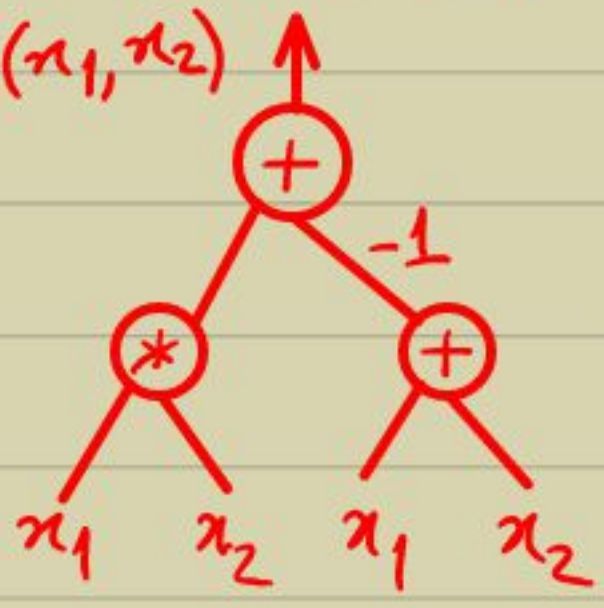
- PIT: Given an arithmetic circuit $C(x_1, \ldots, x_n)$ over $\mathbb{F}$, test if $C = 0$?

- Arithmetic circuit: A rooted tree with inputs as leaves, output as root, $+, *$ as internal nodes, & constants on edges.



- Size of a circuit includes the #edges, #gates, representation-size of the constants.

- Circuit $C(\bar{x})$ can capture very large polynomials, in small size!

- Thus, PIT is nontrivial.
  We want a poly-time algorithm for PIT over fields $\mathbb{Q}, \mathbb{F}_2$.

Theorem: PIT $\in$ BPP.

Pf:

- Let $C(\bar{x})$ be the given circuit, over $\mathbb{F}$, of size s.
- Note that the total-deg of C is $< s^2$.
  (Each multiplication layer increases the degree by a multiple of at most s.)
- We could assume $|\mathbb{F}| > 2 \cdot s^2$, otherwise we use an appropriate field extension as $\mathbb{F}$.
- The algorithm is simply a random evaluation
  (0) Pick a subset $S \subseteq \mathbb{F}$, $|S| = 2 \cdot s^2$.
  (1) Pick a <u>random</u> $(a_1, \ldots, a_n) \in S^n$.
  (2) If $C(a_1, \ldots, a_n) = 0$ then OUTPUT <u>zero</u>, else OUTPUT <u>nonzero</u>.

- Correctness:
  If $C(\bar{x}) = 0$ then $\text{Prob}[\text{correct output}] = 1$.
  Else $\text{Prob}_{\bar{a}}[\text{correct output}] > 1 - \frac{s^2}{2 \cdot s^2} = \frac{1}{2}$
  is given by the following lemma. $\square$

Lemma (DeMillo & Lipton '78, Zippel '79, Schwartz '80):
Let $P \in \mathbb{F}[\bar{x}]$ be a polynomial of degree $d \geq 0$.
Let $S \subseteq \mathbb{F}$ be a finite subset. Then,

$$\Pr_{\bar{a} \in S^n} [P(\bar{a}) = 0] \leq d/|S| .$$

Proof: • For $n=1$, it follows from the fact that $P(x_1)$ can have at most d roots in $\mathbb{F}$. ← Base case

• Assume it to be true for $(n-1)$ variables. ← Induction hypothesis

• Write $P = \sum_{i=0}^{d} x_n^i \cdot P_i(x_1, \ldots, x_{n-1})$.

• As $P \neq 0$, let i_0 be the largest i s.t. $P_i \neq 0$.

$$\Rightarrow \Pr_{\bar{a}} [P(\bar{a}) = 0] = \Pr_{\bar{a}} [P_{i_0}(\bar{a}) = 0] \cdot \Pr_{\bar{a}} [P(\bar{a}) = 0 \mid P_{i_0}(\bar{a}) = 0]$$

($a_1, \ldots, a_{n-1}$)

$$+ \Pr_{\bar{a}} [P_{i_0}(\bar{a}) \neq 0] \cdot \Pr_{\bar{a}} [P(\bar{a}) = 0 \mid P_{i_0}(\bar{a}) \neq 0]$$

$$\leq \Pr [P_{i_0}(\bar{a}) = 0] + \Pr [P(\bar{a}) = 0 \mid P_{i_0}(\bar{a}) \neq 0]$$

(induction; univariate)

$$\leq \frac{d - i_0}{|S|} + \frac{i_0}{|S|} = \frac{d}{|S|} .$$

□

- For the small field case show that:

Exercise: A field $G \supset \mathbb{F}$ of size $> 2 \cdot s^2$ can be constructed in randomized poly(s)-time.

The Circuit Model

- An arithmetic circuit (over $\mathbb{F}$) has $+, *$ gates and field elements.

- A boolean circuit has AND, OR, NOT gates and $\{0,1\}$. (0=false, 1=true)

- An arithmetic circuit outputs a polynomial while a boolean circuit outputs a boolean formula.

- We can use these as a model of computation instead of TMs.

Defn: • A problem $L \subseteq \{0,1\}^*$ is said to be solved by a boolean circuit family $\{C_n(x_1,..,x_n) \mid n \geq 1\}$ if $\forall n, \forall x \in \{0,1\}^n$, $C_n(x) = 1$ iff $x \in L$.

• The computational resources now are: size(C_n), depth(C_n) & fanin / fanout.

Proposition: (1) Any TM can be turned into a circuit family (vice versa ?)

(2) Boolean circuits are inspired from "electronics" & capture parallel computation:

size (C) circuit ⇒ the space requirement of the parallel algorithm.

depth (C) circuit ⇒ the time taken by the parallel algorithm.

(3) Two n-bit integers can be added by a size-poly(n), constant-depth boolean circuit.

(What about multiplication ?)

Circuit Complexity Classes

- Analogous to $\mathrm{Dtime}(T(n))$ we have,
  $\underline{\mathrm{Size}(s(n))} := \{L \subseteq \{0,1\}^* \mid \exists\, O(s(n))\text{-sized boolean circuits } \{C_n\} \text{ solving } L\}$.

$$\underline{\mathrm{P/poly}} := \bigcup_{c \in \mathbb{R}_{>0}} \mathrm{Size}(n^c).$$

- Analogously for arithmetic circuits we can define:
  $\underline{\mathrm{Alg\text{-}Size}(s(n))} = \{ \{f_n\}_n \mid n \in \mathbb{N}, f_n : \mathbb{F}^n \to \mathbb{F},$
  $\exists\, O(s(n))\text{-sized arithmetic circuit } C_n$
  computing f_n $\}$

$$\underline{\mathrm{Alg\text{-}P/poly}} := \bigcup_{c \in \mathbb{R}_{>0}} \mathrm{Alg\text{-}Size}(n^c)$$

Proposition: $\mathrm{P} \subsetneq \mathrm{P/poly}$. But, $\mathrm{AlgP/poly}$ $(\mathbb{F} = \mathbb{F}_2)$ is incomparable.

P/poly is for fns. $f : \{0,1\}^n \to \{0,1\}$. AlgP/poly for $f : \mathbb{F}^n \to \mathbb{F}$.