Math 268 2012-10-16

LFKN '90, JACM 92 $IP \supseteq \#P$, $IP \supseteq PH$

Shamir, '90 FOCS, JACM 92 $IP \supseteq PSPACE$

Shen, JACM 92

O. Meir, unp 2012

PSPACE: space used (in turing machine) $\le p(|x|)$ ← polynomial

so WLOG, time $\le 2^{O[p(|x|)]} = 2^{n^{O(1)}}$

IP: protocol for deciding whether $x \in L$ (language) $x \in \{0,1\}^*$

V - verifier - p polynomial time probabilistic algorithm public coins

P - prover - WLOG a function

V is a turing machine with a special communication tape.

V runs, writes q_i on its communication tape, P responds with $P(q_1, \dots, q_i, \underbrace{r_1, \dots, r_i}_{\text{coins}})$. At some point V must accept/reject.

~~If x∈L~~ V, P is an IP for L: If $x \in L$, $Pr[V \text{ accepts working with } P] = 1$ ($> 2/3$).

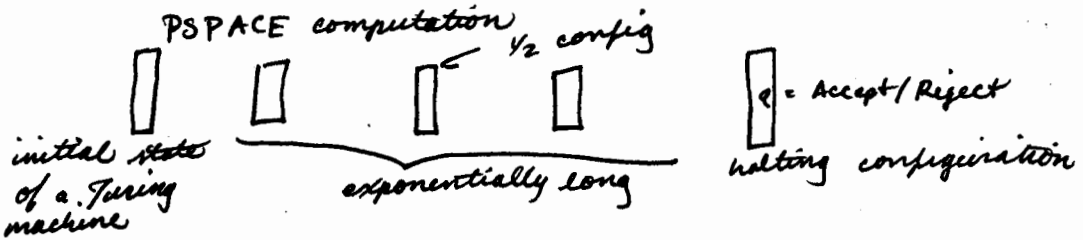
If $x \notin L$, then for all P' (evil provers) $Pr[V, \text{working w/ } P' \text{ accepts}] \le 1/3$ ($< 1/2^n$).

(Think of ourselves as V)

Thm $IP \subseteq PSPACE$ - exhaustively branch over random sequences

Thm $IP \supseteq PSPACE$

First, a proof that doesn't work: Savitch divide-and-conquer



V: ask P for $\frac{1}{2}$ config, P gives it

V: flip $\{0,1\}$ at random; if $r=0$, ask 1/4 config; if $r=0$, ask 1/4 conf
$r=1$, ask 3/4 conf

at the end, get adjacent configs & check if they follow

But an evil prover can pick some random point, and switch to a computation of something in the language - unless V happens to land on the switch point, V will be tricked.

Correct proof: Basically we need more information from the prover.

<u>Quantified Boolean Formulas</u> (QBF)

QBF is PSPACE-complete

Variable $x_1, x_2, \ldots$ $\quad \wedge, \vee, \neg$ $\quad$ e.g., $\forall x_1 \exists x_2 \forall x_3 ((x_1 \vee x_2) \wedge x_3)$

$\forall x_i \in \{0,1\}$ $\quad \exists x_i \in \{0,1\}$

It will suffice to prove QBF $\in$ IP.

Idea: convert a QBF to a polynomial:

(0) WLOG, negatives are only on variables (De Morgan: $\neg(\varphi \wedge \psi) \equiv \neg\varphi \vee \neg\psi$)

(1) For boolean x_i, we have numeric (integer) $z_i \in \{0,1\}$

$x_i \mapsto z_i =: q_{x_i}$ (I.e., for numeric vars / polys

$\neg x_i \mapsto 1 - z_i =: q_{\bar{x}_i}$ $\quad 0$ is false >0 is true)

$q_{\varphi \wedge \psi} := q_\varphi \cdot q_\psi$ $\qquad q_{\varphi \vee \psi} := q_\varphi + q_\psi$. We don't do simplification.

$q_{\forall x_i \varphi} := \prod_{z_i \in \{0,1\}} q_\varphi$ $\qquad q_{\exists x_i \varphi} := \sum_{z_i \in \{0,1\}} q_\varphi$

↑ degrees double

q_φ - polynomial over $\mathbb{N}$, written compactly. Suppose φ has $|\varphi|$ symbols, $\deg(q_\varphi) \le 2^{|\varphi|}$ $\quad$ If no $\forall$ in φ, then $\deg(q_\varphi) \le |\varphi|$.

value $(q_\varphi) \le 2^{2^{|\varphi|}}$

If φ has a free variable x, $\deg_x(q_\varphi) \le 2^{(\#\forall\text{'s } x \text{ is in the scope of})} \cdot |\varphi|$

Goal: (Evil) P wants to convince V that $q_\varphi > 0$, i.e., φ is true (for φ a sentence - all vars bound).

First step: P will pick prime p, send to V. New goal: convince V that $q_\varphi \neq 0 \bmod p$. $\quad$ Else send $b \in \mathbb{Z}_p$, $b \neq 0$, s.t. $q_\varphi \equiv b \pmod p$

WLOG p can be $\le 2^{n^2}$ by CRT and Prime # theorem.

(P sends a Pratt certificate of the primality of p.)

Suppose φ: $\forall x_1 \exists x_2 \forall x_3 \cdots$ $\quad$ subformula ψ: $\forall x_3 \exists x_4 \cdots (x_1, x_2)$

$q_\psi = \varphi_\psi(z_1, z_2)$. $\quad$ ↑ free in φ

$\deg_{z_i}(q_\psi) \le 2^{|\varphi|}$ want $\le |\varphi|^{O(1)}$

<u>Trick</u> WLOG φ is originally in prenex form, but will be tranformed to a nearly prenex form.

Say φ is simple iff for every subformula ψ, if x is free var of ψ, x appears in the scope of at most one $\forall$ in ψ.

$\deg_x(\psi) \le 2|\psi| \le 2|\varphi|$.

Trick: consider ψ subformula of φ of form

$$\forall x_\ell \, \chi(\underbrace{x_1, \dots,}_{\text{free in } \psi} x_{\ell-1}, x_\ell)$$

replace by $\forall x_\ell \, \exists x_1' \exists x_2' \cdots \exists x_{\ell-1}' \left([x_1 \leftrightarrow x_1'] \wedge [x_2 \leftrightarrow x_2'] \wedge \cdots \wedge [x_{\ell-1} \leftrightarrow x_{\ell-1}'] \right) \wedge \chi(x_1', x_2', \dots, x_{\ell-1}', x_\ell))$

perform transformations to φ in parallel & keep names straight.

Thm cl$\varphi \in$ QBF, $\exists \varphi^*$ formed as above s.t. $\deg_x(\psi) \le 2|\varphi|$ for subformulas x of φ^*, x free in ψ, $|\varphi^*| = O(|\varphi|^2)$

IP-Protocol Step 0: φ is WLOG simple and nearly prenex

Step 1: P tells V a prime p (w/ certificate), a value $b \in \mathbb{Z}_p^*$, P claims $g_\varphi \equiv b \pmod p$, $|p| \le |\varphi|^2$. V rejects if any of these dont hold.

Invariant: We have a subformula ψ of φ, value b, $\psi = \chi(x_1, \dots, x_\ell)$ values $a_1 \dots a_\ell \in \mathbb{Z}_p$ and P is claiming $g_\chi(a_1, \dots, a_\ell) = b$.

Step $i+1$: (b) clψ is quantifier-free, then V evaluates g_ψ and accepts/rejects if $g_\psi = b$ or not

(b) ψ is $\forall x_{\ell+1} \chi$. V asks P for $g(z_{\ell+1})$ $\deg(g) \le 2|\varphi|$ poly over $\mathbb{Z}_p$ that is equal to $g_{\forall_{x_{\ell+1}} \chi}(a_1, \dots, a_\ell, z_{\ell+1})$. V checks that $g(0)g(1) = b$, if not V rejects. Otherwise V chooses $a_{\ell+1} \in \mathbb{Z}_p$ at random, compute $b' = g(a_{\ell+1})$, set $b := b'$, $\psi = \chi$, and go to next step. (I.e., P was claiming $g_{a_{\ell+1}} = b = g_\psi(a_1, \dots, a_{\ell+1})$.)

(c) $\exists x_{\ell+1}$ same, but use $g(0) + g(1)$.

(d) ψ is $\chi_0 \wedge \chi$ where χ_0 is quantifier-free. V evaluates $\chi_0(a_1, \dots, a_\ell) = b'$. $b := b(b')^{-1} \bmod p$ (if $b' \neq 0$), $\psi := \chi$. If $b' = 0$, V accepts $b = 0$.

The good news: if $g(z_{\ell+1}) \neq g_\psi(\vec{a}, z_{\ell+1})$, then for $b \in \mathbb{Z}_p$ at random, degrees $\le 2|\varphi|$, the probability $g(a_{\ell+1}) = g_\psi(\vec{a}, a_{\ell+1}) \in \frac{2|\varphi|}{p}$, $p \in [2^n, 2^{n^2}]$ $\le \frac{2n}{2^n}$ where $n = |\varphi|$. So total Pr is $n \cdot \frac{n}{2^n} < 1/3$.

Why do we need polynomials? Lies uses error-correcting codes / finite fields. Take away: why cant we just use boolean computation.