Polynomial Identity Testing

Amir Shpilka

Technion
Goal of talk

• Model: Arithmetic circuits
• Problem: Polynomial Identity Testing
• Example: Depth-3 circuits
• Some open problems
Boolean Complexity

• Holy grail: P vs. NP
• In a nutshell: Show that certain problems (e.g., finding the minimum distance of a binary code given by its parity check matrix) cannot be decided by small Boolean circuits
Boolean Complexity

- Holy grail: $\mathbf{P} \text{ vs. } \mathbf{NP}$
- In a nutshell: Show that certain problems (e.g., finding the minimum distance of a binary code given by its parity check matrix) cannot be decided by small Boolean circuits
- Problem notoriously difficult – with minuscule advance
- Natural idea: consider more structured models
Playground: Arithmetic Circuits

Field: $\mathbb{F}$ (e.g., $\mathbb{F}_2$, $\mathbb{R}$)

Variables: $X_1,...,X_n$

Gates: $+, \times$

Every gate in the circuit computes a polynomial in $\mathbb{F}[X_1,...,X_n]$

Example: $(X_1 \cdot X_2) \cdot (X_2 + 1)$

Size = number of wires

Depth = length of longest input-output path

Degree = max degree of internal gates
Playground: Arithmetic Circuits

In Example:
- Size = 6
- Depth = 2
- Degree = 3

Example: \((X_1 \cdot X_2) \cdot (X_2 + 1)\)

Size = number of wires
Depth = length of longest input-output path
Degree = max degree of internal gates

Example Circuit:
- \(X_1\) and \(X_2\) as inputs
- \(1\) as a constant input
- Two multiplication gates
- One addition gate

Size = 6
Depth = 2
Degree = 3
Why Arithmetic Circuits?

- Structured model (compared to Boolean circuits) $P$ vs. $NP$ may be easier
- Most natural model for computing polynomials
- For many problems (e.g. Matrix Multiplication, Det) best algorithm is an arithmetic circuit
- Great algorithmic achievements:
  - Fourier Transform
  - Matrix Multiplication
  - Polynomial Factorization
Important Problems

• Design new algorithms:
  – $\tilde{O}(n^2)$ for Matrix Multiplication?
  – Understanding $\mathbb{P}$

• Prove lower bounds:
  – Find a polynomial (e.g. Permanent) that requires super-polynomial size or super-logarithmic depth
  – Analog of $\mathbb{P}$ vs. $\mathbb{NP}$

• Derandomize Polynomial Identity Testing:
  – Understanding the power of randomness
  – Analog of $\mathbb{P}$ vs. $\mathbb{BPP}$
ω^n=1. Is the following polynomial identically 0?

\[
\prod_{i=1}^{n} \left( \omega^5 \pi X + \left( \omega^5 e - \omega^i \hbar \right) Y - \omega^i \pi e Z \right) + \\
\prod_{i=1}^{n} \left( -e\omega^i X + \left( \pi\omega^i + \hbar \right) Y + \left( \pi e - \hbar \omega^i \right) Z \right) + \\
\prod_{i=1}^{n} \left( e\omega^2 - \pi \omega^i \right) X - \left( \pi \omega^2 + e\omega^i \right) Y + \hbar \omega^2 Z
\]

Prove it!
Will do so later.
Polynomial Identity Testing

Input: Arithmetic circuit computing $f$
Problem: Is $f \equiv 0$?

Note: $x^2 - x$ is the zero function over $\mathbb{F}_2$ but not the zero polynomial!
Polynomial Identity Testing

Input: Arithmetic circuit computing $f$
Problem: Is $f \equiv 0$?

Randomized algorithm [Schwartz, Zippel, DeMillo-Lipton]: evaluate $f$ at a random point
Goal: A proof. I.e., a deterministic algorithm
Analogy with SAT

Input: Boolean circuit

Decide: is $C = 0$?

Note: SAT does not have randomized algorithms
Black Box PIT $\equiv$ Explicit Hitting Set

Input: A Black-Box circuit computing $f$.

Problem: Is $f \equiv 0$?

$S,Z,DM-L$: Evaluate at a random point

Goal: deterministic algorithm (a.k.a. Hitting Set):
find explicit set $H$: if $f \neq 0$, $\exists a \in H$ with $f(a) \neq 0$
Motivation

• Natural and fundamental problem
• Strong connection to circuit lower bounds
• Algorithmic importance:
  – Primality testing [Agrawal-Kayal-Saxena]
  – Parallel algorithms for finding matching [Karp-Upfal-Wigderson, Mulmuley-Vazirani-Vazirani]
• May help you solve exams!
Talk Overview

✓ Definition of the problem
  • Connection to lower bounds (hardness)
  • Survey of positive results
  • Some proofs
  • Open problems
Hardness: PIT $\equiv$ lower bounds

[Heintz-Schnorr,Agrawal]:
Polynomial time **Black-Box** PIT $\Rightarrow$
Exponential lower bounds for arith. Circuits

[Kabanets-Impagliazzo]:
- **Exponential** lower bound for Permanent $\Rightarrow$
  **Black-Box** PIT in $n^{\text{polylog}(n)}$ time
- **Polynomial** time **White-Box** PIT $\Rightarrow$
  (roughly) super-polynomial lower bounds.

[Dvir-S-Yehudayoff]: (almost) same as K-I for bounded depth circuits

**Lesson**: Derandomizing PIT essentially equivalent to proving lower bounds for arithmetic circuits
Black-Box PIT ⇒ Lower Bounds

[Heintz-Schnorr, Agrawal]:
BB PIT for size $s$ circuits in time $\text{poly}(s)$
(i.e. $\text{poly}(s)$ size hitting set)
⇒ exp. lower bounds for arithmetic circuits.

Proof: Given $\mathcal{H} = \{p_i\}$, find non-zero polynomial $f$ in
$log(|\mathcal{H}|)$ variables, such that $f(p_i) = 0$ for all $i$.

⇒ $f$ does not have size $s$ circuits

Gives lower bounds for $f$ in $\text{EXP (PSPACE)}$

Conjecture [Agrawal]:
$\mathcal{H} = \{(y_1, \ldots, y_n) : y_i = y_{ki \mod r}, k, r < s^{20}\}$ is a hitting set for size $s$ circuits
A short digression

Bounded Depth Circuits
Bounded depth circuits: $\Sigma \Pi$

- $\Sigma \Pi$ circuits: depth-2 circuits with $+$ at the top and $\times$ at the bottom. Size $s$ circuits compute $s$-sparse polynomials.

Example:

$$(-e)x_1 \cdot x_n + 2x_1 \cdot x_2 \cdot x_7 + 5(x_n)^2$$
Bounded depth circuits: $\Sigma\Pi\Sigma$

- $\Sigma\Pi\Sigma$ circuits: + at the top, $\times$ at the middle and + at the bottom: computes sums of products of linear functions.

Example:

$$(-e) \cdot (-2x_1 + x_n) \cdot (x_1 + \pi x_2 + \frac{1}{4} x_7) + \ldots$$
Bounded depth circuits: $\Sigma \Pi \Sigma$

- $\Sigma \Pi \Sigma$ circuits: $\times$ at the top, $\times$ at the middle and $+$ at the bottom: computes sums of products of linear functions.

Another Exam(ple):

\[
\prod_{i=1}^{n} \left( \omega^5 \pi X + \left( \omega^5 e - \omega^i \hbar \right) Y - \omega^i \pi e Z \right) + \\
\prod_{i=1}^{n} \left( -e \omega^i X + (\pi \omega^i + \hbar) Y + (\pi e - \hbar \omega^i) Z \right) + \\
\prod_{i=1}^{n} \left( (e \omega^2 - \pi \omega^i) X - (\pi \omega^2 + e \omega^i) Y + \hbar \omega^2 Z \right)
\]
Bounded depth circuits: $\Sigma\Pi\Sigma\Pi$

- $\Sigma\Pi\Sigma\Pi$ circuits: depth-4 circuits with $+$ at the top, then $\times$, then $+$ and another $\times$ at the bottom. Compute sums of products of sparse polynomials.

\[
\begin{align*}
\Sigma\Pi\Sigma\Pi & : \\
& \text{depth-4 circuits with } + \text{ at the top, then } \times, \text{ then } + \text{ and another } \times \text{ at the bottom. Compute sums of products of sparse polynomials.}
\end{align*}
\]
Back to Hardness-Randomness
Importance of $\Sigma\Pi\Sigma\Pi$ circuits

[Agrawal-Vinay,Raz]: Exponential lower bounds for $\Sigma\Pi\Sigma\Pi$ circuits imply exponential lower bounds for general circuits.

Cor [Agrawal-Vinay]: Polynomial time PIT of $\Sigma\Pi\Sigma\Pi$ circuits gives quasi-polynomial time PIT for general circuits.

Proof: By [Heintz-Schnorr,Agrawal] polynomial time PIT $\Rightarrow$ exponential lower bounds for $\Sigma\Pi\Sigma\Pi$ circuits. By [Agrawal-Vinay, Raz] $\Rightarrow$ exponential lower bounds for general circuits. Now use [Kabanets-Impagliazzo].
Importance of $\Sigma \Pi \Sigma \Pi$ circuits

[Agrawal-Vinay, Raz]: Exponential lower bounds for $\Sigma \Pi \Sigma \Pi$ circuits imply exponential lower bounds for general circuits.

Cor [Agrawal-Vinay]: Polynomial time PIT of $\Sigma \Pi \Sigma \Pi$ circuits gives quasi-polynomial time PIT for general circuits.

Lesson: Understanding small depth (i.e. depth 4) circuits is as important as the general case!
Talk Overview

✓ Definition of the problem
✓ Connection to lower bounds (hardness)
  • Survey of positive results
  • Some proofs
  • Open problems
Deterministic algorithms for PIT

- $\Sigma\Pi$ circuits $[\text{BenOr-Tiwari, Grigoriev-Karpinski, Klivans-Spielman},...]$  
  - Black-Box in polynomial time

- Non-commutative formulas $[\text{Raz-S}]$  
  - White-Box in polynomial time

- $\Sigma\Pi\Sigma\Sigma(k)$ circuits $[\text{Dvir-S, Kayal-Saxena, Karnin-S, Kayal-Saraf, Saxena-Seshadri}]$  
  - Black-Box in time $n^{O(k)}$

- Mult. $\Sigma\Pi\Sigma\Pi(k)$ $[\text{Karnin-Mukhopadhyay-S-Volkovich, Saraf-Volkovich}]$  
  - Black-Box in time $n^{\text{poly}(k)}$

- Read-k multilinear formulas $[\text{S-Volkovich, Anderson-van Melkebeek-Volkovich}]$  
  - White-Box in time $n^{kO(k)}$
  - Black-Box in $n^{O(\log(n)+kO(k))}$
Talk Overview

✓ Definition of the problem
✓ Connection to lower bounds (hardness)
✓ Survey of positive results
  • Some proofs:
    – Depth-3 circuits
  • Open problems
Solution to Exam (ΣΠΣ circuit)

$\omega^n = 1$. Is the following polynomial identically 0?

$$\prod_{i=1}^{n} \left( \omega^5 \pi X + (\omega^5 e - \omega^i \hbar) Y - \omega^i \pi e Z \right) +$$

$$\prod_{i=1}^{n} \left( -e\omega^i X + (\pi\omega^i + \hbar) Y + (\pi e - \hbar \omega^i) Z \right) +$$

$$\prod_{i=1}^{n} \left( (e\omega^2 - \pi\omega^i) X - (\pi\omega^2 + e\omega^i) Y + \hbar \omega^2 Z \right)$$

Prove it!

Will do so later — now
Idea: change of basis

\[ A = \pi \cdot X + e \cdot Y \]
\[ B = \hbar \cdot X + \pi \cdot e \cdot Z \]
\[ C = e \cdot X - \pi \cdot Y + \hbar \cdot Z \]

• Identity becomes

\[
\prod_{i=1}^{n} (A - \omega^i B) + \prod_{i=1}^{n} (B - \omega^i C) + \prod_{i=1}^{n} (C - \omega^i A) = \\
(A^n - B^n) + (B^n - C^n) + (C^n - A^n) = 0
\]

• But surely, this is not the general case. Right?
Depth 3 identities

• What is the structure of a zero circuit?
• If $M_1 + \ldots + M_k = 0$ then
  – Multiplying by a common factor:
    $$\prod x_i \cdot M_1 + \ldots + \prod x_i \cdot M_k = 0$$
  – Adding two identities:
    $$(M_1 + \ldots + M_k) + (T_1 + \ldots + T_{k'}) = 0$$
• How do the most basic identities look like?
• Basic: cannot be `broken’ to pieces (minimal) and no common linear factors (simple).
Depth 3 identities

- \( C = M_1 + ... + M_k \) \( M_i = \prod_{j=1}^{d_i} L_{i,j} \)
- **Rank**: dimension of space spanned by \( \{L_{i,j}\} \)
- In the exam: Rank=3
- Turns out: this is (almost) the general case!
- **Theorem [Dvir S]**: If \( C \equiv 0 \) is a basic identity then \( \dim(C) \leq \text{Rank}(k,d) = \left(\log(d)\right)^k \)
- **White-Box Algorithm**: find partition to sub-circuits of low dimension (after removal of g.c.d.) and brute force verify that they vanish.
- **Improved \( n^{O(k)} \)** algorithm by [Kayal-Saxen].
- **Black-Box**: Similar ideas...
Depth 3 identities

• **Lesson 1**: depth 3 identities are very structured!
• **Lesson 2**: Rank is an important invariant to study.
• **Improvements** [Kayal-Saraf,Saxena-Seshadri]:
  – finite $\mathbb{F}$, $k \cdot \log(d) < \text{Rank}(k,d) < k^3 \cdot \log(d)$
  – over $\mathbb{Q}$, $k < \text{Rank}(k,d) < k^2 \cdot \log(k)$
• Improves [Dvir-S] + [Karnin-S] (plug and play)
• [Saxena-Seshadri] BB-PIT in time $n^{O(k)}$

**Open problem**: remove $k$ from the exponent!
Bounding the rank

Basic observation: Consider $C = M_1 + M_2$

$$M_1 = \begin{bmatrix} L_1 & L_2 & \cdots & L_i & \cdots & L_j & \cdots & L_d \end{bmatrix}$$

$$M_2 = \begin{bmatrix} L'_1 & L'_2 & \cdots & L'_i & \cdots & L'_j & \cdots & L'_d \end{bmatrix}$$

Fact: linear functions are irreducible polynomial.

Corollary: $C \equiv 0$ then $M_1, M_2$ have same factors.

Corollary: $\exists$ matching $i \rightarrow \pi(i)$ s.t. $L_i \sim L'_{\pi(i)}$
Bounding the rank

• Claim: $\text{Rank}(3,d) = O(\log(d))$

Sketch: cover all linear functions in $\log(d)$ steps, where at $m$’th step:
• $\text{dim}$ of cover is $O(m)$
• $\Omega(2^m)$ functions in span
Bounding the rank

• Claim: $\text{Rank}(3,d) = O(\log(d))$

Sketch: cover all linear functions in $\log(d)$ steps, where at $m$'th step:
• $\text{dim}$ of cover is $O(m)$
• $\Omega(2^m)$ functions in span
“Geometric interpretation of $M_1 + M_2 + M_3 = 0$”

- Lets map Linear forms to points in $\mathbb{R}^n$
- The map

$$a_1 x_1 + a_2 x_2 + \ldots + a_n x_n \mapsto (1, \frac{a_2}{a_1}, \ldots, \frac{a_n}{a_1})$$

- Say $L_1 \rightarrow P_1$, $L_2 \rightarrow P_2$, $L_3 \rightarrow P_3$
  - If $L_3 = \alpha L_1 + \beta L_2$ then $P_3$ lies on the line through $P_1$ and $P_2$
$$M_1 + M_2 + M_3 = 0 \rightarrow \text{colored points}$$
$M_1 + M_2 + M_3 = 0 \rightarrow \text{colored points}$

**Question:** If every line containing points of two colors also includes the third, must the points sit in low-dimensional space?
The Sylvester-Gallai Theorem

- **Sylvester-Gallai Theorem:**
  Given a finite set of points $S$ in the plane.
  - $\exists$ line $L$ intersecting exactly two points of $S$
  - or all points in $S$ are collinear

Not good enough!
$L$ may contain only red points
Edelstein-Kelly Theorem

- \textbf{[Edelstein-Kelly 66]}: Let P be a set of points with the following properties:
  - Every point is assigned one of three colors – either \textcolor{red}{Red} or \textcolor{blue}{Blue} or \textcolor{green}{Green}
  - The points span a space of $\leq 4$ dimensions
  - Then there exists a line containing points of exactly 2 distinct colors from P

- Theorem: $\text{Rank}(3,d) \leq 4$ over $\mathbb{R}$

- For $\text{Rank}(k,d)$ generalizations for higher dimensions are used
Summary of depth-3

• Depth-3 important subcase before the general case of $\Sigma\Pi\Sigma\Pi$ circuits

• Demonstrated structure in depth-3 identities that led to beautiful mathematics
  – High dimensional colored versions of Sylvester-Gallai theorem
  – Extensions to finite fields

• Didn’t see it but
  – Problem related to low-rank-recovery in signal processing
  – Reconstruction of depth-3 circuits
Talk Overview

✓ Definition of the problem
✓ Connection to lower bounds (hardness)
✓ Survey of positive results
✓ Some proofs:
  ✓ Depth-3 circuits

• Open problems
Open problems

• Improve PIT of depth-3 circuits
  – e.g. to $f(k) \cdot \text{poly}(n)$

• Give PIT algorithm to $\Sigma \Pi \Sigma \Pi(k)$ circuits
  – Even $n^{f(k)}$ white-box algorithm will be great
  – Related to open problems on factorization of sparse polynomials

• PIT for tensors
  – Special case of depth-3 circuits
  – Related to **Low-Rank-Recovery** in signal processing

• Use PIT to **reconstruct** arithmetic circuits
Thank You!