<u>Thm</u> [Kabanets, Impagliazzo '03]: $PIT \in P \Rightarrow NEXP \not\subseteq P/poly$ or $VNP \neq VP$.

- We will skip this proof & instead focus on the implications of an efficient prg (& a converse!).

<u>Thm</u> [Agrawal '05]: Let f be an $s(n)$-prg against $\mathcal{C}$. Then, there is a multi-linear polynomial computable in poly($s(n)$)-time that is <u>not</u> in $\mathcal{C}$.

Assume $s(n) \leq 2^{n/2}$.

<u>Proof:</u>

- Consider $f(n) = (p_1(y), \dots, p_n(y))$ for a large enough n.
- Define $\underline{\ell(n)} := \lg s(n)$ & $\underline{m} := 2\ell \leq n$.

- The idea is to consider an <u>annihilating polynomial</u> $q(x_1, \dots, x_m)$ for $(p_1(y), \dots, p_m(y))$.

- In particular, $q(\overline{X}) = \sum_{S \subseteq [m]} c_S \cdot X_S$

s.t. $c_S \in \mathbb{F}$ & $q(p_1(y), \ldots, p_m(y)) = 0$.

- This sets up a linear system in the unknowns c_S:

  $\#\text{unknowns} = 2^m$,

  $\#\text{equations} \leq m \cdot s$

$\Rightarrow \exists$ a nontrivial solution ($\because 2^m > ms$).

- Moreover, the solution can be computed in time $\text{poly}(2^m) = \text{poly}(s(n))$.

- Since q vanishes on $f(n)$, we deduce that $q \notin \mathcal{C}$. (Also, q is m-var. & computable in $2^{O(m)}$-time) $\square$

— Is there a converse to this?

Does "VNP $\neq$ VP $\Rightarrow$ efficient prg for VP"?

- We can prove a weaker claim (both strengthening the premise & weakening the conclusion!).

<u>Thm</u> [KI'03, Agrawal-Vinay '08]: Let $\{q_m\}_{m\geq 1}$ be a multilinear polynomial family, computable in EXP, that is not computable by subexponential sized arithmetic circuits.

Then, there is an efficient variable reduction for VP circuits, from n to $O(\lg n)$ variables, that preserves nonzeroness.

Recall: deg & s are poly(n).

(This implies an $n^{O(\lg n)}$-prg for VP circuits.)

<u>Proof:</u>

- Let C be a nonzero circuit of size $s = s(n)$ computing a polynomial of deg $\leq s$ (wlog).
- We wish to reduce its variables,

preserving the nonzeroness.

We will utilize the q_m's, for "small" m, to feed into C.

- For this we need a set-family called Nisan-Wigderson designs.

Defn: Let $\ell > n > d$. A collection $\mathcal{I} = \{I_1, \ldots, I_m\}$ of n-size subsets of $[\ell]$ is an (ℓ, n, d)-design if: $|I_j \cap I_k| \leq d$, $\forall j \neq k \in [m]$.

Lemma [NW'94]: There is an algorithm that on input (ℓ, n, d), $(\ell > 10n^2/d)$, outputs an (ℓ, n, d)-design $\mathcal{I}$ having $m \geq 2^{d/10}$ subsets, in time $2^{O(\ell)}$.

Pf:

- A greedy approach works.
- Details skipped.

$\square$

- Say, C has n variables $z_1, \dots, z_n$.
- Let $\mathcal{I} = \{S_1, \dots, S_n\}$ be a $(c\lg n, d\lg n, 10\lg n)$-design, for suitable constants $c > d > 10$.

Note that by the previous Lemma, $\mathcal{I}$ can be constructed in poly(n)-time.

- Now we map $\{z_1, \dots, z_n\}$ to $\bar{x} := \{x_1, \dots, x_{c\lg n}\}$ as follows:

$$z_i \mapsto p_i := q_{d\lg n}(\bar{x}_{S_i})$$

where $\bar{x}_{S_i}$ is the $(d\lg n)$-tuple given by the indices in S_i.

<u>Claim</u>: $C(p_1, \dots, p_n) \neq 0$.

Pf: • Suppose not.

- As $C(\bar{z}) \neq 0$ but $C(\bar{p}) = 0$, there is a $j \in [n]$ s.t. $C(p_1, \dots, p_j, z_{j+1}, \dots, z_n) = 0$ but $C(p_1, \dots, p_{j-1}, z_j, \dots, z_n) \neq 0$.
- $\Rightarrow (z_j - p_j) \mid C(p_1, \dots, p_{j-1}, z_j, \dots, z_n)$.

- Now we can fix $z_{j+1}, \ldots, z_n$ & the x_i's that do not occur in p_j to random values from the field.
- This reduces us to the case:

$$(z_j - p_j) \mid C'\left(p_1'(\overline{x}_{S_1 \cap S_j}), \ldots, p_{j-1}'(\overline{x}_{S_{j-1} \cap S_j}), z_j\right) \neq 0.$$

- Note that $|S_k \cap S_j| \leq 10 \lg n$, for $k \neq j$.

  $\Rightarrow$ The above circuit $C'(p_1', \ldots, z_j)$ has size $< s + n^{11}$.

  (As p_1' etc. can be written as a sum of $2^{10 \lg n}$ monomials.)

- We could now invoke Kaltofen ('89) VP circuit factorization algorithm (in the blackbox setting!).

  $\Rightarrow$ p_j has a VP circuit of size s^e, where e is a constant independent of d & c.

- Since $p_j = q_{d\lg n}(\bar{x}_{S_j})$ was assumed

complexity $2^{\Omega(d\lg n)}$ → to be a "hard" polynomial, we can deduce a contradiction by taking d suitably larger than e.

$\Rightarrow C(p_1, \ldots, p_n) \neq 0.$

□

- Note that $C(\bar{p})$ is $(c\lg n)$-variate & $\deg = O(s\lg n)$.

- Thus, $C(\bar{p})$ has sparsity at most $(s\lg n)^{O(\lg n)} = n^{O(\lg n)}$.

- Finally, one needs to design an efficient prg for sparse polynomials.

(When the arity m is small then one can simply take $[0 \ldots \deg]^m$ as a hitting-set.)