

Circuit Complexity, Ideals, and Proof Systems: Connections & Recent Results

FOCS '21 [Workshop: Reflections on Propositional
Proofs in Algorithms & Complexity](#)

Feb 2022

Joshua A. Grochow



Main Goal

Get people excited about strengthening the ties between

Algebraic
Proof
Complexity

Algebraic
Circuit
Complexity

Main Goal

Get people excited about strengthening the ties between

Algebraic
Proof
Complexity

Algebraic
Circuit
Complexity



Complexity in Ideals

Complexity in Ideals of Polynomials

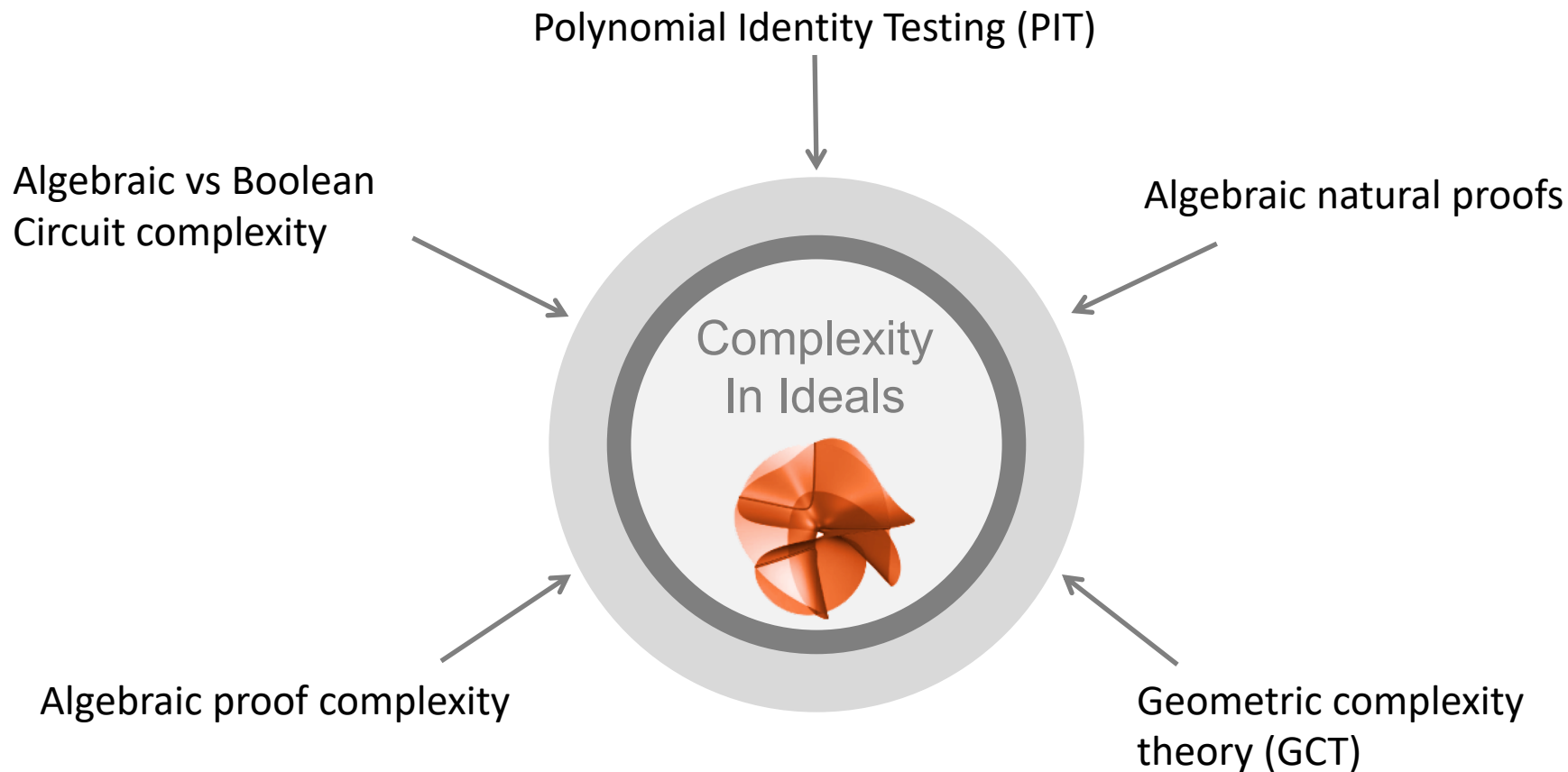
Family of rings $R_n = F[x_1, \dots, x_{v(n)}]$

Family of ideals $I_n \subseteq R_n$, denoted $I = (I_n)_{n=1,2,3,\dots}$

Main motif: Given a family of ideals* I_n , what can be said about the complexity of polynomials $(f_n) \in (I_n)$?

*Or sometimes their cosets $p_n + I_n$

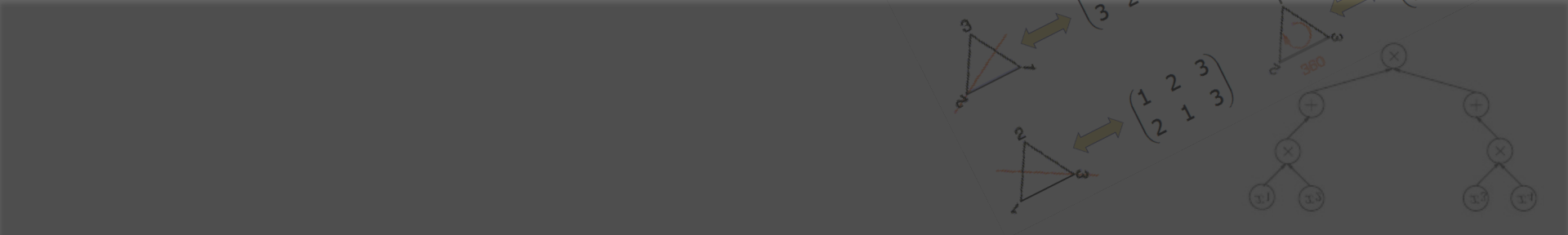
How did we get here?



Rings & Ideals

Definitions:

- A **ring** R has $(+, \times, 0, 1)$ satisfying usual axioms (e.g. distributivity). **Examples:** $\mathbb{Z}, \mathbb{Q}, \mathbb{R}, \mathbb{C}, F_p, F[x_1, \dots, x_n]$
- $I \subseteq R$ is an **ideal** if (1) I is closed under addition & negation
(2) I is closed under multiplication by arbitrary elements of R :
 $(\forall i \in I, r \in R)[ri \in I]$.
- I is **generated** by f_1, \dots, f_k if it is the smallest ideal containing these. In this case we write $I = \langle f_1, \dots, f_k \rangle$. It follows that $I = \{\sum_{i=1}^k f_i g_i \mid g_i \in R\}$. **Examples:** $\langle p \rangle \subseteq \mathbb{Z}, \langle x^2, y \rangle \subseteq F[x, y]$
- A **coset** of I is $r + I = \{r + i : i \in I\}$ for some $r \in R$. The cosets form the quotient ring R/I .

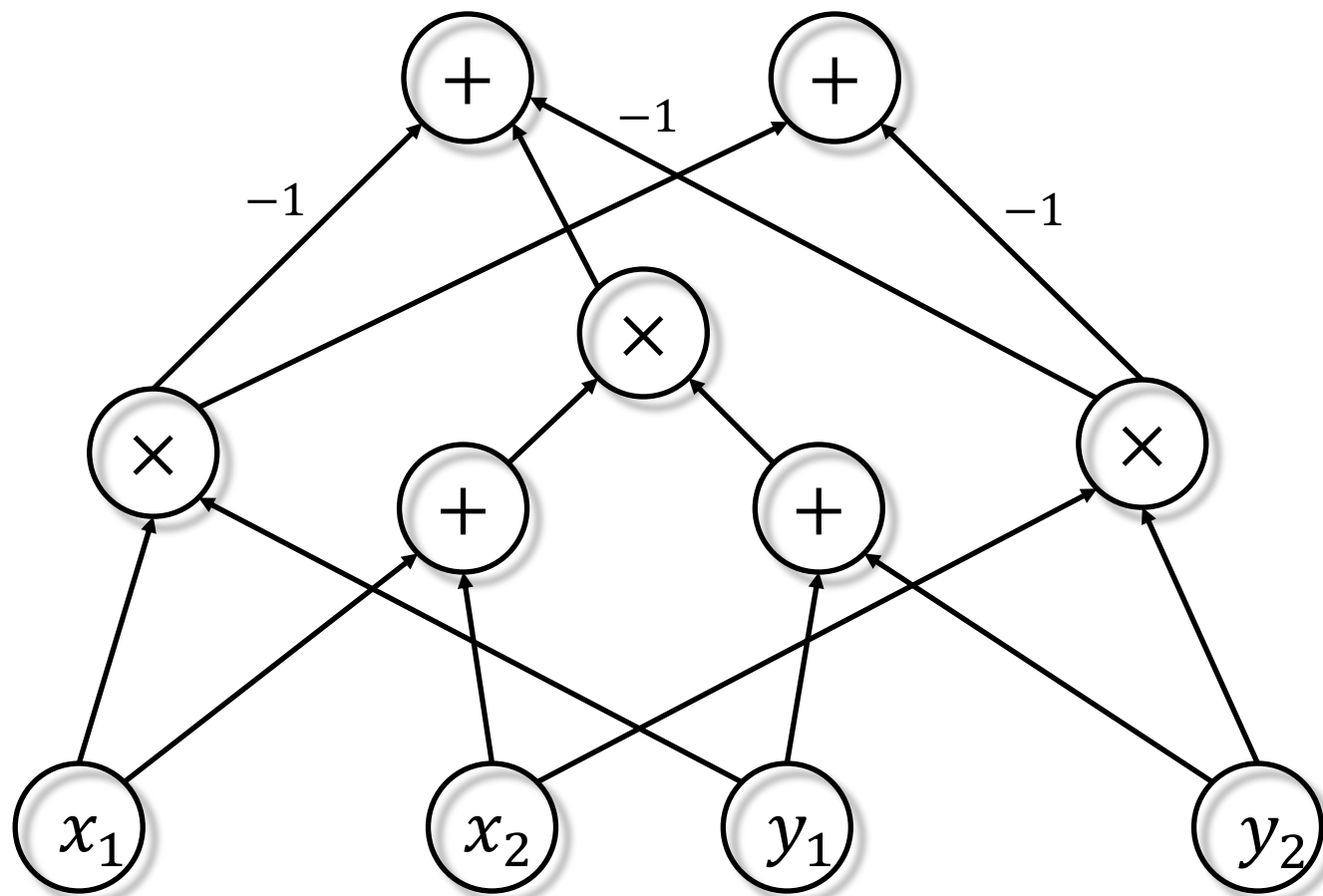


The header features a dark gray background with several mathematical diagrams. On the left, there are two triangles with vertices labeled 1, 2, and 3, and arrows indicating a cycle. In the center, there is a 3x3 matrix: $\begin{pmatrix} 1 & 2 & 3 \\ 2 & 1 & 3 \end{pmatrix}$. On the right, there is a tree diagram representing an algebraic circuit, with nodes labeled with multiplication (x) and addition (+) symbols, and leaf nodes containing the values 1, 2, and 3.

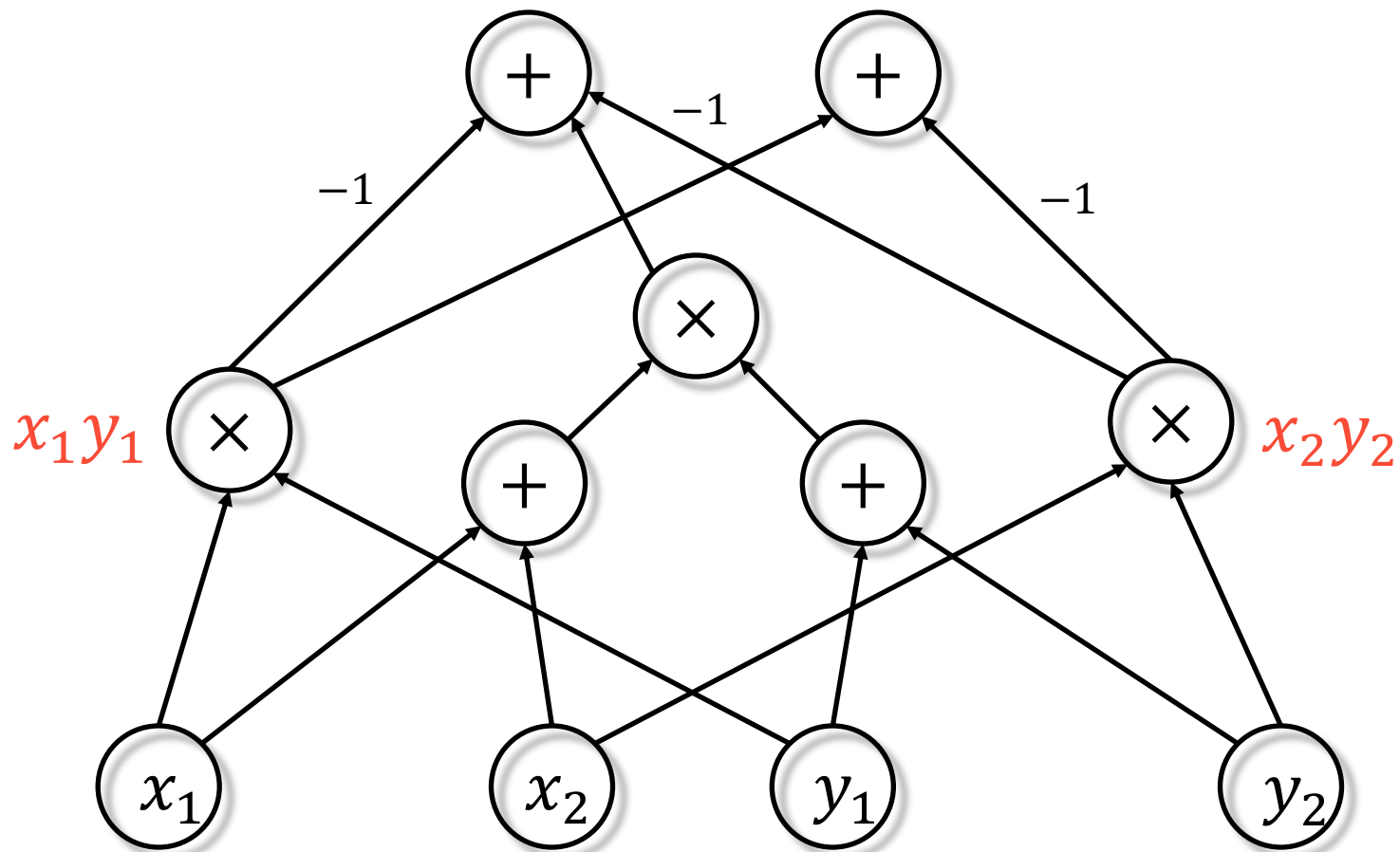
Prologue:

Algebraic circuit complexity

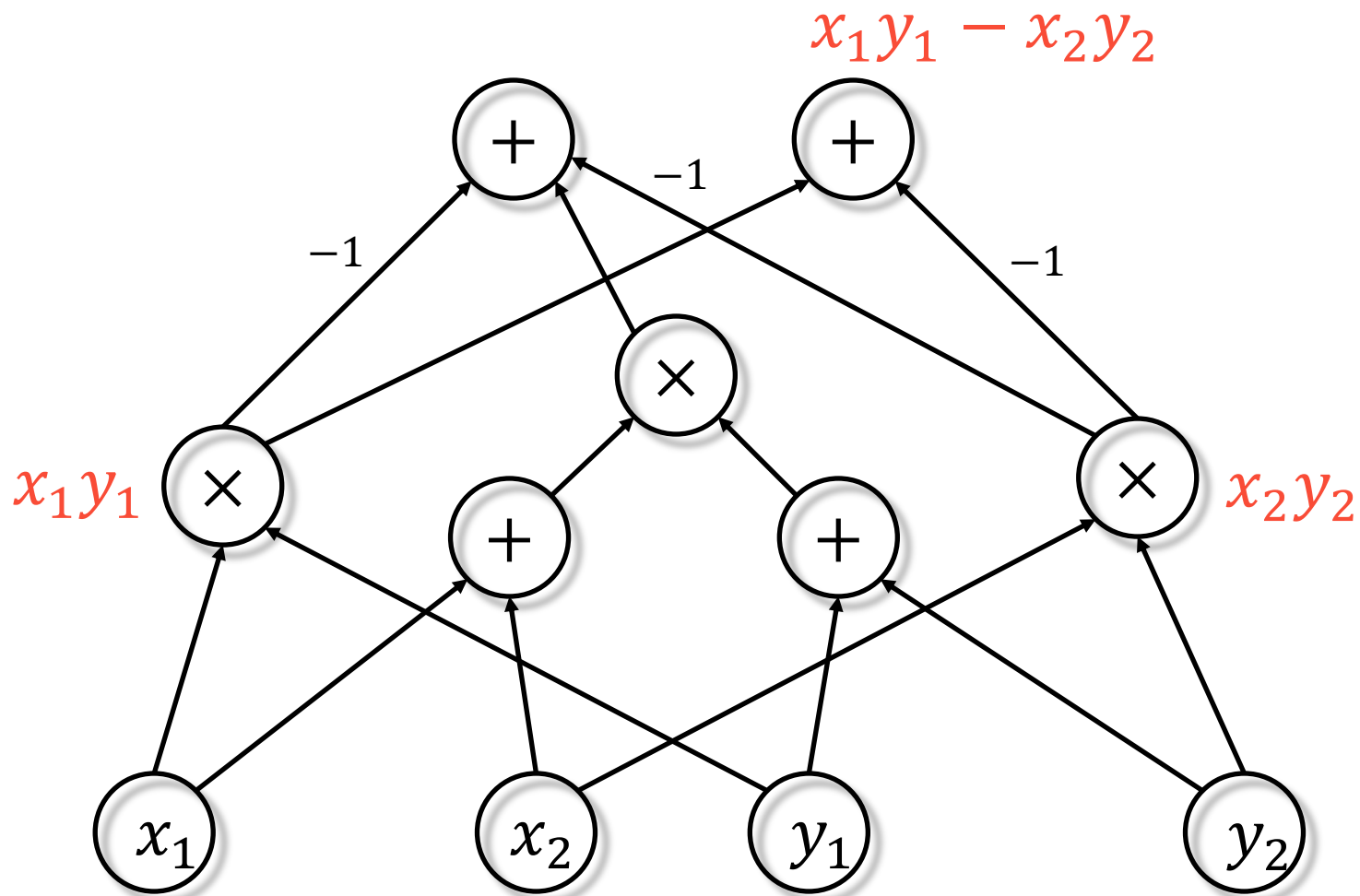
Algebraic Circuits



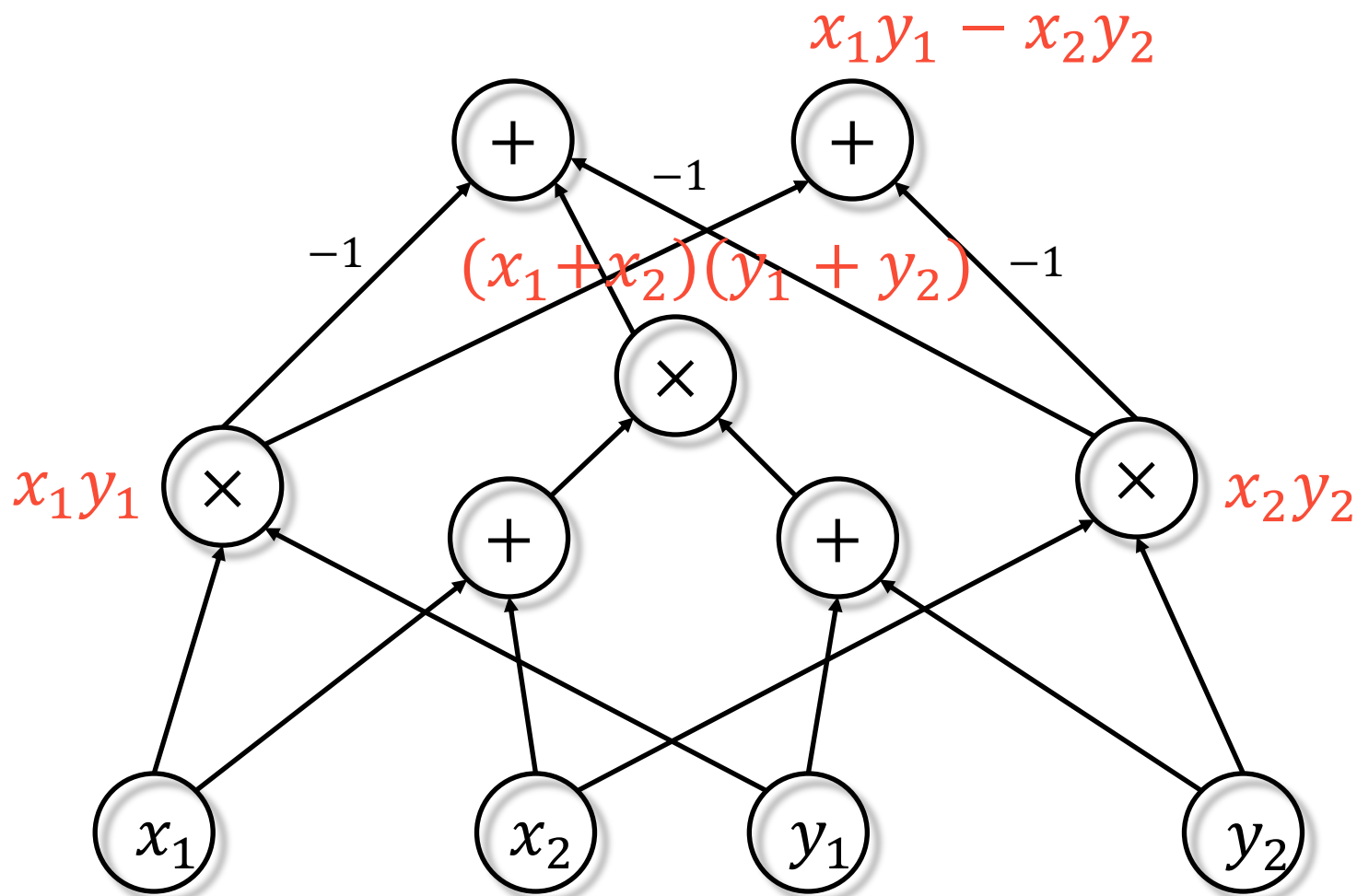
Algebraic Circuits



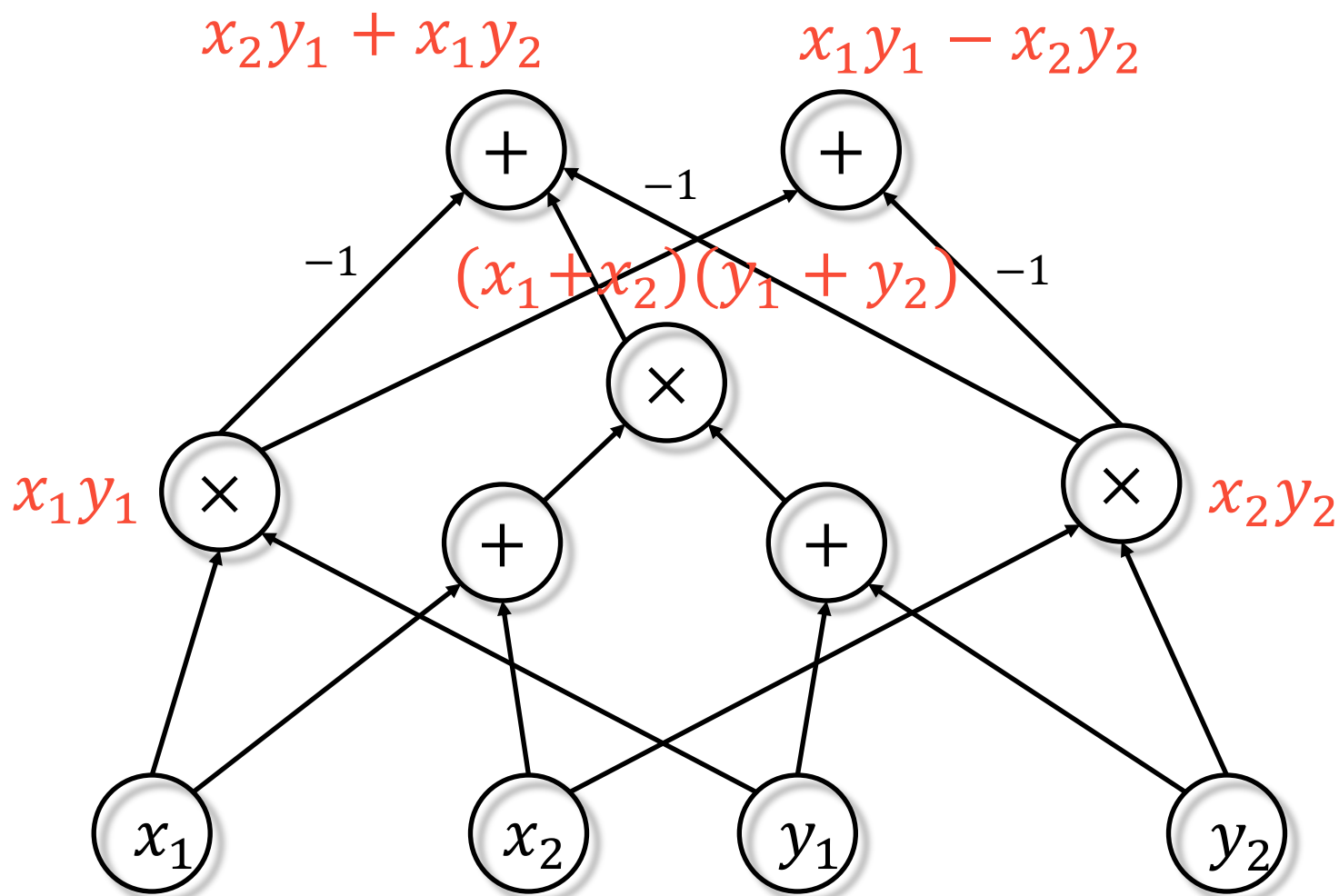
Algebraic Circuits



Algebraic Circuits



Algebraic Circuits



Algebraic Circuits

VP (“Valiant’s P”) is the class of **p-computable** functions, i.e. polynomial families (f_n) such that f_n has:

- $\text{poly}(n)$ variables
- $\text{poly}(n)$ degree
- $\text{poly}(n)$ -size algebraic circuits

Example: $(\det_n(X))$

Algebraic “NP”

VNP is the class of **p-definable** functions, i.e. $(f_n(x))$ such that there is $(g_n(e, x)) \in VP$ with

$$f_n(x) = \sum_{e \in \{0,1\}^{poly(n)}} g_n(e, x)$$

Example: $perm_n(X) =$

$$\sum_{E \in \{0,1\}^{n \times n}} \left(\prod_{\substack{i,j,i',j' \\ i=i' \Leftrightarrow j \neq j'}} (1 - E_{ij}E_{i'j'}) \right) \left(\prod_{i \in [n]} \sum_{j \in [n]} E_{ij} \right) \left(\prod_{i \in [n]} \sum_{j \in [n]} X_{ij}E_{ij} \right)$$

VNP is morally equivalent to counting solutions to NP problems

Boolean \rightarrow Algebraic Complexity

Theorem [[Bürgisser '00](#)]: $NP/\text{poly} \neq P/\text{poly} \Rightarrow VP \neq VNP$

Algebraic complexity:

- Formally necessary for Boolean complexity
- More structured – if we can't even prove lower bounds here...
- Captures key difficulties in Boolean complexity [[Valiant '79](#)]



The header features a dark gray background with several mathematical diagrams. On the left, there are two triangles with vertices labeled 1, 2, and 3, connected by double-headed arrows. In the center, a 3x3 matrix is shown: $\begin{pmatrix} 1 & 2 & 3 \\ 2 & 1 & 3 \end{pmatrix}$. On the right, there is a tree diagram representing a polynomial expression, with nodes labeled with multiplication (x) and addition (+) symbols, and leaf nodes containing the values 1, 1, 1, and 1.

Complexity in Ideals I:

Algebraic vs Boolean complexity

Complexity in Ideals I: Boolean vs Algebraic Complexity

In algebraic complexity, we consider polynomials
symbolically

Example

$x^2 - x$ is different from 0 *as a polynomial* over $\mathbb{Z}/2\mathbb{Z}$, even though they are the same *function* over $\mathbb{Z}/2\mathbb{Z}$

In Boolean complexity, we only care about the Boolean
function

Complexity in Ideals I: Boolean vs Algebraic Complexity

Can translate Boolean operations to algebra:

Boolean	Algebraic
Variable x	Variable x
$\neg f$	$1 - f$
$f \wedge g$	$f \cdot g$
Function $\{0,1\}^n \rightarrow \{0,1\}$	Polynomial over F^n , restricts to a function $\{0,1\}^n \rightarrow \{0,1\}$

But many polynomials define the same (Boolean) function on $\{0,1\}^n \subseteq F^n$. This is one reason algebraic lower bounds don't imply Boolean ones, even over finite fields.

Complexity in Ideals I: Boolean vs Algebraic Complexity

Many polynomials define the same (Boolean) function on $\{0,1\}^n \subseteq F^n$. This is one reason algebraic lower bounds don't imply Boolean ones, even over finite fields.

Ideals to the rescue: $f|_{\{0,1\}^n} = g|_{\{0,1\}^n} \Leftrightarrow f - g \in \langle x_i^2 - x_i | \forall i \rangle$.

Lower bounds on all polynomials in the ideal coset $f + \langle x_i^2 - x_i | \forall i \rangle$ over F_q imply Boolean lower bounds on f .



The header features a dark gray background with several mathematical diagrams. On the left, there are two triangles with vertices labeled 1, 2, and 3, and arrows indicating a cycle. In the center, there is a 3x3 matrix: $\begin{pmatrix} 1 & 2 & 3 \\ 2 & 1 & 3 \end{pmatrix}$. On the right, there is a tree diagram with a root node labeled 'x', branching into two nodes labeled '+', which further branch into nodes labeled 'x' and '1', and finally into leaf nodes labeled '1' and '1'.

Complexity in Ideals II:

Algebraic proof complexity

Algebraic Proof Complexity

Something I like about proof complexity: gives a way of
measuring the complexity of individual instances of SAT

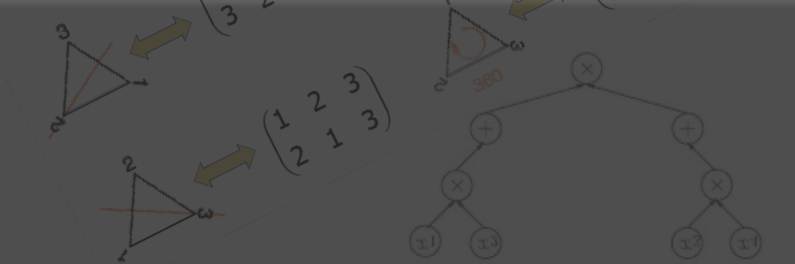
Algebraic proof complexity studies how hard it is to prove there
are no solutions to systems of polynomial equations

$$F_1(\vec{x}) = F_2(\vec{x}) = \dots = F_k(\vec{x}) = 0$$

Arises naturally...

- ...in geometric theorem-proving
- ...as a path towards $AC^0[p]$ -Frege lower bounds, which we're still stuck on (despite Razborov-Smolensky)!

Algebraic Proof Complexity



Systems of polynomial equations can simulate Boolean equations:

$$x_i^2 - x_i = 0 \text{ forces } x_i \in \{0,1\}$$

$$T(x) := 1 - x \text{ (} 1 - x = 0 \text{ is satisfied iff } x \text{ is TRUE)}$$

$$T(\neg f) := 1 - T(f)$$

$$T(f \vee g) := T(f)T(g)$$

Hilbert's Nullstellensatz: $F_1 = F_2 = \dots = F_k = 0$ has no solutions \Leftrightarrow there are polynomials G_1, \dots, G_k such that

$$F_1 G_1 + F_2 G_2 + \dots + F_k G_k = 1$$

$$\text{i.e. } 1 \in \langle F_1, \dots, F_k \rangle$$

The Ideal Proof System [[P96](#), [P98](#), [GP14](#)]

Hilbert's Nullstellensatz: $F_1 = F_2 = \dots = F_k = 0$ has no solutions \Leftrightarrow there are polynomials G_1, \dots, G_k such that

$$F_1 G_1 + F_2 G_2 + \dots + F_k G_k = 1.$$

Introduce **new place-holder variables** y_1, \dots, y_k , get a new polynomial

$$C(\vec{x}, y_1, \dots, y_k) = y_1 G_1(\vec{x}) + \dots + y_k G_k(\vec{x})$$

Definition: $C(\vec{x}, \vec{y})$ is an **IPS certificate** if

1. $C(\vec{x}, \overrightarrow{F(\vec{x})}) = 1$
2. $C(\vec{x}, \vec{y}) \in \langle y_1, \dots, y_k \rangle$

The Ideal Proof System [[P96](#), [P98](#), [GP14](#)]

Definition: $C(\vec{x}, y)$ is an **IPS certificate** if

1. $C(\vec{x}, \overrightarrow{F(\vec{x})}) = 1$
2. $C(\vec{x}, \vec{y}) \in \langle y_1, \dots, y_k \rangle$

Definition: The **IPS complexity** of an unsatisfiable system of equations is the **optimum algebraic complexity of any certificate**.

E.g. algebraic circuit size, formula size, VNP, ...

Default: algebraic circuit size (*no degree bound!*)

Aside:
What about semi-algebraic?

Cone Proof System introduced by [Aleksseev, Grigoriev, Hirsch, Tzameret \(STOC '20\)](#)

Ideal Proof System is to Polynomial Calculus

As

Cone Proof System is to Sum of Squares

What is the analogue of complexity in ideal cosets for the cone proof system?

Randomized Verification of IPS

Polynomial Identity Testing

PIT(C)

Input: an algebraic circuit K from C

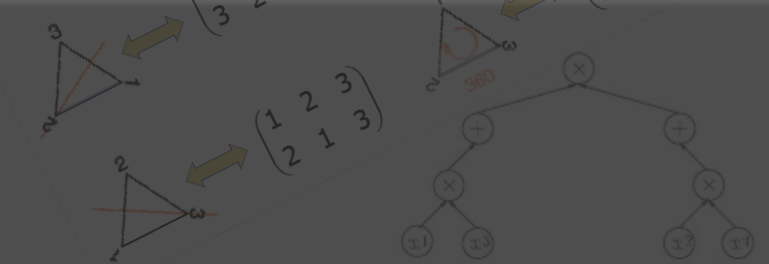
Decide: Does K compute the zero polynomial?

(Remember: polynomial \neq function)

Schwarz-Zippel-DeMillo-Lipton: PIT is in coRP. Proof: evaluate at random points. (1-page proof by [Moshkovitz '10](#))

[Kabanets-Impagliazzo '04](#): Derandomizing PIT implies circuit lower bounds (simplified in [CIKK15](#))

Randomized Verification and IPS versus NP



Proposition [P96]: If $NP \not\subseteq coMA$, then some tautologies require super-polynomial size IPS proofs.

Note: $NP \subseteq coMA \Rightarrow PH$ collapses.

Proof: Merlin guesses the poly-size circuit for a certificate, Arthur verifies it's a certificate using two polynomial identity tests (PIT):

1. $C(\vec{x}, \overrightarrow{F(\vec{x})}) = 1$
2. $C(\vec{x}, \vec{y}) \in \langle y_1, \dots, y_k \rangle$, that is, $C(\vec{x}, \vec{0}) = 0$.

QED

Dealing w/ constants needs more work, coAM, & GRH.



Circuit Lower Bound Implications

Theorem [GP14]: Any super-polynomial lower bound on IPS over a ring R implies $VNP_R^0 \neq VP_R^0$.

Key Lemma: Every unsatisfiable CNF has a VNP^0 certificate.

Proof of Theorem assuming Key Lemma: A super-polynomial size lower bound on IPS means there are tautologies such that *every certificate* requires super-polynomial size. Since some certificate is in VNP^0 , that function requires super-poly size circuits. QED

Complexity in Ideals from The Ideal Proof System

Definition: $C(\vec{x}, \vec{y})$ is an **IPS certificate** if

1. $C(\vec{x}, \overrightarrow{F(\vec{x})}) = 1$
2. $C(\vec{x}, \vec{y}) \in \langle y_1, \dots, y_k \rangle$

Condition 1 is equivalent to:

$$C \equiv 1 \bmod \langle y_1 - F_1(\vec{x}), \dots, y_k - F_k(\vec{x}) \rangle$$

i.e.

$$C \in 1 + \langle y_1 - F_1(\vec{x}), \dots, y_k - F_k(\vec{x}) \rangle$$

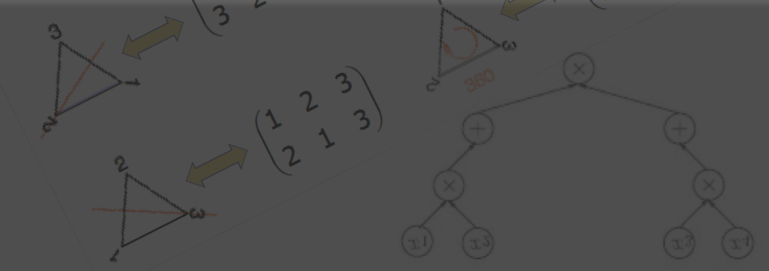
Thus the **set of IPS certificates** is a coset of an ideal:

$$(1 + \langle y_1 - F_1(\vec{x}), \dots, y_k - F_k(\vec{x}) \rangle) \cap \langle y_1, \dots, y_k \rangle$$

Key question: complexity of IPS certificates, i.e. **polynomials in this ideal coset**

What **structure** does the set of all IPS certificates have?

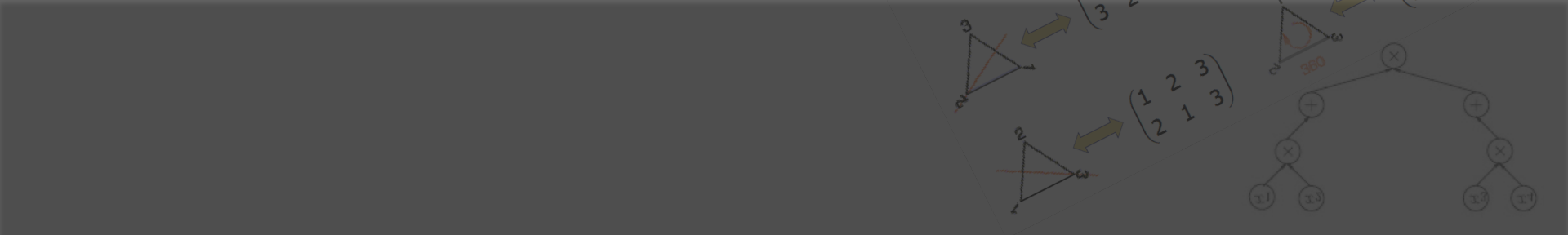
Complexity in Ideals from The Ideal Proof System



[Li-Tzameret-Wang](#) (CCC '15, SICOMP '18): Frege is quasipoly equivalent to noncommutative formula IPS, when given clauses, $x_i^2 - x_i$, and $x_i x_j - x_j x_i$.

Main idea of proof: Show that [Raz-Shpilka '04](#) noncommutative formula PIT algorithm can be formalized in Frege.

The set of noncommutative IPS certificates is still an ideal coset, *but* noncommutative ideals need not be finitely generated! What does this tell us? Still unclear...



The header features a dark gray background with several mathematical diagrams. On the left, there are two triangles with vertices labeled 1, 2, and 3, and arrows indicating a cycle. In the center, there is a 3x3 matrix: $\begin{pmatrix} 1 & 2 & 3 \\ 2 & 1 & 3 \end{pmatrix}$. On the right, there is a tree diagram representing a polynomial expression, with nodes labeled with multiplication (x) and addition (+) symbols, and leaf nodes containing numerical values like 1.1 and 1.3.

Complexity in Ideals III: Polynomial identity testing

Polynomial Identity Testing

Def: A *hitting set generator* for a complexity class \mathcal{C} is a polynomial map

$$G: F^m \rightarrow F^n$$

such that any nonzero $f \in \mathcal{C}$ also has $f \circ G \neq 0$. (Over infinite fields: f is nonzero on $\text{Im}(G)$.)

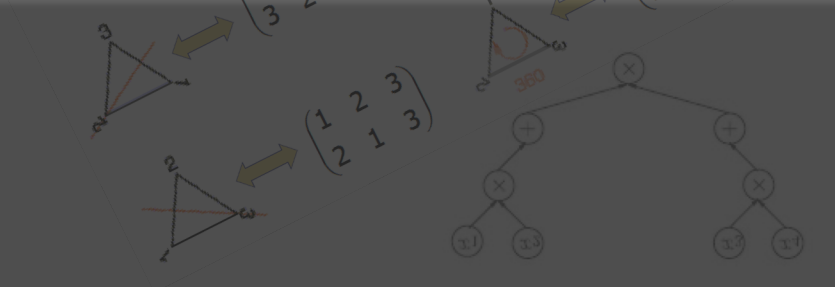
$I_G := \{f \mid f \circ G = 0\}$ is an ideal!

Proof: If $f_i \circ G = 0$, then $(af_1 + bf_2) \circ G = af_1 \circ G + bf_2 \circ G = 0$. If g any poly, then $(gf) \circ G = (g \circ G)(f \circ G) = 0$. QED

G is a hitting set generator for \mathcal{C}



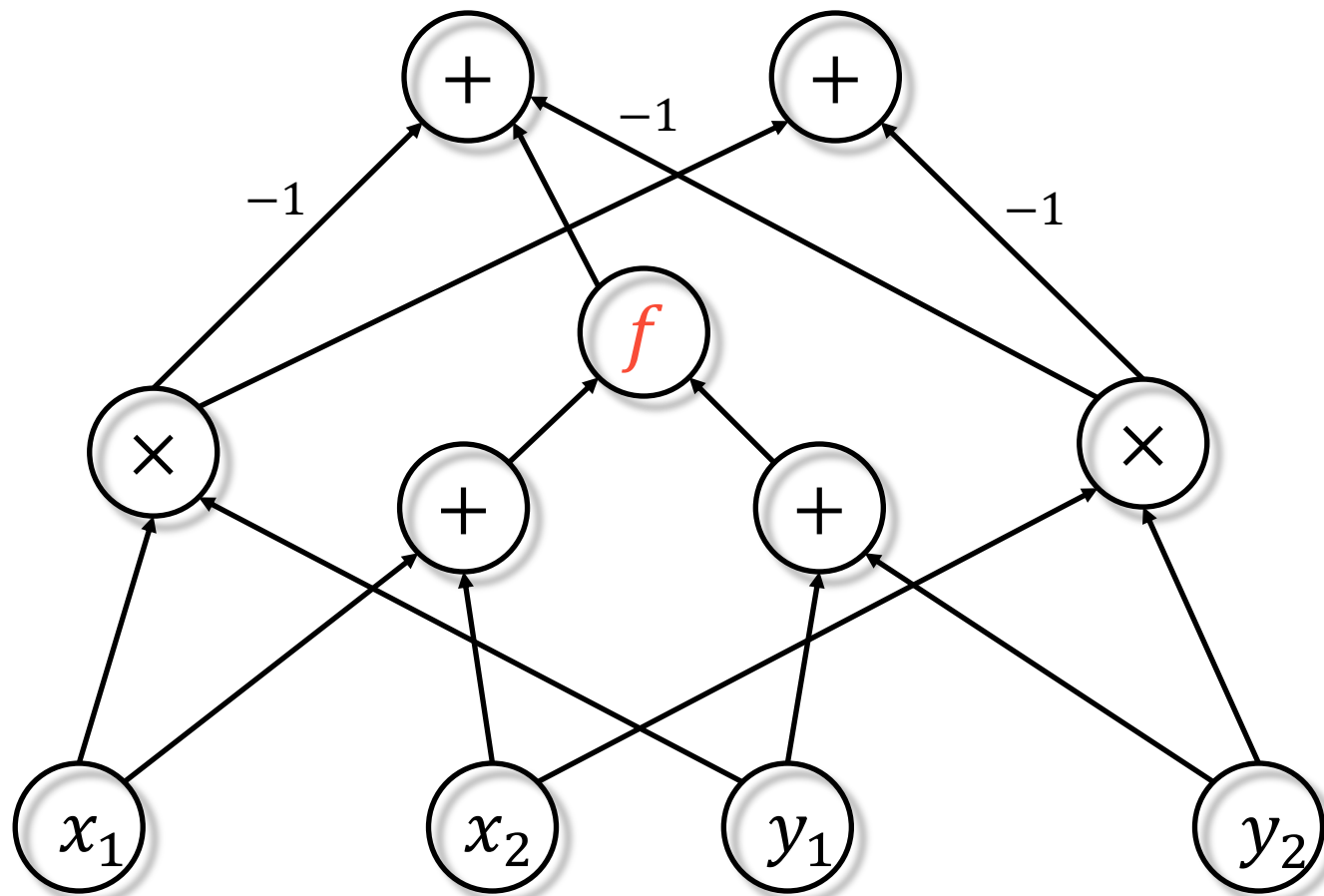
\mathcal{C} lower bound on all polys in the ideal I_G



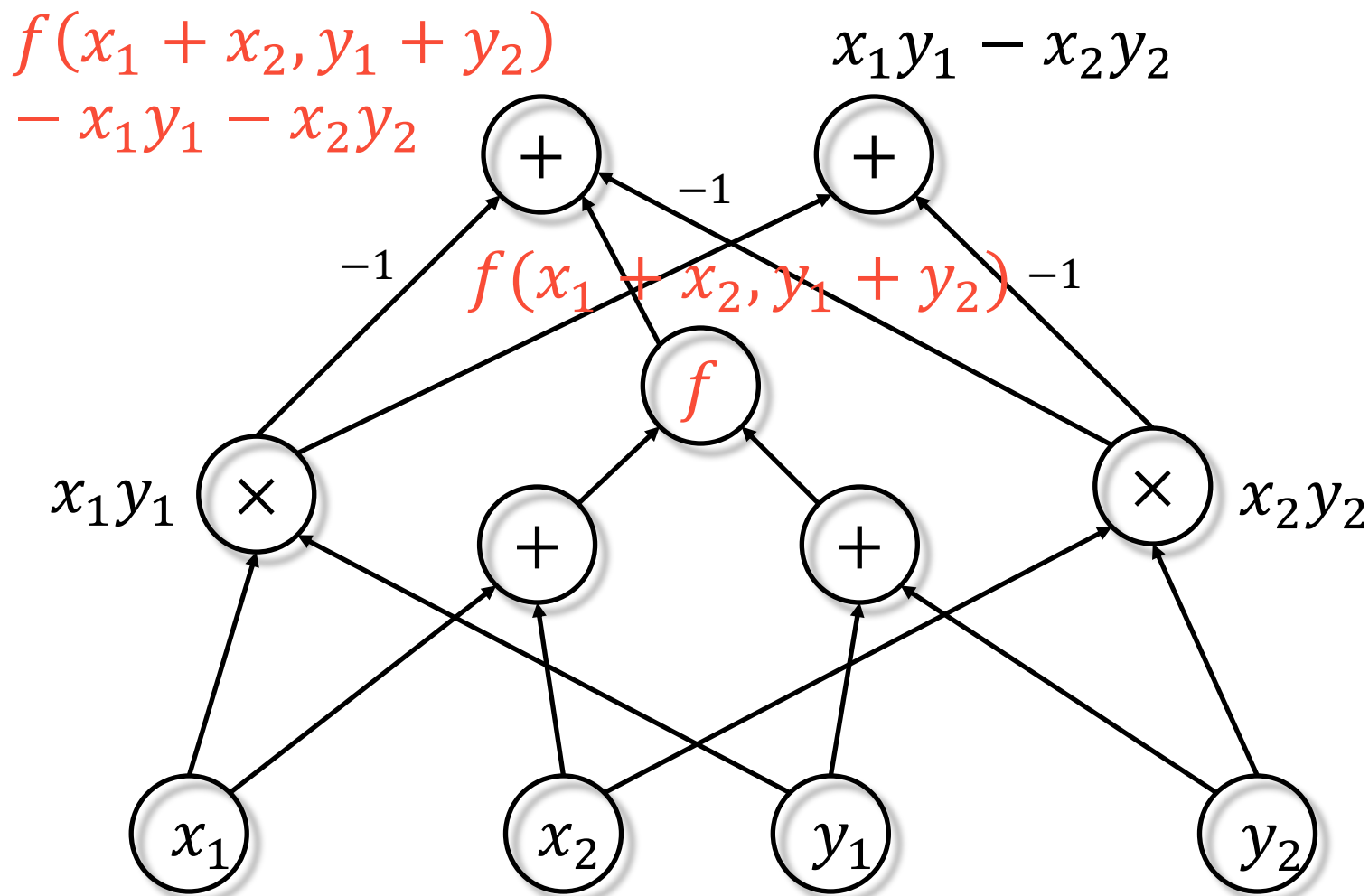
How is this different from circuit lower bounds?

Complexity in complexity classes vs.
Complexity in ideals

Algebraic Circuit Reductions



Algebraic Circuit Reductions



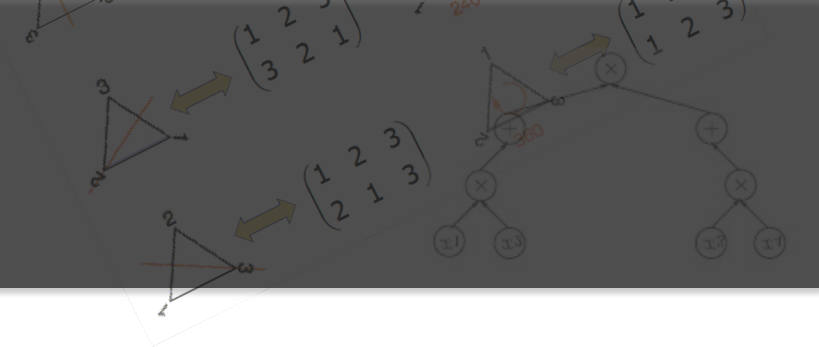
Algebraic Circuit **Reductions**

Definition:

- $C^f(g)$ = min circuit size to compute g , with $(+, \times, f)$ gates.
- $(g_n) \leq_c (f_n)$, g is a **c-reduction** of f , if $C^{f \text{ poly}(n)}(g_n) \leq \text{poly}(n)$.
- The **(c-)degree** of $g = (g_n)$ is the equivalence class of g under \leq_c , that is, $D_c(g) := \{f \mid g \leq_c f \text{ and } f \leq_c g\}$.
- Given two c-degrees D_1, D_2 , $D_1 \leq_c D_2$ is well-defined.

Ladner-Type Theorem [[Bürgisser '00](#)]: If $f <_c g$, then any countable poset embeds in the poset of c-degrees between f, g .

Poset of c-Degrees in Complexity Classes vs. Ideals



In VP vs VNP, suffices to show **one** polynomial in $\text{VNP} \setminus \text{VP}$.

In an ideal I , we want to know if **every** polynomial in I is not in VP.

VNP, VQP, ... have complete problems (top of the poset).
WLOG, look for lower bounds on complete problems.

Do ideals have minimal c-degrees?

Factorization & Principal Ideals

Definition: An ideal is **principal** if it is generated by 1 element.
Consider the principal ideals $I_n = \langle g_n \rangle$.

Theorem [K86, B00]: For $f_n = g_n^{e_n} h_n$ (g, h coprime).
$$C(g_n) \leq C(f_n) \text{poly}(\deg g_n, e_n)$$

In fact, $g \leq_c f$ when $e_n \leq \text{poly}(n)$.

Theorem [B04]: $\underline{C}(g) \leq C(f) \text{poly}(\deg g)$ (indep. of e !)

Factor Conjecture [Bürgisser]: $g \leq_c f$?

IPS Lower Bounds From Principal Ideals

[Forbes-Shpilka-Tzameret-Wigderson](#) (CCC '16, ToC '21)

Lemma [[FSTW16](#)]: Suppose

$F = F_1 = F_2 = \dots = F_m = 0$ is unsatisfiable, but

$F_1 = F_2 = \dots = F_m = 0$ is satisfiable.

Then for any IPS certificate $C(\vec{x}, \vec{y})$, we have

$$0 \neq 1 - C(\vec{x}, 0, F_1, \dots, F_m) \in \langle F \rangle.$$

Strategy: “Lower bounds for multiples” (i.e. the ideal $\langle F \rangle$)

IPS Lower Bounds From Principal Ideals

Using lower bounds for multiples, i.e. on principal ideals, they show:

Theorem [FSTW16]: Exponential lower bounds on $\Sigma \wedge \Sigma$ -IPS proofs for

- $x_1 x_2 \dots x_n - 1, \sum x_i - m, \{x_i^2 - x_i\}$

Exponential lower bounds on roABP-IPS proofs for

- $\prod_{i < j} ((x_i + x_j - 1)), \sum x_i - m, \{x_i^2 - x_i\}$

IPS Lower Bounds From Boolean Ideals

Lemma [FSTW16]: Suppose $F = x_1^2 - x_1 = \dots = x_n^2 - x_n = 0$ is unsatisfiable. If every G that agrees with $1/F$ on $\{0,1\}^n$ is not in C , then this does not have C -IPS refutations.

Lower bounds on a coset of $\langle x_i^2 - x_i | \forall i \rangle$.

What is Known: Algebraic Proof Complexity

Using functional lower bounds = **lower bounds on Boolean ideal cosets**, they get

Theorem [FSTW16]: Exponential lower bounds on $\Sigma \wedge \Sigma$ -IPS proofs for

- $\sum x_i y_i - \beta, \{x_i^2 - x_i\}, \{y_i^2 - y_i\} \ (\beta > n)$

Exponential lower bounds on roABP-IPS proofs for

- $\sum z_{ij} x_i x_j - \beta, \{x_i^2 - x_i\}, \{z_{ij}^2 - z_{ij}\} \ (\beta > \binom{2n}{2})$

Constant-depth IPS lower bounds

From lower bounds on ideals

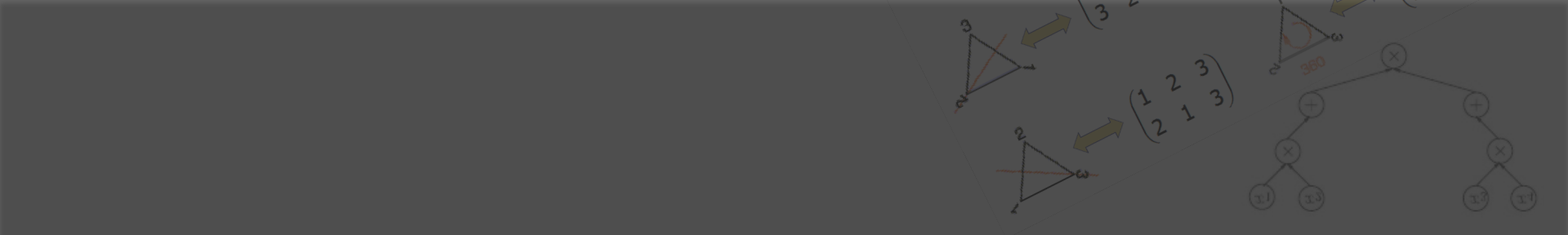
Conjecture [G., 2018]: The ideal of $\frac{n}{2} \times \frac{n}{2}$ minors of an $n \times n$ matrix has the determinant as its unique minimum c-degree.

Andrews & Forbes '21: Conjecture is true! Even for depth-3 (border) c-reductions.

Limaye-Srinivasan-Távenas [FOCS '21]: Det requires large constant depth algebraic circuits.

Andrews & Forbes '21: Constant-depth IPS lower bounds on $\det(X) = 0, XY = I, \{all\ vars\ Boolean\}$.

Proof: Lower bound for multiples of det. QED



Lots of Open Questions on
Complexity in Ideals

Open Questions

- Must ideals/cosets have a unique minimum c-degree? (Surely not!)
 - Must they have only **finitely many** minimal c-degrees?
 - Must they have **any** minimal c-degree?
 - Can they contain an infinite descending chain of degrees?
- OPEN** even for 2-generated ideals $I_n = \langle g_n, f_n \rangle$. (In n vars, max # generators is ε^{n+1} Grzegorzczyk primitive recursive hierarchy. [[Seidenberg '71](#), [Simpson '88](#)].)

Open Questions

Get an IPS lower bound (for some restricted system, say roABP, depth 3, constant depth) on the translation of Boolean tautologies.

What is the proof complexity of formalizing other PIT algorithms?

Does the p-simulation of IPS by any deterministically verifiable (Cook-Reckhow) proof system imply some derandomization of PIT?

EXTRA SLIDES

Open Questions

Definition: A c-degree D is **saturated** in an ideal I if every $D' \geq_c D$ appears in I .

Observation [see [G. '18](#)]: In any ideal, every p-bounded c-degree is saturated, assuming the Factor Conjecture.

OPEN: What about in cosets of ideals?

How many generators can there be?

Intuition: In n variables, m equations should have a solution set of dimension $n - m$, so should never need more than n equations.

True if equations are “generic” (=chosen randomly).

False in general!

Example: twisted cubic $\{(s^3, s^2t, st^2, t^3) : s, t \in F\} \subseteq F^4$
2D variety in 4D, “should” only need 2 equations.

But needs 3:

$$xz - y^2, yw - z^2, xw - yz$$

How many generators can there be?

Grzegorzczyk primitive recursive hierarchy ε^n .

ε^1 =linear

ε^2 =polynomial

ε^3 =tower of exponentials of const. height

ε^n =primitive recursive over ε^{n-1} .

Theorem [Seidenberg '71; Simpson '88]: An ideal in $F[x_1, \dots, x_n]$ generated in degrees $\leq n$ needs no more than ε^{n+1} generators.

How many generators can there be?

Grzegorzczyk primitive recursive hierarchy ε^n .

ε^1 =linear

ε^2 =polynomial

ε^3 =tower of exponentials of const. height

ε^n =primitive recursive over ε^{n-1} .

Theorem [[Seidenberg '71](#); [Simpson '88](#)]: An ideal in $F[x_1, \dots, x_n]$ generated in degrees $\leq n$ needs no more than ε^{n+1} generators. And this is tight!

Circuit Lower Bound Implications: Proof of Key Fact

Key Lemma: Every DNF tautology has a VNP^0 certificate.

Proof:

1. $x_1 + (1 - x_1) = 1$

$$x_1x_2 + x_1(1 - x_2) + (1 - x_1)x_2 + (1 - x_1)(1 - x_2) = 1$$

$$\sum_{e \in \{0,1\}^n} \prod_i \begin{cases} x_i & \text{if } e_i = 0 \\ 1 - x_i & \text{if } e_i = 1 \end{cases} = 1$$

2. Turn this into a certificate.

3. Show the certificate is in VNP .

Circuit Lower Bound Implications: Proof of Key Fact

2. Turn this into a certificate.

$$x_1 = 0 \wedge (1 - x_1)(1 - x_2) = 0 \wedge x_2(1 - x_3) = 0 \wedge x_3 = 0$$

$$\begin{aligned} 1 &= x_1(x_2x_3 + x_2(1 - x_3) + (1 - x_2)x_3 + (1 - x_2)(1 - x_3)) \\ &+ (1 - x_1)(1 - x_2)(x_3 + (1 - x_3)) \\ &+ (1 - x_1)x_2(1 - x_3) \\ &+ (1 - x_1)x_2x_3 \end{aligned}$$

Circuit Lower Bound Implications: Proof of Key Fact

2. Turn this into a certificate.

$$x_1 = 0 \wedge (1 - x_1)(1 - x_2) = 0 \wedge x_2(1 - x_3) = 0 \wedge x_3 = 0$$

$$\begin{aligned} & x_1(x_2x_3 + x_2(1 - x_3) + (1 - x_2)x_3 + (1 - x_2)(1 - x_3)) \\ & + (1 - x_1)(1 - x_2)(x_3 + (1 - x_3)) \\ & + (1 - x_1)x_2(1 - x_3) \\ & + (1 - x_1)x_2x_3 \end{aligned}$$

Circuit Lower Bound Implications: Proof of Key Fact

2. Turn this into a certificate.

$$x_1 = 0 \wedge (1 - x_1)(1 - x_2) = 0 \wedge x_2(1 - x_3) = 0 \wedge x_3 = 0$$

$$\begin{aligned} & y_1(x_2x_3 + x_2(1 - x_3) + (1 - x_2)x_3 + (1 - x_2)(1 - x_3)) \\ & + (1 - x_1)(1 - x_2)(x_3 + (1 - x_3)) \\ & + (1 - x_1)x_2(1 - x_3) \\ & + (1 - x_1)x_2x_3 \end{aligned}$$

Circuit Lower Bound Implications: Proof of Key Fact

2. Turn this into a certificate.

$$x_1 = 0 \wedge (1 - x_1)(1 - x_2) = 0 \wedge x_2(1 - x_3) = 0 \wedge x_3 = 0$$

$$\begin{aligned} & y_1(x_2x_3 + x_2(1 - x_3) + (1 - x_2)x_3 + (1 - x_2)(1 - x_3)) \\ & + (1 - x_1)(1 - x_2)(x_3 + (1 - x_3)) \\ & + (1 - x_1)x_2(1 - x_3) \\ & + (1 - x_1)x_2x_3 \end{aligned}$$

Circuit Lower Bound Implications: Proof of Key Fact

2. Turn this into a certificate.

$$x_1 = 0 \wedge (1 - x_1)(1 - x_2) = 0 \wedge x_2(1 - x_3) = 0 \wedge x_3 = 0$$

$$\begin{aligned} & y_1(x_2x_3 + x_2(1 - x_3) + (1 - x_2)x_3 + (1 - x_2)(1 - x_3)) \\ & + y_2(x_3 + (1 - x_3)) \\ & + (1 - x_1)x_2(1 - x_3) \\ & + (1 - x_1)x_2x_3 \end{aligned}$$

Circuit Lower Bound Implications: Proof of Key Fact

2. Turn this into a certificate.

$$x_1 = 0 \wedge (1 - x_1)(1 - x_2) = 0 \wedge x_2(1 - x_3) = 0 \wedge x_3 = 0$$

$$\begin{aligned} & y_1(x_2x_3 + x_2(1 - x_3) + (1 - x_2)x_3 + (1 - x_2)(1 - x_3)) \\ & + y_2(x_3 + (1 - x_3)) \\ & + (1 - x_1)x_2(1 - x_3) \\ & + (1 - x_1)x_2x_3 \end{aligned}$$

Circuit Lower Bound Implications: Proof of Key Fact

2. Turn this into a certificate.

$$x_1 = 0 \wedge (1 - x_1)(1 - x_2) = 0 \wedge x_2(1 - x_3) = 0 \wedge x_3 = 0$$

$$\begin{aligned} & y_1(x_2x_3 + x_2(1 - x_3) + (1 - x_2)x_3 + (1 - x_2)(1 - x_3)) \\ & + y_2(x_3 + (1 - x_3)) \\ & + (1 - x_1)y_3 \\ & + (1 - x_1)x_2x_3 \end{aligned}$$

Circuit Lower Bound Implications: Proof of Key Fact

2. Turn this into a certificate.

$$x_1 = 0 \wedge (1 - x_1)(1 - x_2) = 0 \wedge x_2(1 - x_3) = 0 \wedge \textcolor{red}{x}_3 = 0$$

$$\begin{aligned} & y_1(x_2x_3 + x_2(1 - x_3) + (1 - x_2)x_3 + (1 - x_2)(1 - x_3)) \\ & + y_2(x_3 + (1 - x_3)) \\ & + (1 - x_1)y_3 \\ & + (1 - x_1)x_2\textcolor{red}{x}_3 \end{aligned}$$

Circuit Lower Bound Implications: Proof of Key Fact

2. Turn this into a certificate.

$$x_1 = 0 \wedge (1 - x_1)(1 - x_2) = 0 \wedge x_2(1 - x_3) = 0 \wedge x_3 = 0$$

$$\begin{aligned} & y_1(x_2x_3 + x_2(1 - x_3) + (1 - x_2)x_3 + (1 - x_2)(1 - x_3)) \\ & + y_2(x_3 + (1 - x_3)) \\ & + (1 - x_1)y_3 \\ & + (1 - x_1)x_2y_4 \end{aligned}$$

Circuit Lower Bound Implications: Proof of Key Fact

$$x_1 = 0 \wedge (1 - x_1)(1 - x_2) = 0 \wedge x_2(1 - x_3) = 0 \wedge x_3 = 0$$

$$\begin{aligned} & y_1(x_2x_3 + x_2(1 - x_3) + (1 - x_2)x_3 + (1 - x_2)(1 - x_3)) \\ & + y_2(x_3 + (1 - x_3)) \\ & + y_3(1 - x_1) \\ & + y_4(1 - x_1)x_2 \end{aligned}$$

3. Show the certificate is in VNP.

$$\sum_{\text{clauses } k} f_k \left[\sum_{e \in \{0,1\}^n} (1 - F_k(\vec{e})) \left(\prod_{i < k} F_i(\vec{e}) \right) \prod_{j \notin F_i} \begin{cases} x_j & \text{if } e_j = 0 \\ 1 - x_j & \text{if } e_j = 1 \end{cases} \right]$$

Circuit Lower Bound Implications: Proof of Key Fact

$$x_1 = 0 \wedge (1 - x_1)(1 - x_2) = 0 \wedge x_2(1 - x_3) = 0 \wedge x_3 = 0$$

$$\begin{aligned} & y_1(x_2x_3 + x_2(1 - x_3) + (1 - x_2)x_3 + (1 - x_2)(1 - x_3)) \\ & + y_2(x_3 + (1 - x_3)) \\ & + y_3(1 - x_1) \\ & + y_4(1 - x_1)x_2 \end{aligned}$$

3. Show the certificate is in VNP.

$$\sum_{\text{clauses } k} f_k \left[\sum_{e \in \{0,1\}^n} (1 - F_k(\vec{e})) \left(\prod_{i < k} F_i(\vec{e}) \right) \prod_{j \notin F_i} (e_j + (1 - 2e_j)x_j) \right]$$

QED