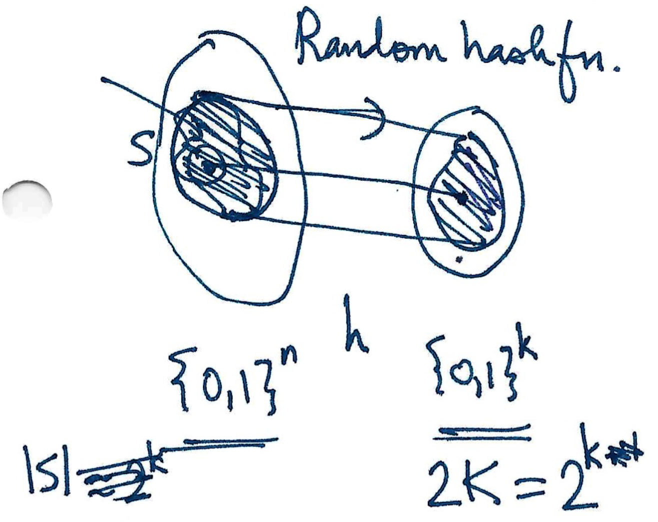
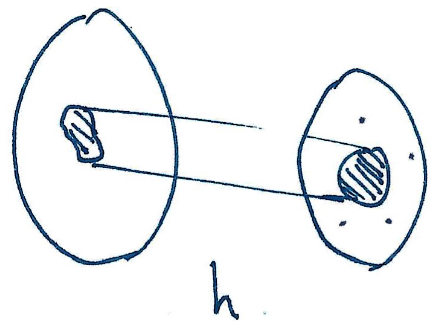


YES



NO

Arthur: (1) Picks random hash fn. h

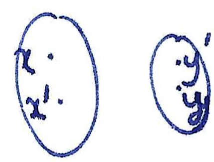
(2) Pick random y in range $\{0,1\}^k$

Asks Merlin "give me $x \in S$ s.t. $h(x) = y$ & give me certificate that $x \in S$."

Def: Pairwise independent hash fn. family

Let $\mathcal{H}_{n,k}$ be a ~~set~~ family of fns from $\{0,1\}^n \to \{0,1\}^k$.

This family is Pairwise Independent if



$\forall \underbrace{x \neq x'}_{\in \{0,1\}^n}, \forall y, y' \in \{0,1\}^k$

$$\Pr_{h \in \mathcal{H}_{n,k}} [h(x) = y \wedge h(x') = y'] = 1/2^{2k}$$

Clm: Let $\mathcal{H}_{n,k}$ be the family of $k \times n$ Boolean matrices

$$\begin{array}{c} \overset{k}{\longleftrightarrow} \\ n \updownarrow \begin{bmatrix} & & \\ & & \\ & & \end{bmatrix} \end{array} \begin{bmatrix} x \end{bmatrix} = \begin{bmatrix} y \end{bmatrix}$$

$h \equiv$ Pick M uar from $\mathcal{H}_{n,k}$

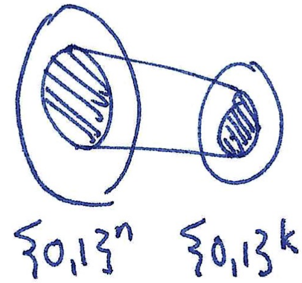
$h(x) = Mx$

$\mathcal{H}_{n,k}$ is a pairwise ind. hash family.

~~$x \in L \Longrightarrow$~~

YES : $|S| \geq 2K$ NO: $|S| \leq K$

Choose k s.t. $2^k \in [4K, 8K)$



Clm: In YES case, $\dfrac{|h(S)|}{2^k} \geq \dfrac{3}{2}\left(\dfrac{K}{2^k}\right)$

In NO case, $\dfrac{|h(S)|}{2^k} \leq \left(\dfrac{K}{2^k}\right)$

Proof: In NO case, $|h(S)| \leq |S| \leq K$ $\quad \dfrac{|h(S)|}{K} \leq \dfrac{K}{2^k}$

In YES case,

$$\Pr_{\substack{y \in \{0,1\}^k \\ h \sim \mathcal{H}_{n,k}}} [\exists\, x \in S, \text{ s.t. } h(x) = y]$$

I give a lower bound when $|S| = 2K$. This lower bound holds when $|S| \geq 2K$.
(by monotonicity)

$$\Pr_{h,y}\left[\bigcup_{x\in S}(h(x)=y)\right] \geqslant \sum_{x\in S}\Pr_{h,y}[h(x)=y]$$

$$- \frac{1}{2}\sum_{\substack{x_1,x_2\in S\\ x_1\neq x_2}} \Pr\left[h(x_1)=y \wedge h(x_2)=y\right]$$

$$\geqslant \frac{|S|}{2^k} - \frac{1}{2}\frac{|S|^2}{2^{2k}} = \frac{|S|}{2^k}\underbrace{\left(1-\frac{|S|}{2\times 2^k}\right)}_{\geqslant 3/4}$$

$|S| = 2K$

$2^k \geqslant 4K$

$$= 2\left(\frac{K}{2^k}\right)\times\frac{3}{4} = \frac{3}{2}\left(\frac{K}{2^k}\right)$$

∎

<u>Clm</u>: For the GS protocol for GNI ; $(p=\Theta(1))$

Arthur computes a value $p\left(=\frac{K}{2^k}\right)$. $p \geqslant \frac{1}{8}$

If $G_0 \not\cong G_1$, $\Pr[\text{Arthur accepts}] \geqslant \left(\frac{3}{2}\right)p$

If $G_0 \cong G_1$, $\Pr[\text{Arthur accepts}] \leqslant p$

$\mathbb{IP} = \mathbb{PSPACE}$

[Lund-Fortnow-Karloff-Nisan 90, Shamir 90]

We will show $\#SAT \in \mathbb{IP}$

→ counting # satisfying assignments

$\#SAT = \{\langle \Phi, K\rangle \mid \Phi$ has K satisfying assignments$\}$ → CNF

subsumes $\mathbb{NP}$ & co-$\mathbb{NP}$

Arithmetization of SAT: Convert 3CNF Φ into a polynomial.

Literal $\quad x_i \rightarrow X_i$ $\qquad$ AND $\rightarrow$ Multiplication

$\qquad\quad \overline{x_i} \rightarrow (1-X_i)$ $\qquad$ Truth table

Clause $\quad x_i \vee \overline{x_j} \vee x_k$

X_i	X_j	X_k	
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1

$$\delta + \sum_{i=1}^{3} \alpha_i X_i + \cancel{\sum_{j=1}^{3} \beta_j}\ \beta_1 X_1 X_2 + \beta_2 X_2 X_3 + \beta_3 X_1 X_3 + \gamma X_1 X_2 X_3$$

$$x_i \vee \overline{x_j} = 1 - (1-X_i) X_j$$

$$A \vee B = 1 - (1-p(A))(1-p(B))$$

$$x_i \vee \overline{x_j} \vee x_k = -(1-X_i)(X_j)(1-X_k) + 1$$

Clause becomes degree 3 polynomial over $\underbrace{X_i, X_j, X_k}_{\text{variables in clause}}$

$$\Phi = C_1 \wedge C_2 - - - \wedge C_m$$

$P(C_k) \leftarrow$ polynomial of clause C_k

$$P(\Phi) \Rightarrow \prod_{i=1}^{m} P(C_i)$$

$$P(\Phi) = \prod_{i=1}^{m} (P(C_i))(X_1, - - , X_n)$$

$P(\Phi)(X_1, \ldots, X_n)$ agrees with $\Phi(x_1, \ldots x_n)$

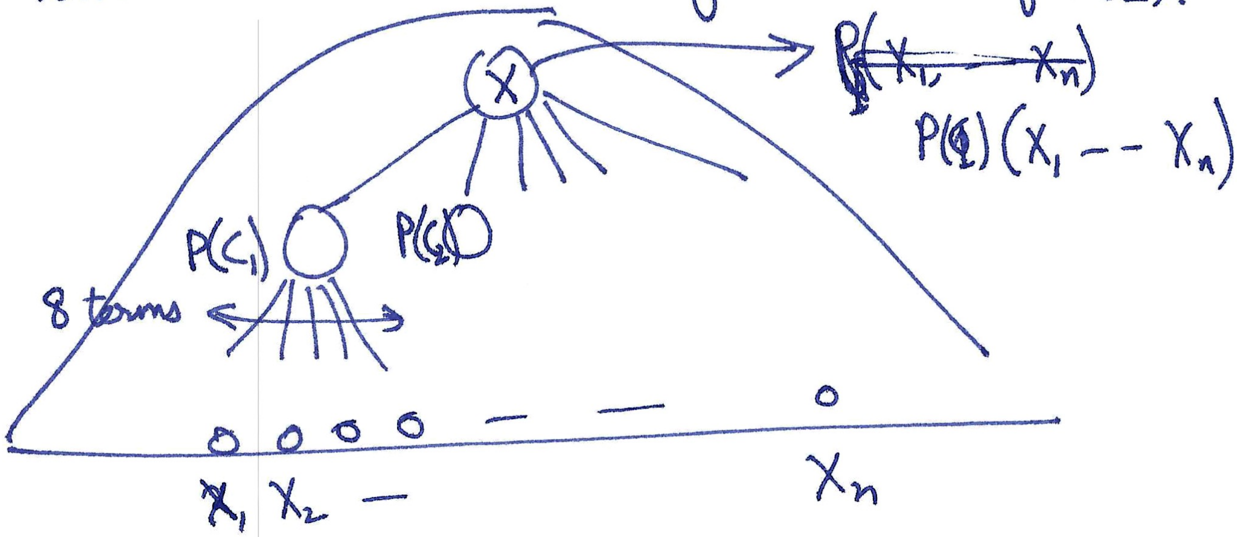
when X_i's are in $\{0,1\}$
$\quad\;\; \overset{\shortparallel}{x_i}$

$P(\Phi)$ has n variables and degree $d \leq \underset{\uparrow \text{ \#clause}}{3m} = \text{poly}(n)$

Expansion of

$P(\Phi)$ may have exponentially many monomials.

We have a concise, circuit representation of $P(\Phi)$.



$\langle \Phi, K \rangle \in \#SAT \iff$

$$\sum_{b_1 \in \{0,1\}} \sum_{b_2 \in \{0,1\}} \sum_{b_n \in \{0,1\}} P_\Phi(b_1, \ldots, b_n) = K$$

$(K \leq 2^n$, so choose $p \in [2^n, 2^{n+1}))$

Sumcheck Protocol

Originally $p > K$

We wish to verify that

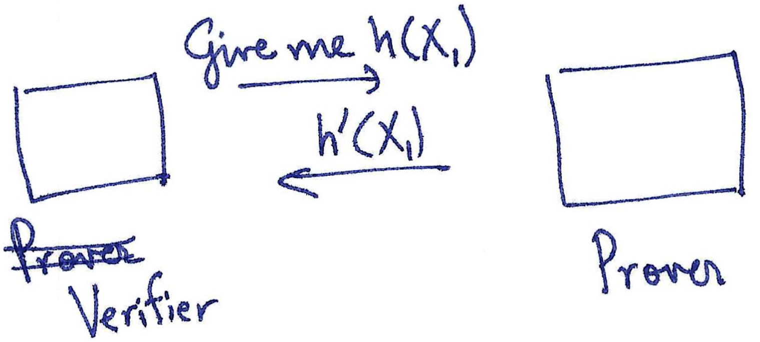
$$\underbrace{\sum_{b_1 \in \{0,1\}} \sum_{b_2 \in \{0,1\}} \cdots \sum_{b_n \in \{0,1\}} g(b_1, \ldots, b_n)}_{= h(0) + h(1)} \equiv K \pmod{p}$$

g has a poly(n) representation (for evaluation)
as a circuit efficient

For any setting of $b_2 \ldots b_n$, g is a univariate polynomial..

$$h(X_1) := \sum_{b_2 \in \{0,1\}} \cdots \sum_{b_n \in \{0,1\}} g(X_1, b_2, \ldots, b_n)$$

$h(0) + h(1)$ should be $K \pmod{p}$



Problem: $h'(0) + h'(1) \equiv K \pmod{p}$

but $h' \neq h$