VPSPACE and a transfer theorem over the reals

Algebraic versions of the question "$\mathrm{P} = \mathrm{PSPACE}$?"

Pascal Koiran Sylvain Perifel

LIP, ENS Lyon

Aachen, February 23rd, 2007

Introduction

- Decision problems
  Languages (over $\mathbb{R}$), Blum-Shub-Smale model
  Example: decide whether a multivariate polynomial has a real root ($\mathrm{NP}_{\mathbb{R}}$-complete)

Introduction

- Decision problems
  Languages (over $\mathbb{R}$), Blum-Shub-Smale model
  Example: decide whether a multivariate polynomial has a real root ($\mathrm{NP}_{\mathbb{R}}$-complete)

- Evaluation problems
  Families of polynomials, Valiant's model
  Example: compute the permanent of a matrix (VNP-complete)

Outline

1. P and PSPACE (boolean case)
2. P and PSPACE in BSS model
3. P and PSPACE in Valiant's model
4. Sign condition
5. An orthogonal vector

if $\mathrm{VP} = \mathrm{VPSPACE}$ then $\mathrm{P}_{\mathbb{R}} = \mathrm{PAR}_{\mathbb{R}}$

P and PSPACE (boolean case)

- P: languages over $\{0,1\}$ recognized in polynomial time by a Turing machine.
- PSPACE: languages over $\{0,1\}$ recognized in polynomial space by a Turing machine.

P and PSPACE (boolean case)

- P: languages over $\{0,1\}$ recognized in polynomial time by a Turing machine.
- PSPACE: languages over $\{0,1\}$ recognized in polynomial space by a Turing machine.

- Turing machines $\longleftrightarrow$ boolean circuits (gates $\wedge$, $\vee$, $\neg$).
- Language recognition: one circuit per input length.

P and PSPACE (boolean case)

- P: languages over $\{0,1\}$ recognized in polynomial time by a Turing machine.
- PSPACE: languages over $\{0,1\}$ recognized in polynomial space by a Turing machine.

- Turing machines $\longleftrightarrow$ boolean circuits (gates $\wedge$, $\vee$, $\neg$).
- Language recognition: one circuit per input length.
- P: languages recognized by boolean circuits of polynomial size (+ uniformity).
- PSPACE: languages recognized by boolean circuits of polynomial *depth* (of possibly exponential size) (+ uniformity).

P and PSPACE in BSS model

- Algebraic circuits: gates $+$, $-$, $\times$ and $\leq$.
- Languages over $\mathbb{R}$: sets of words over the alphabet $\mathbb{R}$, that is, $A \subseteq \cup_{n \geq 0} \mathbb{R}^n$.
- Language recognition over $\mathbb{R}$: one circuit per input length.

P and PSPACE in BSS model

- Algebraic circuits: gates $+$, $-$, $\times$ and $\leq$.
- Languages over $\mathbb{R}$: sets of words over the alphabet $\mathbb{R}$, that is, $A \subseteq \cup_{n\geq 0}\mathbb{R}^n$.
- Language recognition over $\mathbb{R}$: one circuit per input length.
- $\mathrm{P}_{\mathbb{R}}$: languages over $\mathbb{R}$ recognized by algebraic circuits of polynomial size ($+$ uniformity).

P and PSPACE in BSS model

- Algebraic circuits: gates $+$, $-$, $\times$ and $\leq$.
- Languages over $\mathbb{R}$: sets of words over the alphabet $\mathbb{R}$, that is, $A \subseteq \cup_{n \geq 0} \mathbb{R}^n$.
- Language recognition over $\mathbb{R}$: one circuit per input length.
- $\mathrm{P}_{\mathbb{R}}$: languages over $\mathbb{R}$ recognized by algebraic circuits of polynomial size ($+$ uniformity).
- $\mathrm{PAR}_{\mathbb{R}}$: languages over $\mathbb{R}$ recognized by algebraic circuits of polynomial *depth* (of possibly exponential size) ($+$ uniformity).

P and PSPACE in Valiant's model

- Arithmetic circuits: gates $+$, $-$ and $\times$, inputs $x_1, \ldots, x_n$ and constant 1 $\longrightarrow$ multivariate polynomial with integer coefficients.
- Family of polynomials (f_n): one circuit C_n per polynomial $f_n \in \mathbb{Z}[x_1, \ldots, x_{u(n)}]$.

P and PSPACE in Valiant's model

- Arithmetic circuits: gates $+$, $-$ and $\times$, inputs $x_1, \ldots, x_n$ and constant 1 $\longrightarrow$ multivariate polynomial with integer coefficients.
- Family of polynomials (f_n): one circuit C_n per polynomial $f_n \in \mathbb{Z}[x_1, \ldots, x_{u(n)}]$.
- VP: families of polynomials computed by arithmetic circuits of polynomial size (+ uniformity).

$(= \text{Uniform } \mathrm{VP}^0_{\mathrm{nb}})$

P and PSPACE in Valiant's model

- Arithmetic circuits: gates $+$, $-$ and $\times$, inputs $x_1, \ldots, x_n$ and constant 1 $\longrightarrow$ multivariate polynomial with integer coefficients.
- Family of polynomials (f_n): one circuit C_n per polynomial $f_n \in \mathbb{Z}[x_1, \ldots, x_{u(n)}]$.
- VP: families of polynomials computed by arithmetic circuits of polynomial size $(+$ uniformity).

  $(= \mathrm{Uniform\ VP}^0_{\mathrm{nb}})$
- $\mathrm{VPSPACE}$: families of polynomials computed by arithmetic circuits of polynomial *depth* $(+$ uniformity).

Recapitulation

- Decision problems over $\{0, 1\}$: boolean circuits (gates $\wedge$, $\vee$ et $\neg$).
- Decision problems over $\mathbb{R}$ (BSS): algebraic circuits (gates $+$, $-$, $\times$, $\leq$).
- Evaluation problems (Valiant): arithmetic circuits (gates $+$, $-$, $\times$).

Recapitulation

- Decision problems over $\{0, 1\}$: boolean circuits (gates $\wedge$, $\vee$ et $\neg$).
- Decision problems over $\mathbb{R}$ (BSS): algebraic circuits (gates $+$, $-$, $\times$, $\leq$).
- Evaluation problems (Valiant): arithmetic circuits (gates $+$, $-$, $\times$).

- P: circuits of polynomial size.
- PSPACE: circuits of polynomial depth.

Other characterizations of VPSPACE

- Original definition: coefficient function in PSPACE.

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

Function $a : \{0,1\}^* \to \mathbb{Z}$ computable bit by bit in polynomial space.

Other characterizations of $\mathrm{VPSPACE}$

- Original definition: coefficient function in PSPACE.

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

  Function $a : \{0,1\}^* \rightarrow \mathbb{Z}$ computable bit by bit in polynomial space.

- Poizat: circuits of polynomial size endowed with exponential summation gates

Other characterizations of $\mathrm{VPSPACE}$

- Original definition: coefficient function in PSPACE.

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

  Function $a : \{0,1\}^* \to \mathbb{Z}$ computable bit by bit in polynomial space.

- Poizat: circuits of polynomial size endowed with exponential summation gates
  or gates of evaluation at 0 and 1.

Other characterizations of $\mathrm{VPSPACE}$

- Original definition: coefficient function in PSPACE.

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

  Function $a : \{0,1\}^* \to \mathbb{Z}$ computable bit by bit in polynomial space.
- Poizat: circuits of polynomial size endowed with exponential summation gates
  or gates of evaluation at 0 and 1.
- Example: multivariate resultant of a system of polynomials.

Transfer theorem

$$\text{If } \mathrm{VPSPACE} = \mathrm{VP} \text{ then } \mathrm{PAR}_{\mathbb{R}} = \mathrm{P}_{\mathbb{R}}.$$

Outline of the proof:

- Goal: for $A \in \mathrm{PAR}_{\mathbb{R}}$, decide in polynomial time whether $\bar{x} \in A$.
- Find the sign condition of $\bar{x}$
- Simulate the circuit on this sign condition.

Transfer theorem

If $\mathrm{VPSPACE} = \mathrm{VP}$ then $\mathrm{PAR}_{\mathbb{R}} = \mathrm{P}_{\mathbb{R}}$.

Outline of the proof:

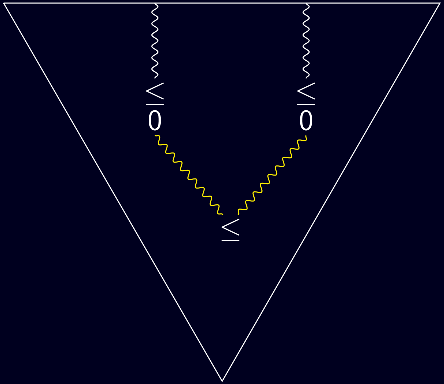
- Goal: for $A \in \mathrm{PAR}_{\mathbb{R}}$, decide in polynomial time whether $\bar{x} \in A$.
- Find the sign condition of $\bar{x}$
  - enumeration of the satisfiable sign conditions (Renegar);
  - binary search (orthogonal vector).
- Simulate the circuit on this sign condition.

Polynomials tested by a circuit

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

If the results of the preceding tests are fixed, f is a polynomial.

$\longrightarrow$ enumeration of all possible polynomials (polynomial space): family $f_1, \ldots, f_s$.

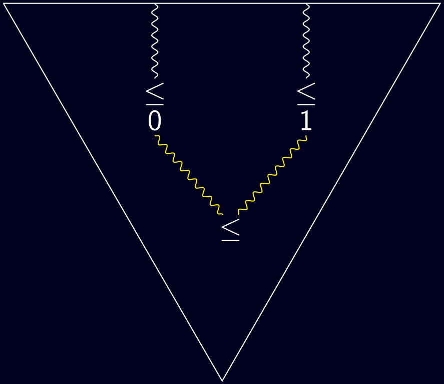


Polynomials tested by a circuit

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

If the results of the preceding tests are fixed, f is a polynomial.

$\rightarrow$ enumeration of all possible polynomials (polynomial space): family $f_1, \ldots, f_s$.



Sign conditions

- Sign condition $S \in \{-1, 0, 1\}^s$: sign of the polynomials $f_1, \ldots, f_s$.
- Sign condition of $\bar{x}$: $(\mathrm{sign}(f_1(\bar{x})), \ldots, \mathrm{sign}(f_s(\bar{x})))$.

Sign conditions

- Sign condition $S \in \{-1, 0, 1\}^s$: sign of the polynomials $f_1, \ldots, f_s$.
- Sign condition of $\bar{x}$: $(\mathrm{sign}(f_1(\bar{x})), \ldots, \mathrm{sign}(f_s(\bar{x})))$.
- If $\bar{x}$ and $\bar{y}$ have the same sign condition then every test gives the same result $\longrightarrow$ $\bar{x}$ and $\bar{y}$ are simultaneously in the language or outside of the language.
- It is enough to study the sign condition (boolean object).

Satisfiable sign conditions

- Sign condition $S \in \{-1, 0, 1\}^s$: sign of the polynomials $f_1, \ldots, f_s$.
- A sign condition is not necessarily satisfiable.
- Example: $x^2 + 1$ always yields 1 (always positive over $\mathbb{R}$).

Satisfiable sign conditions

- Sign condition $S \in \{-1, 0, 1\}^s$: sign of the polynomials $f_1, \ldots, f_s$.
- A sign condition is not necessarily satisfiable.
- Example: $x^2 + 1$ always yields 1 (always positive over $\mathbb{R}$).

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

- *There are $N = (sd)^{O(n)}$ satisfiable sign conditions (s: number of polynomials, n: number of variables, d: max degree).*
- *Satisfiable sign conditions can be enumerated in PSPACE.*

Finding the partial sign condition

- We first want to distinguish only between $= 0$ and $\neq 0$.
- Linear order compatible with inclusion on satisfiable sign conditions:

$$\longmapsto$$

Finding the partial sign condition

- We first want to distinguish only between $= 0$ and $\neq 0$.
- Linear order compatible with inclusion on satisfiable sign conditions:

- Search of the minimal satisfiable sign condition S satisfying

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

Finding the partial sign condition

- We first want to distinguish only between $= 0$ and $\neq 0$.
- Linear order compatible with inclusion on satisfiable sign conditions:

- Search of the minimal satisfiable sign condition S satisfying

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

- Binary search thanks to $\mathrm{VPSPACE}$ tests

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

Finding the partial sign condition

- We first want to distinguish only between $= 0$ and $\neq 0$.
- Linear order compatible with inclusion on satisfiable sign conditions:

- Search of the minimal satisfiable sign condition S satisfying

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

- Binary search thanks to $\mathrm{VPSPACE}$ tests

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

Finding the partial sign condition

- We first want to distinguish only between $= 0$ and $\neq 0$.
- Linear order compatible with inclusion on satisfiable sign conditions:

- Search of the minimal satisfiable sign condition S satisfying

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

- Binary search thanks to $\mathrm{VPSPACE}$ tests

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

Finding the partial sign condition

- We first want to distinguish only between $= 0$ and $\neq 0$.
- Linear order compatible with inclusion on satisfiable sign conditions:

- Search of the minimal satisfiable sign condition S satisfying

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

- Binary search thanks to $\mathrm{VPSPACE}$ tests

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

Complete sign condition

- Partial sign condition is known: we know which polynomials vanish. We are now looking for the sign of the others.
- There is no natural order in which the sign condition would be a maximum.
- Candidates will be eliminated step by step.

Binary search

- New convention: 0 for positive and 1 for negative.
- "Inner product" over $\{0,1\}^s$: $u.v = \sum_{i=1}^{s} u_i v_i \mod 2$.
- Let S be the sign condition of $\bar{x}$. Let $u \in \{0,1\}^s$. We have:

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

Binary search

- New convention: 0 for positive and 1 for negative.
- "Inner product" over $\{0,1\}^s$: $u.v = \sum_{i=1}^{s} u_i v_i \mod 2$.
- Let S be the sign condition of $\bar{x}$. Let $u \in \{0,1\}^s$. We have:

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

- If u is orthogonal to roughly half the satisfiable sign conditions then we have "eliminated" roughly half of the candidates.
  $\longrightarrow$ Logarithmic number of repetitions.

An orthogonal vector

- Problem: given is a family of vectors $S^{(1)}, \ldots, S^{(k)} \in \{0,1\}^s$. Find a vector orthogonal to roughly half of the vectors $S^{(i)}$.

An orthogonal vector

- Problem: given is a family of vectors $S^{(1)}, \dots, S^{(k)} \in \{0,1\}^s$. Find a vector orthogonal to roughly half of the vectors $S^{(i)}$.

- Grigoriev 1998: there always exists a vector orthogonal to at least $k/3$ and at most $2k/3$ vectors. Nonconstructive.

An orthogonal vector

- Problem: given is a family of vectors $S^{(1)}, \ldots, S^{(k)} \in \{0,1\}^s$. Find a vector orthogonal to roughly half of the vectors $S^{(i)}$.
- Grigoriev 1998: there always exists a vector orthogonal to at least $k/3$ and at most $2k/3$ vectors. Nonconstructive.
- Charbit, Jeandel, Koiran, Perifel, Thomassé 2006:
  - a random vector $\rightarrow$ interval $[k/2 - \sqrt{k}; k/2 + \sqrt{k}]$ with probability 3/4 (Chebyshev's inequality, still nonconstructive);

An orthogonal vector

- Problem: given is a family of vectors $S^{(1)}, \dots, S^{(k)} \in \{0,1\}^s$. Find a vector orthogonal to roughly half of the vectors $S^{(i)}$.
- Grigoriev 1998: there always exists a vector orthogonal to at least $k/3$ and at most $2k/3$ vectors. Nonconstructive.
- Charbit, Jeandel, Koiran, Perifel, Thomassé 2006:
  - a random vector $\rightarrow$ interval $[k/2 - \sqrt{k}; k/2 + \sqrt{k}]$ with probability 3/4 (Chebyshev's inequality, still nonconstructive);
  - it can be derandomized in parallel (hence logarithmic space).

Recapitulation

In order to show that $\mathrm{VPSPACE} = \mathrm{VP} \Rightarrow \mathrm{PAR}_{\mathbb{R}} = \mathrm{P}_{\mathbb{R}}$:

- For $A \in \mathrm{PAR}_{\mathbb{R}}$ we want to decide in polynomial time whether $\bar{x} \in A$.
- We enumerate all the polynomials possibly tested in the cricuit (polynomial space).
- Thanks to $\mathrm{VPSPACE}$ tests, a binary search gives the partial sign condition of $\bar{x}$.
- In order to find the complete sign condition of $\bar{x}$:
  - we are back on $\{0, 1\}$;
  - thanks to the orthogonal vector and $\mathrm{VPSPACE}$ tests, we eliminate at each step half of the candidate sign conditions.
- Once the sign condition of $\bar{x}$ is obtained, we can simulate the circuit and conclude.

Conclusion

- Study of the question $\mathrm{P} = \mathrm{PSPACE}$ in different contexts (boolean, BSS, Valiant).
- Similar results over $\mathbb{C}$ but different techniques: a variety requires more than one equation (unlike over $\mathbb{R}$ where we can make sums of squares).
- Converse? Over $\mathbb{C}$, Nullstellensatz $\Rightarrow$ work only up to a multiple.

Outline

1. P and PSPACE (boolean case)
2. P and PSPACE in BSS model
3. P and PSPACE in Valiant's model
4. Sign condition
5. An orthogonal vector