Derandomization via Robust Algebraic Circuit Lower Bounds

Michael Forbes

Princeton University

based on various work with Amir Shpilka

(I am on the job market this year)

September 29, 2015

Theme

Question

*When do lower bounds for a circuit class $\mathcal{C}$ yield efficient **deterministic** algorithms for deciding properties of circuits from $\mathcal{C}$?*

Motivations

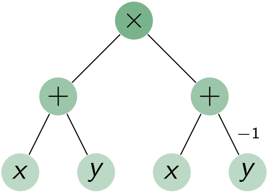
- *Derandomize algorithms*
- *$\mathcal{C}$ may be too weak for derandomization via hardness versus randomness paradigm*
- *Better understanding*

Polynomial Identity Testing (PIT)

Question (Polynomial Identity Testing (PIT))

Given a polynomial $f \in \mathbb{Q}[x_1, \ldots, x_n]$, is $f \not\equiv 0$?

Polynomials are given by a small computational device, e.g.
$x^2 - y^2 = (x + y)(x - y)$ can be given by



Polynomial Identity Testing (ii)

Lemma (Schwartz-Zippel)

$f \not\equiv 0$ iff $f(\vec{\alpha}) \neq 0$ for random $\vec{\alpha}$.

- Gives a randomized algorithm for testing if $f \not\equiv 0$, only uses f as a **black-box**. Deterministic algorithms?
- **hitting set** ($\equiv$ **black-box PIT**): set of points $\mathcal{H} \subseteq \mathbb{Q}^n$ such that

$$f \not\equiv 0 \text{ iff } f|_{\mathcal{H}} \not\equiv 0, \text{ for computationally simple } f.$$

- Non-constructively $|\mathcal{H}| =$ small, constructively?

Depth-3 Powering Formulas

Algebraic formulas typically use addition ($\sum$) and multiplication ($\prod$), but we can also use addition ($\sum$) and **powering** ($\bigwedge$)

$$xy = \frac{1}{4}\left((x+y)^2 - (x-y)^2\right) \ ,$$

Have equivalence for arbitrary formulas, but not for low-depth.
A **depth-3 powering formula** ($\sum \bigwedge \sum$) is a sum of powers of linear forms

$$f(x_1, \ldots, x_n) = \sum_{i=1}^{s} (\alpha_{i,0} + \alpha_{i,1}x_1 + \cdots + \alpha_{i,n}x_n)^{d_i} \ , \qquad \alpha_{i,j} \in \mathbb{Q} \ .$$

$\sum \bigwedge \sum$ are a moral analogue of CNFs/DNFs from boolean complexity.

Lower Bounds for Depth-3 Powering Formulas

Theorem (NisanWigderson96,Kayal08,Sylvester1851)

The monomial $x_1 \cdots x_n$ requires $2^{\Omega(n)}$ size to be computed as a $\sum \bigwedge \sum$ formula.

Theorem (F-Shpilka13)

If $f(x_1, \ldots, x_n) = \sum_{\vec{a}} \alpha_{\vec{a}} x_1^{a_1} \cdots x_n^{a_n}$ is computed by a size-s $\sum \bigwedge \sum$ formula, then f is determined by its restriction to monomials involving $O(\log s)$ variables. This implies a $\mathrm{poly}(n, s)^{O(\lg s)}$ size hitting set.

This is proven by making the above lower bound **robust**.

Lower Bounds for Depth-3 Powering Formulas (ii)

The lower bound follows from a *complexity measure* argument. For $f \in \mathbb{Q}[x_1, \ldots, x_n]$ define

$$\mu(f) := \dim \operatorname{span} \left\{ \frac{\partial}{\partial_{x_1}^{a_1} \cdots \partial_{x_n}^{a_n}} f \right\}_{a_1, \ldots, a_n \geq 0}$$

facts:

- $\mu(f + g) \leq \mu(f) + \mu(g)$.
- $\mu\left((\alpha_0 + \alpha_1 x_1 + \cdots + \alpha_n x_n)^d\right) \leq d + 1$.
- $\mu(x_1 \cdots x_n) = 2^n$.

$\Longrightarrow$ $x_1 \cdots x_n$ needs size $2^{\Omega(n)}$ to be computed as

$$x_1 \cdots x_n = \sum_{i=1}^{s} (\alpha_{i,0} + \alpha_{i,1} x_1 + \cdots + \alpha_{i,n} x_n)^{d_i} \ ,$$

if $d_i \leq \operatorname{poly}(n)$.

Lower Bounds for Depth-3 Powering Formulas (iii)

lower bound: small $\sum \bigwedge \sum$ formula cannot *exactly* equal large monomials. Approximately?

$$(x_1 + \cdots + x_n)^n = x_1 \cdots x_n + \cdots + x_1^n + \cdots + x_n^n + \cdots \ .$$

Express $f \neq 0$ as

$$f = \alpha x_1^{a_1} \cdots x_n^{a_n} + \text{lower order terms} \ ,$$

where monomials are ordered lexicographically with $x_1 \succ \cdots \succ x_n$

fact: $\mu(f) \geq \mu(x_1^{a_1} \cdots x_n^{a_n})$ — the measure μ is robust

$\implies$ the leading monomial of a small $\sum \bigwedge \sum$ formula involves few variables [**F**-Shpilka13]

$\implies$ quasipolynomial time deterministic blackbox PIT for $\sum \bigwedge \sum$

Conclusions

Robust Lower Bound:

$$\mu(\text{extrema}(f) + \text{lower terms}) \geq \mu(\text{extrema}(f)) \; .$$

Other PIT via Robust Lower Bounds:

- [SV09]: Read-Once Formula
- [**F**S12]: (commutative) read-once algebraic branching programs
- [MRS14,**F**15]: sums of powers of low-degree polynomials
- [GKST15,**F**15]: sparse polynomials

Open questions:

- polynomial-size hitting set for $\sum \bigwedge \sum$ formula? best known is $s^{\mathcal{O}(\lg \lg s)}$ for size s [**F**SS14]

TOC