Electronic Colloquium on Computational Complexity, Report No. 62 (2012)

Average-Case Lower Bounds for Formula Size

Ilan Komargodski* Ran Raz*

Abstract

We give an explicit function $h : \{0,1\}^n \to \{0,1\}$ such that any deMorgan formula of size $O(n^{2.499})$ agrees with h on at most $\frac{1}{2} + \epsilon$ fraction of the inputs, where ϵ is exponentially small (i.e. $\epsilon = 2^{-n^{\Omega(1)}}$). Previous lower bounds for formula size were obtained for exact computation.

The same technique also shows that any boolean formula of size $O(n^{1.999})$ over the complete basis, agrees with h on at most $\frac{1}{2} + \epsilon$ fraction of the inputs, where ϵ is exponentially small (i.e. $\epsilon = 2^{-n^{\Omega(1)}}$).

Our construction is based on Andreev's $\Omega(n^{2.5-o(1)})$ formula size lower bound that was proved for the case of exact computation [And87].

1 Introduction

In this paper we shall deal with deMorgan formulas. A *deMorgan formula* is a boolean formula over the basis $B_2 = \{\vee, \wedge, \neg\}$ with fan in at most 2. A deMorgan formula is represented by a tree such that every leaf is labeled by an input variable and every internal node is labeled by an operation from B_2. A formula is said to compute a function $f : \{0,1\}^n \to \{0,1\}$ if on input $x \in \{0,1\}^n$ it outputs $f(x)$. The computation is done in the natural way from the leaves to the root. The size of a formula is defined as the number of leaves it contains.

The research on lower bounds for deMorgan formulas has focused on worst case computation. A worst case computation of a function $f : \{0,1\}^n \to \{0,1\}$ is a computation in which a formula F has to compute f correctly on *every* input. Various results for specific functions with polynomial lower bounds have been obtained in this model. The earliest results were of [Sub61] that proved an $\Omega(n^{1.5})$ lower bound and [Khr71] that proved an $\Omega(n^2)$ lower bound. Later on, Andreev proved an $\Omega(n^{2.5-o(1)})$ lower bound [And87]. Andreev's result was gradually improved by [IN93, PZ93], and was further improved by Håstad to an $\Omega(n^{3-o(1)})$ lower bound [Has98], which is the best known to date.

An approximate computation of a function $f : \{0,1\}^n \to \{0,1\}$ by a formula F is a computation in which F computes f correctly on some fraction larger than $1/2$ of the inputs (rather than on all inputs). Besides being interesting in their own right, lower bounds for

[*] Department of Computer Science and Applied Mathematics, Weizmann Institute of Science, Rehovot 76100, Israel. Email:{ilan.komargodski,ran.raz}@weizmann.ac.il. Research supported by an Israel Science Foundation grant and by the I-CORE Program of the Planning and Budgeting Committee and the Israel Science Foundation.

ISSN 1433-8092

approximate computation have proved useful in many fields of complexity theory, such as derandomization (e.g, [Nis91, NW94]). Lower bounds for approximate computation are also known as correlation bounds and average-case hardness.

In this paper, we focus on lower bounds for approximation by deMorgan formulas. We construct an explicit function $f : \{0,1\}^n \to \{0,1\}$ such that any deMorgan formula of size at most $O(n^{2.499})$ computes f correctly on a fraction of at most $\frac{1}{2} + 2^{-n^{\Omega(1)}}$ of the inputs. The same techniques can be used to show that any boolean formula of size $O(n^{1.999})$ over the complete basis, computes f correctly on at most $\frac{1}{2} + 2^{-n^{\Omega(1)}}$ fraction of the inputs.

1.1 Techniques

The average-case hard function that we construct is based on a construction known as Andreev's function introduced in [And87], and used to prove the lower bounds in [And87, IN93, PZ93, Has98]. Andreev's function is a function $A : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ that works as follows. Split the second input into $\log n$ parts of equal size. In each part compute the XOR of the input bits. Use the resulting $\log n$ bits to address an index in the first input ($\log n$ bits are enough to represent a cell in a vector of length n) and return that bit. The analysis of [And87, IN93, PZ93, Has98] relies on the fact that most n bit vectors represent boolean functions which are hard to compute by formulas of size $o(n)/\log\log n$.

Our construction of a function $h : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ which is hard to approximate by deMorgan formulas is a variant of the function A. We first need to make sure that most of the functions represented by the first input are hard to approximate by deMorgan formulas. This task is accomplished by a good error correcting code (ECC). We prove that if we encode the first input with a good ECC, it is almost always correct that the resulting string represents a function which is hard to approximate by deMorgan formulas of size roughly $o(n)/\log n$. This fact is proved using the Johnson bound. Since the first input, after being encoded, is longer than n bits, we need to split the second input into $r > \log n$ parts of equal size (rather than $\log n$ parts as in the function A). In conclusion, applying an ECC on the first input, and then splitting the second input into r sets of equal sizes, gives the construction as in Andreev's function. For a formal definition of h, see Section 4. For a proof that most strings $x \in \{0,1\}^n$ after being encoded by a good ECC represent functions which are hard to approximate, see Section 5.

With this construction in mind, we follow the general method introduced by [And87] to prove the lower bound. [And87] used the shrinkage property of deMorgan formulas proved by [Sub61]. In the proof of [And87] it was enough that a formula shrinks well in expectation. Since we are dealing with lower bounds for approximation, we need to improve this shrinkage property. We show that a formula shrinks well with probability exponentially close to 1. In order to prove this result, we analyze the shrinkage process in a more delicate way by breaking it into steps such that in every step only one variable is restricted. Having this process, we use the Azuma inequality to prove that with very high probability the formula shrinks well.

Although this technique seems natural, we run into technical issues which make the details of the proof non-trivial. One such technical issue stems from the fact that in order to get significant results after applying Azuma inequality, we need to keep the difference between the formula sizes in every two consecutive steps as small as possible. In a formula F it is possible that a variable appears in many leaves (we call such a variable a heavy variable).

Restricting by a heavy variable makes F shrink by more than one expects when restricting according to a random variable. In other words the difference between the size of F before the restriction and the size of F after the restriction may be relatively large. It follows, that applying the Azuma inequality on the naïve sequence of random restrictions (at every step restricting according to a random variable), does not give results which are good enough.

In order to solve this problem we define a random restriction process which at every step takes into account the structure of the formula as follows. At every step, if there are heavy variables, it restricts according to one of them, and if there is no heavy variable, it restricts according to a random variable. We define our steps that go into the Azuma inequality to contain only those steps in which non-heavy variables were removed. In this way we ensure that the sequence of chosen steps is both a supermartingale and has bounded (small) difference. For the formal definition of the process and the proof that deMorgan formulas shrink well with very high probability see Section 6.

Recall the definition of the hard function h that we construct. h splits the second input into r parts and XORs each part. These r bits are used to address an entry in the first input of h after being encoded by an ECC. Recall that the restriction process described above is not completely random and depends on the structure of the formula. We think of every variable that was chosen to be restricted because of being heavy, as chosen by an adversary (rather than at random). Since many of the restricted variables were chosen by an adversary, one could think that after the restriction process, the adversary can fix a large number of the parts, with non-negligible probability. If this happens for a large enough number of parts, then the function h may become easy to approximate. We prove that a large number of parts remain with at least one variable unassigned, with very high probability. We prove that by a series of reductions to bins and balls adversary games. For the details see Section 7.

1.2 Related Works

We have learnt that an independent work by Impagliazzo, Meka and Zukerman [IMZ12] also proves a theorem that shows that (in several models of computation) shrinkage occurs with very high probability (rather than in expectation). Their proof is obtained more generally for any model with shrinkage properties, and in particular for deMorgan formulas. Their theorem is related to our Theorem 6.6 (that is obtained for deMorgan formulas).

Moreover, in [IMZ12] the theorem is proved for certain pseudorandom distributions and is used to construct pseudorandom generators with seed of length $O(s)$ for deMorgan formulas of size $s^{3-o(1)}$, for boolean formulas of size $s^{2-o(1)}$ over the complete basis, as well as for several other models. Their result can also be used to prove average-case lower bounds[1].

2 Preliminaries

We start with some general notations. Throughout the paper we will only consider deMorgan formulas and not always explicitly mention it. We denote by $[n]$ the set of numbers $\{1, 2, \dots, n\}$. For $i \in [n]$ and for $x \in \{0,1\}^n$, denote by x_i the i-th bit of x.

[1] Private communication with the authors.

Boolean Formulas

Definition 2.1. *A deMorgan formula is a boolean formula with AND, OR and NOT gates with fan in at most 2.*

Definition 2.2. *The size of a formula F is the number of leaves in it and is denoted by $L(F)$. For a function $f : \{0,1\}^n \to \{0,1\}$, we will denote by $L(f)$ the size of the smallest formula computing the function f.*

Consider a formula F. Let q be a node in F (q can be either an internal node or a leaf). We refer to the tree rooted at q as a subformula of F or a subtree of F.

Let x_i be a variable that appears as a leaf in a formula F. Let g be a subtree (rooted at any internal node of F) of the formula F of the form $g = x_i \vee g_1$ or $g = x_i \wedge g_1$ where g_1 is a subformula of g. We call g_1 a sibling subtree of a leaf labeled by x_i (a sibling subtree of x_i, in short) or a neighbor subtree of x_i.

Average-Case Hardness

Definition 2.3. *A function $f : \{0,1\}^n \to \{0,1\}$ is said to be (s, ε)-hard if for any deMorgan formula F of size at most s*

$$\Pr_{x \in \{0,1\}^n}[F(x) = f(x)] \leq \frac{1}{2} + \varepsilon$$

Probability

We begin by stating some well known variants of Chernoff bound.

Proposition 2.4 (Chernoff Bound)**.** *Let $X = \sum_{i=1}^{n} X_i$ be a sum of identically distributed independent random variables $X_1, \ldots, X_n \in \{0,1\}$. Let $\mu = \mathbb{E}[X] = \sum_{i=1}^{n} \mathbb{E}[X_i]$. It holds that for $\delta \in (0,1)$,*

$$\Pr[X < (1-\delta)\mu] \leq \exp\left(-\delta^2 \mu/2\right)$$

and

$$\Pr[X > (1+\delta)\mu] \leq \exp\left(-\delta^2 \mu/3\right)$$

and for $a > 0$,

$$\Pr[X > \mu + a] \leq \exp(-2a^2/n)$$

We define the concept of a supermartingale.

Definition 2.5. *A* ***supermartingale*** *is a sequence of random variables $X_0, X_1, \ldots$ such that*

$$\mathbb{E}[X_i | X_0, \ldots, X_{i-1}] \leq X_{i-1}$$

The Azuma inequality (see e.g. in [DP09]) gives a concentration result for the value of supermartingales that have bounded differences. Formally,

Proposition 2.6 (Azuma Inequality). *Let $X_0, X_1, \dots$ be a supermartingale such that for every $i \in \{1, 2, \dots\}$ there exists some nonnegative c_i such that $|X_i - X_{i-1}| \leq c_i$. Then, for every $t > 0$ and every k ,*

$$\Pr[X_k \geq X_0 + t] \leq e^{\frac{-t^2}{2\sum_{i=1}^{k} c_i^2}}$$

We define hypergeometric distribution.

Definition 2.7 (Hypergeometric Distribution). *The hypergeometric distribution $H(N, M, n)$ describes the number of red balls drawn in an experiment where n balls are sampled without replacement from a bin containing N balls, M of which are red.*

We state a concentration of measure theorem for hypergeometric distributions (see [DP09], Chapter 7).

Proposition 2.8. *Let $H = H(N, M, n)$ be a hypergeometric distribution as in definition 2.7. Let X be a random variable distributed according to H. It holds that,*

$$\Pr\left[|X - \mathbb{E}[X]| > t\right] \leq \exp\left(\frac{-2(N-1)t^2}{(N-n)(n-1)}\right)$$

Using $t = \epsilon\,\mathbb{E}[X] = \epsilon\frac{M}{N}n$ we get that

$$\Pr\left[\left|X - \frac{M}{N}n\right| > \epsilon\frac{M}{N}n\right] \leq \exp\left(\frac{-2(N-1)\left(\epsilon\frac{M}{N}n\right)^2}{(N-n)(n-1)}\right) \leq \exp\left(-2\left(1 - \frac{1}{N}\right)\epsilon^2\frac{M^2 n}{N(N-n)}\right)$$

Assuming $N > 2$, we get

$$\Pr\left[\left|X - \frac{M}{N}n\right| > \epsilon\frac{M}{N}n\right] \leq \exp\left(-\epsilon^2\frac{M^2 n}{N(N-n)}\right)$$

We state Jensen inequality.

Proposition 2.9 (Jensen inequality). *If X is a random variable and f is concave, then*

$$\mathbb{E}[f(X)] \leq f(\mathbb{E}[X])$$

Coding Theory

Definition 2.10. *A linear code C over $\{0, 1\}$ that has block length n, dimension k and minimal distance d is denoted as an $[n, k, d]_2$ code. A linear code C can be thought of as a linear mapping from k bits to n bits such that every two output strings of the mapping differ in at least d bits. The mapping procedure is sometimes referred to as the encoding function of C. The relative distance of C is $\delta = d/n$*

Definition 2.11. *Let $0 \leq \rho \leq 1$ and $L \geq 1$. A code $C \subset \{0, 1\}^n$ is (ρ, L)-list decodable if for every $y \in \{0, 1\}^n$,*

$$|\{c \in C\,|\Delta(y, c) \leq \rho n\}| \leq L$$

where Δ denotes the Hamming distance.

Next, we state the well known Johnson bound for codes with binary alphabet. This version of the bound was taken from [Rud07] for the case of binary alphabet.

Proposition 2.12 (Johnson Bound)**.** *Let $C \subseteq \{0,1\}^n$ be an $[n,k,d]_2$ code with relative distance $\delta = d/n$. It holds that C is $(\rho, 2dn)$-list decodable for any*

$$\rho < \frac{1}{2}\left(1 - \sqrt{1-2\delta}\right)$$

3 Main Theorem

In this section we state our main theorem.

Theorem 3.1. *There exists an explicit function $h : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ such that for every formula F of size at most $O(n^{2.499})$, it holds that*

$$\Pr_{x,y \in \{0,1\}^n}[F(x,y) = h(x,y)] \le \frac{1}{2} + \frac{1}{2^{n^{\Omega(1)}}}$$

4 Definition of h

In this section we define the function h.

The function $h : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ that we will consider is defined as follows. Let r be such that $100 \log n \le r \le o(n)$. We assume for simplicity that r divides n. Let $\mathcal{C}$ be a $[2^r, n, d]_2$ code with relative distance $\delta = \frac{d}{2^r}$. Let $r' = r/4$. Assume that $d = \left(\frac{1}{2} - \frac{1}{2 \cdot 2^{r'-2}}\right) 2^r$ and $\delta = \frac{1}{2} - \frac{1}{2 \cdot 2^{r'-2}}$.

Denote the encoding of $x \in \{0,1\}^n$ using $\mathcal{C}$ by $Enc_x^{\mathcal{C}}$. We may view $Enc_x^{\mathcal{C}}$ both as a truth table of a function from r bits to 1 bit, or as a vector of length 2^r.

Denote by $x \in \{0,1\}^n$ and $y \in \{0,1\}^n$ the first and second inputs of h, respectively. Split y into parts of size $\frac{n}{r}$. For each part calculate the XOR of its bits. Using the resulting r bits, return the appropriate value, indexed by the binary value of the bits, in $Enc_x^{\mathcal{C}}$. Formally, for r as above and for $m = \frac{n}{r}$,

$$h(x,y) = Enc_x^{\mathcal{C}}\left(\bigoplus_{j=1}^{m} y_j, \bigoplus_{j=m+1}^{2m} y_j, \ldots, \bigoplus_{j=(r-1)m+1}^{rm} y_j\right)$$

5 Hardness of Most Inputs

In this section we prove two theorems. The first theorem states that for most of the inputs $x \in \{0,1\}^n$, $Enc_x^{\mathcal{C}}$ represents a hard to approximate function of r bits.

Theorem 5.1. *Recall that $r' = r/4$. Let $n' = \frac{o(n)}{\log r}$. Denote by $\mathcal{H} \subseteq \{0,1\}^n$ the following set of vectors.*

$$\mathcal{H} = \left\{x \in \{0,1\}^n \middle| Enc_x^{\mathcal{C}} \text{ is } \left(n', \frac{1}{2^{r'/2}}\right)\text{-hard}\right\}$$

It holds that for some fixed $\delta \leq 2^{o(n)}$,

$$|\mathcal{H}| \geq 2^n - \delta$$

The second theorem states that if $h(x, y)$ is easy to compute, then there must be some $x_0 \in \mathcal{H}$ such that h_{x_0} is also quite easy to compute.

Theorem 5.2. *Let $\delta/2^n \leq \epsilon < \frac{1}{2}$ (the δ is the same δ as in Theorem 5.1). Let $F(x, y)$ be a formula whose size is s such that $\Pr_{x,y\in\{0,1\}^n}[F(x, y) = h(x, y)] \geq \frac{1}{2} + \epsilon$. Denote by $h_{x_0}(y) : \{0,1\}^n \to \{0,1\}$ the function $h(x, y)$ when the input x is fixed to x_0. There exists some $x_0 \in \mathcal{H}$ such that $\Pr_{y\in\{0,1\}^n}[F(x_0, y) = h_{x_0}(y)] \geq \frac{1}{2} + \frac{\epsilon}{2}$.*

First, we prove the first theorem.

Proof of Theorem 5.1. In order to analyze the size of $\mathcal{H}$, we will use the Johnson bound stated in Proposition 2.12. Applying it to $\mathcal{C}$ we get that for $\rho < \frac{1}{2}\left(1 - \sqrt{1 - 2\left(\frac{1}{2} - \frac{1}{2\cdot 2^{r'-2}}\right)}\right) = \frac{1}{2}\left(1 - \sqrt{\frac{1}{2^{r'-2}}}\right) = \frac{1}{2} - \frac{1}{2^{r'/2}}$ it holds that $\mathcal{C}$ is $(\rho, \mathrm{poly}(2^r))$-list decodable.

The theorem is proved by counting the number of possible easy to approximate functions compared to the number of different functions that $Enc_x^{\mathcal{C}}$ can represent.

Denote by D the set of all functions $f : \{0,1\}^r \to \{0,1\}$ such that $L(f) \leq n'$. We will now upper bound the size of the set D. Following the calculation in [Juk12] (see Theorem 1.23), we get that for n' there are at most $4^{n'}(2\cdot r+2)^{n'} < (9r)^{n'}$ different deMorgan formulas for functions of r variables and with at most n' leaves.

Think of every function $f \in D$, as above, as a vector of length 2^r. For every such vector, denote by $B(f, \gamma)$ the hamming ball that contains all vectors whose hamming distance from f is at most $\gamma 2^r$. From the calculation above, we know that for every function, f, $\left|B\left(f, \frac{1}{2} - \frac{1}{2^{r'/2}}\right) \cap \mathcal{C}\right| \leq \mathrm{poly}(2^r)$. Hence, by a simple union bound,

$$\left|\bigcup_{f\in D} B\left(f, \frac{1}{2} - \frac{1}{2^{r'/2}}\right) \cap \mathcal{C}\right| \leq \mathrm{poly}(2^r)\cdot|D| \leq \mathrm{poly}(2^r)\cdot 9^{n'}2^{n'\log r} \tag{5.1}$$

Recall that $r \leq o(n)$ and $n' = \frac{o(n)}{\log r}$. It follows, that there exists some fixed $\delta = \delta(n) \leq 2^{o(n)}$ such that the expression in equation 5.1 is not larger than δ. This implies that $|\mathcal{H}|$ is at least $2^n - \delta$. □

Next, we prove the second theorem of this section. We begin with a simple averaging lemma. The proof of this lemma can be found in the appendix.

Lemma 5.3. *Let $g, f : \{0,1\}^m \times \{0,1\}^{d-m} \to \{0,1\}$ be functions, and assume that*

$$\Pr_{\substack{u\in\{0,1\}^m \\ w\in\{0,1\}^{d-m}}}[g(u, w) = f(u, w)] \geq \gamma$$

Let $H \subseteq \{0,1\}^m$. Then, there exists $u_0 \in H$ such that

$$\Pr_{w\in\{0,1\}^{d-m}}[g(u_0, w) = f(u_0, w)] \geq (\gamma - 1)\frac{2^m}{|H|} + 1$$

Using this lemma, we can prove Theorem 5.2 that intuitively states that if $h(x, y)$ is easy to compute, then there must be some $x_0 \in \mathcal{H}$ such that h_{x_0} is also quite easy to compute.

Proof of Theorem 5.2. Let $F(x, y)$ be a formula whose size is s such that $\Pr_{x,y\in\{0,1\}^n}[F(x, y) = h(x, y)] \geq \frac{1}{2} + \epsilon$. From the averaging lemma (Lemma 5.3) there must exist some $x_0 \in \mathcal{H}$ such that

$$\Pr_{y\in\{0,1\}^n}[F(x_0, y) = h_{x_0}(y)] \geq \left(\frac{1}{2} + \epsilon - 1\right)\frac{2^n}{|\mathcal{H}|} + 1 = \left(\epsilon - \frac{1}{2}\right)\frac{2^n}{|\mathcal{H}|} + 1$$

Notice that for $\epsilon < \frac{1}{2}$ it holds that $\epsilon - \frac{1}{2} < 0$. Recall that we proved that $|\mathcal{H}|$ is at least $2^n - \delta$ which is at least $(1 - \epsilon)\,2^n$. Plugging this in we get that

$$\Pr_{y\in\{0,1\}^n}[F(x_0, y) = h_{x_0}(y)] \geq \frac{2\epsilon - 1}{2(1 - \epsilon)} + 1 = \frac{1}{2 - 2\epsilon} > \frac{1}{2} + \frac{\epsilon}{2}$$

as needed.

□

6 Shrinkage with High Probability

In this section our goal is to prove that deMorgan formulas shrink well with high probability (rather than in expectation). For this purpose we consider the restriction process as an iterative process that restricts only one variable at a time. After breaking up the restriction process into steps, we use the Azuma inequality to prove that large shrinkage happens with high probability. Azuma inequality doesn't work well enough as is on a standard random restriction process (in which a restriction is done uniformly at random at every step), so we have to overcome this technical difficulty by defining a more delicate random restriction process.

We begin with a standard definition and a warm-up application that will be useful during this section.

Definition 6.1. *Let $\mathcal{R}_k$ be the set of all partial assignments on n variables which leave exactly k variables unassigned. The probability distribution of restrictions from $\mathcal{R}_k$ is as follows: randomly choose $n - k$ variables and assign them to be 0 or 1 randomly and independently.*

We sketch the proof of the next simple lemma which is just one step of the proof of the shrinkage property of [Sub61].

Lemma 6.2. *Let $f : \{0,1\}^n \to \{0,1\}$ be a function. Let $\rho \in \mathcal{R}_{n-1}$ be a random restriction that assigns one random variable to 0 or 1 at random. It holds that given $L(f)$,*

$$\mathop{\mathbb{E}}_{\rho}[L(f_\rho)] \;\leq\; \left(1 - \frac{3}{2n}\right)L(f)$$

where f_ρ denotes the function f restricted by ρ.

Sketch. Observe that in a minimal size deMorgan formula F that computes f, it is not possible that a sibling of some leaf x_i will also contain x_i (otherwise, it is not minimal). In F a random input variable appears $L(f)/n$ times as a leaf in expectation. All of these disappear after applying the restriction. Moreover, we expect half of their siblings in the formula to disappear, since this is a deMorgan formula. In total, we stay with a formula of at most

$$L(f) - \frac{L(f)}{n} - \frac{L(f)}{2n} = L(f)\left(1 - \frac{3}{2n}\right)$$

leaves in expectation, as required by the lemma. □

Let $f : \{0,1\}^n \to \{0,1\}$ be a function computed by a formula F of size $L(F) = L(f)$. We break down the process of large random restrictions into small steps where in each step one variable is restricted according to some rules that we define later. We restrict the formula until there are exactly k variables left.

We define a set of restriction rules that will enable us to define a sequence of random variables that are supermartingale, and on the other hand have bounded difference. This will be helpful since we need to apply Azuma inequality on a sequence which is required to be both a supermartingale and has bounded difference. If we use a standard random restriction (in which we choose uniformly and randomly variables and restrict them) then we fail to have the bounded differences property.

Recall that k is defined as the number of variables left after the process finish. For every $i \in \{0, \ldots, n-k\}$ define the following. Let $f_i : \{0,1\}^{n-i} \to \{0,1\}$ and F_i be a sequence of functions and formulas, respectively. Assume $f_0 = f$ and $F_0 = F$. We define a process where in every step we assign exactly one variable such that after i steps we denote the resulting function by f_i and the resulting formula by F_i. We stress that for $i > 0$ the formula F_i is a result of some restriction of F_{i-1}.

Let # be a dummy variable that is not part of the inputs to the function f. In each step i we will have a set D_i of dummy leaves #. Initially, $D_0 = \emptyset$. We denote by F_i^* the formula F_i combined with the set of dummy leaves, D_i.

A restriction (assignment) of a specific variable is (unless otherwise stated) an assignment of 0 or 1 at random.

Let $k' = n^\alpha$ where $\alpha > 0$ is a constant that will be defined later. We will have that $2k' < k$. For $i \in \{1, \ldots, n-k\}$ denote by T_i the set of variables that appear in F_{i-1} at least $t = \frac{200k' L(F_{i-1})}{n-i+1}$ times as a leaf. We will refer to these variables as heavy variables. Now we are ready to define the set of restriction rules.

1. If $T_i \neq \emptyset$. Eliminate $\frac{3|D_{i-1}|}{2(n-i+1)}$ dummy leaves[2]. Then, assign the first variable from T_i at random and restrict the formula.

2. If $T_i = \emptyset$. Eliminate $\frac{3|D_{i-1}|}{2(n-i+1)}$ dummy leaves. Then, assign a random variable in the formula. Denote the variable we assigned by x_w.

   We allow the removal of all appearances of x_w from the formula as well as one of the leaves in a sibling subtree of x_w for every appearance of x_w. For any additional leaf that is eliminated due to the restriction, we add a dummy leaf.

[2] We assume for simplicity that the number of dummy leaves is divided by $2(n-i+1)$. If not, we can work with fractions of leaves.

This process defines a distribution on restrictions such that after applying it $n-k$ times, a formula is left with k unassigned variables. Denote this distribution on restrictions by $\mathcal{T}_k$.

We begin by proving that the shrinkage property of formulas after being restricted by one step of the process defined above still holds, in expectation.

Lemma 6.3. *Let $i \in \{1,\dots,n-k\}$. For a given F^*_{i-1},*

$$\mathbb{E}[L(F^*_i)] \leq L(F^*_{i-1}) \left(1 - \frac{1}{n-i+1}\right)^{3/2}$$

Where the expectation is taken over the random process described above.

Proof. Let $i \in \{1,\dots,n-k\}$. We will analyze each case separately and prove that the expected size of the formula at step i is bounded by what we need. Recall that $F^*_i = F_i \cup D_i$, hence, $L(F^*_i) = L(F_i) + |D_i|$.

Case 1 Assume that there is a variable that appears in at least t leaves in the formula F_{i-1}. Without loss of generality denote it by x_j. Notice that since x_j appears more than $t = \frac{200k' L(F_{i-1})}{n-i+1}$ times in the formula, $L(F_i) \leq L(F_{i-1}) \left(1 - \frac{200k'}{n-i+1}\right) \leq L(F_{i-1}) \left(1 - \frac{1}{n-i+1}\right)^{3/2}$.

Using linearity of expectation, it follows that,

$$\begin{aligned}
\mathbb{E}[L(F^*_i)] &= \mathbb{E}[L(F_i)] + \mathbb{E}[|D_i|] \\
&\leq L(F_{i-1}) \left(1 - \frac{1}{n-i+1}\right)^{3/2} + |D_{i-1}| \left(1 - \frac{3}{2(n-i+1)}\right) \\
&\leq (L(F_{i-1}) + |D_{i-1}|) \left(1 - \frac{1}{n-i+1}\right)^{3/2} \\
&= L(F^*_{i-1}) \left(1 - \frac{1}{n-i+1}\right)^{3/2}
\end{aligned}$$

Case 2 Assume that $T_i = \emptyset$. By rule number 2, we restrict F_{i-1} according to a random variable. Denote it by x_w. We allow the removal of the leaves labeled by x_w as well as one of the leaves in a sibling subtree of x_w, for every appearance of x_w. For any additional leaf that is eliminated due to the restriction, we add a dummy leaf. From Lemma 6.2, we know that

$$\begin{aligned}
\mathbb{E}[L(F^*_i)] &= \mathbb{E}[L(F_i)] + \mathbb{E}[|D_i|] \\
&\leq L(F_{i-1}) \left(1 - \frac{3}{2(n-i+1)}\right) + |D_{i-1}| \left(1 - \frac{3}{2(n-i+1)}\right) \\
&= (L(F_{i-1}) + |D_{i-1}|) \left(1 - \frac{3}{2(n-i+1)}\right) \\
&\leq L(F^*_{i-1}) \left(1 - \frac{1}{n-i+1}\right)^{3/2}
\end{aligned}$$

Since in both cases $\mathbb{E}[L(F^*_i)] \leq L(F^*_{i-1}) \left(1 - \frac{1}{n-i+1}\right)^{3/2}$, the claim follows. □

After breaking up the process of restrictions to small steps where in each step where we apply case 2 the size of the formula decreases by a bounded expression, we can use the Azuma inequality to prove that with very high probability the formula shrinks well.

Define the following sequence of random variables. Define $Z_0 = 0$. For $i \in \{1, \dots, n-k\}$,

$$Z_i = \log L(F_i^*) - \log L(F_{i-1}^*) - \frac{3}{2} \log \left(1 - \frac{1}{n-i+1}\right)$$

Let $S^1 \subseteq \{1, \dots, n-k\}$ be the set that contains indexes of the steps in which we chose case 1. Denote by $S^2 \subseteq \{0\} \cup \{1, \dots, n-k\}$ the set of indexes in which we chose case 2. We will denote the i-th element of S^1 by S_i^1 and the i-th element of S^2 by S_i^2. Define $S_0^2 = 0$.

Define the following set of random variables. Define $Y_0 = 0$. For $i \in S^2$ define

$$Y_i = \sum_{j \in S^2, j \leq i} Z_j$$

Next, we show that since we don't allow the number of leaves to decrease by too much in case 2, the sequence has bounded differences.

Lemma 6.4. *Let $i \in \{1, \dots, |S^2|\}$. There exists some large enough constant $c > 0$ such that*

$$\left| Y_{S_i^2} - Y_{S_{i-1}^2} \right| \leq c \frac{k'}{n - S_i^2 + 1}$$

Proof. Let $i \in \{1, \dots, |S^2|\}$. Recall that the $Y_{S_i^2}$ sequence was defined only on steps of the process in which a random variable was restricted (i.e. when case 2 was applied). By the definition of the process in case 2, we choose some random variable x_w and restrict according to it. We allow the removal of the leaves labeled by x_w as well as one of the leaves in a sibling subtree of x_w (for every appearance of x_w). For any additional leaf that is eliminated due to the restriction, we add a dummy leaf.

It follows that, every leaf labeled by x_w can at most eliminate one additional leaf (except for x_w itself). Then,

$$\log L(F_{S_i^2}^*) \geq \log L(F_{S_i^2 - 1}^*) + \log \left(1 - \frac{400k'}{n - S_i^2 + 1}\right)$$

It follows that

$$\begin{aligned} |Y_{S_i^2} - Y_{S_{i-1}^2}| &= \left| \log L(F_{S_i^2}^*) - \log L(F_{S_i^2 - 1}^*) - \frac{3}{2} \log \left(1 - \frac{1}{n - S_i^2 + 1}\right) \right| \\ &\leq \left| \log \left(1 - \frac{400k'}{n - S_i^2 + 1}\right) - \frac{3}{2} \log \left(1 - \frac{1}{n - S_i^2 + 1}\right) \right| \\ &\leq c \frac{k'}{n - S_i^2 + 1} \end{aligned}$$

where the last inequality holds for some large enough constant $c > 0$ and follows from the Taylor series for the ln function (recall that $2k' < k$).

□

Lemma 6.5. *For every $i \in \{1, \ldots, |S^2|\}$*

$$E[Y_{S^2_i} | Y_{S^2_0}, Y_{S^2_1}, \ldots, Y_{S^2_{i-1}}] \leq Y_{S^2_{i-1}}$$

In other words, the sequence $Y_{S^2_0}, Y_{S^2_1}, Y_{S^2_2}, \ldots$ is a supermartingale.

Proof. Let $i \in \{1, \ldots, |S^2|\}$. Recall that the Y's sequence was defined only on steps of the process in which a random variable was restricted, as follows.

$$Y_i = \sum_{j \in S^2, j \leq i} Z_j$$

Hence,

$$\begin{aligned}
\mathbb{E}[Y_{S^2_i} | Y_{S^2_0}, Y_{S^2_1}, \ldots, Y_{S^2_{i-1}}] &= \sum_{j=1}^{i} \mathbb{E}[Z_{S^2_j} | Y_{S^2_0}, Y_{S^2_1}, \ldots, Y_{S^2_{i-1}}] \\
&= \mathbb{E}[Z_{S^2_i} | Y_{S^2_0}, Y_{S^2_1}, \ldots, Y_{S^2_{i-1}}] + \sum_{j=1}^{i-1} Z_{S^2_j}
\end{aligned}$$

Since $\sum_{j=1}^{i-1} Z_{S^2_j} = Y_{S^2_{i-1}}$, it is enough to prove that $\mathbb{E}[Z_{S^2_i} | Y_{S^2_0}, Y_{S^2_1}, \ldots, Y_{S^2_{i-1}}] \leq 0$. By Jensen inequality,

$$\begin{aligned}
\mathbb{E}[Z_{S^2_i} | Y_{S^2_0}, Y_{S^2_1}, \ldots, Y_{S^2_{i-1}}] &= \mathbb{E}[\log L(F^*_{S^2_i}) | Y_{S^2_0}, Y_{S^2_1}, \ldots, Y_{S^2_{i-1}}] - \\
&\quad \mathbb{E}[\log L(F^*_{S^2_i - 1}) | Y_{S^2_0}, Y_{S^2_1}, \ldots, Y_{S^2_{i-1}}] - \frac{3}{2} \log \left(1 - \frac{1}{n - S^2_i + 1}\right) \\
&\leq \log \mathbb{E}[L(F^*_{S^2_i}) | Y_{S^2_0}, Y_{S^2_1}, \ldots, Y_{S^2_{i-1}}] - \\
&\quad \mathbb{E}[\log L(F^*_{S^2_i - 1}) | Y_{S^2_0}, Y_{S^2_1}, \ldots, Y_{S^2_{i-1}}] - \frac{3}{2} \log \left(1 - \frac{1}{n - S^2_i + 1}\right)
\end{aligned}$$

Using Lemma 6.3, we get that,

$$\begin{aligned}
& \log \mathbb{E}[L(F^*_{S_i^2})|Y_{S_0^2}, Y_{S_1^2}, \ldots, Y_{S_{i-1}^2}] - \mathbb{E}[\log L(F^*_{S_i^2-1})|Y_{S_0^2}, Y_{S_1^2}, \ldots, Y_{S_{i-1}^2}] = \\
= \; & \mathop{\mathbb{E}}_{F^*_{S_i^2-1}|Y_{S_0^2}, Y_{S_1^2}, \ldots, Y_{S_{i-1}^2}} \log \mathbb{E}[L(F^*_{S_i^2})|Y_{S_0^2}, Y_{S_1^2}, \ldots, Y_{S_{i-1}^2}, F^*_{S_i^2-1}] - \\
& \mathop{\mathbb{E}}_{F^*_{S_i^2-1}|Y_{S_0^2}, Y_{S_1^2}, \ldots, Y_{S_{i-1}^2}} \mathbb{E}[\log L(F^*_{S_i^2-1})|Y_{S_0^2}, Y_{S_1^2}, \ldots, Y_{S_{i-1}^2}, F^*_{S_i^2-1}] \\
= \; & \mathop{\mathbb{E}}_{F^*_{S_i^2-1}|Y_{S_0^2}, Y_{S_1^2}, \ldots, Y_{S_{i-1}^2}} \log \mathbb{E}[L(F^*_{S_i^2})|F^*_{S_i^2-1}] - \mathop{\mathbb{E}}_{F^*_{S_i^2-1}|Y_{S_0^2}, Y_{S_1^2}, \ldots, Y_{S_{i-1}^2}} \mathbb{E}[\log L(F^*_{S_i^2-1})|F^*_{S_i^2-1}] \\
\leq \; & \mathop{\mathbb{E}}_{F^*_{S_i^2-1}|Y_{S_0^2}, Y_{S_1^2}, \ldots, Y_{S_{i-1}^2}} \left[\log \left(L(F^*_{S_i^2-1}) \left(1 - \frac{1}{n - S_i^2 + 1}\right)^{3/2} \right)\right] - \\
& \mathop{\mathbb{E}}_{F^*_{S_i^2-1}|Y_{S_0^2}, Y_{S_1^2}, \ldots, Y_{S_{i-1}^2}} \log \text{\L}(F^*_{S_i^2-1}) \\
= \; & \mathop{\mathbb{E}}_{F^*_{S_i^2-1}|Y_{S_0^2}, Y_{S_1^2}, \ldots, Y_{S_{i-1}^2}} \left[\log \left(L(F^*_{S_i^2-1}) \left(1 - \frac{1}{n - S_i^2 + 1}\right)^{3/2} \right) - \log L(F^*_{S_i^2-1})\right] \\
= \; & \frac{3}{2} \log \left(1 - \frac{1}{n - S_i^2 + 1}\right)
\end{aligned}$$

which proves the claim. □

Recall that the process of random restrictions is executed until the formula is left with k variables. We are now ready to apply Azuma inequality on the sequence $Y_{S_0^2}, Y_{S_1^2}, \ldots$ and to prove the main theorem of this section.

Theorem 6.6. *Let c be the same constant as in Lemma 6.4. Then,*

$$\Pr\left[L(F^*_{n-k}) < 2^{\sqrt{2}c} \left(\frac{k}{n}\right)^{3/2} L(F_0)\right] > 1 - 2^{-k/k'^2}$$

Proof of Theorem 6.6. Denote $|S^2| = \mathcal{M} \leq n - k$. Let the sequence of random variables $Y_{S_0^2}, \ldots, Y_{S_{\mathcal{M}}^2}$ be defined as before. From Lemma 6.4 we know that for every $i \in [\mathcal{M}]$ it holds that for some constant $c > 0$ and large enough n

$$|Y_{S_i^2} - Y_{S_{i-1}^2}| \leq c \cdot \frac{k'}{n - S_i^2 + 1}$$

Using this combined with the fact that the sequence $Y_{S_0^2}, \ldots, Y_{S_{\mathcal{M}}^2}$ is a supermartingale (Lemma 6.5), we can apply Azuma inequality. For every $t \geq 0$

$$\Pr[Y_{S_{\mathcal{M}}^2} - Y_{S_0^2} \geq t] \leq \exp\left(\frac{-t^2}{2\sum_{i=1}^{\mathcal{M}} \left(\frac{ck'}{n - S_i^2 + 1}\right)^2}\right)$$

Notice that for $w \geq 2$ it holds that $\frac{1}{w^2} \leq \frac{1}{w-1} - \frac{1}{w}$. Then,

$$\begin{aligned}
\sum_{i=1}^{\mathcal{M}} \left(\frac{ck'}{n - S_i^2 + 1}\right)^2 &\leq \sum_{i=1}^{n-k} \left(\frac{ck'}{n-i+1}\right)^2 \\
&\leq (ck')^2 \sum_{i=1}^{n-k} \left(\frac{1}{n-i} - \frac{1}{n-i+1}\right) \\
&= (ck')^2 \left(\frac{1}{k} - \frac{1}{n}\right) \leq \frac{(ck')^2}{k}
\end{aligned}$$

It follows that,

$$\Pr[Y_{S^2_{\mathcal{M}}} - Y_{S^2_0} \geq t] \;\leq\; \exp\left(\frac{-kt^2}{2(ck')^2}\right) \tag{6.1}$$

Notice that all the Z_i's for $i \in S^1$ correspond to steps in which we chose variables that appear in many leaves (heavy variables). Then for $i \in S^1$ it holds that $\log L(F_i^*) - \log L(F_{i-1}^*) \leq \frac{3}{2} \log\left(1 - \frac{1}{n-i+1}\right)$. It follows that,

$$Z_i = \log L(F_i^*) - \log L(F_{i-1}^*) - \frac{3}{2}\log\left(1 - \frac{1}{n-i+1}\right) \leq 0$$

Since $Y_{S^2_0} = 0$ we get,

$$\begin{aligned}
\Pr\left[Y_{S^2_{\mathcal{M}}} - Y_{S^2_0} \geq t\right] &= \Pr\left[\sum_{i \in S^2 \setminus \{0\}} \left(\log L(F_i^*) - \log L(F_{i-1}^*) - \frac{3}{2}\log\left(\frac{n-i}{n-i+1}\right)\right) \geq t\right] \\
&\geq \Pr\left[\sum_{i \in S^1} \left(\log L(F_i^*) - \log L(F_{i-1}^*) - \frac{3}{2}\log\left(\frac{n-i}{n-i+1}\right)\right) + \right. \\
&\qquad \left. \sum_{i \in S^2 \setminus \{0\}} \left(\log L(F_i^*) - \log L(F_{i-1}^*) - \frac{3}{2}\log\left(\frac{n-i}{n-i+1}\right)\right) \geq t\right] \\
&= \Pr\left[\sum_{i=1}^{n-k} \left(\log L(F_i^*) - \log L(F_{i-1}^*) - \frac{3}{2}\log\left(\frac{n-i}{n-i+1}\right)\right) \geq t\right] \\
&= \Pr\left[\log L(F_{n-k}^*) - \log L(F_0^*) - \frac{3}{2}\sum_{i=1}^{n-k}\log\left(\frac{n-i}{n-i+1}\right) \geq t\right] \\
&= \Pr\left[\log L(F_{n-k}^*) - \log L(F_0^*) - \frac{3}{2}\log\left(\frac{k}{n}\right) \geq t\right] \\
&= \Pr\left[L(F_{n-k}^*) \geq 2^t \left(\frac{k}{n}\right)^{3/2} L(F_0)\right]
\end{aligned}$$

Combining this with equation (6.1) we get that for $t = \sqrt{2}c$,

$$\Pr\left[L(F_{n-k}^*) \geq 2^t \left(\frac{k}{n}\right)^{3/2} L(F_0)\right] \leq e^{-k/k'^2} \leq 2^{-k/k'^2}$$

Or, equivalently,

$$\Pr\left[L(F^*_{n-k}) < 2^{\sqrt{2}c}\left(\frac{k}{n}\right)^{3/2} L(F_0)\right] > 1 - 2^{-k/k'^2}$$

□

7 Most Restrictions are Good Enough

We now turn to proving that most restrictions leave at least one variable alive in a large enough fraction of the r parts that we XOR in h.

Theorem 7.1. *Let $f : \left(\{0,1\}^{n/r}\right)^r \to \{0,1\}$. We view f as a function whose input variables are partitioned into r bins with exactly n/r balls in each. Let $\rho \in \mathcal{T}_k$ be a restriction as described in Section 6.*

Recall that k represents the number of variables left in the formula after the restriction, and $2k' < k$. Then, with probability at least $1 - r2^{-k/(4r^2)} - (\log n) \cdot 2^{-k/k'^2}$ the restriction leaves at least $\left(1 - \frac{220\log^2 n}{k'}\right)$ fraction of the r bins with at least one variable unset.

First, let us prove the following claim that states that if a function $f : \{0,1\}^r \to \{0,1\}$ is very hard to approximate, then it is also hard to approximate when some of its inputs are fixed.

Claim 7.2. *Let $R = p \cdot r$ where $0 < p \leq 1$. Let $\epsilon > 0$. Let $f : \{0,1\}^r \to \{0,1\}$ be a function. Let $f' : \{0,1\}^R \to \{0,1\}$ be a function, which is f restricted to R input bits (that is, the other $r - R$ bits are fixed to some values). Assume that there exists a formula F' such that*

$$\Pr[F'(x) = f'(x)] \geq \frac{1}{2} + \epsilon$$

It holds that there exists a formula F of size $L(F') + 2(r - R)$ such that

$$\Pr[F(x) = f(x)] \geq \frac{1}{2} + \frac{\epsilon}{2^{r-R}}$$

Proof. Assume without loss of generality that f depends on $x_1, \ldots, x_r \in \{0,1\}$ and that f' depends only on $x_1, \ldots, x_R$. This means that there is some assignment to $x_{R+1}, \ldots, x_r$, denoted by $y_{R+1}, \ldots, y_r$ such that for every $x_1, \ldots, x_R \in \{0,1\}$

$$f(x_1, \ldots, x_R, y_{R+1}, \ldots, y_r) = f'(x_1, \ldots, x_R)$$

Assume that there exists a formula F' of size $L(F')$ such that

$$\Pr_{x \in \{0,1\}^R}[F'(x) = f'(x)] \geq \frac{1}{2} + \epsilon$$

Denote by $e \in \{0,1\}$ the following value

$$e = \text{majority}\{f(x_1, \ldots, x_r) \,|x_1, \ldots, x_r \in \{0,1\}, x_{R+1} \ldots x_r \neq y_{R+1} \ldots y_r\}$$

We construct the following simple formula F for f. On input $x \in \{0,1\}^r$ such that $x_{R+1} \dots x_r = y_{R+1} \dots y_r$ return $F'(x_1, \dots, x_R)$. Otherwise, return e.

$$\begin{aligned}\Pr_{x \in \{0,1\}^r}[f(x) = F(x)] &= \Pr[x_{R+1} \dots x_r = y_{R+1} \dots y_r] \Pr[f(x) = F(x) | x_{R+1} \dots x_r = y_{R+1} \dots y_r] + \\ &\quad \Pr[x_{R+1} \dots x_r \neq y_{R+1} \dots y_r] \Pr[f(x) = F(x) | x_{R+1} \dots x_r \neq y_{R+1} \dots y_r] \\ &\geq \frac{1}{2^{r-R}} \left(\frac{1}{2} + \epsilon \right) + \left(1 - \frac{1}{2^{r-R}} \right) \cdot \frac{1}{2} \geq \frac{1}{2} + \frac{\epsilon}{2^{r-R}}\end{aligned}$$

To perform the check $x_{R+1} \dots x_r = y_{R+1} \dots y_r$, F only needs additional $2(r-R)$ leaves. Notice that the calculation of e is not part of F. It follows that the size of F is $L(F') + 2(r-R)$. □

At this point we are ready to start working on proving Theorem 7.1. Recall the definition of the hard function h. There are r sets of size n/r such that h XORs every such set. We want to prove that with high probability not too many sets are completely assigned (restricted) and then using Claim 7.2, we will get that with high probability the restricted formula must still be hard.

We analyze the restriction process by partitioning it into $M = \log \frac{n}{k}$ intervals. We assume that n and k are powers of 2 for simplicity. The length of the first interval is defined to be $n/2$, the length of the second is defined to be $n/4$ and so on. In general, the length of the i-th interval ($i \in [M]$) is defined to be $n/2^i$. Since the process stops when there are k variables left, the last interval consists of k steps. In total we have M intervals. Denote by $I_i = n/2^i$ the length of the i-th interval. After the M-th interval there are k variables left, which we will call the leftover variables.

We begin by a simple lemma that states that if we apply a restriction $\rho \in \mathcal{T}_k$ to a formula F of size at most n^9, it is not possible that too many heavy variables are assigned in one interval.

Lemma 7.3. *Let F be a formula of size at most n^9. Let $\rho \in \mathcal{T}_k$ be a restriction. At any interval $i \in [M]$, it is not possible that more than $\frac{100 I_i \log n}{k'}$ heavy variables are restricted.*

Proof. Let $i \in [M]$. Assume that more than $\frac{100 I_i \log n}{k'}$ heavy variables are assigned during interval i. Assume we begin interval i with a formula F'. Notice that at any step during the i-th interval there are at least I_i variables in the formula and at most $2I_i$ that are still not restricted. So, the size of the formula at the end of the interval is at most

$$\begin{aligned}L(F') \left(1 - \frac{200k'}{2I_i} \right)^{\frac{I_i}{k'} 100 \log n} &= L(F') \left(1 - \frac{100k'}{I_i} \right)^{\frac{I_i}{k'} 100 \log n} \\ &\leq L(F') e^{-10000 \log n} \\ &\leq L(F') n^{-10000} < 1\end{aligned}$$

This result means that the size of the formula F is now 0 and it cannot be possible that an additional heavy variable was restricted. This is a contradiction which proves the claim. □

From now on we will assume that this is indeed the case. Meaning that, there is no interval $i \in [M]$ in which more than $\frac{100 I_i \log n}{k'}$ heavy variables were assigned. This assumption is valid, since the formulas that we will be working with are of size $O(n^{2.499})$.

Recall that we are trying to prove that not too many parts which h XORs are completely restricted by a random restriction $\rho \in \mathcal{T}_k$. We can restate this problem in equivalent terms of an adversarial game G, as follows. There are r bins, in each $\frac{n}{r}$ balls. The goal of the adversary is to completely empty more than $p \cdot r$ bins. The adversary can choose one of the following two moves at each step. The first (r-move) is to choose a random ball to remove (uniformly at random from all the balls that are left). The second (s-move) is to remove a specific ball that the adversary chooses. The only restriction is that at any interval $i \in [M]$ (as defined above) the adversary is allowed to apply the s-move only $\frac{100 I_i \log n}{k'}$ times. Recall that for $\rho \in \mathcal{T}_k$ we do $n-k$ restriction steps.

We now turn to the analysis of the game. Define the *score* of adversary $\mathcal{A}$ playing game G by,

$$\mathit{score}_G^{\mathcal{A}} = \Pr\left[\text{More than } p \cdot r \text{ bins are completely empty after } \mathcal{A} \text{ plays } G\right]$$

where the probability is over the random choices made during the game. An optimal player for the game G is defined as a player that maximizes the *score* of the game G (over all adversaries). Denote by score_G the score of the game G when played by an optimal adversary.

We first prove that we can simulate the restriction process by playing G.

Lemma 7.4. *Let $f : \left(\{0,1\}^{n/r}\right)^r \to \{0,1\}$. We view f as a function whose input variables are partitioned into r bins with exactly n/r balls in each.*

$$\Pr_{\rho \in \mathcal{T}_k}\left[\textit{More than } p \cdot r \textit{ bins are completely empty after applying } \rho \textit{ to } f\right] \le \mathit{score}_G$$

Proof. In order to prove the claim we present an adversary $\mathcal{A}$ that plays the game G and has exactly the same probability to empty more than $p \cdot r$ bins after playing G as the probability to empty more than $p \cdot r$ bins after applying ρ to f. In order to prove this lemma we construct an adversary $\mathcal{A}$ that plays G and simulates every step in the execution of the restriction $\rho \in \mathcal{T}_k$. $\mathcal{A}$ is defined as follows. If the restriction process decides on Case 1, the adversary will also choose an s-move and remove the ball corresponding to the restricted variable. If the restriction process decides on Case 2, then the adversary will also choose to do an r-move.

It follows from Lemma 7.3 that it is not possible that the restriction process chooses more than $\frac{100 I_i \log n}{k'}$ of specific variables from interval $i \in [M]$. This is exactly the limitation for our adversary $\mathcal{A}$ playing G.

Since an optimal adversary for G is at least as good as $\mathcal{A}$, it follows that the probability to completely empty more than $p \cdot r$ bins after applying ρ to f is at most score_G, as needed. □

Now we can assume that the adversary plays G which simulates the random process. We will now prove that in G the adversary $\mathcal{A}$ can choose the specific balls at the end of every interval, while not decreasing $\mathit{score}_G^{\mathcal{A}}$.

Lemma 7.5. *Let $i \in [M]$ be an interval. Assume that $\mathcal{A}$ is playing G and makes the following two consecutive moves in interval i. First, it removes a specific ball (an s-move).*

Then, it chooses a random ball and removes it (an r-move). We claim that if $\mathcal{A}$ plays them in a reverse order, $score_G^{\mathcal{A}}$ can only increase.

Proof. Let $\mathcal{A}_1$ be an adversary playing G. Assume that during interval $i \in [M]$ adversary $\mathcal{A}_1$ first removes a specific ball (s-move) and then a ball at random (r-move). Without loss of generality, denote the variable (corresponding to the ball) that $\mathcal{A}_1$ removed in the s-move by x_1. Observe that the choice of the random ball is done at random from the remaining balls after removing the ball corresponding to the variable x_1.

We will construct an adversary $\mathcal{A}_2$ with $score_G^{\mathcal{A}_2} \geq score_G^{\mathcal{A}_1}$ which does the choice above in a reverse order, as follows. $\mathcal{A}_2$ completely simulates the execution of $\mathcal{A}_1$ until the point $\mathcal{A}_1$ chooses x_1. At that point $\mathcal{A}_2$ chooses a random ball (corresponding to some variable). If the chosen variable happens to be x_1, then from that point $\mathcal{A}_2$ can simulate the exact same execution of $\mathcal{A}_1$ (without even using his additional s-move). On the other hand, if the chosen variable is not x_1, then $\mathcal{A}_2$ chooses in its s-move the ball corresponding to x_1. We can see that $score_G^{\mathcal{A}_2} \geq score_G^{\mathcal{A}_1}$, as needed. □

By repeated applications of this lemma together with Lemma 7.3, we get,

Corollary 7.6. *Let $\mathcal{A}$ be an optimal adversary playing G. Let $i \in [M]$ be some interval. We can assume that during the interval i the optimal adversary $\mathcal{A}$ first does all the r-moves and then it does at most $\frac{100 I_i \log n}{k'}$ s-moves.*

Recall that after the M-th interval ($M = \log \frac{n}{k}$) there are k balls left and the adversary is not allowed to remove any of them. Recall that we call the k variables, that are left after the M-th interval, the leftover variables (when we speak about balls and bins, we call them leftover balls).

Our next goal is to define a sequence of games such that the first game in the sequence is a game in which all the r-moves are done before the s-moves.

First, denote by G_{M+1} the game G. For $j \in \{0, \ldots, M-1\}$ define a game G_{M-j} in which the adversary $\mathcal{A}_{M-j}$ has the same rules as G except that it is not allowed to remove any specific balls in all intervals i such that $M - j \leq i \leq M$, but instead, after it restricts $n - k$ variables (that is, after the M-th interval), it is allowed to remove additional $\frac{110 \cdot j \cdot k \cdot \log n}{k'}$ specific balls from the leftover balls.

We prove that we can switch from G_{M+1} to G_1 with only small loss in the *score* by doing it in small steps as follows.

Lemma 7.7. *Let $j \in [M]$. It holds that,*

$$score_{G_j} \geq score_{G_{j+1}} - 2^{-k/(k'^2)}$$

Proof. Let $\mathcal{A}_j$ and $\mathcal{A}_{j+1}$ be two adversaries playing G_j and G_{j+1}, respectively. Assume $\mathcal{A}_{j+1}$ is an optimal adversary. We prove that $\mathcal{A}_j$ can simulate with high probability the execution of $\mathcal{A}_{j+1}$. The idea is that until the j-th interval (excluding the j-th interval itself) $\mathcal{A}_j$ can simulate the game played by $\mathcal{A}_{j+1}$ exactly. During interval j, $\mathcal{A}_j$ will only mark the specific balls that $\mathcal{A}_{j+1}$ removes using s-moves (recall that $\mathcal{A}_j$ is not allowed to remove them).

Recall that in the j-th interval the adversary $\mathcal{A}_{j+1}$ chooses $\frac{100 I_j \log n}{k'}$ specific balls. Recall that we have at the end of the M-th interval k balls and recall that at the end of the j-th interval, there are I_j balls. We claim that since all the choices of $\mathcal{A}_j$, starting in the $(j+1)$-th interval, until the M-th interval (including) are r-moves, and since we are left with k leftover balls, the expected number of the marked balls which are not removed (at the random steps until the end of the M-th interval) is at most $\frac{100 I_j \log n}{k'} \cdot \frac{k}{I_j} = \frac{100 k \log n}{k'}$.

Denote by Q the number of marked balls that are not chosen before the end of the M-th interval. Notice that Q is distributed according to a hypergeometric distribution $H\left(I_j, \frac{100 I_j \log n}{k'}, k\right)$ (see definition 2.7). Due to Proposition 2.8, we get that

$$\begin{aligned}
\Pr\left[Q \geq (1+0.1)\frac{100 \cdot k \cdot \log n}{k'}\right] &\leq \exp\left(\frac{-0.1^2\left(\frac{100 \cdot I_j \cdot \log n}{k'}\right)^2 k}{(I_j - k)I_j}\right) \\
&= \exp\left(\frac{-100 \cdot I_j \cdot k \cdot \log^2 n}{(I_j - k)k'^2}\right) \\
&\leq \exp\left(\frac{-k}{k'^2}\right) \leq 2^{-k/(k'^2)}
\end{aligned}$$

We got that with probability at most $2^{-k/(k'^2)}$ more than $E = \frac{110 \cdot k \cdot \log n}{k'}$ of the specific balls that $\mathcal{A}_{j+1}$ chooses, are not *randomly* chosen by $\mathcal{A}_j$ before reaching the end of the M-th interval. Recall that $\mathcal{A}_j$ has additional $\frac{110 k \log n}{k'}$ specific choices of balls to remove (over the amount that $\mathcal{A}_{j+1}$ has) from the leftover variables. The additional choices can be used to remove the E balls that were not removed until the end of the M-th interval. So, we got that with probability at least $1 - 2^{-k/(k'^2)}$ adversary $\mathcal{A}_j$ can simulate the execution of $\mathcal{A}_{j+1}$. In other words,

$$score_{G_j} \geq score_{G_j}^{\mathcal{A}_j} \geq score_{G_{j+1}} - 2^{-k/(k'^2)}$$

□

Notice that the game G_1 by definition is a game in which the adversary first does all the random choices of balls to remove until the end of the M-th interval. After the M-th interval it is allowed to remove at most $\frac{110 k \log^2 n}{k'}$ specific balls from the leftover balls. By Lemma 7.7 it follows that (recall that there are at most $\log n$ intervals)

Corollary 7.8. *It holds that,*

$$score_{G_1} \geq score_G - (\log n) \cdot 2^{-k/k'}$$

With the order of removals in mind, we can bound the number of bins an optimal adversary $\mathcal{A}_1$ playing G_1 can completely empty. We do this by first bounding (with high probability) the number of balls removed from each bin during the M intervals. Then we show that every adversary $\mathcal{A}_1$ playing G_1 cannot eliminate too many bins using at most $\left(\frac{110 \cdot k \cdot \log^2 n}{k'}\right)$ balls that it can eliminate from the leftover balls.

We bound the number of balls from a specific bin that stay after the M-th interval.

Lemma 7.9. *Let $i \in [r]$. Denote by B_i the amount of balls from the i-th bin that are not removed during the first $(n-k)$ r-moves (random removal steps) in the game G_1.*

$$\Pr\left[B_i < \frac{k}{2r}\right] \leq 2^{-k/(4r^2)}$$

Proof. Notice that B_i is distributed according to a hypergeometric distribution $H(n, \frac{n}{r}, k)$. From Proposition 2.8, we get that

$$\Pr\left[B_i < \left(1 - \frac{1}{2}\right)\frac{k}{r}\right] \leq \exp\left(-\frac{\left(\frac{n}{r}\right)^2 k}{4(n-k)n}\right) \leq \exp\left(-\frac{k}{4r^2}\right) \leq 2^{-k/(4r^2)}$$

which proves the claim. □

We are now ready to prove the main theorem of this section.

Proof of Theorem 7.1. Let $f : \left(\{0,1\}^{n/r}\right)^r \to \{0,1\}$. We view f as a function whose domain consists of r bins with exactly n/r balls in each. Let $\rho \in \mathcal{T}_k$ be a restriction as described in Section 6.

From Lemma 7.9 we know that with probability at most $2^{-k/(4r^2)}$ a specific bin has less than $\frac{k}{2r}$ balls left in it after playing all M intervals of G_1. So with probability at least $1 - r2^{-k/(4r^2)}$ all the bins contain at least $\frac{k}{2r}$ balls at the end of the random choices part.

We let the adversary to complete its set of moves by removing $\frac{110k\log^2 n}{k'}$ specific balls from the leftover balls. Since in all bins there are at least $\frac{k}{2r}$ balls, it can remove at most $\frac{\frac{110k\log^2 n}{k'}}{\frac{k}{2r}} = \frac{220r\log^2 n}{k'}$ bins.

We get that (in G_1) with probability at least $1 - r2^{-k/(4r^2)}$ we are left with at least one ball in $r - \frac{220r\log^2 n}{k'}$ bins. Since this is correct for the game G_1, from Lemma 7.7, it follows that with probability at least $1 - r2^{-k/(4r^2)} - (\log n) \cdot 2^{-k/k'^2}$ there is at least one ball in $r - \frac{220r\log^2 n}{k'}$ bins when the adversary plays G.

Recall that G was based on our restriction process (Lemma 7.4). Hence, it follows that with probability at least $1 - r2^{-k/(4r^2)} - (\log n) \cdot 2^{-k/k'^2}$ at most a fraction of $\frac{220\log^2 n}{k'}$ of the r bins in the definition of f are completely restricted after the process of the restriction $\rho \in \mathcal{T}_k$. □

8 Proof of Main Theorem 3.1

Proof of Theorem 3.1. Let $\tau > 0$ be a small constant. Set $r = n^\tau$, $\epsilon = 4 \cdot 2^{-r} + 2 \cdot 2^{-r/12}$ and $k = n^{10\tau}$, $k' = n^{\tau/100}$.

Let $h : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ be the function that is defined in Section 4. Let $F(x,y)$ be a formula of size $L(F(x,y))$ such that

$$\Pr_{x,y\in\{0,1\}^n}[F(x,y) = h(x,y)] \geq \frac{1}{2} + \epsilon.$$

Recall the set $\mathcal{H}$ from Theorem 5.1. By Theorem 5.2 we know that there exists some $x_0 \in \mathcal{H}$ such that

$$\Pr_{y \in \{0,1\}^n}[F(x_0, y) = h_{x_0}(y)] \geq \frac{1}{2} + \epsilon/2.$$

Let ρ be a random restriction distributed according to $\mathcal{T}_k$. Denote by $h_{x_0}(y)|_\rho$ the function $h_{x_0}(y)$ restricted by ρ. Denote by $F(x_0, y)|_\rho$ the formula $F(x_0, y)$ restricted by ρ. Denote by $S_\rho \subseteq [n]$ the subset of coordinates that are unassigned by ρ. Since once a variable is chosen to be restricted it is randomly assigned, it follows that,

$$\mathop{\mathbb{E}}_{\rho \in \mathcal{T}_k} \Pr_{y \in \{0,1\}^{S_\rho}}[F(x_0, y)|_\rho = h_{x_0}(y)|_\rho] \geq \frac{1}{2} + \epsilon/2 \tag{8.1}$$

Denote by $A \subseteq \mathcal{T}_k$ the subset of restrictions that shrink well. Formally, $\rho \in A \iff L(F(x_0, y)|_\rho) \leq 2^{\sqrt{2}c}\left(\frac{k}{n}\right)^{3/2} L(F(x_0, y))$ (where c is the same constant as in Theorem 6.6). By Theorem 6.6, it follows that

$$\Pr_{\rho \in \mathcal{T}_k}[\rho \in A] \geq 1 - 2^{-k/k'^2} \geq 1 - 2^{-n^\tau}$$

Denote by $B \subseteq \mathcal{T}_k$ the subset of restrictions that leave at least a $\left(1 - \frac{220 \log^2 n}{k'}\right)$ fraction of the bins with at least one ball unassigned. By Theorem 7.1, it follows that

$$\Pr_{\rho \in \mathcal{T}_k}[\rho \in B] \geq 1 - r2^{-k/(4r^2)} - (\log n) \cdot 2^{-k/k'^2} \geq 1 - 2^{-n^\tau}$$

With these bounds, equation (8.1) can be rewritten as

$$\begin{aligned}
\frac{1}{2} + \epsilon/2 \quad \leq \quad & \Pr_{\rho \in \mathcal{T}_k}[\rho \in A \cap B] \cdot \mathop{\mathbb{E}}_{\rho \in \mathcal{T}_k | \rho \in A \cap B} \Pr_{y \in \{0,1\}^{S_\rho}}[F(x_0, y)|_\rho = h_{x_0}(y)|_\rho] + \\
& \Pr_{\rho \in \mathcal{T}_k}[\rho \notin A \cap B] \cdot \mathop{\mathbb{E}}_{\rho \in \mathcal{T}_k | \rho \notin A \cap B} \Pr_{y \in \{0,1\}^{S_\rho}}[F(x_0, y)|_\rho = h_{x_0}(y)|_\rho] \\
\leq \quad & \mathop{\mathbb{E}}_{\rho \in \mathcal{T}_k | \rho \in A \cap B} \Pr_{y \in \{0,1\}^{S_\rho}}[F(x_0, y)|_\rho = h_{x_0}(y)|_\rho] + 2 \cdot 2^{-n^\tau}
\end{aligned}$$

We got,

$$\mathop{\mathbb{E}}_{\rho \in \mathcal{T}_k | \rho \in A \cap B} \Pr_{y \in \{0,1\}^{S_\rho}}[F(x_0, y)|_\rho = h_{x_0}(y)|_\rho] \geq \frac{1}{2} + \epsilon/2 - 2 \cdot 2^{-n^\tau} = \frac{1}{2} + 2^{-r/12}$$

Consequently, there must exist some $\rho \in A \cap B$ for which

$$\Pr_{y \in \{0,1\}^{S_\rho}}[F(x_0, y)|_\rho = h_{x_0}(y)|_\rho] \geq \frac{1}{2} + 2^{-r/12}$$

First, $\rho \in A$, so the formula shrinks well after applying ρ. In other words,

$$L(F(x_0, y)|_\rho) \leq 2^{\sqrt{2}c}\left(\frac{k}{n}\right)^{3/2} L(F(x_0, y))$$

which means that

$$L(F(x_0,y)) \geq 2^{-\sqrt{2}c}\left(\frac{n}{k}\right)^{3/2} L(F(x_0,y)|_\rho) \tag{8.2}$$

Second, $\rho \in B$, thus there is at most $\frac{220\log^2 n}{k'} \leq \frac{1}{24}$ fraction of the r XORs that are completely eliminated. The others contain at least one variable that is alive. Applying Claim 7.2, gives us that there exists a formula of size $L(F(x_0,y)|_\rho) + 2r$ that computes $h_{x_0}(y)$ with probability at least $\frac{1}{2} + \frac{2^{-r/12}}{2^{r/24}} \geq \frac{1}{2} + \frac{1}{2^{r/8}}$.

Recall that $h_{x_0}(y)$ is the function represented by $Enc^{\mathcal{C}}_{x_o}$. By the definition of $\mathcal{H}$ in Theorem 5.1, we get that every formula computing $Enc^{\mathcal{C}}_{x_o}$ with probability at least $\frac{1}{2} + \frac{1}{2^{r/8}}$ must be of size at least $\Omega(n')$, for $n' = \frac{n}{\log^2 n}$. Hence, $L(F(x_0,y)|_\rho) > \Omega(n') - 2r = \Omega(n')$. Plugging this into equation (8.2) we get,

$$L(F(x,y)) \geq L(F(x_0,y)) \geq 2^{-\sqrt{2}c}\left(\frac{n}{k}\right)^{3/2}\Omega(n') \geq \Omega(n^{2.499})$$

for small enough τ. □

9 Formulas over Complete Basis

We note that our result can easily be adapted to get a lower bound for boolean formulas over the complete basis. Formally, we can show that the function h we constructed in Section 4 is hard to approximate by any boolean formula of size $O(n^{1.999})$ over the complete basis, within $\frac{1}{2} + \epsilon$ fraction of the inputs, where ϵ is exponentially small (i.e. $\epsilon = 2^{-n^{\Omega(1)}}$).

The proof of this statement follows the lines of the proof for deMorgan formulas, except for a minor change. For formulas over the complete basis, there is no (known) non-trivial shrinkage property, so the constant 3/2 needs to be changed to 1 in Lemma 6.2 and all subsequent claims in Section 6.

References

[And87] Alexander E. Andreev. On a method for obtaining more than quadratic effective lower bounds for the complexity of π-schemes. *Moscow Univ. Math. Bull.*, 42:63–66, 1987. In Russian.

[DP09] Devdatt P. Dubhashi and Alessandro Panconesi. *Concentration of Measure for the Analysis of Randomized Algorithms.* Cambridge University Press, 2009.

[Has98] Johan Håstad. The shrinkage exponent of de morgan formulas is 2. *SIAM J. Comput.*, 27(1):48–64, 1998.

[IMZ12] Russell Impagliazzo, Raghu Meka, and David Zuckerman. Pseudorandomness from shrinkage. *Electronic Colloquium on Computational Complexity (ECCC)*, 7th May 2012. TR12-057.

[IN93] Russell Impagliazzo and Noam Nisan. The effect of random restrictions on formula size. *Random Struct. Algorithms*, 4(2):121–134, 1993.

[Juk12] Stasys Jukna. *Boolean Function Complexity: Advances and Frontiers*. Springer Berlin Heidelberg, 2012.

[Khr71] V.M. Khrapchenko. A method of determining lower bounds for the complexity of π schemes. *Matematischi Zametki*, 10"1:83–92, 1971. In Russian.

[NW94] Noam Nisan and Avi Wigderson. Hardness vs randomness. *J. Comput. Syst. Sci.*, 49(2):149–167, 1994.

[Nis91] Noam Nisan. Pseudorandom bits for constant depth circuits. *Combinatorica*, 11(1):63–70, 1991.

[PZ93] Mike Paterson and Uri Zwick. Shrinkage of de morgan formulae under restriction. *Random Struct. Algorithms*, 4(2):135–150, 1993.

[Rud07] Atri Rudra. Lecture notes on coding theory, 2007. `http://www.cse.buffalo.edu/~atri/courses/coding-theory/lectures/lect19.pdf`.

[Sub61] B.A Subbotovskaya. Realizations of linear function by formulas using $+, \cdot, -$. *Doklady Akademii Nauk SSSR*, 136:3:553–555, 1961. In Russian.

Appendix

Proof of Lemma 5.3. Denote by U, W the *uniform independent random* variables for $u \in \{0,1\}^m$ and $w \in \{0,1\}^{d-m}$, respectively. Denote by $E(u,w)$ the characteristic function of the event $g\left(u,w\right) = f\left(u,w\right)$. Denote by $\bar{H} = \{0,1\}^m \setminus H$.

$$\begin{aligned}
\gamma &\leq \Pr_{\substack{u \in \{0,1\}^m \\ w \in \{0,1\}^{d-m}}} [g(u,w) = f(u,w)] \\
&= \sum_{u\in\{0,1\}^m, w\in\{0,1\}^{d-m}} \Pr[W = w, U = u] \cdot E(u,w) \\
&= \sum_{u\in\{0,1\}^m} \Pr[U = u] \cdot \sum_{w\in\{0,1\}^{d-m}} \Pr[W = w] \cdot E(u,w) \\
&= \sum_{u\in\{0,1\}^m} \Pr[U = u] \cdot \mathbb{E}_W[E(u,W)] \\
&= \sum_{u\in H} \Pr[U = u] \cdot \mathbb{E}_W[E(u,W)] + \sum_{u\in\bar{H}} \Pr[U = u] \cdot \mathbb{E}_W[E(u,W)] \\
&= \frac{1}{2^m} \sum_{u\in H} \mathbb{E}_W[E(u,W)] + \frac{1}{2^m} \sum_{u\in\bar{H}} \mathbb{E}_W[E(u,W)] \\
&\leq \frac{1}{2^m} \sum_{u\in H} \mathbb{E}_W[E(u,W)] + \frac{1}{2^m} \sum_{u\in\bar{H}} 1 \\
&= \frac{1}{2^m} \sum_{u\in H} \mathbb{E}_W[E(u,W)] + \frac{|\bar{H}|}{2^m}
\end{aligned}$$

Rearranging the inequality, we get that

$$\left(\gamma - \frac{|\bar{H}|}{2^m}\right) 2^m \leq \sum_{u\in H} \mathbb{E}_W[E(u,W)]$$

By an averaging argument, we can finally state that there exists at least one $u_0 \in H$ such that

$$\mathbb{E}_W[E(u_0,W)] \geq \frac{\left(\gamma - \frac{|\bar{H}|}{2^m}\right) 2^m}{|H|} = \frac{\gamma 2^m - |\bar{H}|}{|H|} = \frac{\gamma 2^m - 2^m + |H|}{|H|} = (\gamma - 1)\frac{2^m}{|H|} + 1$$

as needed. □

ECCC ISSN 1433-8092
http://eccc.hpi-web.de