# Boolean function complexity

*Lecturer:* *Nitin Saurabh* Meeting: 8
*Scribe:* *Nitin Saurabh* 12.06.2019

In this lecture we will prove nearly tight lower bound on the size of a constant depth circuit computing parity of $n$ bits. Recall, by a *constant-depth* circuit we mean that the depth of the circuit is bounded by a universal constant, say, 3, 4, etc. The fan-in of each gate in a constant-depth circuit is allowed to be unbounded. We denote the class of functions computable by depth $d$ circuits of size at most $s$ by $\mathsf{AC}[s,d]$. Thus, the class of constant-depth circuits are denoted by $\mathsf{AC}[s,O(1)]$. There are two methods to prove lower bounds for constant-depth circuits, namely *polynomial method* and *Håstad's Switching Lemma*. In this lecture we see the polynomial method.

Let us consider the *parity* function, $\mathsf{Parity}_n\colon \{0,1\}^n \to \{0,1\}$. A depth 2 circuit is basically a CNF or DNF representation. Therefore, $\mathsf{Parity}_n$ requires $2^{n-1}$ clauses (or, terms) in a depth 2 representation. What is the size of depth 3 circuit computing $\mathsf{Parity}_n$?

**Depth 3:** Consider the following circuit construction: Construct a $\sqrt{n}$-ary tree with parity gates computing parity over $n$ variables (see Fig. 1). Represent the parity gate at the root with a DNF of size $2^{\sqrt{n}-1}$, and the bottom level parity gates as a CNF of size $2^{\sqrt{n}-1}$. Then, collapse the $\mathsf{OR}$ gates in the middle layers to get a maximum depth of 3 (see Fig. 1). It is easily seen that the size of the circuit thus constructed is $2^{O(\sqrt{n})}$.

**Depth d:** For a generic depth $d$, we can generalize the construction in Fig. 1. Construct a $k$-ary tree of depth $d-1$ where $k = n^{1/(d-1)}$. The internal nodes of the tree are labeled by parity gates. Now simulate parity gates in each level in the tree with brute-force DNF/CNF representation alternately so that one can collapse consecutive levels of $\mathsf{OR}$ gates (or, $\mathsf{AND}$ gates). Thus the total depth of the circuit thus constructed is at most $d$. We can also easily compute the size of the circuit to be $O(n2^{n^{1/(d-1)}})$.

Thus, we obtain the following theorem.

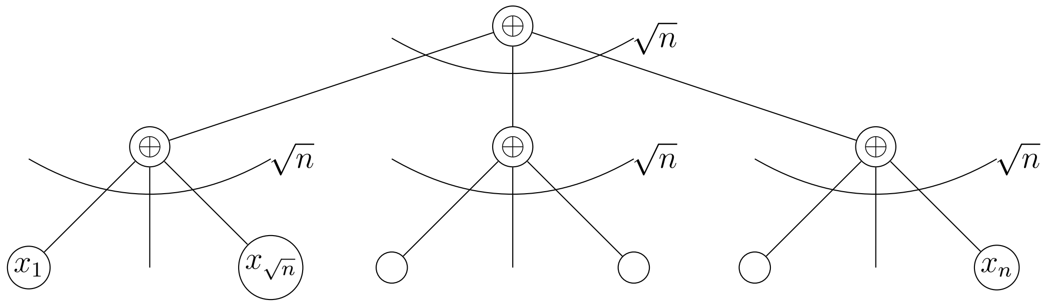
**Theorem 1.** *For every constant $d \geq 2$, there are circuits of size $O(n2^{n^{1/(d-1)}})$ and depth $d$ that computes $\mathsf{Parity}_n$.*

In this lecture we will prove a weaker bound of $\Omega(2^{n^{1/4d}})$ using polynomial methods.

# 1 Razborov-Smolensky's polynomial method

The idea behind this method is that circuits of small size and constant depth can be represented by low degree polynomials that errs at only a small fraction of points. To formalise this notion we need the following definitons.

**Definition 2** (probabilistic polynomials). *An $\epsilon$-error* probabilistic polynomial *of degree $d$ for a function $f\colon \{0,1\}^n \to \{0,1\}$ is a random polynomial $\mathbf{P}$ of degree $d$, chosen according*




$\sqrt{n}$-ary depth-2 circuit computing $\mathsf{Parity}_n$ using parity gates

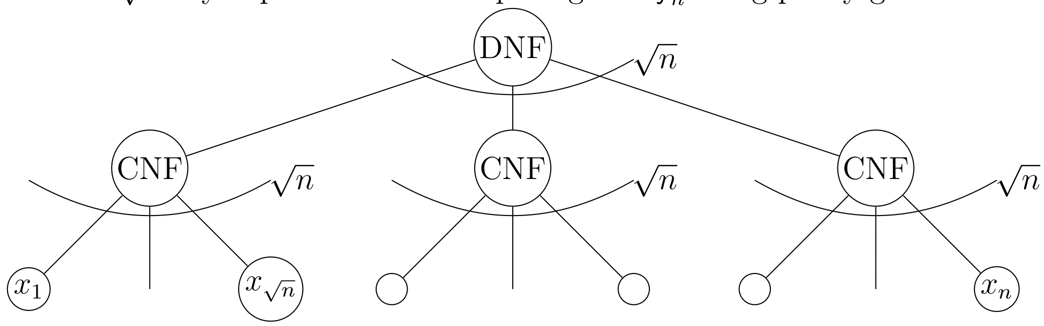



Figure 1: depth-3 circuit computing $\mathsf{Parity}_n$

*to some distribution $\mathcal{D}$ over polynomials of degree at most $d$, such that for any $x \in \{0,1\}^n$, we have*

$$\Pr_{\mathbf{P}\sim\mathcal{D}}[f(x) = \mathbf{P}(x)] \geq 1 - \epsilon.$$

**Definition 3** (approximating polynomials)**.** *An $\epsilon$-error* approximating polynomial *for a function $f\colon \{0,1\}^n \to \{0,1\}$ is a polynomial $p$ such that*

$$\Pr_{x\in\{0,1\}^n}[f(x) \neq p(x)] \leq \epsilon,$$

*where $x \in \{0,1\}^n$ is chosen uniformly at random.*

We now see that the existence of $\epsilon$-error probabilistic polynomials implies the existence of $\epsilon$-error approximating polynomial.

**Lemma 4.** *Let $f\colon \{0,1\}^n \to \{0,1\}$ be a Boolean function. Then, $\epsilon$-error probabilistic polynomials of degree $d$ for $f$ implies there exist an $\epsilon$-error approximating polynomial of degree $d$ for $f$.*

*Proof.* We prove it when $\mathcal{D}$, the distribution of the probabilistic polynomial, is uniform. For arbitrary distribution, it is left as an exercise.

Let $\mathrm{supp}(\mathcal{D})$ denote the set of polynomials with non-zero probability of being sampled. Consider the $\{0,1\}$-matrix $E$ of dimension $2^n \times |\mathrm{supp}(\mathcal{D})|$ where the rows are labeled by $x \in \{0,1\}^n$ and columns are labeled by polynomials $p$ in $\mathrm{supp}(\mathcal{D})$. An entry $E(x,p)$ in the matrix is 1 if $f(x) \neq p(x)$ and 0 otherwise. Since $\mathcal{D}$ is an $\epsilon$-error probabilistic polynomial with uniform distribution on its support, we have at most $\epsilon \cdot |\mathrm{supp}(\mathcal{D})|$ many ones in each row $x$. Therefore, the total number of ones in $E$ is at most $\epsilon \cdot |\mathrm{supp}(\mathcal{D})| \cdot 2^n$. Thus, there exists a column $p$ such that it has at most $\epsilon \cdot 2^n$ ones. The polynomial $p$ labeling the column is an $\epsilon$-error approximating polynomial of degree $d$ for $f$.

□

## 1.1 Approximating $\mathsf{OR}$

We know that the degree of a polynomial that exactly represents $\mathsf{OR}_n$ must be $n$. In this section we will see that if we are fine with making errors at a small fraction of points then we can bring down the degree substantially.

**Lemma 5.** *For all $n$ and $\epsilon > 0$, there exists an $\epsilon$-error probabilistic polynomial of degree $O((\log 1/\epsilon)\log n)$ for $\mathsf{OR}_n$.*

*Proof.* We want a random polynomial of low degree that computes OR on most of the inputs. Recall,

$$\mathsf{OR}_n(x) = 1 - \prod_{i=1}^{n}(1 - x_i).$$

The above expression checks if there exist any $x_i$ that is set to 1. To bring down the degree, the idea is to do few tests of random batches of variables. We now formalize this.

Let $m := \log n$, and consider $m+1$ random subsets $S_0, S_1, \ldots, S_m$ of $[n]$ where $S_i$ is defined as follows: independently for each $j \in [n]$ include it in $S_i$ with probability $2^{-i}$. For each $i$, $0 \leq i \leq m$, define $q_i(x) := \sum_{j \in S_i} x_j$. Now consider the random polynomial

$$q(x) = 1 - \prod_{i=0}^{m}(1 - q_i(x)).$$

Clearly its degree is $m+1$. Moreover, if $\mathsf{OR}_n(x) = 0$, then $q(x) = 0$. We now show that $q$ is correct with at least a constant probability when $\mathsf{OR}_n(x) = 1$.

**Claim 1.1.** *If $x \neq 0^n$, then* $\Pr[q(x) = 1] \geq 1/6$.

*Proof.* Let $w = \sum_{i=1}^{n} x_i$. Since $x \neq 0^n$, $1 \leq w \leq n$. Let $0 \leq k \leq m$ be such that $2^{k-1} < w \leq 2^k$. Clearly,

$$\Pr[q(x) = 1] \geq \Pr[q_k(x) = 1].$$

We now lower bound the $\Pr[q_k(x) = 1]$.

$$\begin{aligned}
\Pr[q_k(x) = 1] &= \Pr\left[\sum_{j \in S_k} x_j = 1\right] \\
&= w \cdot \frac{1}{2^k} \cdot \left(1 - \frac{1}{2^k}\right)^{w-1} \geq \frac{1}{2} \cdot \left(1 - \frac{1}{2^k}\right)^{2^k - 1} \geq \frac{1}{2e} > \frac{1}{6}.
\end{aligned}$$

The second inequality follows from $(1 + \frac{1}{n})^n < e < (1 + \frac{1}{n})^{n+1}$ for all positive $n$. □

Thus, the random polynomial $q(x)$ is correct with probability at least $1/6$. To reduce the error probability to $\epsilon$, we sample $r$ independent copies of $q$, say $p_1(x), \ldots, p_r(x)$, and consider the random polynomial

$$p(x) = 1 - \prod_{i=1}^{r}(1 - p_i(x)).$$

The probability the $p(x)$ errs is at most $(5/6)^r$ and the degree of $p$ is at most $r(\log n + 1)$. Choosing $r$ such that $(5/6)^r \leq \epsilon$, gives us a probabilistic polynomial $p$ of degree $O((\log 1/\epsilon)\log n)$ for $\mathsf{OR}_n$. $\square$

Note that from the above lemma we also obtain an $\epsilon$-error probabilistic polynomial of degree $O((\log 1/\epsilon)\log n)$ for $\mathsf{AND}_n$. Using the probabilistic polynomial for both $\mathsf{OR}$ and $\mathsf{AND}$ we now obtain an approximating polynomial for the whole circuit.

**Theorem 6.** *For every circuit $C$ of size $s$ and depth $d$, there exists an approximating polynomial $p$ of degree $O((\log s)^{2d})$ such that*

$$\Pr_x[C(x) \neq p(x)] \leq \frac{1}{4}.$$

*Proof.* Using Lemma 5 we construct a $\frac{1}{4s}$-error probabilistic polynomial for each gate in the circuit. We compose probabilistic polynomials of all gate to obtain a probabilistic polynomial $\mathbf{P}$ for the circuit. Clearly, then for any input $x$,

$$\begin{aligned}\Pr[\mathbf{P}(x) \neq C(x)] &\leq \text{ prob. that polynomial representing some gate is wrong} \\ &\leq \frac{1}{4s} \cdot s = \frac{1}{4}\end{aligned}$$

Now using Lemma 4 we obtain an $\frac{1}{4}$-approximating polynomial of degree $O((\log s)^{2d})$ for $C$. $\square$

## 1.2 Parity requires large degree for approximation

In this section we show that if a polynomial represents parity at most of the inputs then its degree must be at least $\sqrt{n}$.

**Theorem 7.** *Let $p(x)$ be a polynomial of degree $d$ such that $\Pr_x[p(x) = \mathsf{Parity}_n(x)] \geq 3/4$. Then, $d = \Omega(\sqrt{n})$.*

*Proof.* We change from $\{0, 1\}$ basis to Fourier $\{-1, 1\}$ basis. Therefore, define

$$q(x) = 1 - 2p\left(\frac{1 - x_1}{2}, \frac{1 - x_2}{2}, \ldots, \frac{1 - x_n}{2}\right).$$

Clearly, $\deg(q) \leq d$ and $\Pr[q(x) = \prod_{i=1}^{n} x_i] \geq 3/4$. Recall $\prod_i x_i$ is the representation of parity in the Fourier basis.

Let $A := \{x \in \{-1,1\}^n \mid q(x) = \prod_{i=1}^n x_i\}$. Then $|A| \geq (3/4)2^n$. Consider the set of all functions $f\colon A \to \mathbb{R}$. It is easily seen to be a vector space of dimension $|A|$ over $\mathbb{R}$. Thus, any $f$ can be represented by a polynomial $\sum_{S\subseteq[n]} c_S \prod_{i\in S} x_i$. We now claim the following upper bound on the degree of a polynomial representation of any $f\colon A \to \mathbb{R}$.

**Claim 1.2.** *Any $f\colon A \to \mathbb{R}$ can be represented by a polynomial of degree at most $d + (n/2)$.*

*Proof.* For all $x \in \{-1,1\}^n$ and $S \subseteq [n]$, $\prod_{i\in S} x_i = (\prod_{i=1}^n x_i) \cdot \left(\prod_{i\notin S} x_i\right)$. Therefore, for all $x \in A$, $\prod_{i\in S} x_i = q(x) \cdot \left(\prod_{i\notin S} x_i\right)$. In a polynomial representation of $f\colon A \to \mathbb{R}$ replace all monomials $\prod_{i\in T} x_i$ of degree $> n/2$ by $q(x) \cdot \left(\prod_{i\notin T} x_i\right)$. Thus, all monomials in the representation will have degree at most $d + (n/2)$. □

Since this is true for every $f$, the dimension of the vector space of all functions $f\colon A \to \mathbb{R}$ is upper bounded by $\sum_{k=0}^{d+(n/2)} \binom{n}{k}$. Therefore, we have

$$\frac{3}{4} \cdot 2^n \leq |A| \leq \sum_{k=0}^{d+(n/2)} \binom{n}{k} \leq \sum_{k=0}^{n/2} \binom{n}{k} + \sum_{k=n/2}^{d+(n/2)} \binom{n}{k}.$$

Thus,

$$\sum_{k=n/2}^{d+(n/2)} \binom{n}{k} \geq \frac{1}{4} \cdot 2^n.$$

This implies

$$d \cdot \binom{n}{n/2} \geq \frac{1}{4} \cdot 2^n.$$

Hence we obtain using Stirling's approximation,

$$d = \Omega(\sqrt{n}).$$

□

## 2 Parity $\notin$ AC$^0$

**Theorem 8** (Razborov'87, Smolensky'87)**.** *For every constant $d \geq 2$, if $C$ is a circuit of size $s$ and depth $d$ that computes* $\mathsf{Parity}_n$*, then $s \geq 2^{\Omega(n^{1/4d})}$.*

*Proof.* From Theorems 6 and 7, we have $(\log s)^{2d} = \Omega(\sqrt{n})$. Hence the theorem follows. □