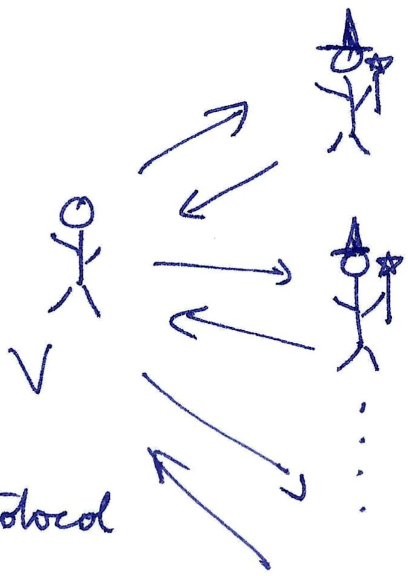
PCP Theorem & Hardness of Approximation

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

Multiple provers?

$\mathbb{MIP} \subseteq \mathbb{NEXP}$

Guess the whole protocol



Provers cannot communicate after Protocol Starts

$\mathbb{MIP} = \mathbb{NEXP}$

Thm [FRS94]: Two provers suffice for a $\mathbb{MIP}$ protocol.

Proof sketch: Given x, Verifier flips all random r bits of an $\mathbb{MIP}$ protocol, gives all bits to Prover 1 and asks "give me the full transcript of the $\mathbb{MIP}$ protocol".

Prover 1 gives $\mathcal{T} =$ | OP_1 | OP_k | OP_1 | OP_2 | OP_2 | OP_n | – | – | |

\# old provers = poly(n)

$\mathcal{T}$ = transcript. If $\mathcal{T}$ is incorrect, it is incorrect on messages of some prover OP_i.

Verifier simulates (MIP) protocol with Prover 2 playing role of OP_i, where i is chosen at random.

If transcript differs from responses of Prover 2, reject.
$\boxed{\text{If any transcript ends up in rejection, reject.}}$
With prob. $\geq \frac{1}{\text{poly}(n)}$, Verifier rejects.

(Repeat entire approach poly(n) times to boost probability.) ∎

We know $\mathbb{NP} \subsetneq \mathbb{NEXP}$.

Hence, $\mathbb{MIP}$ contains languages outside $\mathbb{NP}$.

We known deterministic $\mathbb{MIP} = \mathbb{NP}$.

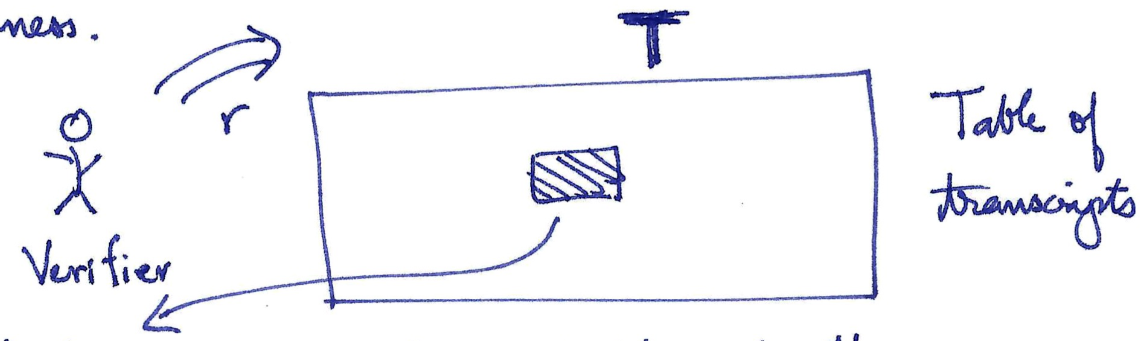
Hence, randomness (in interactive protocols) have provable power.

What was the 2-prover protocol doing?

Prover 1 was just a "lookup" using randomness.

Prover 2 was used only cross checking the response of Prover 1.

There is a notion of the "right transcript" for a given randomness.



Acc/rej based on $\mathbb{T}(r)$

$|r| = \text{poly}(n)$ ~~$|\mathbb{T}(r)|$~~

$|\mathbb{T}(r)| = \text{poly}(n)$

$L \in \mathbb{NEXP}$

$x \in L \Rightarrow \exists$ table $\mathbf{T}$ s.t. $\Pr_{r}[\mathbf{T}(r)$ leads to accept$] \cancel{\geq} = 1$

$x \notin L \Rightarrow \forall$ tables $\mathbf{T}$ $\Pr_{r}[\mathbf{T}(r)$ leads to accept$] \leq 1/3$

$\mathbf{T}$ certificate

What is the size of $\mathbf{T}$? ~~exp~~ 2^{n^c} $\rightarrow$ size of transcript

How many queries? $q \cancel{\leq} = poly(n)$

How much randomness? $r = poly(n)$

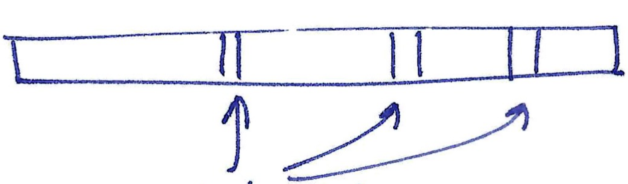
$\mathbb{PCP}[r, q] \longrightarrow$ Proof of length 2^r

Queries q

$$\mathbb{NEXP} = \bigcup_{c \geq 1} \mathbb{PCP}[n^c, n^c]$$

By a (non-trivial) scaling argument Completeness = 1

$$\mathbb{NP} = \bigcup_{c \geq 1} \mathbb{PCP}[\log^c n, \log^c n] \quad \text{Soundness} = \rho$$

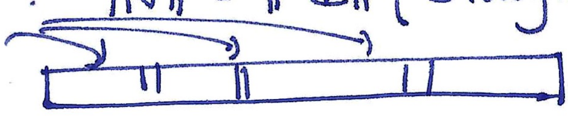


Proof size $= n^{poly(\log n)}$

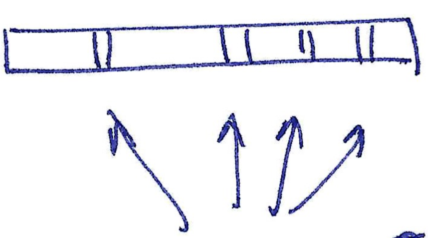
$poly(\log n)$

<u>PCP Theorem</u>: $\mathbb{NP} = \mathbb{PCP}[O(\log n), O(1)]$

$3 = q = O(1)$



size $= n^{O(1)}$



n^c

Verifier flip r random bits and makes q queries
Depending on these bits, acc/rej.

Total possible subsets of queries $\leq n^{cq} = n^{O(1)}$

For any subset Q of queries

Verifier runs some fn. on Q and x to acc/rej.

For Q, there are 2^q possible outcomes. as a truth table

$y_{i_1}, y_{i_2} \cdots y_{i_q}$ are the bits of the proof in Q

Acc, rej is some $f_Q(y_{i_1}, y_{i_2} - , y_{i_q})$ = CNF with a constant # of clauses

assume CNF

Like a clause

"Behavior of verifier" is encoded by ~~a CNF~~

set ~~A~~ $\{f_Q\}$ of CNFs.

If $x \in L$, all f_Q's are satisfiable (together)

$x \notin L$, at most p fraction of f_Q's are satisfiable.

$(L \in NP)$

Look at CNF $\boxed{\bigwedge_Q f_Q}$

Given $x \in L$, we can construct in polynomial time
a 3-CNF $f_x = \bigwedge_Q f_Q$ s.t.

If $x \in L \Rightarrow f_x$ is satisfiable

$x \notin L \Rightarrow$ At most ρ-fraction of clauses are satisfiable.

Thm: Getting a ~~$(1-\rho)$-approx. (for some~~ ρ-approx (for some $\rho \in [0,1]$) for # satisfiable clauses is NP-hard.

$\text{MAXSAT}(\phi) = $ Max # of satisfiable clauses.

There is a 7/8-approx for MAXSAT
(in poly time, we can satisfy $\geq \frac{7}{8}$ (OPT number of satisfied clauses))

[Håstad] 7/8-hardness for MAXSAT $\quad \forall \varepsilon > 0$
3-query PCP with soundness $= 7/8 + \varepsilon$

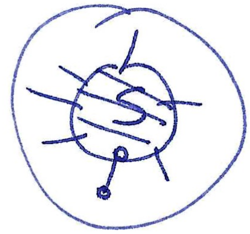
7/8-approx: Suppose we have 3SAT instance.
All clauses with exactly 3 literals.
$\geq$

$(x_i \vee \bar{x}_j \vee x_k)$

Random assignment satisfies an expected 7/8 of all clauses.

General 7/8 requires more machinery.

(VC) Vertex cover: S is a vertex cover if all edges have endpoint in S



Min Vertex Cover

Easy 2-approx in linear time (take ~~a maximal~~ all endpoints of a maximal matching)

Using ~~vertx~~ 3SAT to VC reduction and PCP theorem, we can prove some non-trivial hardness

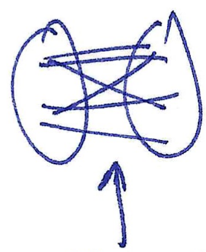
<u>Unique Games Conjecture</u> (UGC): Conjectures approx-hardness of ~~certain~~ specific NP problems. Has been used to prove optimal hardness for many problems.

<u>Hardness of MaxCut:</u>

$\frac{1}{2}$-approx : random cut

$\boxed{0.878}$-approx : Semidefinite Prog

Assuming UGC, this is optimal



Max. # edges crossing bipartition