Toward Better Depth Lower Bounds: Strong Composition of XOR and a Random Function

Nikolai Chukhin
Neapolis University Pafos, Cyprus
JetBrains Research, Paphos, Cyprus

Alexander S. Kulikov
JetBrains Research, Paphos, Cyprus

Ivan Mihajlin
JetBrains Research, Paphos, Cyprus

Abstract

Proving formula depth lower bounds is a fundamental challenge in complexity theory, with the strongest known bound of $(3 - o(1)) \log n$ established by Håstad over 25 years ago. The Karchmer–Raz–Wigderson (KRW) conjecture offers a promising approach to advance these bounds and separate P from NC^1. It suggests that the depth complexity of a function composition $f \diamond g$ approximates the sum of the depth complexities of f and g.

The Karchmer–Wigderson (KW) relation framework translates formula depth into communication complexity, restating the KRW conjecture as $\mathsf{CC}(\mathsf{KW}_f \diamond \mathsf{KW}_g) \approx \mathsf{CC}(\mathsf{KW}_f) + \mathsf{CC}(\mathsf{KW}_g)$. Prior work has confirmed the conjecture under various relaxations, often replacing one or both KW relations with the universal relation or constraining the communication game through strong composition.

In this paper, we examine the strong composition $\mathsf{KW}_{\mathsf{XOR}} \circledast \mathsf{KW}_f$ of the parity function and a random Boolean function f. We prove that with probability $1 - o(1)$, any protocol solving this composition requires at least $n^{3-o(1)}$ leaves. This result establishes a depth lower bound of $(3 - o(1)) \log n$, matching Håstad's bound, but is applicable to a broader class of inner functions, even when the outer function is simple. Though bounds for the strong composition do not translate directly to formula depth bounds, they usually help to analyze the standard composition (of the corresponding two functions) which is directly related to formula depth.

Our proof utilizes formal complexity measures. First, we apply Khrapchenko's method to show that numerous instances of f remain unsolved after several communication steps. Subsequently, we transition to a different formal complexity measure to demonstrate that the remaining communication problem is at least as hard as $\mathsf{KW}_{\mathsf{OR}} \circledast \mathsf{KW}_f$. This hybrid approach not only achieves the desired lower bound, but also introduces a novel technique for analyzing formula depth, potentially informing future research in complexity theory.

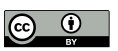
2012 ACM Subject Classification Theory of computation → Circuit complexity

Keywords and phrases complexity, formula complexity, lower bounds, Boolean functions, depth

Digital Object Identifier 10.4230/LIPIcs.STACS.2025.26

1 Introduction

Proving formula depth lower bounds is an important and difficult challenge in complexity theory: the strongest known lower bound $(3-o(1)) \log n$ proved by Håstad [6] (following a line of works starting from Subbotovskaya [17, 9, 16]) remains unbeaten for more than 25 years already (in 2014, Tal [18] improved lower order terms in this lower bound). One of the most actively studied approaches to this problem is the one suggested by Karchmer, Raz, and Wigderson [11]. They conjectured that the naive approach of computing a composition of two functions is close to optimal. Namely, for two Boolean functions $f\colon \{0,1\}^m \to \{0,1\}$ and $g\colon \{0,1\}^n \to \{0,1\}$, define their composition $f \diamond g\colon \{0,1\}^{m\times n} \to \{0,1\}$ as a function that first applies g to every row of the input matrix and then applies f to the resulting



column vector. The KRW conjecture then states that $\mathsf{D}(f \diamond g)$ is close to $\mathsf{D}(f) + \mathsf{D}(g)$, where $\mathsf{D}(\cdot)$ denotes the minimum depth of a de Morgan formula computing the given function. Karchmer, Raz, and Wigderson [11] proved that if the conjecture is true, then $\mathsf{P} \not\subseteq \mathsf{NC}^1$, that is, there are functions in P that cannot be computed in logarithmic parallel time.

A convenient way of studying the KRW conjecture is through the framework of Karchmer–Wigderson relation [12]. It not only allows one to apply the tools from communication complexity, but also suggests various important special cases of the conjecture. For a function $f\colon \{0,1\}^n \to \{0,1\}$, the relation KW_f is defined as follows:

$$\mathsf{KW}_f = \{(a,b,i)\colon a \in f^{-1}(1), b \in f^{-1}(0), i \in [n], a_i \neq b_i\}.$$

The communication complexity $\mathsf{CC}(\mathsf{KW}_f)$ of this relation is the minimum number of bits that Alice and Bob need to exchange to solve the following communication problem: Alice is given $a \in f^{-1}(1)$, Bob is given $b \in f^{-1}(0)$, and their goal is to find an index $i \in [n]$ such that $(a,b,i) \in \mathsf{KW}_f$ (i.e., $a_i \neq b_i$). Karchmer and Wigderson [12] proved that, for any function f, the communication complexity of KW_f is equal to the depth complexity of f: $\mathsf{CC}(\mathsf{KW}_f) = \mathsf{D}(f)$. Within this framework, the KRW conjecture is restated as follows: $\mathsf{CC}(\mathsf{KW}_f \diamond \mathsf{KW}_g)$ is close to $\mathsf{CC}(\mathsf{KW}_f) + \mathsf{CC}(\mathsf{KW}_g)$ (where $\mathsf{KW}_f \diamond \mathsf{KW}_g$ is another name for $\mathsf{KW}_{f \diamond g}$).

One natural way of relaxing the conjecture is to replace one or both of the two relations KW_f and KW_g by the universal relation, defined as follows:

$$U_n = \{(a,b,i)\colon a,b \in \{0,1\}^n, a \neq b, i \in [n], a_i \neq b_i\}.$$

Using a universal relation instead of the Karchmer–Wigderson relation makes the corresponding communication game only harder, hence proving lower bounds for it is potentially easier and could lead to the resolution of the original conjecture. For this reason, such relaxations have been studied intensively.

Edmonds et al. [4] proved the KRW conjecture for the composition $U_m \diamond U_n$ of two universal relations using communication complexity methods. Håstad and Wigderson [7] improved it for a higher degree of composition using a different approach. Karchmer et al. [11] extended this result to monotone functions. Håstad [6] demonstrated the conjecture for the composition $f \diamond \mathsf{XOR}_n$ of an arbitrary function $f\colon \{0,1\}^m \to \{0,1\}$ with the parity function XOR_n. This was later reaffirmed by Dinur and Meir [3] through a communication complexity approach. Further advancements were made by Gavinsky et al. [5] who established the conjecture for the composition $f \diamond U_n$ of any non-constant function $f\colon \{0,1\}^m \to \{0,1\}$ with the universal relation U_n. Mihajlin and Smal [15] proved the KRW conjecture for the composition of a universal relation with certain hard functions using XOR-composition. Subsequently, Wu [20] improved this result by extending it to the composition of a universal relation with a wider range of functions (though still not with the majority of them). de Rezende et al. [2] proved the conjecture in a semi-monotone setting for a wide range of functions g.

Another natural way of relaxing the initial conjecture is to constrain the communication game (instead of allowing for more inputs for the game). In the *strong composition* $\mathsf{KW}_f \circledast \mathsf{KW}_g$, Alice receives $X \in (f \diamond g)^{-1}(1)$ and Bob receives $Y \in (f \diamond g)^{-1}(0)$, and their objective is to identify a pair of indices (i,j) such that $X_{i,j} \neq Y_{i,j}$, similar to the regular composition. However, this time it must hold additionally that $g(X_i) \neq g(Y_i)$.

This way of relaxing the conjecture was considered in a number of previous papers and was formalized recently by Meir [14]. Håstad and Wigderson, in their proof of the lower bound for two universal relations, initially establish the result for what they call the extended

universal relation, a concept closely related to strong composition. Similarly, Karchmer et al. [11] demonstrate that, in the monotone setting, strong composition coincides with the standard composition. de Rezende et al. [2] utilized this notion, although without explicitly naming it. Meir [14] formalized the notion of strong composition in his proof of the relaxation of the KRW conjecture.

▶ **Theorem 1** (Meir, [14]). *There exists a constant $\gamma > 0.04$ such that for every non-constant function $f\colon \{0,1\}^m \to \{0,1\}$ and for all $n \in \mathbb{N}$, there exists a function $g\colon \{0,1\}^n \to \{0,1\}$ such that*

$$\mathsf{CC}(\mathsf{KW}_f \circledast \mathsf{KW}_g) \geq \log \mathsf{CC}(\mathsf{KW}_f) - (1-\gamma)m + n - O(\log(mn)).$$

1.1 Our Result

Håstad [6] proved the KRW conjecture for $\mathsf{KW}_f \diamond \mathsf{KW}_{\mathsf{XOR}}$. However it is still an open question to prove the KRW conjecture for $\mathsf{KW}_{\mathsf{XOR}} \diamond \mathsf{KW}_f$. In this paper, we study the strong composition $\mathsf{KW}_{\mathsf{XOR}_m} \circledast \mathsf{KW}_f$ of the parity function XOR_m with a random function $f\colon \{0,1\}^{\log m} \to \{0,1\}$. Since Alice and Bob receive an input of size $m \log m$, we estimate the size of $\mathsf{KW}_{\mathsf{XOR}_m} \circledast \mathsf{KW}_f$ in terms of $n = m \log m$. It is not difficult to see that the communication complexity of the corresponding game is at most $3 \log n$: KW_f can be solved in $\log m$ bits of communication, whereas $\mathsf{KW}_{\mathsf{XOR}_m}$ can be solved in $2 \log m$ bits of communication, using the standard divide-and-conquer approach (Alice sends the parity of the first half, Bob then identifies the half in which the parity differs, thus, by utilizing 2 bits of communication, the input size is reduced by a factor of two). We prove that if the function f is well balanced and hard to approximate (which happens with probability $1 - o(1)$), then the bound $3 \log n$ is essentially optimal. Below, we state the result in terms of the protocol size (i.e., the number of leaves), rather than depth, since this gives a more general lower bound. In particular, it immediately implies a $(3 - o(1)) \log n$ depth lower bound.

▶ **Corollary 2.** *With probability $1 - o(1)$, for a random function $f\colon \{0,1\}^{\log m} \to \{0,1\}$, any protocol solving $\mathsf{KW}_{\mathsf{XOR}_m} \circledast \mathsf{KW}_f$ has at least $n^{3-o(1)}$ leaves, where $n = m \log m$.*

In turn, this result follows from the following general lower bound, given in terms of $\mathsf{L}_{\frac{3}{4}}$ that stands for the smallest size of a formula that agrees with f on a $\frac{3}{4}$ fraction of inputs.

▶ **Theorem 3.** *For any 0.49-balanced function $f\colon \{0,1\}^{\log m} \to \{0,1\}$, any protocol solving $\mathsf{KW}_{\mathsf{XOR}_m} \circledast \mathsf{KW}_f$ has at least $n^{2-o(1)} \cdot \mathsf{L}_{\frac{3}{4}}(f)$ leaves, where $n = m \log m$.*

In contrast to many results mentioned above and similarly to the bound by de Rezende at al. [2], our result works for a wide range of inner functions f, what brings us closer to resolving KRW, which makes a claim about the complexity of composing any pair of functions. Also, many of the previous techniques work well in the regime where the outer function is hard and give no strong lower bounds when the outer function is easy (as it is the case with the XOR function). For example, random restrictions (as one of the most successful methods for proving lower bounds) does not seem to give meaningful lower bounds for $\mathsf{KW}_{\mathsf{XOR}_m} \circledast \mathsf{KW}_f$, as under a random restriction this composition turns into a XOR of a small number of variables which is easy to compute. The lower bound by Meir (see Theorem 1) also gives strong lower bounds in the regime where the outer function is hard (and only gives a trivial lower bound of the form $o(\log n)$ for the function that we study).

To prove the lower bound, we exploit formal complexity measures. As in [4, 15], we consider two stages of a protocol solving $\mathsf{KW}_{\mathsf{XOR}_m} \circledast \mathsf{KW}_f$. During the first stage, we track the progress using the classical measure by Khrapchenko [13] and ensure that even after many steps of the

protocol, there are still many instances of f that need to be solved. At the second stage, we switch to another formal complexity measure and show that the remaining communication problem is, roughly, not easier than $\mathsf{KW}_{\mathsf{OR}} \circledast \mathsf{KW}_f$. We believe that this proof technique is interesting on its own, since it is not only easy to show that Khrapchenko's measure cannot give superquadratic size lower bounds, but it is also known that natural generalizations of this measure are also unable to give stronger than quadratic lower bounds [8]. For more details on Khrapchenko's measure and its limitations, see Sections 2.1 and 2.5.

2 Notation, Known Facts, and Technical Lemmas

Throughout the paper, log denotes the binary logarithm whereas ln denotes the natural logarithm. By n we usually denote the size of the input. All asymptotic estimates are given under an implicit assumption that n goes to infinity. By $[n]$, we denote the set $\{1, 2, \ldots, n\}$. By R_+ we denote the set $\{x \in \mathsf{R} \colon x > 0\}$. We utilize the following asymptotic estimates for binomial coefficients. For any constant $0 < \alpha < 1$,

$$\Omega(n^{-1/2})2^{h(\alpha)n} \leq \binom{n}{\alpha n} \leq 2^{h(\alpha)n}, \tag{1}$$

where $h(x) = -x \log x - (1-x)\log(1-x)$ denotes the binary entropy function.

For a string $x \in \{0,1\}^n$, its i-th bit of x is denoted by x_i. For a matrix $X \in \{0,1\}^{m \times n}$, by X_i we denote the i-th row of X and by $X_{i,j}$ we denote the bit of X in the intersection of the i-th row and the j-th column.

2.1 Graphs

For a rooted tree, the *depth* of its node is the number of edges on the path from the node to the root; the depth of the tree is the maximum depth of its nodes.

Let $G(V, E)$ be a graph and $\varnothing \neq A \subseteq V$ be its nonempty subset of nodes. By $G[A]$, we denote a subgraph of G induced by A. By $\operatorname{avgdeg}(G, A)$, we denote the average degree of A:

$$\operatorname{avgdeg}(G, A) = \frac{1}{|A|} \sum_{v \in A} \deg(v)\,. \tag{2}$$

For a biparite graph $G(A \sqcup B, E)$ with nonempty parts, let

$$\psi(G) = \operatorname{avgdeg}(G, A) \cdot \operatorname{avgdeg}(G, B)\,. \tag{3}$$

Clearly, $\psi(G) \leq |A| \cdot |B|$. The lemma below shows that this graph measure is subadditive.

▶ **Lemma 4.** *Let $G(A \sqcup B, E)$ be a bipartite graph and $A = A_L \sqcup A_R$ be a partition of A into two parts. Let $G_L = G[A_L \sqcup B]$ and $G_R = G[A_R \sqcup B]$. Then,*

$$\psi(G) \leq \psi(G_L) + \psi(G_R).$$

Proof. Let E_L and E_R be the set of edges of G_L and G_R, respectively. Clearly, $E = E_L \sqcup E_R$. Then,

$$\begin{aligned}
\psi(G) \leq \psi(G_L) + \psi(G_R) &\iff \frac{|E|^2}{(|A_L| + |A_R|)|B|} \leq \frac{|E_L|^2}{|A_L||B|} + \frac{|E_R|^2}{|A_R||B|} \\
&\iff \frac{|E_L|^2 + |E_R|^2 + 2|E_L||E_R|}{|A_L| + |A_R|} \leq \frac{|E_L|^2}{|A_L|} + \frac{|E_R|^2}{|A_R|} \\
&\iff 2|E_L||E_R||A_L||A_R| \leq |E_R|^2|A_L|^2 + |E_L|^2|A_R|^2 \\
&\iff 0 \leq (|E_R||A_L| - |E_L||A_R|)^2.
\end{aligned}$$

◀

The next lemma shows that if G contains a node of small enough degree, then deleting it not only does not drop ψ, but also does not drop too much the average degree of the parts.

▶ **Lemma 5.** *Let a node $a \in A$ of a bipartite graph $G(A \sqcup B, E)$ satisfy $\deg(G, a) \leq \operatorname{avgdeg}(G, A)/2$ and let $A' = A \setminus \{a\}$ and $G'(A' \sqcup B, E') = G[A \setminus \{a\} \sqcup B]$. Then,*

$$\psi(G') \geq \psi(G), \tag{4}$$

$$\operatorname{avgdeg}(G', A') \geq \operatorname{avgdeg}(G, A), \tag{5}$$

Proof. The inequality $\operatorname{avgdeg}(G', A') \geq \operatorname{avgdeg}(G, A)$ holds since A' results from A by removing a node of degree less than the average degree.

To prove the inequality (4), let $d = \deg(G, a)$. Then, $|E'| = |E| - d$ and

$$\begin{aligned}
\psi(G') \geq \psi(G) &\iff \frac{(|E| - d)^2}{(|A| - 1)|B|} \geq \frac{|E|^2}{|A||B|} \\
&\iff \frac{|E|^2 - 2|E|d + d^2}{(|A| - 1)|B|} \geq \frac{|E|^2}{|A||B|} \\
&\impliedby \frac{|E| - 2d}{|A| - 1} \geq \frac{|E|}{|A|} \\
&\iff |E||A| - 2d|A| \geq |E|(|A| - 1) \\
&\iff d \leq \frac{|E|}{2|A|} = \frac{\operatorname{avgdeg}(G, A)}{2}.
\end{aligned}$$

◀

2.2 Boolean Functions

By $\mathbb{B}_n$, we denote the set of all Boolean functions on n variables. For two disjoint sets $A, B \subseteq \{0,1\}^n$, the set $A \times B$ is called a *combinatorial rectangle*, and it is called *full* if A and B form a partition of $\{0,1\}^n$. Clearly, there is a bijection between $\mathbb{B}_n$ and full combinatorial rectangles. For $f \in \mathbb{B}_n$, by $R_f = f^{-1}(1) \times f^{-1}(0)$, we denote the corresponding full rectangle. We say that a Boolean function f is *balanced* if $|f^{-1}(0)| = |f^{-1}(1)|$.

In this paper, it will prove convenient to apply a function $g \in \mathbb{B}_m$ not only to Boolean vectors $x \in \{0,1\}^m$, but also to matrices $X \in \{0,1\}^{n \times m}$:

$$g(X) = (g(X_1), \ldots, g(X_n)),$$

i.e., $g(X) \in \{0,1\}^n$ results by applying g to every row of X. This allows to define a composition in a natural way. For $f \in \mathbb{B}_m$ and $g \in \mathbb{B}_n$, their *composition* $f \diamond g \colon \{0,1\}^{m \times n} \to \{0,1\}$ treats the input as an $m \times n$ matrix and first applies g to all its rows and then applies f to the resulting column-vector:

$$f \diamond g(X) = f(g(X)) = f(g(X_1), \ldots, g(X_m)).$$

For a set of matrices $\mathcal{X} \subseteq \{0,1\}^{m \times n}$, by i-th projection $\operatorname{proj}_i \mathcal{X}$, we denote the set of all i-th rows among the matrices of $\mathcal{X}$:

$$\operatorname{proj}_i \mathcal{X} = \{X_i \colon X \in \mathcal{X}\} = \{t \in \{0,1\}^n \colon \exists X \in \mathcal{X} \colon t = X_i\}. \tag{6}$$

In the proof of the main result, we will be dealing with Boolean matrices of dimension $n \times \log n$. Let $\mathcal{X} \subseteq \{0,1\}^{n \times \log n}$ be a set of such matrices. We say that $\mathcal{X}$ is α-*bounded* if $|\operatorname{proj}_i \mathcal{X}| \leq \alpha n$, for all $i \in [n]$. The i-th projection of $\mathcal{X}$ is called *sparse* if $|\operatorname{proj}_i \mathcal{X}| < \frac{3}{8}n$, and *dense* otherwise. The following lemma shows that if $|\mathcal{X}|$ is large and $\mathcal{X}$ is α-bounded,

then the number of sparse projections of $\mathcal{X}$ is low. Later on, we will be applying this lemma for $\mathcal{X}$ which is almost 0.5-bounded and whose size gradually decreases to argue that the number of sparse projections cannot grow too fast.

▶ **Lemma 6.** *Let $k \in \mathbb{N}$ and $\alpha \in \left(\frac{3}{8}, \frac{1}{2}\right]$. If $\mathcal{X} \subseteq \{0,1\}^{n \times \log n}$ is α-bounded and $|\mathcal{X}| \geq \alpha^n \frac{n^n}{2^k}$, then the number of sparse projections of $\mathcal{X}$ does not exceed $k \log^{-1} \frac{8\alpha}{3}$.*

Proof. Let $\beta_i \in [0,1]$ be such that $|\operatorname{proj}_i \mathcal{X}| = \beta_i \alpha n$. The i-th projection is sparse if and only if $\beta_i < \frac{3}{8\alpha}$. Let t be the number of sparse projections and assume, without loss of generality, that the first t projections are sparse. Then,

$$\alpha^n \frac{n^n}{2^k} \leq |\mathcal{X}| \leq \prod_{i=1}^{n} |\operatorname{proj}_i \mathcal{X}| = (\alpha n)^n \prod_{i=1}^{n} \beta_i \iff$$

$$\frac{1}{2^k} \leq \prod_{i=1}^{n} \beta_i = \prod_{i=1}^{t} \beta_i \cdot \prod_{i=t+1}^{n} \beta_i < \left(\frac{3}{8\alpha}\right)^t \implies k \log^{-1} \frac{8\alpha}{3} \geq t. \qquad \blacktriangleleft$$

2.3 Boolean Formulas

The computational model studied in this paper is *de Morgan formulas*: it is a binary tree whose leaves are labeled by variables $x_1, \ldots, x_n$ and their negations whereas internal nodes (called gates) are labeled by $\vee$ and $\wedge$ (binary disjunction and conjunction, respectively). Such a formula *computes* a Boolean function $f(x_1, \ldots, x_n) \in \mathbb{B}_n$. We also say that a formula F *separates* a rectangle $A \times B$, if $f(a) = 1$ and $f(b) = 0$, for all $(a, b) \in A \times B$. This way, if a formula F computes a function f, then it separates R_f.

For a formula F, the *size* $\mathsf{L}(F)$ is defined as the number of leaves in F. This extends to Boolean functions: for $f \in \mathbb{B}_n$, by $\mathsf{L}(f)$ we denote the smallest size of a formula computing f. Similarly, the *depth* $\mathsf{D}(F)$ is the depth of the tree whereas $\mathsf{D}(f)$ is the smallest depth of a formula computing f.

By $\mathsf{L}_{\frac{3}{4}}(f)$, we denote the smallest size of a formula F that agrees with f on a $\frac{3}{4}$ fraction of inputs, i.e.,

$$\Pr_{x \in \{0,1\}^n}[F(x) = f(x)] \geq \frac{3}{4}.$$

We say that F *approximates* f.

It is known that formulas can be balanced: $\mathsf{D}(f) = \Theta(\log \mathsf{L}(f))$ (see references in [10, Section 6.1]): this is proved by showing that, for any formula F, there exists an equivalent formula F' with $\mathsf{L}(F') \leq \mathsf{L}(F)^{O(1)}$ and $\mathsf{D}(F') \leq O(\log \mathsf{L}(F))$. The following theorem further refines this: by allowing a larger constant in the depth upper bound, one can control the size of the resulting balanced formula.

▶ **Theorem 7** ([1])**.** *For any $k \geq 2$ and any formula F, there exists an equivalent formula F' satisfying $\mathsf{D}(F') \leq 3 \ln 2 \cdot k \cdot \log \mathsf{L}(F)$ and $\mathsf{L}(F') \leq \mathsf{L}(F)^\gamma$, where $\gamma = 1 + \frac{1}{1+\log(k-1)}$.*

Using a counting argument, one can show that, with probability $1 - o(1)$, for a random Boolean function $f \in \mathbb{B}_n$, $\mathsf{L}(f) = \Omega(2^n / \log n)$. To prove this, one compares the number of small size formulas with the number of Boolean functions $|\mathbb{B}_n| = 2^{2^n}$, using the following estimate (see [10, Lemma 1.23]). It ensures that the number of formulas of size at most $\frac{2^n}{100 \log n}$ is $o(|\mathbb{B}_n|)$:

$$(17n)^{\frac{2^n}{100 \log n}} = 2^{\frac{\log(17n)}{100 \log n} 2^n}.$$

▶ **Lemma 8.** *For all large enough l, the number of Boolean formulas over n variables with at most l leaves is at most*

$$(17n)^l. \tag{7}$$

Proof. The number of binary trees with l leaves is at most 4^l. For each such tree, there are at most $(4n)^l$ ways to convert it into a de Morgan formula: there are $2n$ input literals for the leaves and two operations for each internal gate. Consequently, the total number of formulas with at most l leaves is at most

$$l \cdot 4^l \cdot (4n)^l = l \cdot 16^l \cdot n^l \leq (17n)^l,$$

which is true for $l \geq 71$. ◀

We say that $f \in \mathbb{B}_n$ is *α-balanced* if

$$\alpha \cdot 2^n \leq |f^{-1}(0)|, |f^{-1}(1)| \leq (1-\alpha) \cdot 2^n$$

i.e., $\big||f^{-1}(0)| - |f^{-1}(1)|\big| \leq (1-2\alpha)2^n$.

▶ **Lemma 9.** *For all sufficiently large n and any constant $\frac{3}{8} < \alpha < \frac{1}{2}$, a random function $f \in \mathbb{B}_n$ is α-balanced and $\mathsf{L}_{\frac{3}{4}}(f) = \Omega(\frac{2^n}{\log n})$, with probability $1 - o(1)$.*

Proof. For a formula over n variables, the number of Boolean functions it approximates is at most (by the estimate (1))

$$\sum_{d=3\cdot 2^n/4}^{2^n} \binom{2^n}{d} = \sum_{d=0}^{2^n/4} \binom{2^n}{d} \leq 2^n \binom{2^n}{2^n/4} \leq 2^n \cdot 2^{h(1/4)2^n}.$$

Combining this with (7), we get that the number of functions approximated by formulas of size $\beta\frac{2^n}{\log n}$ is at most

$$(17n)^{\beta\frac{2^n}{\log n}} 2^n 2^{h(1/4)2^n} = 2^{2^n\left(\beta\frac{\log(17n)}{\log n} + h(1/4)\right)+n}.$$

For any constant $0 < \beta < 1 - h(1/4)$, this is a $o(1)$ fraction of $\mathbb{B}_n$.

Now, the probability that a random $f \in \mathbb{B}_n$ is not α-balanced (i.e., $||f^{-1}(0)| - |f^{-1}(1)|| > (1-2\alpha)\cdot 2^n$) is at most

$$\frac{1}{2^{2^n}} \cdot 2 \cdot \sum_{i=0}^{\alpha\cdot 2^n - 1} \binom{2^n}{i} \leq \frac{1}{2^{2^n}} \cdot 2 \cdot \alpha \cdot 2^n \cdot \binom{2^n}{\alpha \cdot 2^n} \leq 2^{2^n(h(\alpha)-1)+n+1} = o(1).$$

Thus, with probability $1-o(1)$, a random $f \in \mathbb{B}_n$ is α-balanced and hard to approximate. ◀

2.4 Karchmer–Wigderson Games

Karchmer and Wigderson [12] came up with the following characterization of Boolean formulas. For a Boolean function $f \in \mathbb{B}_n$, the *Karchmer–Wigderson game* KW_f is the following communication problem. Alice is given $a \in f^{-1}(1)$, whereas Bob is given $b \in f^{-1}(0)$, and their goal is to find an index $i \in [n]$ such that $a_i \neq b_i$. A *communication protocol for* KW_f is a rooted binary tree whose leaves are labeled with indices from $[n]$ and each internal node v is labeled either by a function $A_v \colon f^{-1}(1) \to \{0,1\}$ or by a function $B_v \colon f^{-1}(0) \to \{0,1\}$. For any pair $(a,b) \in f^{-1}(1) \times f^{-1}(0)$, one can reach a leaf of the protocol by traversing

a path from the root to a leaf to determine to which of the two children to proceed from a node v, one computes either $A_v(a)$ or $B_v(b)$. We say that a protocol *solves* KW_f, if for any $(a,b) \in f^{-1}(1) \times f^{-1}(0)$, one reaches a leaf $i \in [n]$ such that $a_i \neq b_i$. Similarly to formulas, we say that a protocol separates a combinatorial rectangle $A \times B$, if it works correctly for all pairs $(a,b) \in A \times B$.

Karchmer and Wigderson showed that formulas computing f and protocols solving KW_f can be transformed (even mechanically) into one another. In particular, the smallest number of leaves in the protocol solving KW_f is equal to $\mathsf{L}(f)$, whereas the smallest depth of a protocol (also known as the communication complexity of KW_f, denoted by $\mathsf{CC}(\mathsf{KW}_f)$) is nothing else but $\mathsf{D}(f)$. By $\mathsf{L}(A \times B)$ for a combinatorial rectangle $A \times B$, we denote the minimum number of leaves in a protocol separating A and B.

With each node of a protocol solving KW_f, one can associate a combinatorial rectangle in a natural way. The root of the protocol corresponds to R_f. For the two children of Alice's node v with a rectangle $A \times B$, one associates two rectangles $A_0 \times B$ and $A_1 \times B$, where $A_i = \{a \in A \colon A_v(a) = i\}$. This way, Alice splits the current rectangle horizontally. Similarly, when Bob speaks, he splits the current rectangle vertically. Each leaf of a protocol solving KW_f is associated with a *monochromatic* rectangle, i.e., a rectangle $A \times B$ such that there exists $i \in [n]$ for which $a_i \neq b_i$ for all $(a,b) \in A \times B$.

For functions $f \in \mathbb{B}_m$ and $g \in \mathbb{B}_n$, the *strong composition* of KW_f and KW_g, denoted as $\mathsf{KW}_f \circledast \mathsf{KW}_g$, is the following communication problem: Alice and Bob receive inputs $X \in (f \diamond g)^{-1}(1)$ and $Y \in (f \diamond g)^{-1}(0)$, respectively, and need to find indices (i,j) such that $X_{i,j} \neq Y_{i,j}$ and $g(X_i) \neq g(Y_i)$. We say that a protocol *strongly separates* sets $\mathcal{X} \subseteq (f \diamond g)^{-1}(1)$ and $\mathcal{Y} \subseteq (f \diamond g)^{-1}(0)$, if it solves the strong composition $\mathsf{KW}_f \circledast \mathsf{KW}_g$ on inputs $\mathcal{X} \times \mathcal{Y}$.

2.5 Formal Complexity Measures

For $f \in \mathbb{B}_n$, define a bipartite graph $G_f(f^{-1}(1) \sqcup f^{-1}(0), E_f)$ as follows:

$$E_f = \{\{u,v\} \colon u \in f^{-1}(1), v \in f^{-1}(0), d_H(u,v) = 1\},$$

where d_H is the Hamming distance. Khrapchenko [13] proved that, for any $f \in \mathbb{B}_n$, $\psi(G_f) \leq \mathsf{L}(f)$ (recall (3) for the definition of $\psi(G)$). This immediately gives a lower bound $\mathsf{L}(\mathsf{XOR}_n) \geq n^2$. Note the two useful properties of $\psi(G_f)$: on the one hand, it is a lower bound to $\mathsf{L}(f)$, on the other hand, it is much easier to estimate than $\mathsf{L}(f)$.

Paterson [19, Section 8.8] noted that Khrapchenko's approach can be cast as follows. A function $\mu \colon \mathbb{B}_n \to \mathsf{R}_+$ is called a *formal complexity measure* if it satisfies the following two properties:

1. normalization: $\mu(x_i), \mu(\overline{x_i}) \leq 1$, for all $i \in [n]$,
2. subadditivity: $\mu(f \vee g) \leq \mu(f) + \mu(g)$ and $\mu(f \wedge g) \leq \mu(f) + \mu(g)$, for all $f, g \in \mathbb{B}_n$.

Note that Khrapchenko's measure can be defined in this notation as $\phi(f) = \psi(G_f)$. Its subadditivity is shown in Lemma 4, whereas the normalization property can be easily seen.

It is not difficult to see that L itself is a formal complexity measure. Moreover, it turns out that it is the largest formal complexity measure.

▶ **Lemma 10** (Lemma 8.1 in [19]). *For any formal complexity measure $\mu \colon \mathbb{B}_n \to \mathsf{R}$ and any $f \in B_n$, $\mu(f) \leq \mathsf{L}(f)$.*

3 Proof of the Main Result

In this section, we prove the main result of the paper.

▶ **Theorem 3.** *For any* 0.49*-balanced function* $f\colon \{0,1\}^{\log m} \to \{0,1\}$*, any protocol solving* $\mathsf{KW}_{\mathsf{XOR}_m} \circledast \mathsf{KW}_f$ *has at least* $n^{2-o(1)} \cdot \mathsf{L}_{\frac{3}{4}}(f)$ *leaves, where* $n = m \log m$*.*

▶ **Corollary 2.** *With probability* $1 - o(1)$*, for a random function* $f\colon \{0,1\}^{\log m} \to \{0,1\}$*, any protocol solving* $\mathsf{KW}_{\mathsf{XOR}_m} \circledast \mathsf{KW}_f$ *has at least* $n^{3-o(1)}$ *leaves, where* $n = m \log m$*.*

Proof. Lemma 9 guarantees that for a random function $f\colon \{0,1\}^{\log m} \to \{0,1\}$, f is 0.49-balanced and $\mathsf{L}_{\frac{3}{4}}(f) = \Omega(m/\log\log m)$ with probability $1-o(1)$. Plugging this into Theorem 3 gives the required lower bound. ◀

3.1 Proof Overview

We start by proving a lower bound on the size of any protocol solving $\mathsf{KW}_{\mathsf{XOR}_m} \circledast \mathsf{KW}_f$ and having a logarithmic depth. Then, using balancing techniques (see Theorem 7), we generalize the size lower bound to all protocols.

Fix a set $\mathcal{Z} \subseteq \{0,1\}^{\log m}$ such that $|\mathcal{Z}| = 0.98m$ and f is balanced on $\mathcal{Z}$: $|\mathcal{X}_0| = |\mathcal{Y}_0| = 0.49m$, where $\mathcal{X}_0 = f^{-1}(1) \cap \mathcal{Z}$ and $\mathcal{Y}_0 = f^{-1}(0) \cap \mathcal{Z}$.

We prove a lower bound for any protocol that strongly separates $\mathsf{KW}_{\mathsf{XOR}_m} \circledast \mathsf{KW}_f$ on inputs $\mathcal{X}_T \times \mathcal{Y}_T$ (which are defined later). To this end, we associate, with nodes of the protocol, a graph similar to G_{XOR_m} and use Khrapchenko's measure to track the progress of the protocol. A node of the graph is associated with all inputs X having the same vector $f(X)$. The reasoning is that, in a natural scenario, the protocol will first solve XOR_m, followed by solving f, implying that the protocol does not need to distinguish between X and X' in the initial rounds, if $f(X) = f(X')$. We connect two graph nodes by an edge if their vectors differ in exactly one coordinate.

We aim to ensure that each edge in the graph has a large projection: for any two nodes connected by an edge, the elements of blocks associated with them cover a substantial number of inputs for the function f. There will be no small protocol capable of solving the problem within these two blocks since f is hard to approximate. This is the rationale behind ensuring that all edges in the graph have large projections on both sides. To achieve this, we enforce that each block that is associated with a node shrinks by at most a factor of two at each step of the protocol. This process ensures that a significant number of edges in the graph will maintain large projections on both sides.

Once the Khrapchenko measure becomes sufficiently small, we can assert that XOR_m is nearly solved, and the protocol, in a sense, must now solve an instance of $\mathsf{OR}_d \circledast f$. Using the fact that solving each edge independently is hard, we conclude that solving an $\mathsf{OR}_d \circledast f$ over these edges should be as difficult as approximately $d \cdot \mathsf{L}_{\frac{3}{4}}(f)$.

3.2 Proof

Throughout this section, we assume that m is large enough and $f \in \mathbb{B}_{\log m}$ is a fixed function that is 0.49-balanced. Fix sets $\mathcal{X}_0 \subseteq f^{-1}(1), \mathcal{Y}_0 \subseteq f^{-1}(0)$ of size $0.49m$ and let

$$\mathcal{X}_T = \{X \in \{0,1\}^{m \times \log m}\colon (\mathsf{XOR}_m \diamond f)(X) = 1 \wedge X_i \in \mathcal{X}_0 \sqcup \mathcal{Y}_0,\ \forall i \in [m]\}, \tag{8}$$

$$\mathcal{Y}_T = \{Y \in \{0,1\}^{m \times \log m}\colon (\mathsf{XOR}_m \diamond f)(Y) = 0 \wedge Y_i \in \mathcal{X}_0 \sqcup \mathcal{Y}_0,\ \forall i \in [m]\}. \tag{9}$$

Let $\alpha > 0$ be a constant and P be a protocol that strongly separates $\mathcal{X}_T \times \mathcal{Y}_T$ and has depth at most $\alpha \log m$. Recall that each node S of P is associated with a rectangle $\mathcal{X}_S \times \mathcal{Y}_S$. We build a subtree D of P having the same root and associate a graph G_N to every node N of D. The graphs G_N are built inductively from the graphs associated with the parents of N

as explained below, but all these graphs are subsets of the m-dimensional hypercube: the set of nodes of each such graph is a subset of $\{0,1\}^m$ and for each edge $\{u,v\}$ it holds that $d_H(u,v)=1$.

For the root T of the protocol P, the graph G_T is simply G_{XOR_m} (which is nothing else but the m-dimensional hypercube): its set of nodes is $\{0,1\}^m$, two nodes are joined by an edge with label i if they differ in the i-th coordinate.

For any node v of the graph G_S, we associate the following set of inputs called *block*:

$$\mathcal{B}_S(v) = \{X \in \{0,1\}^{m\times \log m} \colon X \in \mathcal{X}_S \sqcup \mathcal{Y}_S \text{ and } f(X) = v\}.$$

We say that an edge $\{u,v\}$ with label i of G_S is *heavy* if the projection of both $\mathcal{B}_S(u)$ and $\mathcal{B}_S(v)$ onto the ith coordinate is dense, i.e.,

$$|\operatorname{proj}_i \mathcal{B}_S(u)|, |\operatorname{proj}_i \mathcal{B}_S(v)| \geq \frac{3}{8}m,$$

and *light* otherwise.

Since the nodes of the graph G_S form a subset of $\{0,1\}^m$, we can naturally divide them into two parts, as their blocks correspond to subsets of either $\mathcal{X}_S$ or $\mathcal{Y}_S$.

$$A_S = \{v \in V(G_S) \mid \mathsf{XOR}_m(v) = 1\}$$
$$B_S = \{v \in V(G_S) \mid \mathsf{XOR}_m(v) = 0\}$$

For a graph G_S, we define $d_A(G_S)$ as the average degree of the part A_S and $d_B(G_S)$ as the average degree of the part B_S. We say that a graph G_S is *special* if

$$\min\{d_A(G_S), d_B(G_S)\} \leq 12\alpha \log^2 m.$$

We will construct the tree D inductively. For a node S in the tree D, we either stop the process if G_S is special, or construct the two children of S from the protocol P and their graphs. We continue building D on these two children inductively. Hence, all graphs corresponding to internal nodes of the tree D are not special, while all graphs associated with leaves of D are special.

▶ **Definition 11.** *A graph G_S, associated with a node S in the tree D, is* adjusted *if all its edges are heavy and*

$$\deg(v) > \frac{d_A(G_S)}{2},\ \forall v \in A_S \qquad \text{and} \qquad \deg(v) > \frac{d_B(G_S)}{2},\ \forall v \in B_S. \tag{10}$$

We will ensure that all graphs G_S for any node S in the tree D are adjusted.

▶ **Lemma 12.** *For each node v of the graph G_T (associated with the root T of the protocol P),*
1. *the degree of v is m;*
2. *$|\operatorname{proj}_i \mathcal{B}_T(v)| = 0.49m$, for all $i \in [m]$;*
3. *$|\mathcal{B}_T(v)| = \frac{(0.98m)^m}{2^m}$.*

Proof. Nodes of G_T are m-dimensional binary vectors, hence $\deg(v) = m$.

To prove the second property, recall that f is balanced on $\mathcal{X}_0 \sqcup \mathcal{Y}_0$. If $v_i = 1$ (or $v_i = 0$), for some $i \in [m]$, the i-th projection can take any value from $\mathcal{X}_0$ ($\mathcal{Y}_0$, respectively). Hence, $|\operatorname{proj}_i \mathcal{B}_T(v)| = 0.49m$.

Finally, to prove the third property, note the G_T has 2^m nodes and for each vertex v the size of the block $\mathcal{B}_T(v)$ is at most $(0.49m)^m$. Therefore, since each input from $\mathcal{X}_T \sqcup \mathcal{Y}_T$ belongs to exactly one block that is associated with a node from G_T and $|\mathcal{X}_T \sqcup \mathcal{Y}_T| = |\mathcal{X}_0 \sqcup \mathcal{Y}_0|^m = (0.98m)^m$, $|\mathcal{B}_T(v)| = \frac{(0.98m)^m}{2^m}$. ◀

Lemma 12 ensures that the graph G_T is adjusted and not special, thus the root T has two children. Using the function $\mathcal{B}$, we show how to construct an intermediate graph H_N for some child of a node S in the tree D and then we apply some cleanup procedures for the graph H_N to construct a graph G_N. Recall that each step of P partitions the set of either Alice's or Bob's inputs into two parts. Let G_S be a graph for some node S of the protocol P that is associated with a rectangle $\mathcal{X}_S \times \mathcal{Y}_S$ and assume, without loss of generality, that it is Alice's turn. Therefore, graph G_S is not special, otherwise we will stop the building process of the subtree of S. Let S_L be the left child of S in the protocol P and S_R be the right child. We add the same children of the node S in the tree D. Then, we put v from B_S into both H_{S_L} and H_{S_R} (since the block $\mathcal{B}_S(v)$ has not changed). For each node $v \in A_S$ we decide in which of the two graphs we will put it. The block $\mathcal{B}_S(v)$ is also split into two: $\mathcal{B}_{S_L}(v)$ and $\mathcal{B}_{S_R}(v)$, corresponding to the two ways of the protocol. We assign v to the left graph H_{S_L} if $2 \cdot |\mathcal{B}_{S_L}(v)| \geq |\mathcal{B}_S(v)|$, and to the right graph H_{S_R} if $2 \cdot |\mathcal{B}_{S_R}(v)| > |\mathcal{B}_S(v)|$. An edge $\{u, v\}$ from the edges of G_S goes to H_{S_L} if and only if both u and v are assigned to H_{S_L}. The same rule applies for edges in H_{S_R}. This approach ensures that the size of each block $\mathcal{B}_S(v)$ shrinks by at most a factor of two when transitioning from a parent to a child in the tree D. Then, the graphs G_{S_L} and G_{S_R} will be built using graphs H_{S_L} and H_{S_R}, respectively.

The idea of the structure of the graph G_S arises from Khrapchenko's graph, so we will use the same measure:

$$\psi(G_S) = d_A(G_S) \cdot d_B(G_S).$$

Lemma 4 states that ψ is subadditive.

After obtaining the graph H_C for a node C of the tree D, we make our first cleanup by deleting all light edges: let H'_C be a graph resulting from H_C by removing all its light edges. The next lemma shows that this does not drop the measure ψ too much.

▶ **Lemma 13.**

$$\psi(H'_C) \geq \psi(H_C)\left(1 - \frac{1}{\log m}\right).$$

Proof. Let S be the parent of C in D. Since S is not a leaf, we have that $\min\{d_A(G_S), d_B(G_S)\} > 12\alpha \log^2 m$ and the degree of every node in G_S is at least half of the average degree of its part. Without loss of generality, assume that inputs were deleted from $\mathcal{X}_S$, and therefore $d_A(H_C) \geq \frac{d_A(G_S)}{2} > 6\alpha \log^2 m$.

An edge $\{u, v\}$ can become light because of only one of its endpoints, because the blocks on the other side remain unchanged. From Lemma 12, we know that the initial size of each block is $(0.49m)^m$, and after each step of the protocol, the size of a block shrinks by at most a factor of two. Hence, for any node v, the size of its block $\mathcal{B}_S(v)$ is at least $\frac{(0.49m)^m}{2^{\alpha \log m}}$, because the protocol depth is bounded by $\alpha \log m$. Hence, we can bound the number of light edges incident to v by $3\alpha \log m$ using Lemma 6 (since $\log^{-1}(8 \cdot 0.49/3) < 3$). Therefore,

$$d_A(H'_C) \geq d_A(H_C) - 3\alpha \log m.$$

Now, consider $d_B(H'_C)$. Let E_C be the set of edges in H_C, whereas A_C and B_C be its parts of nodes. Then,

$$d_B(H'_C) \geq \frac{E_C - |A_C| \cdot 3\alpha \log m}{|B_C|} = d_B(H_C) - 3\alpha \log m \frac{|A_C|}{|B_C|}.$$

Hence,

$$\begin{aligned}\psi(H'_C) = d_A(H'_C)d_B(H'_C) &\geq (d_A(H_C) - 3\alpha\log m)\left(d_B(H_C) - 3\alpha\log m\frac{|A_C|}{|B_C|}\right)\\ &\geq \psi(H_C) - 3\alpha d_B(H_C)\log m - \frac{3\alpha|A_C|d_A(H_C)\log m}{|B_C|}\\ &= \psi(H_C)\left(1 - \frac{3\alpha\log m}{d_A(H_C)} - \frac{3\alpha|A_C|\log m}{|E_C|}\right)\\ &= \psi(H_C)\left(1 - \frac{6\alpha\log m}{d_A(H_C)}\right)\\ &> \psi(H_C)\left(1 - \frac{6\alpha\log m}{6\alpha\log^2 m}\right) = \psi(H_C)\left(1 - \frac{1}{\log m}\right).\end{aligned}$$

◀

The next lemma shows how to construct an adjusted graph G_C, from the intermediate graph H'_C.

▶ **Lemma 14.** *There exists a subgraph G_C of the graph H'_C such that G_C is adjusted and $\psi(G_C) \geq \psi(H_C)\left(1 - \frac{1}{\log m}\right)$.*

Proof. To get G_C, we keep removing nodes from H'_C until it satisfies (10). If (10) is violated, there exists, without loss of generality, a node $v \in A_C$ such that $\deg(v) \leq \frac{d_A(G_C)}{2}$. Let $G'_C = G_C \setminus \{v\}$. Lemma 5 guarantees that this does not decrease the measure. This process is clearly finite. ◀

This way, we construct the graph G_C for the node C. If C is not special, we continue expanding the subtree rooted at C. Recall also that, for each internal node S of the tree D, whose children are S_L and S_R, the following holds:

$$\psi(G_S) \leq \psi(H_{S_L}) + \psi(H_{S_L}).$$

Hence, combining it with Lemma 14 we have:

$$\psi(G_S)\left(1 - \frac{1}{\log m}\right) \leq \psi(G_{S_L}) + \psi(G_{S_R}). \tag{11}$$

On the other hand, if S is special, we will use the following two lemmas to argue that strongly separating $\mathcal{X}_S \times \mathcal{Y}_S$ is still difficult.

▶ **Lemma 15.** *Let S be a node of the tree D such that it has a node $v \in G_S$ having d adjacent edges. Then, any protocol that strongly separates $\mathcal{X}_S$ and $\mathcal{Y}_S$ has at least $\Omega\left(d \cdot \mathsf{L}_{\frac{3}{4}}(f)\right)$ leaves.*

Proof. Consider the subgraph of G_S induced by v and its neighbors $u_1, \ldots, u_d$ connected to v. Denote by l_i the label of the edge $\{v, u_i\}$. Define a measure ξ on subrectangles of $\mathcal{X}_S \times \mathcal{Y}_S$:

$$\xi(\mathcal{X} \times \mathcal{Y}) = \sum_{i=1}^{d} \mathsf{L}\left(\mathrm{proj}_{l_i}\,\mathcal{B} \times \mathrm{proj}_{l_i}\,\mathcal{B}_i\right),$$

where $\mathcal{X} \subseteq \mathcal{X}_S$, $\mathcal{Y} \subseteq \mathcal{Y}_S$, $\mathcal{B} = \mathcal{X} \cap \mathcal{B}_S(v)$ and $\mathcal{B}_i = \mathcal{Y} \cap \mathcal{B}_S(u_i)$, for all $i \in [d]$. By $\mathsf{KW}(A \times B)$, for any $A \cap B = \varnothing$, we denote a Karchmer-Wigderson communication game where Alice gets $a \in A$, Bob gets $b \in B$, and they need to find $i\colon a_i \neq b_i$. We prove that any protocol strongly separating $\mathcal{X}_S \times \mathcal{Y}_S$ requires at least $\xi(\mathcal{X}_S \times \mathcal{Y}_S)$ leaves.

It is easy to see that ξ is subadditive, being a sum of subadditive measures: if $\mathcal{X} = \mathcal{X}' \sqcup \mathcal{X}''$, then $\xi(\mathcal{X} \times \mathcal{Y}) \le \xi(\mathcal{X}' \times \mathcal{Y}) + \xi(\mathcal{X}'' \times \mathcal{Y})$ and the same applies when we split $\mathcal{Y}$. Namely, let $\mathcal{Y} = \mathcal{Y}' \sqcup \mathcal{Y}''$, $\mathcal{B}_i = \mathcal{Y}' \cap \mathcal{B}_S(u_i)$, and $\mathcal{B}''_i = \mathcal{Y}'' \cap \mathcal{B}_S(u_i)$. Then,

$$\begin{aligned}\xi(\mathcal{X} \times \mathcal{Y}' \sqcup \mathcal{Y}'') &= \sum_{i=1}^{d} \mathsf{L}\left(\operatorname{proj}_{l_i} \mathcal{B} \times \operatorname{proj}_{l_i} \mathcal{B}'_i \sqcup \mathcal{B}''_i\right) \\ &\le \sum_{i=1}^{d} \mathsf{L}\left(\operatorname{proj}_{l_i} \mathcal{B} \times \operatorname{proj}_{l_i} \mathcal{B}'_i\right) + \sum_{i=1}^{d} \mathsf{L}\left(\operatorname{proj}_{l_i} \mathcal{B} \times \operatorname{proj}_{l_i} \mathcal{B}''_i\right) \\ &= \xi(\mathcal{X} \times \mathcal{Y}') + \xi(\mathcal{X} \times \mathcal{Y}'').\end{aligned}$$

Consider a protocol P' strongly separating $\mathcal{X}_S \times \mathcal{Y}_S$ and its leaf L associated with a rectangle of inputs ${\mathcal{X}'}_L \times {\mathcal{Y}'}_L$. We show that $\xi({\mathcal{X}'}_L \times {\mathcal{Y}'}_L) \le 1$. Since L is a leaf, there exists i, j such that for each $X \in {\mathcal{X}'}_L$ and $Y \in {\mathcal{Y}'}_L$:

$$X_{i,j} \ne Y_{i,j} \quad \text{and} \quad f(X_i) \ne f(Y_i).$$

Let k be such that $\mathcal{B}_k \ne \varnothing$ (if all $\mathcal{B}_t$ are empty, then $\xi = 0$). Then, $\mathcal{B}(u_t) = \varnothing$, for all $t \ne k$, as otherwise there would be no i such that $f(X_i) \ne f(Y_i)$ for all $(X, Y) \in {\mathcal{X}'}_L \times {\mathcal{Y}'}_L$, since u_k differs from v in the position l_k, and u_t differs from v in the position l_t and $l_k \ne l_t$. Thus, if $\xi({\mathcal{X}'}_L \times {\mathcal{Y}'}_L) > 1$, then $\mathsf{L}(\operatorname{proj}_{l_k} \mathcal{B} \times \operatorname{proj}_{l_k} \mathcal{B}_k) > 1$, which contradicts to the existence of a pair (i, j).

Thus, ξ is normal (has the value at most 1 for any leaf of any protocol that strongly separates $\mathcal{X}_S \times \mathcal{Y}_S$) and subadditive. Hence, its value for the whole protocol P' is a lower bound on the size of P'. Thus, it remains to estimate ξ for P'.

Since all d edges are heavy, we have:

$$|\operatorname{proj}_{l_i} \mathcal{B}_S(v)| + |\operatorname{proj}_{l_i} \mathcal{B}_S(u_i)| \ge \frac{3}{4}m, \quad \forall i \in [d].$$

Hence,

$$\mathsf{L}(\operatorname{proj}_{l_i} \mathcal{B}_S(v) \times \operatorname{proj}_{l_i} \mathcal{B}_S(u_i)) = \Omega\left(\mathsf{L}_{\frac{3}{4}}(f)\right),$$

for all $i \in [d]$. Summing over all $i \in [d]$, gives the desired lower bound. ◀

▶ **Lemma 16.** *For a special node S of the tree D, the number of leaves in any protocol strongly separating $\mathcal{X}_S \times \mathcal{Y}_S$ is*

$$\Omega\left(\frac{\psi(G_S) \cdot \mathsf{L}_{\frac{3}{4}}(f)}{\log^2 m}\right).$$

Proof. Assume, without loss of generality, that

$$d_A(G_S) \ge d_B(G_S) \quad \text{and} \quad d_B(G_S) \le 12\alpha \log^2 m.$$

Applying Lemma 15 to a node of degree at least $d_A(G_S)$, we get that the number of leaves is at least

$$\Omega\left(d_A(G_S) \cdot \mathsf{L}_{\frac{3}{4}}(f)\right) = \Omega\left(\frac{\psi(G_S)}{d_B(G_S)} \cdot \mathsf{L}_{\frac{3}{4}}(f)\right) = \Omega\left(\frac{\psi(G_S) \cdot \mathsf{L}_{\frac{3}{4}}(f)}{\log^2 m}\right).$$ ◀

At this point, everything is ready to lower bound the size of any protocol of logarithmic depth.

▶ **Theorem 17.** *The size of the protocol P (strongly separating $\mathcal{X}_T \times \mathcal{Y}_T$) is*

$$\Omega\left(\frac{m^2 \cdot \mathsf{L}_{\frac{3}{4}}(f)}{\log^2 m}\left(1-\frac{1}{\log m}\right)^{\alpha \log m}\right).$$

Proof. Lemma 16 states that the number of leaves needed to resolve any leaf S of the tree D is $\Omega\left(\psi(G_S) \cdot \mathsf{L}_{\frac{3}{4}}(f)/\log^2 m\right)$. Let $\mathcal{S}$ be the set of all leaves of the tree D. Using estimate (11), we have:

$$\psi(G_T) \cdot \left(1-\frac{1}{\log m}\right)^{\alpha \log m} \leq \sum_{S \in \mathcal{S}} \psi(G_S).$$

Since $\psi(G_T) = m^2$ (by Lemma 12), Then, the number of leaves in P is

$$\Omega\left(\sum_{S \in \mathcal{S}} \frac{\psi(G_S) \cdot \mathsf{L}_{\frac{3}{4}}(f)}{\log^2 m}\right) \geq \Omega\left(\frac{m^2 \cdot \mathsf{L}_{\frac{3}{4}}(f)}{\log^2 m}\left(1-\frac{1}{\log m}\right)^{\alpha \log m}\right). \quad \blacktriangleleft$$

Recall that α is a constant. Assuming $m \geq 4$, we have $\log m \geq 2$, and thus $1 - \frac{1}{\log m} \geq e^{-\frac{2}{\log m}}$. Then,

$$\frac{m^2 \cdot \mathsf{L}_{\frac{3}{4}}(f)}{\log^2 m}\left(1-\frac{1}{\log m}\right)^{\alpha \log m} \geq \frac{m^2 \cdot \mathsf{L}_{\frac{3}{4}}(f)}{\log^2 m} e^{-\frac{2}{\log m} \cdot \alpha \log m} \geq m^{2-\varepsilon} \cdot \mathsf{L}_{\frac{3}{4}}(f),$$

for any constant $\varepsilon > 0$ when m is sufficiently large. Hence, the number of leaves needed for a protocol P is $m^{2-o(1)} \cdot \mathsf{L}_{\frac{3}{4}}(f)$.

Finally, we get rid of the assumption that the depth of P is logarithmic and prove the main result.

Proof of Theorem 3. Let P be a protocol with $m^{2-\varepsilon} \cdot \mathsf{L}_{\frac{3}{4}}(f)$ leaves, for some $\varepsilon > 0$, solving $\mathsf{KW}_{\mathsf{XOR}_m} \circledast \mathsf{KW}_f$. We transform it into a protocol P' with $(m^{(2-\varepsilon)} \cdot \mathsf{L}_{\frac{3}{4}})^\gamma$ leaves and depth bounded by $3(3-\varepsilon)k \ln 2 \cdot \log m$, by applying Theorem 7, where $\gamma = 1 + \frac{1}{1+\log(k-1)}$. (Theorem 7 is stated in terms of formulas, but it is not difficult to see that it works also for protocols for strong composition.)

Since $\varepsilon > 0$ and $\lim_{k\to\infty} \gamma = 1$, there exist k and $\varepsilon' > 0$ such that

$$\left(m^{2-\varepsilon} \cdot \mathsf{L}_{\frac{3}{4}}(f)\right)^\gamma \leq m^{2-\varepsilon'} \cdot \mathsf{L}_{\frac{3}{4}}(f),$$

since $\mathsf{L}_{\frac{3}{4}}(m) \leq m$. Hence, protocol P' has logarithmic depth and at most $m^{2-\varepsilon'} \cdot \mathsf{L}_{\frac{3}{4}}(f)$ leaves, which contradicts Theorem 17. Therefore, P has $\Omega\left(m^{2-o(1)} \cdot \mathsf{L}_{\frac{3}{4}}(f)\right) = \Omega(n^{2-o(1)} \cdot \mathsf{L}_{\frac{3}{4}}(f))$ leaves. ◀

References

1 Maria Luisa Bonet and Samuel R. Buss. Size-depth tradeoffs for boolean fomulae. *Inf. Process. Lett.*, 49(3):151–155, 1994. doi:10.1016/0020-0190(94)90093-0.

2 Susanna F. de Rezende, Or Meir, Jakob Nordström, Toniann Pitassi, and Robert Robere. KRW composition theorems via lifting. *Comput. Complex.*, 33(1):4, 2024. doi:10.1007/s00037-024-00250-7.

3 Irit Dinur and Or Meir. Toward the KRW composition conjecture: Cubic formula lower bounds via communication complexity. *Comput. Complex.*, 27(3):375–462, 2018. doi:10.1007/s00037-017-0159-x.

4 Jeff Edmonds, Russell Impagliazzo, Steven Rudich, and Jirí Sgall. Communication complexity towards lower bounds on circuit depth. *Comput. Complex.*, 10(3):210–246, 2001. doi:10.1007/s00037-001-8195-x.

5 Dmitry Gavinsky, Or Meir, Omri Weinstein, and Avi Wigderson. Toward better formula lower bounds: The composition of a function and a universal relation. *SIAM J. Comput.*, 46(1):114–131, 2017. doi:10.1137/15M1018319.

6 Johan Håstad. The shrinkage exponent of de morgan formulas is 2. *SIAM J. Comput.*, 27(1):48–64, 1998. doi:10.1137/S0097539794261556.

7 Johan Håstad and Avi Wigderson. Composition of the universal relation. In Jin-Yi Cai, editor, *Advances In Computational Complexity Theory, Proceedings of a DIMACS Workshop, New Jersey, USA, December 3-7, 1990*, volume 13 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 119–134. DIMACS/AMS, 1990. doi:10.1090/dimacs/013/07.

8 Pavel Hrubes, Stasys Jukna, Alexander S. Kulikov, and Pavel Pudlák. On convex complexity measures. *Theor. Comput. Sci.*, 411(16-18):1842–1854, 2010. doi:10.1016/j.tcs.2010.02.004.

9 Russell Impagliazzo and Noam Nisan. The effect of random restrictions on formula size. *Random Struct. Algorithms*, 4(2):121–134, 1993. doi:10.1002/rsa.3240040202.

10 Stasys Jukna. *Boolean Function Complexity - Advances and Frontiers*, volume 27 of *Algorithms and combinatorics*. Springer, 2012. doi:10.1007/978-3-642-24508-4.

11 Mauricio Karchmer, Ran Raz, and Avi Wigderson. Super-logarithmic depth lower bounds via the direct sum in communication complexity. *Comput. Complex.*, 5(3/4):191–204, 1995. doi:10.1007/BF01206317.

12 Mauricio Karchmer and Avi Wigderson. Monotone circuits for connectivity require super-logarithmic depth. *SIAM J. Discret. Math.*, 3(2):255–265, 1990. doi:10.1137/0403021.

13 V. M. Khrapchenko. Method of determining lower bounds for the complexity of p-schemes. *Mathematical notes of the Academy of Sciences of the USSR*, 10(1):474–479, 1971. doi:10.1007/BF01747074.

14 Or Meir. Toward better depth lower bounds: A krw-like theorem for strong composition. In *64th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2023, Santa Cruz, CA, USA, November 6-9, 2023*, pages 1056–1081. IEEE, 2023. doi:10.1109/FOCS57990.2023.00064.

15 Ivan Mihajlin and Alexander Smal. Toward better depth lower bounds: The XOR-KRW conjecture. In Valentine Kabanets, editor, *36th Computational Complexity Conference, CCC 2021, July 20-23, 2021, Toronto, Ontario, Canada (Virtual Conference)*, volume 200 of *LIPIcs*, pages 38:1–38:24. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021. doi:10.4230/LIPIcs.CCC.2021.38.

16 Mike Paterson and Uri Zwick. Shrinkage of de morgan formulae under restriction. *Random Struct. Algorithms*, 4(2):135–150, 1993. doi:10.1002/rsa.3240040203.

17 B. A. Subbotovskaya. Realization of linear functions by formulas using $\vee$, &, $^-$. *Dokl. Akad. Nauk SSSR*, 136(3):553–555, 1961. URL: http://mi.mathnet.ru/dan24539.

18 Avishay Tal. Shrinkage of de morgan formulae by spectral techniques. In *55th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2014, Philadelphia, PA, USA, October 18-21, 2014*, pages 551–560. IEEE Computer Society, 2014. doi:10.1109/FOCS.2014.65.

19 Ingo Wegener. *The complexity of Boolean functions*. Wiley-Teubner, 1987. URL: http://ls2-www.cs.uni-dortmund.de/monographs/bluebook/.

20 Hao Wu. An improved composition theorem of a universal relation and most functions via effective restriction. *CoRR*, abs/2310.07422, 2023. doi:10.48550/arXiv.2310.07422.