CS 388T: Advanced Complexity Theory Spring 2020

AC^0 lower bounds for PARITY

Prof. Dana Moshkovitz *Scribe: Maruth Goyal*

1 Overview

Complexity theorists have been trying to attack the question of whether $\mathsf{NP} \not\subseteq \mathsf{P/poly}$ after shifting their attention to circuits instead of Turing Machines. However, we are (as of writing) no where close to showing this. Nonetheless, all hope is not lost. *Some* progress has been made in showing separations of classical complexity classes and circuit complexity classes. Some of them are "embarassing", such as Ryan Williams' result that $\mathsf{NEXP} \not\subseteq \mathsf{ACC}^0$ [Wil14]. However, even that is a very non-trivial result. Today we shall look at such a separation for P and AC^0. In particular, we will see that PARITY $\notin \mathsf{AC}^0$.

There are several proofs of this fact. Ajtai [A$^+$83], and independently Furst, Saxe, and Sipser [FSS84] first showed this result. Later, Yao, and Hastad [Has86] improved upon these results. These proofs generally use variants of "switching lemmas". However, we present a proof credit to Razborov [Raz87], and Smolensky [Smo87] based on polynomials.

Very roughly, the proof proceeds by constructing low-degree polynomial approximations for all circuits in AC^0. We then show that PARITY cannot be computed by any low-degree polynomial on even 51% inputs to complete the proof.

Note: *This proof is in fact more general, and extends the lower bound to $\mathsf{AC}^0[p]$, i.e. circuits with mod-p gates. These notes are based on those by Anup Rao [Rao12].*

2 Preliminaries

Before diving into the actual proof, we shall first establish some results regarding polynomials over finite fields. In particular, we will consider $\mathbb{F}_p$ for $p \neq 2$, where p is prime.

Theorem 1. *Any function $f : \mathbb{F}_p^n \to \mathbb{F}_p$, can be computed by a unique polynomial $q \in \mathcal{P}(\mathbb{F}_p^n)$ of degree at most $p-1$.*

Proof. We first show that such a polynomial exists. First, for any $a \in \mathbb{F}_p^n$ we define the function $\mathbb{1}_a$ as follows:

$$\mathbb{1}_a(X) = \prod_{i=1}^{n} \prod_{z_j \neq a_i} \frac{X_i - z_j}{a_i - z_j}$$

Here, $z_j \in \mathbb{F}_p$. Observe that if $X = a$, then $X_i - z_i = a_i - z_i$. Hence, the inner term just becomes 1 for all i. Thus, $\mathbb{1}_a(a) = 1$. On the other hand, if $X \neq a$, Then for each i, there is exactly one z_j such that $X_i = z_j$ since $X_i \neq a_i$. Thus, the inner term becomes 0, and $\mathbb{1}_a(X) = 0$. i.e.,

$$\mathbb{1}_a(X) = \begin{cases} 1 & X = a \\ 0 & \text{otherwise} \end{cases}$$

Moreover, observe that there are exactly $p-1$ values of z such that $z \neq a_i$. Thus the inner term is a product of $p-1$ terms in X_i, and hence has degree $p-1$ in each X_i. We will now use this to construct a degree $p-1$ polynomial for f. Define q such that

$$q(x) = \sum_{a \in \mathbb{F}_p^n} f(a) \cdot \mathbb{1}_a(x)$$

Notice if $x = b$ then only the term $f(b) \cdot \mathbb{1}_b(b)$ survives, and all other terms become 0 by definition. Thus q exactly computes f.

To show that this polynomial is unique observe that $\mathcal{P}(\mathbb{F}_p^n)$ is a vector space. Moreover, it is spanned by all the monomials. For each of the n co-ordinates, there are p possible monomials $x^0, \ldots, x^{p-1}$ (ignoring coefficients because basis). Thus, there are p^n spanning monomials. Now observe that for each of these p^n monomials there are p choices of coefficients, and hence we get that there are p^{p^n} total such polynomials. Also observe that the number of functions $f : \mathbb{F}_p^n \to \mathbb{F}_p$ is precisely p^{p^n} since each of the p^n values in the domain can be assigned to any of p values in the co-domain. Thus the mapping must be bijective. □

We state the following without proof

Theorem 2. $\binom{n}{i}$ *is maximized when* $i = n/2$*, and in that case is at most* $O(2^n/\sqrt{n})$.

3 Low-degree polynomial approximations of AC^0

Let's say we are given some circuit $\mathcal{C} \in \mathsf{AC}^0$. We want to find a low-degree polynomial which computes $\mathcal{C}$ with high accuracy. While this works for any finite field, we will work with polynomials over $\mathbb{F}_3$.

Observe that it is sufficient to be able to represent negation ($\neg$) and disjunction ($\vee$) using low-degree polynomials, as the rest follows by De Morgan's laws. Note we can represent the initial wires for the variables with just corresponding variables $x_i \in \{0, 1\}$. Now, if f_i is any wire in the circuit, we can compute its negation $\neg f_i$ simply by $1 - f_i$. Computing $f_1 \vee \cdots \vee f_k$ is slightly trickier.

Let's start with a simple guess. How about $1 - \prod_{i=1}^{k}(1 - f_i)$. Notice this corresponds to $\neg(\neg f_1 \wedge \cdots \wedge \neg f_k)$. Hence, this definitely exactly computes what we want it to, so what's wrong with it? Well the degree of this polynomial is essentially the fan-in of the gate, which is unbounded in AC^0. Let's try and refine this. Remember, we want the function to be 0 iff all f_i are 0, and 1 when at least one is 1.

First observe that $1^2 \equiv 2^2 \equiv 1 \pmod 3$. Thus, for all $x \neq 0$ in $\mathbb{F}_3$, it is the case that $x^2 = 1$. So what if we consider $(\sum f_i)^2$? Well, this works as long as the number of f_i which are 1 are not multiples of 3. That's no good. But, if there is at least one f_i with value 1, then definitely there is some **subset** where the number of 1s is not a multiple of 3 right?

Ok, so let's independently pick, say, ℓ uniformly random subsets $T_1, \ldots, T_\ell \subseteq [k]$. Then, intuitively, at least one of these should have $(\sum_{i \in T_j} f_i)^2 = 1$. So let's consider

$$1 - \prod_{j=1}^{\ell}(1 - (\sum_{i \in T_j} f_i)^2)$$

If any one of the T_j satisfies $(\sum_{i \in T_j} f_i)^2 = 1$, the product collapses to 0, and the entire thing becomes 1. Similarly if all $f_i = 0$, the product will be 1, and the output will be 1, as desired. Awesome, right?

Well, we are not done yet. We still need to analyze exactly how likely is it that $(\sum_{i \in T_j} f_i)^2 = 1$ happens. i.e., we want to know $\mathbb{P}[\sum_{i \in T} f_i \neq 0]$. We make the following claim:

Claim 3. *If at least one of $f_1, \ldots, f_k$ is non-zero, then for any $T \subseteq [k]$, $\mathbb{P}[\sum_{i \in T} f_i \neq 0] \geq 1/2$.*

Proof. WLOG suppose $f_j \neq 0$. Then, consider all the subsets $S \subseteq [k] \setminus j$. There are 2^{k-1} of these. Worst case, $\sum_{i \in S} f_i = 0$ for all S. i.e., literally everything else is 0. Thus, at most 2^{k-1} of the 2^k possible subsets of 2^k will have sum exactly 0. Hence, $\mathbb{P}[\sum_{i \in T} f_i = 0] \leq 1/2$. □

Equivalently, $\mathbb{P}[\sum_{i \in T} f_i = 0] \leq 1/2$. Therefore, the probability that our polynomial is wrong is at most $2^{-\ell}$. Great! If f_i has degree at most r, then our polynomial has degree at most $(2r) \cdot \ell$, and is correct with probability $1 - 2^{-\ell}$.

Now, if our circuit has depth h, then notice that the degree of the final gate is at most $(2\ell)^h$. Essentially the input wires have degree at most 1, so the gates at depth 1 have degree at most 2ℓ, and so depth 2 gates have degree at most $(2 \cdot \underbrace{2\ell}_{r}) \cdot l = (2\ell)^2$, ... and so on. Let's say the circuit has size s, and pick $\ell = \log^2 n$. Then, we have that the degree of the approximating polynomial is at most $(2 \log^2 n)^h \equiv \mathsf{polylog}(n)$. Furthermore, since each gate has error probability at most $2^{-\ell}$, the probability that at least one gate is erroneous is at most $s2^{-\ell}$ by a union bound. Thus, our polynomial is correct with probability $1 - s2^{-\ell}$. For our choice of parameters, one may observe that this is about a 1% error rate.

4 PARITY $\notin \mathsf{AC}^0$

We are now ready to prove the full result. We do this by showing any polynomial of degree d can compute PARITY correctly on at most $1/2 + O(d/\sqrt{n})$ fraction of inputs.

Claim 4. *If f is any polynomial over $\mathbb{F}_3$ in n variables with degree d. Then it can compute* PARITY *on at most $1/2 + O(d/\sqrt{n})$ fraction of inputs.*

Proof. Suppose f computes PARITY on n bits. Then we will define the polynomial

$$g(y_1, \ldots, y_n) = f(y_1 - 1, \ldots, y_n - 1) + 1$$

Here's the trick: we let $y_i \in \{1, -1\}$. Why? Observe that $1 - 1 = 0$, and $-1 - 1 \equiv -2 \equiv 1 \pmod 3$. Now consider $\prod y_i$. This is 1 if and only if there are even number of -1s. f is 1 iff there are an odd number of 1s, and we've mapped all -1s to 1s in g. So, if the number of -1s is even, then the number of 1s is even for f, and hence f evaluates to 0, and thus g evaluates to 1. TL;DR g is exactly computing $\prod y_i$ when $y_i \in \{1, -1\}$. So, we have expressed a degree n polynomial with a degree d polynomial f.

We now complete the proof by showing that $g(y_1, \ldots, y_n) = \prod y_i$ on only a certain fraction of inputs. Let $T \subseteq \{1, -1\}^n$ be the subset of inputs on which $g(y_1, \ldots, y_n) = \prod y_i$.

Then, consider the set of all functions $q : T \to \mathbb{F}_3$. This has dimension $|T|$. From theorem 2, every q can be computed by a polynomial. Moreover, since $y_i \in \{1, -1\}$ it is always the case that $y_i^2 = 1$. So for functions q, we can assume each variable has degree 1. Consider any term (without its coefficients) corresponding to a subset of indices $I \subseteq [n]$, with $|I| \geq n/2$ as $\prod_{i \in I} y_i$. Then,

$$\prod_{i \in I} y_i = (\prod_{i=1}^{n} y_i)(\prod_{i \notin I} y_i) = g(y)(\prod_{i \notin I} y_i)$$

The first equality holds because all terms that are not in I get killed off because $y_i^2 = 1$. More importantly, we have now expressed all monomials of q using terms of degree at most $n/2 + d$. So we now have a low-degree polynomial computing q.

Observe the polynomials with monomials of degree at most $n/2 + d$ is spanned by $\sum_{i=0}^{n/2+d} \binom{n}{i}$ monomials. Essentially, for each $i \leq n/2 + d$ there are $\binom{n}{i}$ ways to pick monomials of degree i. Finally,

$$|T| \leq \sum_{i=0}^{n/2+d} \binom{n}{i} \leq 2^n/2 + \sum_{i=n/2+1}^{n/2+d} \binom{n}{i} \leq 2^n/2 + O(d \cdot 2^n/\sqrt{n}) = 2^n(1/2 + O(d/\sqrt{n}))$$

The second inequality follows since $\sum_{i=0}^{n} \binom{n}{i} = 2^n$, and third follows from theorem 2 (d terms in the sum, each with value at most $O(2^n/\sqrt{n})$. Thus, f can compute PARITY on at most $(1/2 + O(d/\sqrt{n}))$ fraction of inputs. □

We showed that any AC^0 circuit can be represented by a polynomial of $\mathsf{polylog}(n)$ degree, with very small error. Thus any AC^0 circuit can compute PARITY on at most $(1/2 + O(\mathsf{polylog}(n)/\sqrt{n})) <$ 51% fraction of inputs, since $\log n = O(\sqrt{n})$.

References

[A+83] Miklós Ajtai et al. Sigma 11-formulae on finite structures. 1983.

[FSS84] Merrick Furst, James B Saxe, and Michael Sipser. Parity, circuits, and the polynomial-time hierarchy. *Mathematical systems theory*, 17(1):13–27, 1984.

[Has86] John Hastad. Almost optimal lower bounds for small depth circuits. In *Proceedings of the eighteenth annual ACM symposium on Theory of computing*, pages 6–20, 1986.

[Rao12] Anup Rao. Lecture 10 parity not in ac0 and introducing pcp. https://homes.cs.washington.edu/ anuprao/pubs/CSE531Winter12/lecture10.pdf, 2012.

[Raz87] Alexander A Razborov. Lower bounds on the size of bounded depth circuits over a complete basis with logical addition. *Mathematical Notes of the Academy of Sciences of the USSR*, 41(4):333–338, 1987.

[Smo87] Roman Smolensky. Algebraic methods in the theory of lower bounds for boolean circuit complexity. In *Proceedings of the nineteenth annual ACM symposium on Theory of computing*, pages 77–82, 1987.

[Wil14] Ryan Williams. Nonuniform acc circuit lower bounds. *Journal of the ACM (JACM)*, 61(1):1–32, 2014.