Polynomial Identity Testing

Amit Sinhababu
amitks@cmi.ac.in

$$X^2 - Y^2 = (X - Y)(X + Y).$$
$$= X^2 - \cancel{YX} + \cancel{XY} - Y^2$$
$$= X^2 - Y^2.$$

$$p(x_1, \ldots, x_n) = q(x_1, \ldots, x_n)$$

$$p - q = 0$$

How the polynomial is given.

Univariate Polynomials.

$$a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n.$$

Give all the monomials and its corresponding coefficients.

$a_n x^n$ — n multiplications

$a_{n-1} x^{n-1}$ — $n-1$ multiplications

$\vdots$

$a_1 x$ — 1 〃

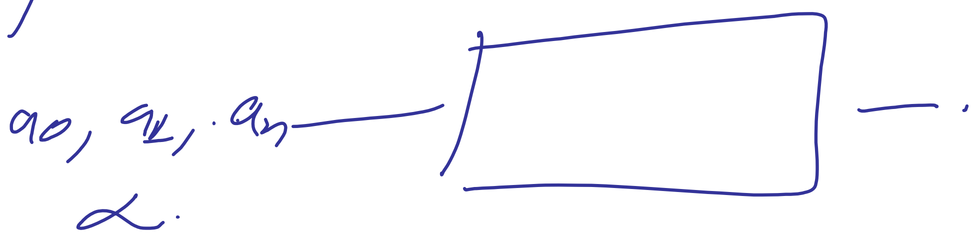
$$1 + \cdots + n = \frac{n(n+1)}{2} \sim n^2.$$

Horner' rule.

$$a_0 + x\left(a_1 + x\left(a_2 + \cdots + x(a_n)\right)\right)$$

n additions n multiplications.

This is optimal.



x^n — Repeated squaring / binary exponentiation.

$x \quad x^2 \quad x^{2^2} \quad x^{2^3} \quad \cdots \quad \approx \log n$ multiplications.

$$1 + x + x^2 + \cdots + x^n = \frac{x^{n+1} - 1}{x - 1}$$

How many additions & multiplications

Exercise. $O(\log n)$ additions & multiplications.

Not every univariate polynomial of degree d can be computed using ~~the~~ $\text{poly}(\log d)$ many $+$ and $\times$.

$$(x+1)(x+2)\cdots(x+d)$$

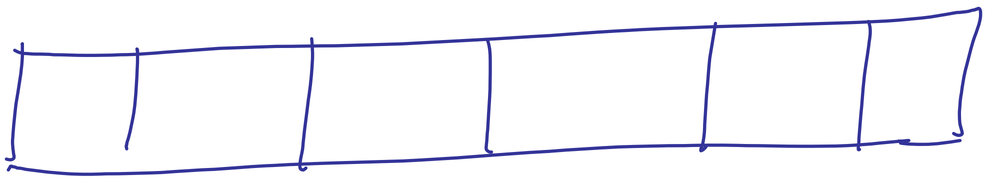
Conjecture: We need $\Omega(d)$ many $+$ and $\times$.

$d!$ in $\text{poly}(\log d)$ is not known.

Ex 2. Suppose you can compute $n!$ in poly$(\log n)$ many arithmetic operations, then integer factoring can be solved efficiently.

Multivariate Polynomials.

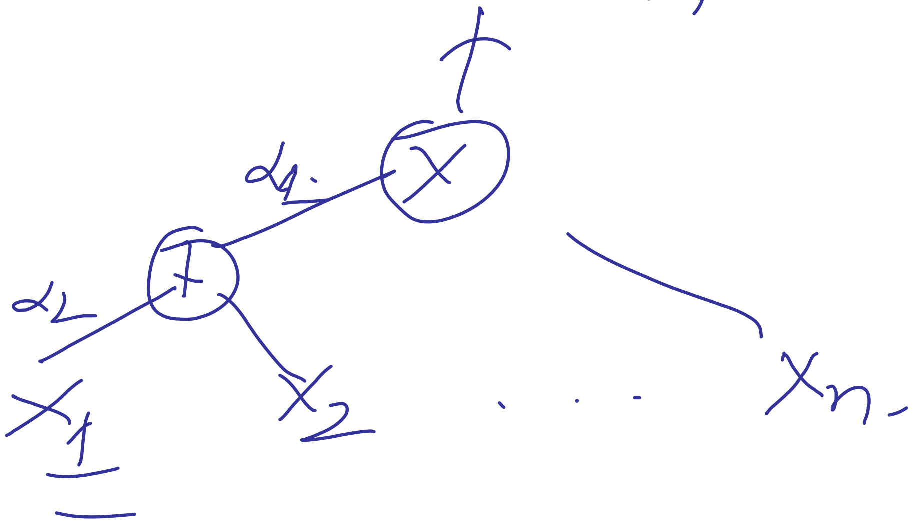
n-variate degree d polynomial can have $\binom{n+d}{d}$ monomials.



$\le \min\{n^d, d^n\}$.

Trees.

$p(x_1, \dots, x_n)$.



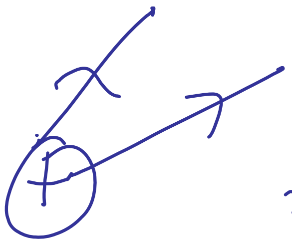
Arithmetic Formulas.

$(1+x_1)(1+x_2)\dots(1+x_n)$

has 2^n many monomials.

but has $O(n)$ size arithmetic formula.

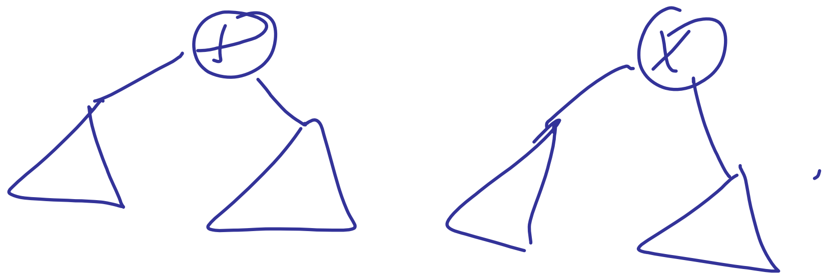
Directed Acyclic Graphs — Circuits.



reuse of computation is allowed.

Repeated Squaring.

Formulas & Circuits as data structures for representing polynomials.



- Evaluation can be done efficiently.

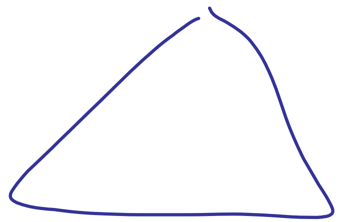
-

Polynomial Identity Testing.

Given an arithmetic formula or circuit, test if the polynomial computed by the formula/circuit is identically zero.

Using randomness it can be solved efficiently.

UNSAT



Randomized Algorithm for PIT.

Input: circuit $C(x_1, \ldots x_n)$

Have a finite set $S \subseteq \mathbb{Q}$. (Rational numbers).

Pick $\alpha_1, \alpha_2, \ldots, \alpha_n$ uniformly and independently at random from S.

Test if $C(\alpha_1, \ldots, \alpha_n) = 0$

if nonzero $\Longrightarrow$ You output "nonzero".

0 — output identically zero.

Schwartz - Zippel lemma.

over a field $\mathbb{F}$.

Any nonzero polynomial $f(x)$ of degree d.
has at most d roots.

Take a set S.

Take random α from S.

$$\Pr(f(\alpha) = 0) \leq \frac{d}{|S|}.$$

Factor theorem $f(\alpha)=0 \Rightarrow x-\alpha \mid f(x)$.

Given a nonzero polynomial $f(x_1, \dots x_n)$

Take any finite set $S \subseteq \mathbb{F}$. Pick $\alpha_1, \dots, \alpha_n$ uniformly

& independently at random from S.

$$\Pr.\left(f(\alpha_1, \dots \alpha_n)=0\right) \le \frac{d}{|S|} \le$$

$$|S| = 2d$$

$$f(x_1, x_2, \dots, x_n) = \underline{\underline{f_r(x_1, \dots, x_{n-1})}} \cdot X_n^r + \dots + f_1()\, x_n$$
$$= \qquad + f_0(x_1, \dots, x_{n-1}).$$

has degree d.

f_r has degree $d - r$.

$\underline{f_r(\alpha_1, \dots, \alpha_{n-1}) \neq 0.}$ 1

$\underline{\underline{f_r(\alpha_1, \dots, \alpha_{n-1}) = 0}}$ $\leq \frac{d-r}{|S|}$

$f(\alpha_1, \dots, \alpha_n) = 0$ $\leq \frac{r}{|S|}$.

0

1

0

$\leq \frac{d}{|S|}$.

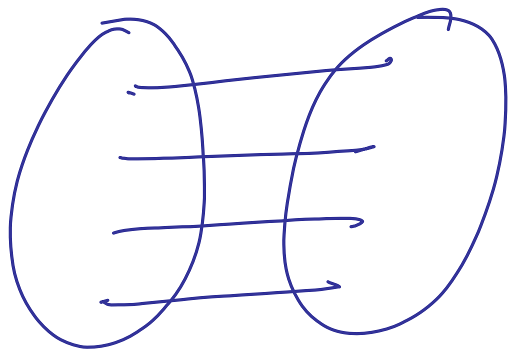
$\leq \frac{r + d - r}{|S|}$.

Can you reduce multivariate PIT to univariate PIT?

$$f(x_1, \ldots, x_n) \sim \hat{f}(x).$$

— $\hat{f}(x)$ is nonzero iff $f(x_1, \ldots, x_n)$ is nonzero.

$x_1 \to$



d-ary expansion.

$x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n}$

$(\alpha_1, \alpha_2, \ldots \alpha_n)$

d-ary

$\boxed{\alpha_1 \alpha_2 \ldots \alpha_n}$ expansion.

$$X_1 \to x$$

$$X_2 \to x^{d}$$

$$X_3 \to x^{d^2}$$

$$\vdots$$

Kronecker Substitution.

Univariate polynomial of exponential degree.

Pick $\omega_1, \omega_2, \ldots, \omega_n \in S \subseteq \mathbb{Q}$ at random.

$$f\left(t^{\omega_1}, t^{\omega_2}, \ldots, t^{\omega_n}\right) \neq 0.$$

Isolation lemma due to Mulmuley, Vazirani, Vazirani.

Application of PIT.

– (1) string comparison / string equality testing.

$$a_0 \ldots\ldots a_n = b_0 \ldots\ldots b_n.$$

$$a_0 + a_1 x + \cdots + a_n x^n = b_0 + b_1 x + \cdots + b_n x^n.$$

Fingerprinting. string hashing.

Where to use and how not to use polynomial string hashing.

— iii) <u>Primality Testing</u> Primes $\in P$.

Given n, test if $\underline{n}$ is prime.

Try out $2, \ldots, \sqrt{n}$, —

Agrawal - Kayal - Saxena 2002.

Primes $\in P$. $\sim (\log n)^6$

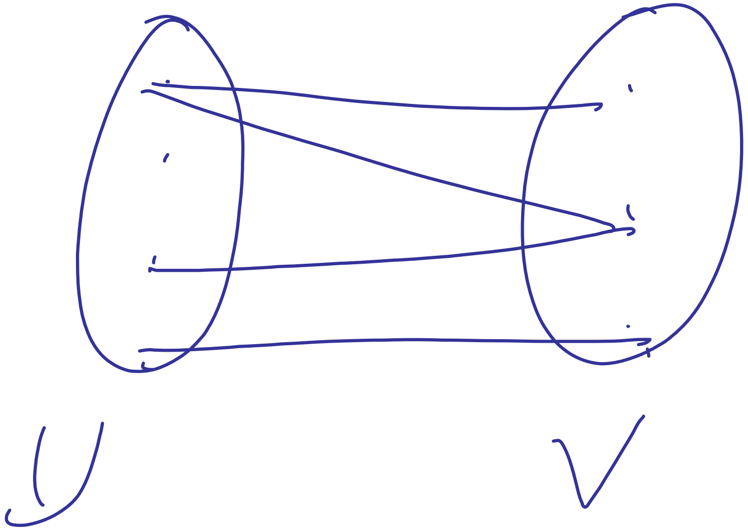
n is prime iff.

$$(x+a)^n \equiv x^n + a \mod n.$$

How to reduce degree? Go modulo low-degree univariates.

Perfect Matching in Bipartite Graphs.

$|U| = |V| = n.$



Matching is a set of edges that do not have any vertex in common.

Given a bipartite graph, is there a matching?

$$M \quad \begin{array}{c|cccc|} & v_1 & \cdots & v_j & v_n \\ \hline u_1 & & & & \\ u_2 & & & & \\ u_i & & & & \\ \vdots & & & & \\ u_n & & & & \end{array}$$

If edge (u_i, v_j) is present.

then put X_{ij}

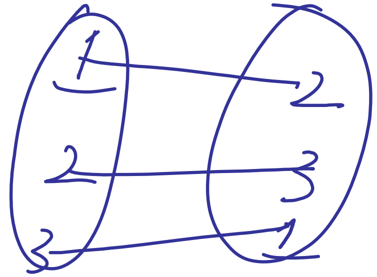
o/w put 0.

There is a perfect matching in the given graph iff. $\text{Det } M \neq 0$.

Matching via Determinant.

$1 \ldots\ldots n$

$\sigma \quad \sigma(1) \qquad \sigma(n).$



$\begin{pmatrix} a & b \\ c & d \end{pmatrix}$

Definitions of Determinant.

$$\text{Determinant}\ A_{n\times n} = \sum_{\sigma \in S_n} \underline{\text{sign}(\sigma)} \prod_{i=1}^{n} \underbrace{A_{i,\sigma(i)}}.$$

$i < j$ but $\sigma(i) > \sigma(j)$.

P

Open Question.

$\in$

NC. $\leftarrow$ class for which we have efficient parallel algorithms.

Bipartite Matching $\in$ NC ?

" Isolating a matching when your coins go missing. "

$\in$ Q~~N~~ quasi-NC.

$n^{\log n}$.

Fenner, Gurjar, Thierauf.