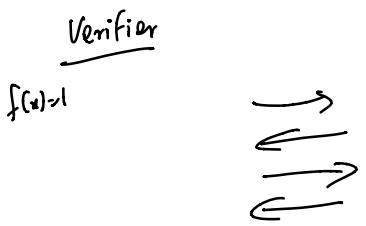
Thm: $IP = PSPACE$ $\quad$ $NP \subseteq IP = PSPACE$.

$f \in IP$ if $\exists$ poly time randomized verifier V

s.t $\quad f(x)=1 \Rightarrow \exists P \; Pr[V^P(x)=1] \geq 2/3$.

$f(x)=0 \Rightarrow \forall P \quad Pr[V^P(x)=0] \leq 1/3$.

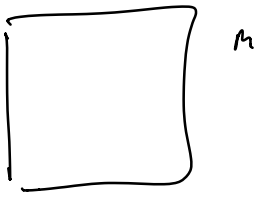


Main challenge: $TQBF \in IP$.

Baby Step: Perm

$$Perm(M) = \sum_{\pi} \prod_{i=1}^{n} M_{i,\pi(i)}$$

$$= \sum_{j=1}^{n} M_{1,j} \cdot Perm(M^{1,j})$$

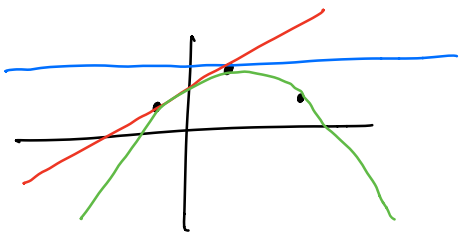


where $M^{1,j}$ is $(n-1) \times (n-1)$ submatrix obtained by deleting row 1, column j from M.

$D(j) = M^{1,j}$: $\quad Perm(M) = \sum_{j=1}^{n} M_{1,j} \cdot Perm(D(j))$

FACT: Given n pairs of points $(a_1, b_1), (a_2, b_2), \ldots, (a_n, b_n)$ with $a_1, \ldots, a_n$ distinct, $\exists$ a degree $(n-1)$ polynomial f s.t

$f(a_1) = b_1$, $f(a_2) = b_2$, $f(a_3) = b_3$, ..., $f(a_n) = b_n$



$(g-f) \in$ deg $n-1$ poly

non-zero

n roots.

$$D(x) = \begin{bmatrix} f_{1,1}(x) & f_{1,2}(x) & \cdots \\ & \ddots & \\ f_{n,1}(x) & & f_{n,n}(x) \end{bmatrix}$$ → deg $n-1$ polynomials.

$\forall \quad D(j) = M^{1,j}$

x ——— X

$$Perm(M) = \sum_j M_{1,j} \; Perm(D(j))$$

$$g(X) = Perm(D(X)) \quad \text{deg } n^2.$$

Goal: To prove $Perm(M) = k$

Verifier | Prover

$g(X)$ ⟵

verify $\sum_j M_{i,j} \cdot g(j) = k$

pick a at random ⟶ a

⋮

recursively prove

$$Perm(D(a)) = g(a)$$

Fine $\quad Perm(M) = k \bmod p$

1. Prover sends prime $p > 2^{n^2}$
2. Prover sends $g(X)$ mod p.
3. Ver. Check $k = \sum_j m_{1,j} \cdot g(j)$.

Ver. picks $a \in \mathbb{F}_p$ uniformly

recursively checks

$$g(a) = Perm(D(a)).$$

If $g(X) \neq Perm(D(X))$

$$\Pr_a[\, g(a) = Perm(D(a)) \,] \leq \frac{n^2}{p} \leq \frac{n^2}{2^{n^2}}$$

Claim: If $Perm(M) \neq k$, prover succeeds with prob. at most $n^3/2^{n^2}$.

TQBF: $\exists x_1 \forall x_2 \exists x_3 \cdots \quad \phi(x_1, \ldots, x_n)$

Baby step 2: Counting # sat. assignments.

$$x \wedge y = xy$$

$$\neg x = 1 - x$$

$$x \vee y = 1 - (1-x)(1-y)$$

$\phi(x_1, \ldots, x_n) = g_\phi(x_1, \ldots, x_n)$ } deg $O(n)$.

Goal: IP protocol to compute. $(p > 2^{2n})$

$$f(0) + f(1) = \sum_{x_1, \ldots x_n} g_\phi(x_1, \ldots, x_n) = k \bmod p.$$

1. Prover sends $k > p$.
2. Prover sends $f(X) \stackrel{?}{=} \sum_{x_2, \ldots, x_n} g_\phi(X, x_2, \ldots, x_n)$
3. Verifier checks $f(0) + f(1) = k$.
4. Verifier picks $a \in \mathbb{F}_p$ uniformly.
   recursively checks
   $$f(a) = \sum_{x_2, \ldots, x_n} g_\phi(a, x_2, \ldots, x_n).$$

$$\Pr(\text{error}) \leq \frac{O(n)}{p} \cdot n \ll 1/3.$$

X ——————— X

$$\exists x_1 \forall x_2 \exists x_3 \cdots \forall x_n \; \phi(x_1, \ldots, x_n)$$

equivalent to showing

$$\sum_{x_1} \prod_{x_2} \sum_{x_3} \cdots \prod_{x_n} g_\phi(x_1, \ldots, x_n) \equiv k \mod p.$$

$$f(X) = \prod_{x_2} \cdots g_\phi(X, x_2, \ldots, x_n).$$