

La question « $P = PSPACE ?$ »
dans les modèles BSS et de Valiant

Pascal Koiran Sylvain Perifel

LIP, ENS Lyon

Montpellier, le 24 janvier 2007

Introduction

- ▶ Problèmes de décision

Langages (sur \mathbb{R}), modèle de Blum-Shub-Smale

Exemple : décider si un polynôme à plusieurs variables a une racine réelle ($\text{NP}_{\mathbb{R}}$ -complet)

Introduction

- ▶ Problèmes de décision

Langages (sur \mathbb{R}), modèle de Blum-Shub-Smale

Exemple : décider si un polynôme à plusieurs variables a une racine réelle ($\text{NP}_{\mathbb{R}}$ -complet)

- ▶ Problèmes d'évaluation

Familles de polynômes, modèle de Valiant

Exemple : calculer le permanent d'une matrice
(VNP-complet)

Plan

1. P et PSPACE
2. P et PSPACE dans le modèle BSS
3. P et PSPACE dans le modèle de Valiant
4. Conditions de signe
5. Un vecteur orthogonal

si $VP = VSPACE$ alors $P_{\mathbb{R}} = PAR_{\mathbb{R}}$

P et PSPACE

- ▶ Classe P : langages sur $\{0, 1\}$ reconnus par machine de Turing en temps polynomial.
- ▶ Classe PSPACE : langages sur $\{0, 1\}$ reconnus par machine de Turing en espace polynomial.

P et PSPACE

- ▶ Classe P : langages sur $\{0, 1\}$ reconnus par machine de Turing en temps polynomial.
- ▶ Classe PSPACE : langages sur $\{0, 1\}$ reconnus par machine de Turing en espace polynomial.
- ▶ Machines de Turing \longleftrightarrow circuits booléens (portes \wedge , \vee , \neg).
- ▶ Reconnaissance de langages : un circuit par taille d'entrée.

P et PSPACE

- ▶ Classe P : langages sur $\{0, 1\}$ reconnus par machine de Turing en temps polynomial.
- ▶ Classe PSPACE : langages sur $\{0, 1\}$ reconnus par machine de Turing en espace polynomial.
- ▶ Machines de Turing \longleftrightarrow circuits booléens (portes \wedge, \vee, \neg).
- ▶ Reconnaissance de langages : un circuit par taille d'entrée.
- ▶ Classe P : langages reconnus par circuits booléens de taille polynomiale (+ uniformité).
- ▶ Classe PSPACE : langages reconnus par circuits booléens de *profondeur* polynomiale (taille éventuellement exponentielle) (+ uniformité).

P et PSPACE dans le modèle BSS

- ▶ Circuits algébriques : portes $+$, $-$, \times et \leq .
- ▶ Langages sur \mathbb{R} : ensemble de mots sur l'alphabet \mathbb{R} , i.e. $A \subseteq \cup_{n \geq 0} \mathbb{R}^n$.
- ▶ Reconnaissance de langages sur \mathbb{R} : un circuit par taille d'entrée.

P et PSPACE dans le modèle BSS

- ▶ Circuits algébriques : portes $+$, $-$, \times et \leq .
- ▶ Langages sur \mathbb{R} : ensemble de mots sur l'alphabet \mathbb{R} , i.e. $A \subseteq \cup_{n \geq 0} \mathbb{R}^n$.
- ▶ Reconnaissance de langages sur \mathbb{R} : un circuit par taille d'entrée.
- ▶ Classe $P_{\mathbb{R}}$: langages sur \mathbb{R} reconnus par circuits algébriques de taille polynomiale (+ uniformité).

P et PSPACE dans le modèle BSS

- ▶ Circuits algébriques : portes $+$, $-$, \times et \leq .
- ▶ Langages sur \mathbb{R} : ensemble de mots sur l'alphabet \mathbb{R} , i.e. $A \subseteq \cup_{n \geq 0} \mathbb{R}^n$.
- ▶ Reconnaissance de langages sur \mathbb{R} : un circuit par taille d'entrée.
- ▶ Classe $P_{\mathbb{R}}$: langages sur \mathbb{R} reconnus par circuits algébriques de taille polynomiale (+ uniformité).
- ▶ Classe $PAR_{\mathbb{R}}$: langages sur \mathbb{R} reconnus par circuits algébriques de *profondeur* polynomiale (taille éventuellement exponentielle) (+ uniformité).

P et PSPACE dans le modèle de Valiant

- ▶ Circuits arithmétiques : portes $+$, $-$ et \times , entrées x_1, \dots, x_n et constante 1 \rightarrow calcul d'un polynôme à coefficients entiers.
- ▶ **Famille de polynômes** (f_n) : un circuit C_n par polynôme $f_n \in \mathbb{Z}[x_1, \dots, x_{u(n)}]$.

P et PSPACE dans le modèle de Valiant

- ▶ Circuits arithmétiques : portes $+$, $-$ et \times , entrées x_1, \dots, x_n et constante 1 \rightarrow calcul d'un polynôme à coefficients entiers.
- ▶ **Famille de polynômes** (f_n) : un circuit C_n par polynôme $f_n \in \mathbb{Z}[x_1, \dots, x_{u(n)}]$.
- ▶ Classe VP : familles de polynômes calculés par circuits arithmétiques de taille polynomiale (**+ uniformité**).
(= Uniform VP_{nb}^0)

P et PSPACE dans le modèle de Valiant

- ▶ Circuits arithmétiques : portes $+$, $-$ et \times , entrées x_1, \dots, x_n et constante 1 \rightarrow calcul d'un polynôme à coefficients entiers.
- ▶ **Famille de polynômes** (f_n) : un circuit C_n par polynôme $f_n \in \mathbb{Z}[x_1, \dots, x_{u(n)}]$.
- ▶ Classe VP : familles de polynômes calculés par circuits arithmétiques de taille polynomiale (**+ uniformité**).
(= Uniform VP_{nb}^0)
- ▶ Classe VPSPACE : familles de polynômes calculés par circuits arithmétiques de *profondeur* polynomiale (+ uniformité).

Récapitulatif

- ▶ Problèmes de décision sur $\{0, 1\}$: circuits booléens (portes \wedge , \vee et \neg).
- ▶ Problèmes de décision sur \mathbb{R} (BSS) : circuits algébriques (portes $+$, $-$, \times , \leq).
- ▶ Problèmes d'évaluation (Valiant) : circuits arithmétiques (portes $+$, $-$, \times).

Récapitulatif

- ▶ Problèmes de décision sur $\{0, 1\}$: circuits booléens (portes \wedge , \vee et \neg).
 - ▶ Problèmes de décision sur \mathbb{R} (BSS) : circuits algébriques (portes $+$, $-$, \times , \leq).
 - ▶ Problèmes d'évaluation (Valiant) : circuits arithmétiques (portes $+$, $-$, \times).
-
- ▶ Classe P : circuits de taille polynomiale.
 - ▶ Classe PSPACE : circuits de profondeur polynomiale.

Autres caractérisations de VPSPACE

- ▶ Définition originale : fonction coefficient PSPACE.

$$f_n(\bar{x}) = \sum_{\alpha} a(\alpha) \bar{x}^{\alpha}$$

Fonction $a : \{0, 1\}^* \rightarrow \mathbb{Z}$ calculable chiffre par chiffre en espace polynomial.

Autres caractérisations de VPSPACE

- ▶ Définition originale : fonction coefficient PSPACE.

$$f_n(\bar{x}) = \sum_{\alpha} a(\alpha) \bar{x}^{\alpha}$$

Fonction $a : \{0, 1\}^* \rightarrow \mathbb{Z}$ calculable chiffre par chiffre en espace polynomial.

- ▶ Poizat : circuits de taille polynomiale munis de portes de sommes exponentielles

Autres caractérisations de VPSPACE

- ▶ Définition originale : fonction coefficient PSPACE.

$$f_n(\bar{x}) = \sum_{\alpha} a(\alpha) \bar{x}^{\alpha}$$

Fonction $a : \{0, 1\}^* \rightarrow \mathbb{Z}$ calculable chiffre par chiffre en espace polynomial.

- ▶ Poizat : circuits de taille polynomiale munis de portes de sommes exponentielles
ou de portes d'évaluation en 0 et 1.

Autres caractérisations de VPSPACE

- ▶ Définition originale : fonction coefficient PSPACE.

$$f_n(\bar{x}) = \sum_{\alpha} a(\alpha) \bar{x}^{\alpha}$$

Fonction $a : \{0, 1\}^* \rightarrow \mathbb{Z}$ calculable chiffre par chiffre en espace polynomial.

- ▶ Poizat : circuits de taille polynomiale munis de portes de sommes exponentielles ou de portes d'évaluation en 0 et 1.
- ▶ Exemple : résultant d'un système de polynômes à plusieurs variables.

Théorème de transfert

Si $\text{VPSPACE} = \text{VP}$ alors $\text{PAR}_{\mathbb{R}} = \text{P}_{\mathbb{R}}$.

Plan de la preuve :

- ▶ On prend $A \in \text{PAR}_{\mathbb{R}}$ et on veut décider en temps polynomial si $\bar{x} \in A$.
- ▶ Trouver la condition de signe de \bar{x}
- ▶ Simuler le circuit sur cette condition de signe.

Théorème de transfert

Si $VPSPACE = VP$ alors $PAR_{\mathbb{R}} = P_{\mathbb{R}}$.

Plan de la preuve :

- ▶ On prend $A \in PAR_{\mathbb{R}}$ et on veut décider en temps polynomial si $\bar{x} \in A$.
- ▶ Trouver la condition de signe de \bar{x}
 - ▶ énumération des conditions de signe satisfaisables (Renegar) ;
 - ▶ recherche dichotomique (vecteur orthogonal).
- ▶ Simuler le circuit sur cette condition de signe.

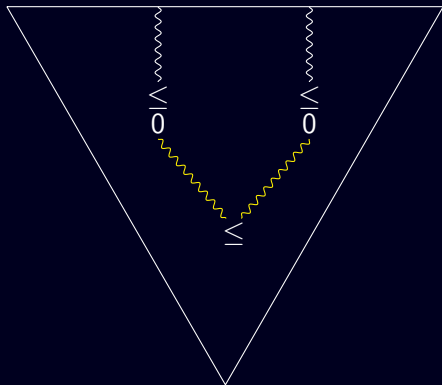
Polynômes testés par un circuit

Porte de test :

$$f(\bar{x}) \leq 0?$$

Si les résultats des tests précédents sont fixés, f est un polynôme.

→ énumération de tous les polynômes possibles (espace polynomial) : famille f_1, \dots, f_s .



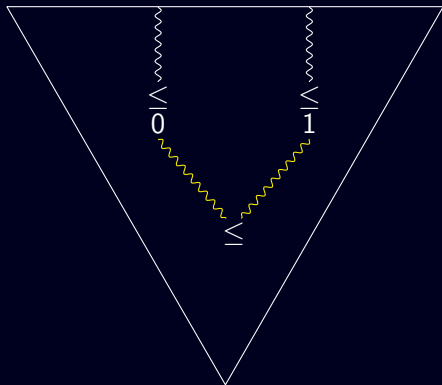
Polynômes testés par un circuit

Porte de test :

$$f(\bar{x}) \leq 0?$$

Si les résultats des tests précédents sont fixés, f est un polynôme.

→ énumération de tous les polynômes possibles (espace polynomial) : famille f_1, \dots, f_s .



Conditions de signe

- ▶ Condition de signe $S \in \{-1, 0, 1\}^s$: signe des polynômes f_1, \dots, f_s .
- ▶ Condition de signe de \bar{x} : $(\text{sign}(f_1(\bar{x})), \dots, \text{sign}(f_s(\bar{x})))$.

Conditions de signe

- ▶ Condition de signe $S \in \{-1, 0, 1\}^s$: signe des polynômes f_1, \dots, f_s .
- ▶ Condition de signe de \bar{x} : $(\text{sign}(f_1(\bar{x})), \dots, \text{sign}(f_s(\bar{x})))$.
- ▶ Si \bar{x} et \bar{y} ont la même condition de signe alors tous les tests donnent le même résultat \longrightarrow \bar{x} et \bar{y} sont simultanément dans le langage ou hors du langage.
- ▶ Il suffit d'étudier la condition de signe (objet booléen).

Conditions de signe satisfaisables

- ▶ Condition de signe $S \in \{-1, 0, 1\}^s$: signe des polynômes f_1, \dots, f_s .
- ▶ Toutes les conditions de signe ne sont pas satisfaisables.
- ▶ Exemple : $x^2 + 1$ donne toujours 1 (toujours positif sur \mathbb{R}).

Conditions de signe satisfaisables

- ▶ Condition de signe $S \in \{-1, 0, 1\}^s$: signe des polynômes f_1, \dots, f_s .
- ▶ Toutes les conditions de signe ne sont pas satisfaisables.
- ▶ Exemple : $x^2 + 1$ donne toujours 1 (toujours positif sur \mathbb{R}).

Théorème (Thom-Milnor 1964, Grigoriev 1988, Renegar 1992)

- ▶ *Il y a $N = (sd)^{O(n)}$ conditions de signe satisfaisables (s : nombre de polynômes, n : nombre de variables, d : degré max).*
- ▶ *On peut énumérer les conditions de signes satisfaisables dans PSPACE.*

Trouver la condition de signe partielle

- ▶ On s'intéresse seulement à savoir si c'est $= 0$ ou $\neq 0$.
- ▶ On ordonne les css partielles « par inclusion » :



Trouver la condition de signe partielle

- ▶ On s'intéresse seulement à savoir si c'est $= 0$ ou $\neq 0$.
- ▶ On ordonne les css partielles « par inclusion » :



- ▶ On cherche la css minimale S qui satisfait

$$\forall k \leq s, S_k = 0 \implies f_k(\bar{x}) = 0.$$

Trouver la condition de signe partielle

- ▶ On s'intéresse seulement à savoir si c'est $= 0$ ou $\neq 0$.
- ▶ On ordonne les css partielles « par inclusion » :



- ▶ On cherche la css minimale S qui satisfait

$$\forall k \leq s, S_k = 0 \implies f_k(\bar{x}) = 0.$$

- ▶ Recherche dichotomique grâce aux tests VPSPACE

$$\prod_{j \leq i} \left(\sum_{S_k^{(j)} = 0} f_k(\bar{x})^2 \right) = 0 \quad (\text{vrai ssi } S \leq i)$$

Trouver la condition de signe partielle

- ▶ On s'intéresse seulement à savoir si c'est $= 0$ ou $\neq 0$.
- ▶ On ordonne les css partielles « par inclusion » :



- ▶ On cherche la css minimale S qui satisfait

$$\forall k \leq s, S_k = 0 \implies f_k(\bar{x}) = 0.$$

- ▶ Recherche dichotomique grâce aux tests VPSPACE

$$\prod_{j \leq i} \left(\sum_{S_k^{(j)} = 0} f_k(\bar{x})^2 \right) = 0 \quad (\text{vrai ssi } S \leq i)$$

Trouver la condition de signe partielle

- ▶ On s'intéresse seulement à savoir si c'est $= 0$ ou $\neq 0$.
- ▶ On ordonne les css partielles « par inclusion » :



- ▶ On cherche la css minimale S qui satisfait

$$\forall k \leq s, S_k = 0 \implies f_k(\bar{x}) = 0.$$

- ▶ Recherche dichotomique grâce aux tests VPSPACE

$$\prod_{j \leq i} \left(\sum_{S_k^{(j)} = 0} f_k(\bar{x})^2 \right) = 0 \quad (\text{vrai ssi } S \leq i)$$

Trouver la condition de signe partielle

- ▶ On s'intéresse seulement à savoir si c'est $= 0$ ou $\neq 0$.
- ▶ On ordonne les css partielles « par inclusion » :



- ▶ On cherche la css minimale S qui satisfait

$$\forall k \leq s, S_k = 0 \implies f_k(\bar{x}) = 0.$$

- ▶ Recherche dichotomique grâce aux tests VPSPACE

$$\prod_{j \leq i} \left(\sum_{S_k^{(j)} = 0} f_k(\bar{x})^2 \right) = 0 \quad (\text{vrai ssi } S \leq i)$$

Condition de signe complète

- ▶ On connaît la condition de signe partielle : on sait quels polynômes s'annulent. On cherche maintenant le signe des autres.
- ▶ Il n'y a pas d'ordre naturel dans lequel la cs serait un maximum.
- ▶ On va éliminer les candidats au fur et à mesure.

Recherche dichotomique

- ▶ Nouvelle convention : 0 pour positif et 1 pour négatif.
- ▶ « Produit scalaire » sur $\{0, 1\}^s$: $u.v = \sum_{i=1}^s u_i v_i \pmod 2$.
- ▶ Soit S la condition de signe de \bar{x} . Soit $u \in \{0, 1\}^s$. On a :

$$u.S = 1 \iff \prod_{i|u_i=1} f_i(\bar{x}) < 0$$

Recherche dichotomique

- ▶ Nouvelle convention : 0 pour positif et 1 pour négatif.
- ▶ « Produit scalaire » sur $\{0, 1\}^s$: $u.v = \sum_{i=1}^s u_i v_i \pmod 2$.
- ▶ Soit S la condition de signe de \bar{x} . Soit $u \in \{0, 1\}^s$. On a :

$$u.S = 1 \iff \prod_{i|u_i=1} f_i(\bar{x}) < 0$$

- ▶ Si u est orthogonal à environ la moitié des css, alors on a « éliminé » environ la moitié des candidats.
→ On répète un nombre logarithmique de fois.

Un vecteur orthogonal

- ▶ Problème : on a une famille de vecteurs $S^{(1)}, \dots, S^{(k)} \in \{0, 1\}^s$. Trouver un vecteur orthogonal à environ la moitié des vecteurs $S^{(i)}$.

Un vecteur orthogonal

- ▶ Problème : on a une famille de vecteurs $S^{(1)}, \dots, S^{(k)} \in \{0, 1\}^s$. Trouver un vecteur orthogonal à environ la moitié des vecteurs $S^{(i)}$.
- ▶ Grigoriev 1998 : il existe toujours un vecteur orthogonal à au moins $k/3$ et au plus $2k/3$ vecteurs. **Non-constructif.**

Un vecteur orthogonal

- ▶ Problème : on a une famille de vecteurs $S^{(1)}, \dots, S^{(k)} \in \{0, 1\}^s$. Trouver un vecteur orthogonal à environ la moitié des vecteurs $S^{(i)}$.
- ▶ Grigoriev 1998 : il existe toujours un vecteur orthogonal à au moins $k/3$ et au plus $2k/3$ vecteurs. **Non-constructif**.
- ▶ Charbit, Jeandel, Koiran, Perifel, Thomassé 2006 :
 - ▶ un vecteur au hasard \rightarrow intervalle $[k/2 - \sqrt{k}; k/2 + \sqrt{k}]$ avec probabilité $1/2$ (Tchebychev, toujours non-constructif) ;

Un vecteur orthogonal

- ▶ Problème : on a une famille de vecteurs $S^{(1)}, \dots, S^{(k)} \in \{0, 1\}^s$. Trouver un vecteur orthogonal à environ la moitié des vecteurs $S^{(i)}$.
- ▶ Grigoriev 1998 : il existe toujours un vecteur orthogonal à au moins $k/3$ et au plus $2k/3$ vecteurs. **Non-constructif**.
- ▶ Charbit, Jeandel, Koiran, Perifel, Thomassé 2006 :
 - ▶ un vecteur au hasard \rightarrow intervalle $[k/2 - \sqrt{k}; k/2 + \sqrt{k}]$ avec probabilité $1/2$ (Tchebychev, toujours non-constructif) ;
 - ▶ on peut dérandomiser en parallèle (donc espace logarithmique).

Récapitulatif

Pour montrer que $VPSPACE = VP \Rightarrow PAR_{\mathbb{R}} = P_{\mathbb{R}}$:

- ▶ On prend $A \in PAR_{\mathbb{R}}$ et on veut décider en temps polynomial si $\bar{x} \in A$.
- ▶ On énumère tous les polynômes pouvant être testés dans le circuit (faisable en espace polynomial).
- ▶ Grâce à des tests $VPSPACE$, on trouve par recherche dichotomique (recherche d'un maximum) la condition de signe partielle de \bar{x} .
- ▶ Pour trouver la condition de signe complète de \bar{x} :
 - ▶ on s'est ramené à $\{0, 1\}$;
 - ▶ grâce au vecteur orthogonal et à des tests $VPSPACE$ on élimine à chaque étape la moitié des cs candidates.
- ▶ Lorsqu'on a la cs de \bar{x} , on peut simuler le circuit et conclure.

Conclusion

- ▶ Étude de la question $P = PSPACE$ dans différents contextes (booléen, BSS, Valiant).
- ▶ Même genre de résultat sur \mathbb{C} , mais techniques différentes : une variété ne s'exprime plus par une seule équation (somme de carrés sur \mathbb{R}).
- ▶ Réciproque ? Sur \mathbb{C} , utilisation du Nullstellensatz \Rightarrow à un multiple près.

Plan

1. P et PSPACE
2. P et PSPACE dans le modèle BSS
3. P et PSPACE dans le modèle de Valiant
4. Conditions de signe
5. Un vecteur orthogonal