

More on $\text{AC}^0[\oplus]$ and Variants of the Majority Function

Nutan Limaye

Department of Computer Science and Engineering, IIT Bombay, India
nutan@cse.iitb.ac.in

Srikanth Srinivasan

Department of Mathematics, IIT Bombay, India
srikanth@math.iitb.ac.in

Utkarsh Tripathi

Department of Mathematics, IIT Bombay, India
utkarshtripathi.math@gmail.com

Abstract

In this paper we prove two results about $\text{AC}^0[\oplus]$ circuits.

- We show that for $d(N) = o(\sqrt{\log N / \log \log N})$ and $N \leq s(N) \leq 2^{dN^{1/4d^2}}$ there is an explicit family of functions $\{f_N : \{0, 1\}^N \rightarrow \{0, 1\}\}$ such that
 - f_N has uniform AC^0 formulas of depth d and size at most s ;
 - f_N does not have $\text{AC}^0[\oplus]$ formulas of depth d and size s^ε , where ε is a fixed absolute constant.

This gives a quantitative improvement on the recent result of Limaye, Srinivasan, Sreenivasaiah, Tripathi, and Venkitesh, (STOC, 2019), which proved a similar *Fixed-Depth Size-Hierarchy theorem* but for $d \ll \log \log N$ and $s \ll \exp(N^{1/2^{\Omega(d)}})$.

As in the previous result, we use the *Coin Problem* to prove our hierarchy theorem. Our main technical result is the construction of uniform size-optimal formulas for solving the coin problem with improved sample complexity $(1/\delta)^{O(d)}$ (down from $(1/\delta)^{2^{O(d)}}$ in the previous result).

- In our second result, we show that randomness buys depth in the $\text{AC}^0[\oplus]$ setting. Formally, we show that for any fixed constant $d \geq 2$, there is a family of Boolean functions that has polynomial-sized randomized uniform AC^0 circuits of depth d but no polynomial-sized (deterministic) $\text{AC}^0[\oplus]$ circuits of depth d .

Previously Viola (Computational Complexity, 2014) showed that an increase in depth (by at least 2) is essential to avoid superpolynomial blow-up while derandomizing randomized AC^0 circuits. We show that an increase in depth (by at least 1) is essential even for $\text{AC}^0[\oplus]$.

As in Viola's result, the separating examples are promise variants of the Majority function on N inputs that accept inputs of weight at least $N/2 + N/(\log N)^{d-1}$ and reject inputs of weight at most $N/2 - N/(\log N)^{d-1}$.

2012 ACM Subject Classification Theory of computation \rightarrow Circuit complexity; Theory of computation \rightarrow Pseudorandomness and derandomization

Keywords and phrases $\text{AC}^0[\oplus]$, Coin Problem, Promise Majority

Digital Object Identifier 10.4230/LIPIcs.FSTTCS.2019.22

Related Version A full version of the paper is available at <https://eccc.weizmann.ac.il/report/2019/133/>.

1 Introduction

This paper addresses questions in the field of *Boolean Circuit complexity*, where we study the complexity of computational problems, modeled as sequences of Boolean functions $f_N : \{0, 1\}^N \rightarrow \{0, 1\}$, in the combinatorially defined Boolean circuit model (see, e.g. [5] for an introduction).



© Nutan Limaye, Srikanth Srinivasan, and Utkarsh Tripathi;
licensed under Creative Commons License CC-BY

39th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2019).

Editors: Arkadev Chattopadhyay and Paul Gastin; Article No. 22; pp. 22:1–22:14



Leibniz International Proceedings in Informatics

Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Boolean circuit complexity is by now a classical research area in Computational complexity, with a large body of upper and lower bound results in many interesting circuit models. The questions we consider here are motivated by two of the most well-studied circuit models, namely AC^0 and $AC^0[\oplus]$. The circuit class AC^0 denotes the class of Boolean circuits of small-depth made up of AND, OR and NOT gates, while $AC^0[\oplus]$ denotes the circuit class that is also allowed the use of parity (addition modulo 2)¹ gates.²

Historically, AC^0 was among the first circuit classes to be studied and for which super-polynomial lower bounds were proved. Building on an influential line of work [2, 9, 24], Håstad [11] showed that any depth- d AC^0 circuit for the Parity function on N variables must have size $\exp(\Omega(N^{1/(d-1)}))$, hence proving an exponential lower bound for constant depths and superpolynomial lower bounds for all depths $d \ll \log N / \log \log N$. Researchers then considered the natural follow-up problem of proving lower bounds for $AC^0[\oplus]$. Soon after, Razborov [17] and Smolensky [21, 22] showed a lower bound of $\exp(\Omega(N^{1/2(d-1)}))$ for computing the Majority function on N inputs, again obtaining an exponential lower bound for constant depths and superpolynomial lower bounds for all depths $d \ll \log N / \log \log N$.

Thus, we have strong lower bounds for both classes AC^0 and $AC^0[\oplus]$. However, in many senses, $AC^0[\oplus]$ remains a much more mysterious class than AC^0 . There are many questions that we have been successfully able to answer about AC^0 but whose answers still evade us in the $AC^0[\oplus]$ setting. This work is motivated by two such questions that we now describe.

Size Hierarchy Theorems. Size Hierarchy theorems are an analogue in the Boolean circuit complexity setting of the classical Time and Space hierarchy theorems for Turing Machines. Formally, the problem is to separate the power of circuits (from some class) of size s from that of circuits of size at most s^ε for some fixed $\varepsilon > 0$. As is usual in the setting of circuit complexity, we ask for *explicit* separations,³ or equivalently, we ask that the separating sequence of functions be computed by a uniform family of circuits of size at most s .

The challenge here is to obtain explicit functions for which we can obtain *tight* (or near-tight) lower bounds, since we want the functions to have (uniform) circuits of size s but no circuits of size at most s^ε .

In the AC^0 setting, Håstad's theorem stated above immediately implies such a tight lower bound, since it is known (folklore) that the Parity function does have depth- d circuits of size $\exp(O(N^{1/d-1}))$ for every d . Varying the number of input variables to the Parity function suitably, this yields a Size Hierarchy theorem for the class of AC^0 circuits of depth d as long as $d \ll \log N / \log \log N$ and $s = \exp(o(N^{1/(d-1)}))$.

For $AC^0[\oplus]$, however, this is not as clear, as explicit *tight* lower bounds are harder to prove. In particular, the lower bounds of Razborov [17] and Smolensky [21, 22] for the Majority function (and other symmetric functions) are not tight; indeed, the exact complexity of these functions in $AC^0[\oplus]$ remains unknown [16]. In a recent result, the authors along with Sreenivasaiiah and Venkitesh [14] were able to show a size hierarchy theorem for $AC^0[\oplus]$ *formulas*⁴ for depths $d \ll \log \log N$ and size $s \ll \exp(N^{1/2^{\Omega(d)}})$. This is a weaker size

¹ Though we state our results only for $AC^0[\oplus]$, they extend in a straightforward way to $AC^0[p]$, where we are allowed gates that add modulo p , for any fixed prime p .

² The formal definitions of AC^0 and $AC^0[\oplus]$ only allow for polynomial-size circuits and constant depth. However, since some of our results apply to larger families of circuits, we will abuse notation and talk about AC^0 circuits of size $s(N)$ and depth $d(N)$ where s and d are growing functions of N .

³ It is trivial to show a non-explicit separation by counting arguments.

⁴ A formula is a circuit where the underlying undirected graph is a tree. For constant-depth, formulas and circuits are interchangeable with a polynomial blowup in depth. However, this is no longer true at superconstant depth [18, 19].

hierarchy theorem than the one that follows from Håstad's theorem for AC^0 , both in terms of the size parameter as well as the depths until which it holds. In this paper, we build upon the ideas in [14] and prove the following result that is stronger in both parameters.

► **Theorem 1.** *The following holds for some absolute constant $\varepsilon > 0$. Let N be a growing parameter and $d = d(N), s = s(N)$ be functions of N with $d = o\left(\sqrt{\frac{\log N}{\log \log N}}\right)$ and $N \leq s \leq 2^{dN^{1/d^2}}$. Then there is a family of functions $\{f_N\}$ such that f_N has uniform AC^0 formulas of depth d and size at most s but no $AC^0[\oplus]$ formulas of depth d and size at most s^ε .*

Randomized versus Deterministic circuits. The study of the relative power of randomized versus deterministic computation is an important theme in Computational complexity. In the setting of circuit complexity, it is known from a result of Adleman [1] that unbounded-depth polynomial-sized *randomized* circuits⁵ are no more powerful than polynomial-sized deterministic circuits.

However, the situation is somewhat more intriguing in the bounded-depth setting. Ajtai and Ben-Or [3] showed that for any randomized depth- d AC^0 circuit of size at most s , there is deterministic AC^0 circuit of depth $d + 2$ and size at most $\text{poly}(s)$ that computes the same function; a similar result also follows for $AC^0[\oplus]$ with the deterministic circuit having depth $d + 3$. This begs the question: is this increase in depth necessary?

For AC^0 circuits of constant depth, Viola [23] gave an optimal answer to this question by showing that an increase of two in depth is necessary to avoid a superpolynomial blow-up in size. To the best of our knowledge, this problem has not been studied in the setting of $AC^0[\oplus]$. In this paper, we show that an increase in depth (of at least one) is required even for $AC^0[\oplus]$. More formally we prove the following theorem.

► **Theorem 2.** *Fix any constant $d \geq 2$. There is a family of Boolean functions that has polynomial-sized randomized uniform AC^0 circuits of depth d but no polynomial-sized (deterministic) $AC^0[\oplus]$ circuits of depth d .*

Theorems 1 and 2 are proved in Sections 2 and 3 respectively. Many proofs are omitted for lack of space.

1.1 Proof Ideas

The proofs of both theorems are based on analyzing the complexity of Boolean functions that are closely related to the Majority function.

Size-Hierarchy Theorem. To prove the size hierarchy theorem for constant-depth $AC^0[\oplus]$ formulas, [14] studied the $AC^0[\oplus]$ complexity of the δ -*coin problem* [7], which is the problem of distinguishing between a coin that is either heads with probability $(1 + \delta)/2$ or is heads with probability $(1 - \delta)/2$, given a sequence of a large number of independent tosses of this coin. This problem has been studied in a variety of computational models [20, 7, 8, 10]. It is known [15, 4] that this problem can be solved by AC^0 formulas of depth d and size $\exp(O(d(1/\delta)^{1/(d-1)}))$ and further [15, 20, 14] that this upper bound is tight up to the constant in the exponent even for $AC^0[\oplus]$ formulas of depth d . This gives a family of functions for which we have tight lower bounds for $AC^0[\oplus]$ formulas.

⁵ A *randomized Boolean circuit* for a Boolean function $f(x)$ is a Boolean circuit C that takes as input variables x and r such that for each setting of x and uniformly random r , $C(x) = f(x)$ with probability at least $3/4$.

Based on this, [14] noted that to prove $AC^0[\oplus]$ size-hierarchy theorems for size $s(N)$ and depth $d(N)$, it suffices to construct a uniform sequence of formulas of size s and depth d solving the coin problem optimally (i.e. for δ such that $s = \exp(O(d(1/\delta)^{1/(d-1)}))$) using at most N samples. Before [14], all known size-optimal formula constructions for solving the δ -coin problem used $N = s = \exp(O(d(1/\delta)^{1/(d-1)}))$ many samples. The work of [14] brought the number of samples down to $N = (1/\delta)^{2^{O(d)}}$. Our main technical result here is an explicit size-optimal formula for solving the δ -coin problem using only $(1/\delta)^{O(d)}$ samples. Plugging this into the framework from [14], we immediately get the improved size-hierarchy theorem.

While the reason for this improvement is rather technical, we try to give a high-level outline here. It was shown by O’Donnell and Wimmer [15] and Amano [4] that the δ -coin problem is solved by read-once AC^0 formulas of depth d with gates of prescribed fan-ins. While the size s of these formulas is optimal, the number of samples is $N = s$, which is too big for our purposes. In [14], this number is brought down by distributing a smaller number of variables across the formula in a pseudorandom way (specifically using a Nisan-Wigderson design). The challenge now is to show that the formula still solves the δ -coin problem: the reason this is challenging is that various subformulas now share variables and hence the events that they accept or reject are no longer independent. However [14] note that *Janson’s inequality* [13], a tool from probabilistic combinatorics, can be used to argue that if the variables are spread out in a suitably “random”-like fashion, then various subformulas at a certain depth may, for our intents and purposes, be treated as “nearly” independent.

This “distance” from independence is determined by a parameter Δ that goes into the statement of Janson’s inequality, and hence let us call it the *Janson parameter*. In [14], this parameter was measured in a very brute-force way, forcing us to square the number of samples every time the depth of the formula increased by 1. This leads to a sample complexity of $(1/\delta)^{2^{O(d)}}$. Here, however, we give a different way of bounding the Janson parameter via a recursive analysis, which works as long as the number of variables grows by a factor of $(1/\delta)$ for each additional depth. This gives the improvement in our construction.

Randomized versus Deterministic circuits. For his separation of deterministic and randomized AC^0 circuits, Viola [23] used the *k-Promise-Majority* functions⁶ which are Boolean functions that accept inputs with at least $N/2 + k$ many 1s and reject inputs with at most $N/2 - k$ many 0s. Building on work of [3, 15, 4], Viola [23] showed that for $k = N/(\log N)^{d-1}$, there are *k-Promise-Majorities* that have uniform polynomial-sized *randomized* depth- d AC^0 circuits. On the other hand, he also showed that the same problem has no deterministic circuit of depth d (and in fact even $d + 1$).

The challenge in proving such a lower bound is that if a Boolean function has a randomized circuit of depth d and size s , then it immediately follows that there is also a deterministic circuit of the same depth and size *approximating* the same Boolean function (i.e. computing it correctly on most inputs). In particular, the lower bound technique must be able to distinguish circuits that are computing the function exactly (since this is hard) from circuits that are merely approximating it (as this is easy). Viola overcomes this hurdle in the case of AC^0 with a clever argument for depth-3 circuits and an inductive use of the Håstad Switching lemma for higher depths. Neither of these techniques is available for $AC^0[\oplus]$ circuits. In fact,

⁶ These are called *Approximate Majorities* in a lot of the earlier literature, including in Viola’s work. We avoid this name, since Approximate Majorities are also used for functions more closely related to the coin problem [15], and in our opinion, the name “Promise Majorities” better describes these functions.

the standard techniques for proving lower bounds against $\text{AC}^0[\oplus]$ involve approximating the circuits to constant error using low-degree polynomials from $\mathbb{F}_2[x_1, \dots, x_N]$. Note that this immediately runs into the obstacle mentioned above since we can then no longer distinguish between circuits that are exactly correct and those that are approximately correct.

The way we get around this argument is to use a recent result of Oliveira, Santhanam and the second author [16] where it is observed that the standard construction of approximating polynomials for $\text{AC}^0[\oplus]$ actually gives polynomials that approximate the given circuit C to very small error on either the zero or the one inputs of C . They are able to use this to improve known $\text{AC}^0[\oplus]$ lower bounds for the Majority function. Our main observation is that this stronger lower bound is actually able to distinguish between circuits that approximate the Majority function to constant error (say from [15, 4]) and those that compute it exactly, thus overcoming the barrier we mentioned above. We then note that their proof can also be made to work for k -Promise-Majorities. This yields the separation.

2 Size hierarchy theorem for $\text{AC}^0[\oplus]$

► **Definition 3** (The δ -Coin Problem). *Let $\delta \in (0, 1)$ be a parameter. Given an $N \in \mathbb{N}$, we define the probability distributions $\mu_{\delta,0}^N$ and $\mu_{\delta,1}^N$ to be the product distributions where each bit is set to 1 with probability $(1 - \delta)/2$ and $(1 + \delta)/2$ respectively. We omit the δ in the subscript and N in the superscript when these are clear from context.*

Given a function $g : \{0, 1\}^N \rightarrow \{0, 1\}$, we say that g solves the δ -coin problem if

$$\Pr_{\mathbf{x} \sim \mu_0^N}[g(\mathbf{x}) = 1] \leq 0.1 \text{ and } \Pr_{\mathbf{x} \sim \mu_1^N}[g(\mathbf{x}) = 1] \geq 0.9. \quad (1)$$

We say that the sample complexity of g is N .

Parameters. Let m, d be growing parameters such that $d = o(m/\log m)$. Let $1/\delta = (m \ln 2)^{d-1}/C_1$, where C_1 is a fixed large constant, to be specified below. Let $M = \lceil m \cdot 2^m \cdot \ln 2 \rceil$ and let $M_1 = 2^m$.

► **Theorem 4.** *For large enough absolute constant C_1 , the following holds. For parameters m, δ, d as above and for $d \geq 2$, there is an explicit depth- d AC^0 formula of size $\exp(O(dm)) = \exp(O(d(1/\delta)^{1/(d-1)}))$ and sample complexity $(1/\delta)^{d+4}$ that solves the δ -coin problem.*

We first show how Theorem 4 implies Theorem 1 stated in the introduction.

Proof of Theorem 1. We use Theorem 4 for a suitable choice of parameters to define the explicit function.

Let $m = \lfloor (\alpha \log s)/d \rfloor$ for some absolute constant $\alpha < 1$ that we fix below. It can be checked that as $s \geq N$ and $d = o(\sqrt{\log N / \log \log N})$, we have $d = o(m/\log m)$. Define δ as above and note that $(1/\delta)^{d+4} \leq m^{2d^2} \leq ((\log s)/d)^{2d^2} \leq N$, where the final inequality uses the given upper bounds on d and s .

We set f_N to be the Boolean function computed by the formula F_d constructed above on the first $(1/\delta)^{d+4}$ of the N input variables. By Theorem 4, the size of F_d is $\exp(O(dm)) \leq s$ for a small enough absolute constant α and F_d solves the δ -coin problem. Moreover, it was shown in [14] that any depth- d $\text{AC}^0[\oplus]$ formula solving the δ -coin problem must have size $\exp(\Omega(d(1/\delta)^{1/(d-1)})) = \exp(\Omega(md)) = s^\varepsilon$ for some absolute constant $\varepsilon > 0$. This proves the theorem. ◀

2.1 Proof of Theorem 4

In this section we give the construction of the explicit formula solving the δ -coin problem and prove Theorem 4. There exist integers Q, D , such that⁷ Q is a prime power, $M \leq Q^D \leq 2M$ and $(m^4/\delta) \leq Q \leq (2m^4/\delta)$. Let \mathbb{F} be a finite field with Q elements and $A \subseteq \mathbb{F}$ be a set of size m . Let \mathcal{P}_D be the lexicographically first M univariate polynomials over \mathbb{F} of degree strictly less than D . Similarly, let \mathcal{P}'_D be the lexicographically first M_1 univariate polynomials over \mathbb{F} of degree less than D .

We now describe the construction of our formula. The variables in the formula correspond to the points in the set $A \times \mathbb{F}^{d-1}$. i.e. for each $(a, c_1, \dots, c_{d-1}) \in A \times \mathbb{F}^{d-1}$, we have a variable $x(a, c_1, \dots, c_{d-1})$. We thus have $m \cdot Q^{d-1}$ many variables, denoted by N .

For each $i \in [d-1]$ and $\bar{P} = (P_i, \dots, P_{d-1}) \in \mathcal{P}_D^{d-i}$, define a depth- i formula $\mathcal{C}_{(P_i, \dots, P_{d-1})}$ inductively as follows.

$$\begin{aligned} \mathcal{C}_{(P_1, \dots, P_{d-1})} &= \bigwedge_{a \in A} x(a, P_1(a), \dots, P_{d-1}(a)), \\ \mathcal{C}_{(P_2, \dots, P_{d-1})} &= \bigvee_{R_1 \in \mathcal{P}_D} \mathcal{C}_{(R_1, P_2, \dots, P_{d-1})}, \quad \mathcal{C}_{(P_3, \dots, P_{d-1})} = \bigwedge_{R_2 \in \mathcal{P}_D} \mathcal{C}_{(R_2, P_3, \dots, P_{d-1})} \end{aligned}$$

and so on, with the gates alternately repeating between **AND** and **OR** until depth $d-1$. Finally, $\mathcal{C}_{(\emptyset)}$ is the output of the formula. If the depth of the formula is odd then $\mathcal{C}_{(\emptyset)}$ is equal to $\bigwedge_{R \in \mathcal{P}'_D} \mathcal{C}_{(R)}$ otherwise it is equal to $\bigwedge_{R \in \mathcal{P}_D} \mathcal{C}_{(R)}$. This finishes the description of our formula. We use $F_d = \mathcal{C}_{(\emptyset)}$ to denote this formula.

Analysis of the construction

Here we present the details regarding the analysis of our construction presented above, which will be used to prove Theorem 4. We will start with some definitions, notations and some useful inequalities.

► **Definition 5.** For $1 \leq i \leq d-1$, we define the following terms.

1. For $\bar{P} = (P_i, \dots, P_{d-1}) \in \mathcal{P}_D^{d-i}$ and $b \in \{0, 1\}$, let

$$\text{Acc}_{\bar{P}, b} := \Pr_{\mu_b}[\mathcal{C}_{(P_i, \dots, P_{d-1})} \text{ accepts}] \text{ and } \text{Rej}_{\bar{P}, b} := \Pr_{\mu_b}[\mathcal{C}_{(P_i, \dots, P_{d-1})} \text{ rejects}].$$

Let $q_{\bar{P}, b} = \text{Acc}_{\bar{P}, b}$ if i is odd and $\text{Rej}_{\bar{P}, b}$ if i is even.

2. For $\bar{P} = (P_i, \dots, P_{d-1}), \bar{P}' = (P'_i, \dots, P'_{d-1}) \in \mathcal{P}_D^{d-i}$, we say that $\bar{P} \sim \bar{P}'$ when $\mathcal{C}_{\bar{P}}$ and $\mathcal{C}_{\bar{P}'}$ are distinct gates which share a common input variable.
3. Fix any $i \in [d-1]$. For $\bar{P} = (P_{i+1}, \dots, P_{d-1}), \bar{P}' = (P'_{i+1}, \dots, P'_{d-1}) \in \mathcal{P}_D^{d-i-1}$, $b \in \{0, 1\}$,

$$\Delta_{\bar{P}, \bar{P}', b} = \begin{cases} \sum_{\substack{R_i, R'_i \in \mathcal{P}_D \\ (R_i, \bar{P}) \sim (R'_i, \bar{P}')}} \Pr_{\mu_b}[\mathcal{C}_{(R_i, \bar{P})} = 0 \text{ AND } \mathcal{C}_{(R'_i, \bar{P}')} = 0] & \text{if } \mathcal{C}_{\bar{P}} \text{ and } \mathcal{C}_{\bar{P}'} \text{ are AND gates} \\ \sum_{\substack{R_i, R'_i \in \mathcal{P}_D \\ (R_i, \bar{P}) \sim (R'_i, \bar{P}')}} \Pr_{\mu_b}[\mathcal{C}_{(R_i, \bar{P})} = 1 \text{ AND } \mathcal{C}_{(R'_i, \bar{P}')} = 1] & \text{if } \mathcal{C}_{\bar{P}} \text{ and } \mathcal{C}_{\bar{P}'} \text{ are OR gates} \end{cases}$$

⁷ Using number-theoretic facts about the density of primes [6] (see for instance [12]), such Q, D can be found in polynomial time.

A useful tool in our analysis of the circuit is Janson's inequality stated here in the language of Boolean circuits.

► **Theorem 6** (Janson's inequality). *Let $\mathcal{C}_1, \dots, \mathcal{C}_M$ be any monotone Boolean circuits over inputs x_1, \dots, x_n and let \mathcal{C} denote $\bigvee_{i \in [M]} \mathcal{C}_i$. For each distinct $i, j \in [M]$, we use $i \sim j$ to denote the fact that \mathcal{C}_i and \mathcal{C}_j share a common variable. Assume each x_j ($j \in [M]$) is chosen independently to be 1 with probability $p_j \in [0, 1]$, and that under this distribution, we have $\max_{i \in [M]} \{\Pr_x[\mathcal{C}_i(x) = 1]\} \leq 1/2$. Then we have*

$$\prod_{i \in [M]} \Pr_x[\mathcal{C}_i(x) = 0] \leq \Pr_x[\mathcal{C}(x) = 0] \leq \left(\prod_{i \in [M]} \Pr_x[\mathcal{C}_i(x) = 0] \right) \cdot \exp(\Delta) \quad (2)$$

where $\Delta := \sum_{i \sim j} \Pr_x[(\mathcal{C}_i(x) = 1) \wedge (\mathcal{C}_j(x) = 1)]$.

Throughout, we use $\log(\cdot)$ to denote logarithm to the base 2 and $\ln(\cdot)$ for the natural logarithm. We use $\exp(x)$ to denote e^x .

► **Fact 7.** *Assume that $x \in [-1/2, 1/2]$. Then we have the following chain of inequalities.*

$$\exp(x - (|x|/2)) \stackrel{(a)}{\leq} \exp(x - x^2) \stackrel{(b)}{\leq} 1 + x \stackrel{(c)}{\leq} \exp(x) \stackrel{(d)}{\leq} 1 + x + x^2 \stackrel{(e)}{\leq} 1 + x + (|x|/2) \quad (3)$$

We define a few parameters which will be useful in the main technical lemma that helps in proving Theorem 4.

For $i \in [d-1]$, let $\alpha_i = m^i \cdot (\ln 2)^{i-1} \cdot \delta$. Also define $\beta_1 = 2\alpha_1$ and $\beta_i = \beta_{i-1} + 2\alpha_i + \frac{2}{m^i (\ln 2)^{i-1}}$ for $2 \leq i \leq d-2$.

► **Observation 8.** *For all $i \in [d-2]$, $\alpha_i, \beta_i \leq O(1/m)$. Also, $\alpha_{d-1} = \Theta(C_1) = \Theta(1)$. Finally, for $i \in [d-2]$, using Fact 7 above, we get $\exp(-\beta_{i-1}) - \exp(-\beta_i) \geq \alpha_i/2$.*

► **Lemma 9.** *Assume $d \geq 3$ and $q_{\bar{P}, b}$ and formula $\mathcal{C}_{(\emptyset)}$ defined as before. We have the following properties.*

1. For $b \in \{0, 1\}$, $i \in [d-2]$ such that $i \equiv b \pmod{2}$,

$$\begin{aligned} \frac{1}{2^m} \cdot (1 + \alpha_i \exp(-\beta_i)) &\leq q_{\bar{P}, b} \leq \frac{1}{2^m} \cdot (1 + \alpha_i \exp(\beta_i)) \\ \frac{1}{2^m} \cdot (1 - \alpha_i \exp(\beta_i)) &\leq q_{\bar{P}, (1-b)} \leq \frac{1}{2^m} \cdot (1 - \alpha_i \exp(-\beta_i)) \end{aligned}$$

2. Say $d-1 \equiv b \pmod{2}$. Then

$$q_{\bar{P}, b} \geq \frac{1}{2^m} \cdot \exp(\alpha_{d-1}/4) \text{ and } q_{\bar{P}, 1-b} \leq \frac{1}{2^m} \cdot \exp(-\alpha_{d-1}/4)$$

3. For all $i \in [d-1]$, $b \in \{0, 1\}$ and $\bar{P}, \bar{P}' \in \mathcal{P}_D^{d-i-1}$, $\Delta_{\bar{P}, \bar{P}', b} < \delta$.

Assuming that the above lemma holds for now, we will prove Theorem 4.

Proof of Theorem 4. We start by bounding the size of $F_d = \mathcal{C}_{(\emptyset)}$. As per our construction, the gates at level 1 are **AND** gates with fan-in m each. For all $2 \leq i \leq d-1$, the fan-in of each gate on level i is $M = \lceil m \cdot 2^m \cdot \ln 2 \rceil$ and the top fan-in is $M_1 = 2^m$. Therefore, the total number of gates in the formula is $m \cdot M^{d-2} \cdot M_1$. We can trivially bound this

22:8 AC⁰[⊕] and Variants of the Majority Function

by $M^d = O(m^d 2^{dm})$. As $d = o(m/\log m)$, we get that the size is bounded by $\exp(O(dm))$. Recall that $1/\delta = (m \ln 2)^{d-1}/C_1$, where C_1 is an appropriately chosen constant. Hence $\exp(O(dm)) = \exp(O(d(1/\delta)^{1/(d-1)}))$.

We will now bound the number of variables N used by the formula. As mentioned above, $N = m \cdot Q^{d-1}$. As Q is chosen such that $Q = \Theta(m^4/\delta)$, there exists a constant C' such that $N \leq m \cdot (C' m^4/\delta)^{d-1}$.

$$\begin{aligned} N &\leq m \cdot (C' m^4/\delta)^{d-1} \leq (1/\delta)^{d-1} \cdot (m^{d-1})^4 \cdot m \cdot C'^{d-1} \\ &\leq \left(\frac{1}{\delta}\right)^{d-1} \cdot \left(\frac{1}{\delta}\right)^4 \cdot m \cdot (C'')^{d-1} \quad (\text{for some constant } C'' \text{ as } 1/\delta = (m \ln 2)^{d-1}/C_1) \\ &\leq \left(\frac{1}{\delta}\right)^{d+3} \cdot \frac{1}{\delta} = \left(\frac{1}{\delta}\right)^{d+4} \end{aligned}$$

Finally, we will show that the formula solves the δ -coin problem. Let us assume that d is even. In that case, the output gate $\mathcal{C}_{(\emptyset)}$ is an **OR** gate. (When it is an **AND** gate, the analysis is very similar.) We bound the probabilities $\Pr_{a \in \mu_0}[F_d(a) = 1]$ and $\Pr_{a \in \mu_1}[F_d(a) = 0]$ by $1/10$ each.

$$\begin{aligned} \Pr_{a \in \mu_0}[F_d(a) = 1] &\leq \sum_{R \in \mathcal{P}'_D} \Pr_{a \in \mu_0}[\mathcal{C}_{(R)}(a) = 1] && \text{Using a Union bound} \\ &\leq 2^m \cdot \frac{1}{2^m} \cdot \exp(-\alpha_{d-1}/4) && |\mathcal{P}'_D| = 2^m, \text{ using Lemma 9, (2)} \\ &\leq \exp(-\Omega(C_1)) && \text{Using the value of } \alpha_{d-1} \\ &\leq 1/10. && \text{for large enough } C_1 \end{aligned}$$

$$\begin{aligned} \Pr_{a \in \mu_1}[F_d(a) = 0] &\leq \prod_{\bar{P} \in \mathcal{P}'_D} \Pr_{a \in \mu_1}[\mathcal{C}_{(\bar{P})}(a) = 0] \cdot \exp(\delta) && \text{Using Janson's inequality} \\ & && \text{and Lemma 9, (3)} \\ &\leq \prod_{\bar{P} \in \mathcal{P}'_D} (1 - \Pr_{a \in \mu_1}[\mathcal{C}_{(\bar{P})}(a) = 1]) \cdot \exp(\delta) \\ &\leq \left(1 - \frac{1}{2^m} \cdot \exp(\alpha_{d-1}/4)\right)^{2^m} \cdot \exp(\delta) && |\mathcal{P}'_D| = 2^m, \text{ using Lemma 9, (2)} \\ &\leq \exp\left(-\frac{2^m}{2^m} \cdot \exp(\alpha_{d-1}/4)\right) \cdot 2 && \text{As } \exp(\delta) \leq 2 \\ &\leq 1/10. && \text{Using the value of } \alpha_{d-1} \\ & && \text{and for large enough } C_1 \end{aligned}$$

This finishes the proof of Theorem 4 assuming Lemma 9. ◀

We now give the proof of Lemma 9. The proof is by induction on the depth of the circuit.

Proof of Lemma 9. The lemma has three parts. As mentioned above, we proceed by induction on the depth.

Base case ($i = 1$): Here let us first assume that we are working with μ_1^N . We start with part (1). We wish to bound $q_{\bar{P},1}$. From the construction of our formula, we know that the formula has **AND** gates at layer 1 and the inputs to these are distinct variables and hence independent. Therefore, $q_{\bar{P},1} = \left(\frac{1+\delta}{2}\right)^m$. We will upper and lower bound this quantity.

$$\left(\frac{1+\delta}{2}\right)^m \geq \frac{1}{2^m} \cdot (1 + \delta m) \geq \frac{1}{2^m} \cdot (1 + \alpha_1 \cdot \exp(-\beta_1)) \quad (\text{As } \alpha_1 = \delta m, \beta_1 > 0)$$

$$\begin{aligned}
 \left(\frac{1+\delta}{2}\right)^m &= \frac{1}{2^m} \cdot (1+\delta)^m \leq \frac{1}{2^m} \cdot \exp(\delta m) && \text{Fact 7 (c)} \\
 &\leq \frac{1}{2^m} \cdot (1+\delta m + (\delta m)^2) && \text{Fact 7 (d)} \\
 &\leq \frac{1}{2^m} \cdot (1+\delta m \cdot \exp(2\delta m)) && \text{Fact 7 (c)} \\
 &= \frac{1}{2^m} \cdot (1+\alpha_1 \cdot \exp(\beta_1)) && \text{As } \alpha_1 = \delta m, \beta_1 = 2\alpha_1
 \end{aligned}$$

In the case of μ_0^N , we get $q_{\bar{P},0} = \left(\frac{1-\delta}{2}\right)^m$ and a very similar computation can be used to upper and lower bound this quantity.

There is nothing to prove for part (2) in the base case. We now prove the base case for part (3). Let $\bar{P} = (P_2, \dots, P_{d-1})$, $\bar{P}' = (P'_2, \dots, P'_{d-1}) \in \mathcal{P}_D^{d-2}$. We will analyse $\Delta_{\bar{P},\bar{P}',1}$ here. The analysis for $\Delta_{\bar{P},\bar{P}',0}$ is very similar. Let λ denote $(1+\delta)/2$. For a formula F , let $\mathbf{Var}(F)$ denote the set of variables appearing in it.

$$\begin{aligned}
 \Delta_{\bar{P},\bar{P}',1} &= \sum_{\substack{R,R' \in \mathcal{P}_D \\ (R,\bar{P}) \sim (R',\bar{P}')}} \Pr_{\mu_1}[C_{(R,\bar{P})} = 1 \text{ AND } C_{(R',\bar{P}')} = 1] \\
 &= \sum_{\substack{R,R' \\ (R,\bar{P}) \sim (R',\bar{P}')}} \lambda^{|\mathbf{Var}(C_{(R,\bar{P})}) \cup \mathbf{Var}(C_{(R',\bar{P}')})|} \\
 &= \lambda^{2m} \cdot \sum_{\substack{R,R' \\ (R,\bar{P}) \sim (R',\bar{P}')}} \left(\frac{1}{\lambda}\right)^{|\mathbf{Var}(C_{(R,\bar{P})}) \cap \mathbf{Var}(C_{(R',\bar{P}')})|}. \tag{4}
 \end{aligned}$$

To bound the above term, we use the following technical claim (proof omitted).

▷ **Claim 10.** Fix any $i \leq d-1$ and any $\bar{P}, \bar{P}' \in \mathcal{P}_D^{d-i-1}$, we have

$$\sum_{\substack{R,R' \\ (R,\bar{P}) \sim (R',\bar{P}')}} 3^{|\mathbf{Var}(C_{(R,\bar{P})}) \cap \mathbf{Var}(C_{(R',\bar{P}')})|} \leq Q^{2D} \cdot O\left(\frac{m}{Q}\right).$$

Using Claim 10 and (4), we can immediately bound $\Delta_{\bar{P},\bar{P}',1}$ as follows.

$$\Delta_{\bar{P},\bar{P}',1} \leq \lambda^{2m} \cdot \sum_{\substack{R,R' \\ (R,\bar{P}) \sim (R',\bar{P}')}} 3^{|\mathbf{Var}(C_{(R,\bar{P})}) \cap \mathbf{Var}(C_{(R',\bar{P}')})|} \leq \lambda^{2m} Q^{2D} \cdot O\left(\frac{m}{Q}\right) \leq O\left(\frac{m^3}{Q}\right) < \delta$$

where we have used the fact that $\lambda \geq 1/3$, $Q^D = O(M^2) = O(m^2 2^{2m})$, and $Q = \Theta(m^4/\delta)$. This concludes the bound on $\Delta_{\bar{P},\bar{P}',1}$ and hence concludes the proof of the base case.

Inductive case: The proof of parts (1) and (2) are similar to the base case and hence omitted.

Finally, we prove the inductive statement about $\Delta_{\bar{P},\bar{P}',1}$ in the case that i is odd. Fix any $\bar{P}, \bar{P}' \in \mathcal{P}_D^{d-i-1}$ (in the case that $i = d-1$, we will have $\bar{P} = \bar{P}' = (\emptyset)$). The computation goes as follows. *The crucial steps are the second equality and first inequality, where we interpret each term in the sum as the probability that a depth $i-1$ circuit takes the value 0, which is bounded using Janson's inequality and the induction hypothesis.*

22:10 AC⁰[⊕] and Variants of the Majority Function

$$\begin{aligned}
\Delta_{\bar{P}, \bar{P}', 1} &= \sum_{\substack{R, R' \\ (R, \bar{P}) \sim (R', \bar{P}')}} \Pr_{\mu_1^n} [C_{(R, \bar{P})} = 0 \text{ AND } C_{(R', \bar{P}')} = 0] \\
&= \sum_{\substack{R, R' \\ (R, \bar{P}) \sim (R', \bar{P}')}} \Pr_{\mu_1^n} [\bigvee_S C_{(S, R, \bar{P})} \vee \bigvee_{S'} C_{(S', R', \bar{P}')} = 0] \\
&\leq \sum_{\substack{R, R' \\ (R, \bar{P}) \sim (R', \bar{P}')}} \prod_S \Pr[C_{(S, R, \bar{P})} = 0] \cdot \prod_{S'} \Pr[C_{(S', R', \bar{P}')} = 0] \cdot \exp(4\delta) \\
&\leq \exp(4\delta) \cdot \sum_{\substack{R, R' \\ (R, \bar{P}) \sim (R', \bar{P}')}} \left(1 - \frac{(1 - 2\alpha_{i-1})}{2^m}\right)^{2M} = \exp(4\delta) \left(1 - \frac{(1 - 2\alpha_{i-1})}{2^m}\right)^{2M} \cdot \sum_{\substack{R, R' \\ (R, \bar{P}) \sim (R', \bar{P}')}} 1 \\
&\leq 2 \exp(-2m \ln 2 + O(\alpha_i)) \cdot \sum_{\substack{R, R' \\ (R, \bar{P}) \sim (R', \bar{P}')}} 1 = O\left(\frac{1}{2^{2m}}\right) \cdot \sum_{\substack{R, R' \\ (R, \bar{P}) \sim (R', \bar{P}')}} 1
\end{aligned}$$

where the first inequality is just Janson's inequality applied to the formula $\bigvee_S C_{(S, R, \bar{P})} \vee \bigvee_{S'} C_{(S', R', \bar{P}'})$; the second inequality follows from the induction hypothesis applied to level $i - 1 \leq d - 2$ (we have used a slightly weaker bound that is applicable also to other cases such as when $b = 0$); and the last inequality follows from our choice of M and the fact that $\alpha_i = \alpha_{i-1} \cdot (m \ln 2)$. The sum in the final term may be bounded by $Q^{2D} \cdot O(m/Q)$ by Claim 10. We thus get

$$\Delta_{\bar{P}, \bar{P}', 1} \leq O\left(\frac{Q^{2D}}{2^m}\right) \cdot \frac{m}{Q} = O\left(\frac{M^2}{2^m}\right) \cdot \frac{m}{Q} \leq \frac{O(m^3)}{Q} < \delta$$

as $Q \geq m^4/\delta$. This finishes the analysis of $\Delta_{\bar{P}, \bar{P}', 1}$. \blacktriangleleft

3 Randomized vs. Deterministic AC⁰[⊕] circuits

For $a \in \{0, 1\}^n$, let $|a|$ denote the Hamming weight of a , i.e. the number of 1s in a .

► **Definition 11.** Let $k, \ell \leq n/2$. The Promise Majority problem, $\text{PrMaj}_{k, \ell}^n$, is a promise problem of distinguishing n -bit strings of Hamming weight less than $n/2 - k$ from those with Hamming weight more than $n/2 + \ell$. Formally,

$$\text{PrMaj}_{k, \ell}^n(a) = \begin{cases} 0 & \text{if } |a| < (\frac{n}{2} - k) \\ 1 & \text{if } |a| \geq (\frac{n}{2} + \ell) \end{cases}$$

If the length of the input is clear from the context then we drop the superscript n . If $k = 0$ then we denote $\text{PrMaj}_{0, \ell}$ by LowPrMaj_ℓ . Similarly, $\ell = 0$ then we denote $\text{PrMaj}_{k, 0}$ by UpPrMaj_k . When both k, ℓ are zero, $\text{PrMaj}_{0, 0}$ is the Majority function. If $k = \ell$ then we use PrMaj_k to denote $\text{PrMaj}_{k, k}$.

Let $\text{Yes}_\ell^n, \text{No}_k^n$ denote the yes and no instances of $\text{PrMaj}_{k, \ell}^n$. That is, $\text{Yes}_\ell^n = \{a \in \{0, 1\}^n \mid |a| \geq n/2 + \ell\}$ and $\text{No}_k^n = \{a \in \{0, 1\}^n \mid |a| < n/2 - k\}$. In [23], the following theorem was proved.

► **Theorem 12** (Theorem 1.2 [23]). For any $d \geq 2$ and $k(N) = \Omega(N/(\log N)^{d-1})$, there is a uniform family of randomized AC⁰ circuits of depth d and $\text{poly}(N)$ size computing $\text{PrMaj}_{k(N)}^N$.

Here, we prove the following theorem.

► **Theorem 13.** *For any $d \geq 2$, say \mathcal{C} is a (deterministic) $\text{AC}^0[\oplus]$ circuit of depth d computing $\text{PrMaj}_{N/2, (\log N)^{d-1}}^N$, then \mathcal{C} must have size $N^{\omega(1)}$.*

It is easy to see that using Theorem 12 and Theorem 13, we immediately get Theorem 2. In order to prove Theorem 13 we need the following claim. This is our main technical claim.

▷ **Claim 14.** Let $n \in \mathbb{N}$ and let $k = \Theta(n/(\log n)^c)$. Let $p \in \mathbb{F}[x_1, \dots, x_n]$ be a (deterministic) polynomial such that it satisfies one of the following two conditions

$$\text{either } \Pr_{a \in \text{No}_k^n} [p(a) = 1] \leq 1/n, \quad \Pr_{a \in \text{Yes}_0^n} [p(a) = 0] \leq 1/10 \quad (5)$$

$$\text{or } \Pr_{a \in \text{No}_0^n} [p(a) = 1] \leq 1/10 \quad \Pr_{a \in \text{Yes}_k^n} [p(a) = 0] \leq 1/n \quad (6)$$

Then $\deg(p) = \Omega(\log^{c+1} n)$.

Proof of Theorem 13 using Claim 14. We will first show that Theorem 13 follows from the above claim. We will do this using the following two step argument.

- (I) Let us assume for now that \mathcal{C} is a circuit of size s and depth d with either OR gate or \oplus gate as its output gate. Let us call the output gate G_{out} . We will show that if \mathcal{C} computes PrMaj_k^N then we have a circuit \mathcal{C}' of size s , depth d and with output gate G_{out} , such that it computes UpPrMaj_{2k}^n , where $n = N - 2k$.⁸
- (II) We will then show that any depth d circuit with OR or \oplus output gate computing UpPrMaj_{2k}^n must have size $n^{\omega(1)}$.

As we will invoke this for $k = N/2(\log N)^c$, which is $o(N)$, an $n^{\omega(1)}$ lower bound on UpPrMaj_{2k}^n will imply a $N^{\omega(1)}$ lower bound on PrMaj_k^N , thereby proving the theorem.

Here, (I) can be shown by simply fixing some of the input bits to the constant 1. Specifically, let us set $2k$ bits out of the N bits to 1s. Let $n = N - 2k$. It is easy to see that if $x \in \{0, 1\}^n$ has Hamming weight at least $n/2$, then in fact $y = x \cdot 1^{2k}$ has $N/2 + k$ many 1s. Similarly, if $x \in \{0, 1\}^n$ has Hamming weight at most $n/2 - 2k$ then the Hamming weight of $y = x \cdot 1^{2k}$ is at most $N/2 - k$.

To show (II) requires a little more work. In particular, to show (II), we use a result from [16] about degree of polynomials approximating $\text{AC}^0[\oplus]$ circuits. To state their result, we will introduce some notation.

► **Definition 15.** *Let $f : \{0, 1\}^n \rightarrow \{0, 1\}$ be a Boolean function. For any parameters $\varepsilon_0, \varepsilon_1$, $(\varepsilon_0, \varepsilon_1)$ -error probabilistic polynomial for f is a random multilinear polynomial P chosen from $\mathbb{F}_2[x_1, \dots, x_n]$, such that for any $b \in \{0, 1\}$ and any $a \in f^{-1}(b)$, $\Pr[P(a) \neq f(a)] \leq \varepsilon_b$.*

A probabilistic polynomial is said to have degree at most d if the underlying distribution is supported on monomials of degree at most d .

We define the $(\varepsilon_0, \varepsilon_1)$ -error probabilistic polynomial degree of a Boolean function f , denoted as $\text{pdeg}_{\varepsilon_0, \varepsilon_1}(f)$, to be the smallest d such that there is an $(\varepsilon_0, \varepsilon_1)$ -error probabilistic polynomial of degree d for f .

⁸ As PrMaj is a self-dual function and UpPrMaj and LowPrMaj are duals of each other, we can assume that the output gate of \mathcal{C} is OR or \oplus without loss of generality.

22:12 AC⁰[⊕] and Variants of the Majority Function

► **Lemma 16** (Corollary 15, [16]). *Let \mathcal{C} be a size s , depth d circuit with OR or \oplus as its output gate. Then there is a probabilistic polynomial \bar{p} approximating \mathcal{C} such that $\text{pdeg}_{1/n^2, 1/100}(\bar{p})$ is at most $O(\log s)^{d-1}$.*

► **Remark 17.** Let \mathcal{C} be a circuit of size s and depth d (with any output gate). It is known that if \bar{p} is a probabilistic polynomial for \mathcal{C} such that $\varepsilon_0 = \varepsilon_1 = 1/s^{O(1)}$, then $\text{pdeg}_{1/s^{O(1)}, 1/s^{O(1)}}(\bar{p})$ is $O(\log s)^d$. The above lemma says that if we need only constant error on one of sides, i.e. say if either ε_0 or ε_1 is $\Omega(1)$, then we can get a better degree upper bound. Instead of having d in the exponent, we get $d - 1$ in the exponent. This is crucial.

Note that, if the output gate of \mathcal{C} is OR (AND) then we can ensure that $\varepsilon_0 = 1/n^2$ ($\varepsilon_1 = 1/n^2$, resp.). If it is a \oplus gate, then either can be ensured.

Suppose there is an AC⁰[⊕] circuit \mathcal{C} of size $s = n^t$ and depth d with top gate OR or \oplus and computing UpPrMaj_{2k} .

Applying Lemma 16 and by standard averaging arguments we can show that there is a fixed polynomial $P \in \mathbb{F}[X]$ that satisfies conditions (5) for $c = d - 1$ and has the same degree as the degree of \bar{p} . Therefore on the one hand, we know that $\deg(P)$ is less than or equal to $O(t \log n)^{d-1}$, while on the other hand using Claim 14 we get that $\deg(P)$ is at least $\Omega(\log n)^d$. (As $N/(\log N)^c = \Theta(n/(\log n)^c)$, Claim 14 is applicable.) Thus, $O(t \log n)^{d-1} \geq \Omega(\log n)^d$ and hence we get $t \geq \Omega(\log n)^{1/d-1}$. Therefore we get (II). This finishes the proof of Theorem 13. ◀

We now proceed with the proof of Claim 14. We will use the following fact in the proof of Claim 14.

► **Fact 18.** *Say $R \in \mathbb{F}[X]$ is a non-zero polynomial that vanishes on No_k^n , then degree of R is at least $n/2 - k$.*

Proof of Claim 14. We will show that if a deterministic polynomial $p \in \mathbb{F}[X]$ satisfies condition (5), then it has degree $C \cdot \log^{c+1} n$ for some constant C . The proof for the lower bound on the degree of p assuming condition (6) is similar. For simplicity we will work out the proof when $k = n/(\log n)^c$. The proof is similar when $k = \Theta(n/(\log n)^c)$.

Let us use D to denote $C \cdot \log^{c+1} n$. Consider a polynomial p satisfying condition (5). Let \mathcal{E}_0 and \mathcal{E}_1 be error sets of this polynomial on no and yes instances respectively, i.e. $\mathcal{E}_0 = \{a \in \text{No}_{n/(\log n)^c}^n \mid p(a) = 1\}$ and $\mathcal{E}_1 = \{a \in \text{Yes}_0^n \mid p(a) = 0\}$. From condition (5) we have a bound on the cardinalities of $\mathcal{E}_0, \mathcal{E}_1$.

We will first observe that in order to prove the claim, it suffices to show the existence of a polynomial $Q \in \mathbb{F}[X]$ with the following three properties.

(a) $Q(a) = 0$ for all $a \in \mathcal{E}_0$.

(b) $Q \cdot p \neq 0$.

(c) $\deg(Q) \leq r - D$, where $r = n/2 - n/(\log n)^c$ and D is as defined above.

Suppose we have such a Q then let $R = Q \cdot p$. Now R is a polynomial that vanishes on $\text{No}_{n/(\log n)^c}^n$. This is because either p vanishes on $\text{No}_{n/(\log n)^c}^n \setminus \mathcal{E}_0$ or Q vanishes on \mathcal{E}_0 . Due to property (b), R is also a non-zero polynomial. Therefore using Fact 18, we know that it has degree at least r . Now assuming property (c) we get that p must have degree at least D , thereby proving the claim.

The existence of such a Q can be proved using arguments similar to those in [16]. The proof is omitted for lack of space. ◀

References

- 1 Leonard M. Adleman. Two Theorems on Random Polynomial Time. In *19th Annual Symposium on Foundations of Computer Science, Ann Arbor, Michigan, USA, 16-18 October 1978*, pages 75–83, 1978.
- 2 Miklós Ajtai. \sum^1_1 -formulae on finite structures. *Ann. Pure Appl. Logic*, 24(1):1–48, 1983. doi:10.1016/0168-0072(83)90038-6.
- 3 Miklós Ajtai and Michael Ben-Or. A Theorem on Probabilistic Constant Depth Computations. In *STOC*, pages 471–474. ACM, 1984.
- 4 Kazuyuki Amano. Bounds on the Size of Small Depth Circuits for Approximating Majority. In *ICALP (1)*, Lecture Notes in Computer Science, pages 59–70. Springer, 2009.
- 5 Sanjeev Arora and Boaz Barak. *Computational complexity : a modern approach*. Cambridge University Press, 2009.
- 6 Roger C. Baker, Glyn Harman, and János Pintz. The difference between consecutive primes, II. *Proceedings of the London Mathematical Society*, 83:532–562, 2001.
- 7 Joshua Brody and Elad Verbin. The Coin Problem and Pseudorandomness for Branching Programs. In *FOCS*, pages 30–39. IEEE Computer Society, 2010.
- 8 Gil Cohen, Anat Ganor, and Ran Raz. Two Sides of the Coin Problem. In *APPROX-RANDOM*, volume 28 of *LIPICs*, pages 618–629. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2014.
- 9 Merrick L. Furst, James B. Saxe, and Michael Sipser. Parity, Circuits, and the Polynomial-Time Hierarchy. *Mathematical Systems Theory*, 17(1):13–27, 1984. doi:10.1007/BF01744431.
- 10 Alexander Golovnev, Rahul Ilango, Russell Impagliazzo, Valentine Kabanets, Antonina Kolokolova, and Avishay Tal. $AC^0[p]$ Lower Bounds Against MCSP via the Coin Problem. In *46th International Colloquium on Automata, Languages, and Programming, ICALP 2019, July 9-12, 2019, Patras, Greece.*, pages 66:1–66:15, 2019.
- 11 John Hastad. Almost Optimal Lower Bounds for Small Depth Circuits. *Advances in Computing Research*, 5:143–170, 1989.
- 12 Pavel Hrubes, Sivaramakrishnan Natarajan Ramamoorthy, Anup Rao, and Amir Yehudayoff. Lower Bounds on Balancing Sets and Depth-2 Threshold Circuits. In *46th International Colloquium on Automata, Languages, and Programming, ICALP 2019, July 9-12, 2019, Patras, Greece.*, pages 72:1–72:14, 2019.
- 13 Svante Janson. Poisson Approximation for Large Deviations. *Random Struct. Algorithms*, 1(2):221–230, 1990.
- 14 Nutan Limaye, Karteek Sreenivasaiah, Srikanth Srinivasan, Utkarsh Tripathi, and S. Venkitesh. A fixed-depth size-hierarchy theorem for $AC^0[\oplus]$ via the coin problem. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC 2019, Phoenix, AZ, USA, June 23-26, 2019.*, pages 442–453, 2019.
- 15 Ryan O’Donnell and Karl Wimmer. Approximation by DNF: Examples and Counterexamples. In *ICALP*, volume 4596 of *Lecture Notes in Computer Science*, pages 195–206. Springer, 2007.
- 16 Igor Carboni Oliveira, Rahul Santhanam, and Srikanth Srinivasan. Parity Helps to Compute Majority. In *34th Computational Complexity Conference, CCC 2019, July 18-20, 2019, New Brunswick, NJ, USA.*, pages 23:1–23:17, 2019.
- 17 Alexander A. Razborov. On the Method of Approximations. In *STOC*, pages 167–176. ACM, 1989.
- 18 Benjamin Rossman. On the constant-depth complexity of k-clique. In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing, Victoria, British Columbia, Canada, May 17-20, 2008*, pages 721–730, 2008. doi:10.1145/1374376.1374480.
- 19 Benjamin Rossman and Srikanth Srinivasan. Separation of $AC^0[\oplus]$ Formulas and Circuits. In *ICALP*, volume 80 of *LIPICs*, pages 50:1–50:13. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2017.
- 20 Ronen Shaltiel and Emanuele Viola. Hardness Amplification Proofs Require Majority. *SIAM J. Comput.*, 39(7):3122–3154, 2010. doi:10.1137/080735096.

22:14 $AC^0[\oplus]$ and Variants of the Majority Function

- 21 Roman Smolensky. Algebraic Methods in the Theory of Lower Bounds for Boolean Circuit Complexity. In *STOC*, pages 77–82. ACM, 1987.
- 22 Roman Smolensky. On Representations by Low-Degree Polynomials. In *FOCS*, pages 130–138. IEEE Computer Society, 1993.
- 23 Emanuele Viola. Randomness Buys Depth for Approximate Counting. *Computational Complexity*, 23(3):479–508, 2014. doi:10.1007/s00037-013-0076-6.
- 24 Andrew Chi-Chih Yao. Separating the Polynomial-Time Hierarchy by Oracles (Preliminary Version). In *26th Annual Symposium on Foundations of Computer Science, Portland, Oregon, USA, 21-23 October 1985*, pages 1–10, 1985.