Polynomial identity testing (PIT)

- PIT is the following algorithmic problem:

  Given an arithmetic circuit $C(\bar{x})$, over a ring R, test whether C is identically zero.

  (We want an algorithm that runs in time polynomial in size(C).)

- We will focus on the case of R being a field $\mathbb{F} = \mathbb{Q}$ or $\mathbb{F}_q$.

Theorem [Schwartz, Zippel et al] $PIT \in coRP$.

Proof:

- Let $C(\bar{x})$ be the given circuit of size s, over $\mathbb{F} = \mathbb{F}_q$.
- $\Rightarrow \deg C < s^s$.
- We could assume $|\mathbb{F}| > 2 \cdot s^s$, otherwise we can use an appropriate field extension.

(Fast constructions are known due to [Adleman, Lenstra '86])

- The algorithm is simply a random evaluation:

  0) Pick an $S \subseteq \mathbb{F}$ of size $2 \cdot s^s$.

  1) Pick a <u>random</u> $(a_1, \ldots, a_n) \in S^n$.

  2) If $C(\bar{a}) = 0$ then OUTPUT <u>Zero</u>

     else " <u>nonZero</u>.

- It has been proved before (in an Assignment) that: if C is a nonzero polynomial then

$$\Pr_{\bar{a} \in S^n}[C(\bar{a}) \neq 0] > 1 - \frac{\deg C}{2s^s} > \frac{1}{2}.$$

- Clearly, $C(\bar{a})$ can be computed in time $\mathrm{poly}(s, \lg|\mathbb{F}|)$.

- In the case when $\mathbb{F} = \mathbb{Q}$, $C(\bar{a})$ may be doubly-exp. large!

In that case, we pick a random prime p & evaluate $C(\bar{a}) \bmod p$.

(Exercise: compute the error probability.)

no mistake on identities (coRP)

• Thus, in all cases PIT has a randomized poly-time algorithm. $\square$

- Note that in the above algorithm the specifics of the circuit C were not used. (Only the size bound was needed.)

- Such an algorithm is called a blackbox identity test.

(One can only evaluate a blackbox.)

Definition: For a family $\mathcal{C}$ of circuits (n-variate) of size s, a hitting-set $\mathcal{H} \subseteq \mathbb{F}^n$ is a poly(s)-sized set of points such that: If $C \in \mathcal{C}$ is nonzero then $\exists\, \bar{\alpha} \in \mathcal{H}$, $C(\bar{\alpha}) \neq 0$.

Or, $\mathcal{H}$ hits $\mathcal{C}$.

Lemma: Let $S \subseteq \mathbb{F}_q$ be of size s^{3s} & $\mathcal{C}$ be the family of size-s circuits, n-variate, over $\mathbb{F}_q$.
Then, a random $\bar{a} \in S^n$ hits $\mathcal{C}$.

Proof:

- $$\Pr_{\bar{a} \in S^n} [\exists\, 0 \neq C \in \mathcal{C},\ C(\bar{a}) = 0]$$
$$\leq |\mathcal{C}| \cdot \frac{s^s}{|S|} < s^s \cdot \frac{s^s}{s^{3s}} = s^{-s}.$$
$$\Rightarrow \Pr_{\bar{a}} [\forall\, 0 \neq C \in \mathcal{C},\ C(\bar{a}) \neq 0] > 1 - s^{-s}.$$
$\square$

OPEN (Derandomization): Can a hitting-set be computed in det. poly-time ?

- Given $\mathcal{H}$, by interpolation, we can find polynomials $(p_1(y), \ldots, p_n(y)) =: \bar{p}(y)$ such that their first few values, on fixing y, give us the points in $\mathcal{H}$.

Also, $\deg p_i \leq |\mathcal{H}|$.

- This motivates us to define arithmetic analogs of prgs (pseudorandom generators).

<u>Defn</u>: $\{\underbrace{(p_1^n(y), \ldots, p_n^n(y))}_{=: f(n)} \mid n \in \mathbb{N}\} =: f$ is called an $s(n)$-prg against $\mathcal{C}$, if

- each $p_j^n(y)$ has deg $\leq s(n)$ & is computable in time poly$(s(n))$,
- for any nonzero $C \in \mathcal{C}$ on n-variables, $C(p_1^n(y), \ldots, p_n^n(y)) \neq 0$.

depending on $\mathcal{C}$ one might want to go $\rightarrow$ mod $g(y)$.

Derandomization Qn: Do efficient prgs exist?

- Apart from being a fundamental qn., this is also related to proving lower bounds (close to VP $\neq$ VNP).

- A PIT algorithm would imply some lower bound:

<u>Thm</u> [Kabanets, Impagliazzo '03]: $PIT \in P \implies$ $NEXP \not\subseteq P/poly$ 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 $s(n) \le 2^{n/2}$.

<u>Proof:</u>

- Consider $f(n) = (p_1(y), \ldots, p_n(y))$ for a large enough n.
- Define $\underline{\ell(n)} := \lg s(n)$ & $\underline{m} := 2\ell \le n$.
- The idea is to consider an <u>annihilating polynomial</u> $q(X_1, \ldots, X_m)$ for $(p_1(y), \ldots, p_m(y))$.