CSE 575: Advanced Cryptography
Fall 2024
Lecture 20

Paul Grubbs

paulgrub@umich.edu

Beyster 4709

Agenda for this lecture

- Announcements
- A first IOP: polynomial identity testing
- Arithmetic circuits
- Plonk’s constraint system
- Arithmetizing constraints

Agenda for this lecture

- Announcements
- A first IOP: polynomial identity testing
- Arithmetic circuits
- Plonk’s constraint system
- Arithmetizing constraints

Announcements

- HW4 will be assigned this week
- Rest of class: no typeset lecture notes.
  - Can get extra credit for typesetting

Agenda for this lecture

- Announcements
- A first IOP: polynomial identity testing
- Arithmetic circuits
- Plonk's constraint system
- Arithmetizing constraints

Polynomial identity testing

- Protocol w/ $\mathbb{P}$, $\mathbb{V}$. Interactive
- $\mathbb{P}$ has two msg. types: normal and "oracle"
  oracle message: label/handle to $\mathbb{P}$ value
- $\mathbb{V}$ msgs: normal or "query oracle i on x"

$\Pi_{IOP} = \langle \mathbb{P}(x, w), \mathbb{V}(x) \rangle$ for relation R

Polynomial identity testing

$R = ((\mathbb{F}, d), (q, r))$: q, r have degree $\leq d$ and $q \equiv r$ over $\mathbb{F}$

$P((\mathbb{F}, d), (P_1, P_2))$ $\quad$ $V((\mathbb{F}, d))$

$\xrightarrow{\ell_1, \ell_2}$

$c \leftarrow \mathbb{F}$

$\xleftarrow{\text{query } \ell_1, \ell_2 \text{ at } c}$

$r_1 = P_1(c)$

$r_2 = P_2(c)$

$\xrightarrow{r_1, r_2}$

Ret_2 $\quad$ $r_1 \stackrel{?}{=} r_2$

Completeness: if $P_1 = P_2$

Soundness? If $P_1 \neq P_2$ $\Pr[\text{out}_V\langle P, V\rangle = 1] < \frac{d}{|\mathbb{F}|}$

Polynomial identity testing

If $P_1 \neq P_2$,

$$\Pr[\text{out}_V\langle P, V\rangle = 1] \leq \frac{d}{|\mathbb{F}|}$$

$P_1 \neq P_2$

$P_1 - P_2$ is degree-d poly over $\mathbb{F}$

$P_1(c) = P_2(c) \Rightarrow P_1 - P_2(c) = 0$

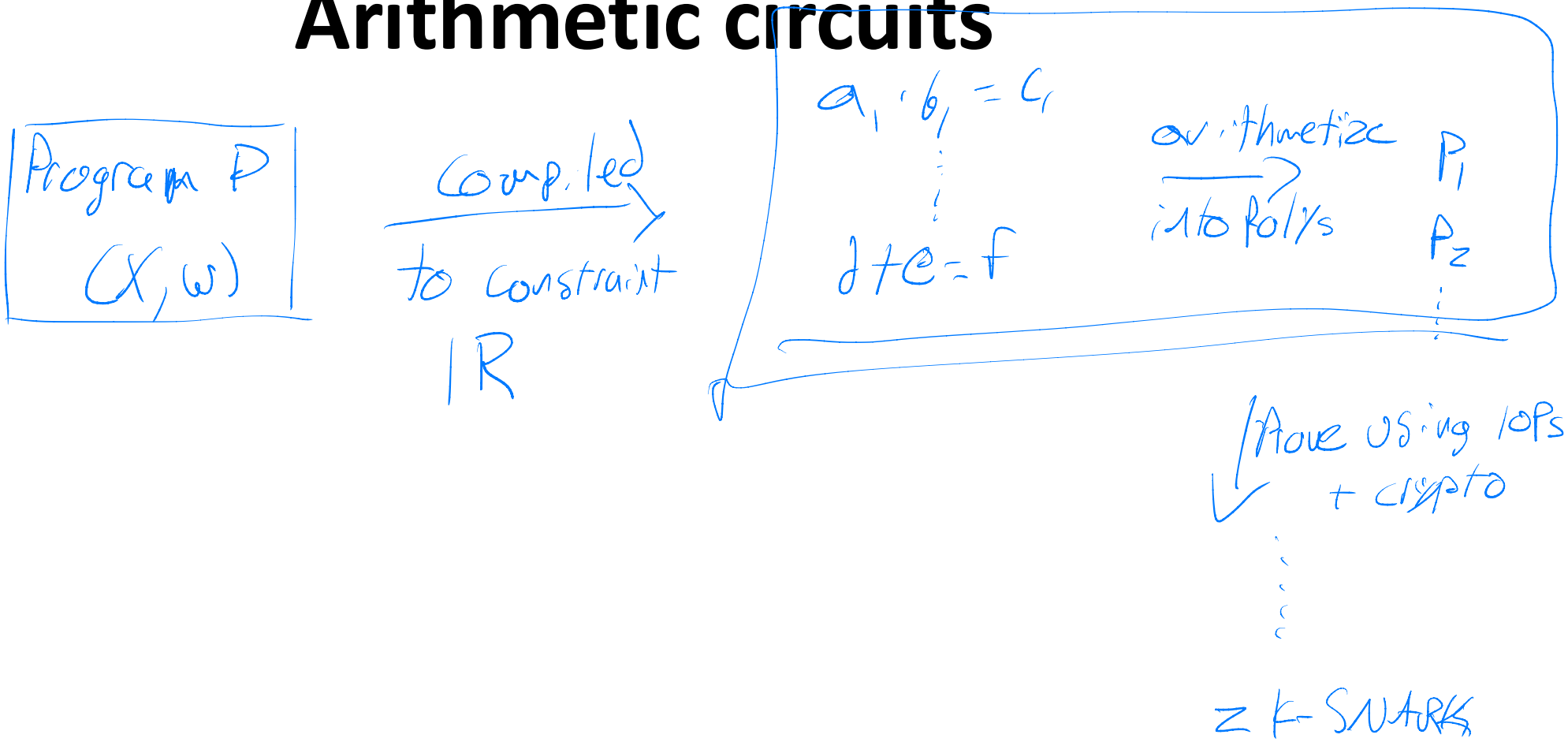
c must be root of $P_1 - P_2$. Happens w.p. $\leq \frac{d}{|\mathbb{F}|}$ by S-Z / FTA

$\square$

Agenda for this lecture

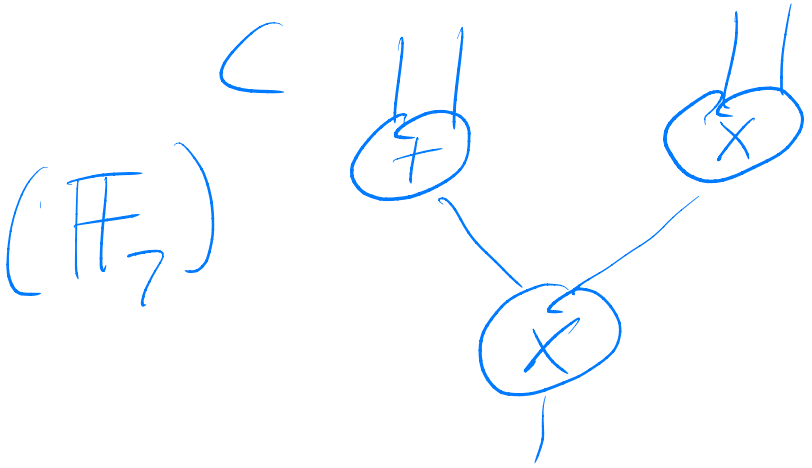
- Announcements
- A first IOP: polynomial identity testing
- Arithmetic circuits
- Plonk's constraint system
- Arithmetizing constraints

Arithmetic circuits



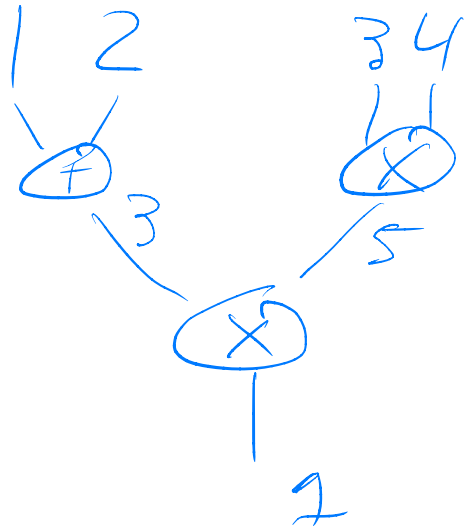
Arithmetic circuits

Let $\mathbb{F}$ be finite field. An AC over $\mathbb{F}$ is DAG where nodes $+/\times$ and edges are "wires"



"Evaluate"

$C(1,2,3,4)$



Agenda for this lecture

- Announcements
- A first IOP: polynomial identity testing
- Arithmetic circuits
- Plonk’s constraint system
- Arithmetizing constraints

Plonkish constraints

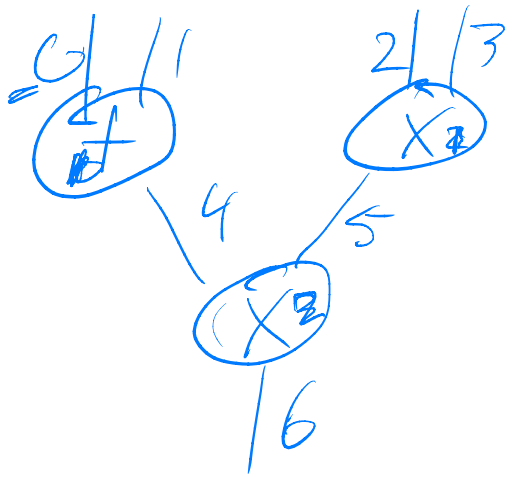
- $\mathcal{C} = (\mathcal{V}, \$)$ n gates, m wires w/ index $[m]$
  - $\mathcal{V}$ is $\vec{a}, \vec{b}, \vec{c} \in [m]^n$. wiring info:
    - $\vec{a}[i]$ is left input wire of gate i
    - $\vec{b}[i]$ is right " " " "
    - $\vec{c}[i]$ is output
  - $\$ \in \mathbb{F}^n$ s.t.
    - $\$[i] = 1$ if gate i is MUL
    - $\$[i] = 0$ if " " ADD

Say $X \in \mathbb{F}^m$ satisfies $\mathcal{C}$ if $\forall i \in [n]$,

$$\$[i](X[\vec{a}[i]] \cdot X[\vec{b}[i]]) + (1 - \$[i])(X[\vec{a}[i]] + X[\vec{b}[i]]) - X[\vec{c}[i]] = 0$$

Plonkish constraints

ACs to

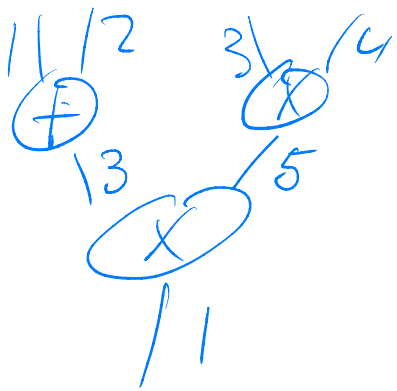


$m = 7, \quad n = 3$

$\vec{a} = [0 \quad 2 \quad 4]$

$\vec{b} = [1 \quad 3 \quad 5]$

$\vec{c} = [4 \quad 5 \quad 6]$



$\$ = [0 \quad 1 \quad 1]$

$i = 0$

$X = [1 \quad 2 \quad 3 \quad 4 \quad 3 \quad 5 \quad 1]$

$\$[0] \cdot (X[a[0]] \cdot X[b[0]]) + (1 - \$)(X[a[0]] + X[b[0]]) - X[c[0]] = 0$

Plonkish constraints

1	2	3
3	4	5
3	5	1

Claim ✓ is correct (gate eval -by-gate) of some AC (or constraint system)

How to check?

- Gate-by-gate
- Copy constraints

Plonkish constraints

wiring permutation: check $n \times 3$ table is invariant under σ.

$$T[\sigma[i]] = T[i]$$

Checks copy constraints

How to compute?

- $\vec{w} = \vec{a} \| \vec{b} \| \vec{c}$
- For each $i \in [m]$, compute indices of $\vec{w}$ where $\vec{w}[j] = i$
  Let B_i be the indices
- Define σ: B_i's are cycles

Agenda for this lecture

- Announcements
- A first IOP: polynomial identity testing
- Arithmetic circuits
- Plonk’s constraint system
- Arithmetizing constraints

Arithmetizing Plonkish constraints