Toward the KRW Composition Conjecture: Cubic Formula Lower Bounds via Communication Complexity*

Irit Dinur[1] and Or Meir[2]

1 **Faculty of Computer Science and Mathematics, Weizmann Institute, Rehovot, Israel**
irit.dinur@weizmann.ac.il

2 **Department of Computer Science, Haifa University, Haifa, Israel**
ormeir@cs.haifa.ac.il

Abstract

One of the major challenges of the research in circuit complexity is proving super-polynomial lower bounds for de-Morgan formulas. Karchmer, Raz, and Wigderson [20] suggested to approach this problem by proving that formula complexity behaves "as expected" with respect to the composition of functions $f \diamond g$. They showed that this conjecture, if proved, would imply super-polynomial formula lower bounds.

The first step toward proving the KRW conjecture was made by Edmonds et al. [10], who proved an analogue of the conjecture for the composition of "universal relations". In this work, we extend the argument of [10] further to $f \diamond g$ where f is an arbitrary function and g is the parity function.

While this special case of the KRW conjecture was already proved implicitly in Håstad's work on random restrictions [14], our proof seems more likely to be generalizable to other cases of the conjecture. In particular, our proof uses an entirely different approach, based on communication complexity technique of Karchmer and Wigderson [21]. In addition, our proof gives a new structural result, which roughly says that the naive way for computing $f \diamond g$ is the *only* optimal way. Along the way, we obtain a new proof of the state-of-the-art formula lower bound of $n^{3-o(1)}$ due to [14].

1998 ACM Subject Classification F.1.3 Complexity Measures and Classes, F.2 Analysis of Algorithms and Problem Complexity

Keywords and phrases Formula lower bounds, communication complexity, Karchmer-Wigderson games, KRW composition conjecture

Digital Object Identifier 10.4230/LIPIcs.CCC.2016.3

1 Introduction

One of the major challenges in the quest for proving lower bounds is to find an explicit function that requires formulas of super-polynomial size. Formally, (de Morgan) formulas are defined as circuits with AND, OR, and NOT gates that have fan-out 1, or in other words, their underlying graph is a tree.

* This research was partially supported by Irit Dinur's ERC grant number 239986.

31st Conference on Computational Complexity (CCC 2016).
Editor: Ran Raz; Article No. 3; pp. 3:1–3:51

Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

The state-of-the-art in this direction is a lower-bound of $\tilde{\Omega}(n^3)$ due to Håstad [14][1], building on earlier work by [32, 1, 16, 27]. This result was achieved by the celebrated method of random restrictions, and in particular, by providing a lower-bound on the shrinkage exponent, which is the parameter controlling the effect of random restrictions. Håstad's lower bound on the shrinkage exponent is known to be best possible, so improving the cubic lower-bound requires a new approach.

In this work we pursue a different approach following the KRW conjecture, named after Karchmer, Raz, and Wigderson who suggested this conjecture in [20]. The KRW conjecture is about composed functions of the form $f \diamond g : \{0,1\}^{mn} \to \{0,1\}$ defined by

$$f \diamond g(x_1, \ldots, x_m) = f(g(x_1), \ldots, g(x_m)),$$

where $f : \{0,1\}^m \to \{0,1\}$ and $g : \{0,1\}^n \to \{0,1\}$. The conjecture says roughly[2] that

$$\mathsf{L}(f \diamond g) \approx \mathsf{L}(f) \cdot \mathsf{L}(g)$$

where $\mathsf{L}(\cdot)$ denotes the formula size of a function, namely, the number of leaves in the underlying tree. In other words, the conjecture says that the naive way of computing $f \diamond g$, by first computing g on each component and then f, is essentially the best way to do it. In addition to being interesting in its own right, the KRW conjecture is particularly important due to the fact that it implies super-polynomial lower bounds for an explicit function [20].

Despite some early successes in the study of the KRW conjecture [10, 15], so far it has not bore new lower bounds. Recently, Gavinsky et al. [12] have made the first progress in two decades in this direction. In this work, we push this direction further, and obtain a new proof of the state-of-the-art cubic lower bound on the formula size of Andreev's function.

▶ **Theorem 1.1.** *Let* $And_n : \{0,1\}^n \to \{0,1\}$ *be Andreev's function* [1] *over* n *bits. Then,*

$$\mathsf{L}(And_n) \geq n^{3-o(1)}.$$

Although this was already proved by [14], our proof is based on an entirely different method – specifically, the communication-complexity technique of Karchmer and Wigderson [21]. Unlike the proof by random restrictions, this method does not seem to have any inherent limitation, and we do not see a reason why it should not be able to prove stronger lower bounds. More importantly, we see this work as a step toward proving the KRW conjecture.

Toward proving the KRW conjecture

As a first step toward proving their conjecture, [20] suggested to study the composition of *universal relations*, which are objects that are similar to functions but are easier to analyze in this context. Let us denote the universal relation by U. Then, [20] suggested to prove an analogue of their conjecture for the composition $U \diamond U$. This challenge was met by Edmonds et al. [10], and an alternative proof was discovered later by Håstad and Wigderson [15].

Recently, Gavinsky et al. [12] made further progress and proved an analogue of the KRW conjecture for $f \diamond U$: the composition of an arbitrary function f with the universal relation. Thus, the next step to proving the KRW conjecture would be to replace the universal relation in their result with a function g, for every choice of g. In this work, we do it for the special case where g is the parity function over n bits, denoted $\oplus_n$.

---

[1] Recently, Tal [34] provided a new proof of the lower bound on the shrinkage exponent, and along the way improved the lower order factors in Håstad's lower bound.

[2] The original KRW conjecture was formulated in terms of formula depth, this variant with formula size is from [12].

▶ **Theorem 1.2** (Main theorem). *Let $f : \{0,1\}^m \to \{0,1\}$ be a non-constant function. Then,*

$$\mathsf{L}(f \diamond \oplus_n) \geq \frac{\mathsf{L}(f) \cdot \mathsf{L}(\oplus_n)}{2^{\tilde{O}(\sqrt{m+\log n})}}.$$

To summarize, the KRW conjecture has been verified on $U \diamond U$ [10], and then $f \diamond U$ [12]. In this work we prove it for $f \diamond \oplus_n$, and one would hope that the next step(s) would lead to $f \diamond g$ for every g.

It is important to note that lower bounds on the composition $f \diamond \oplus_n$ were already proved implicitly in the aforementioned works on the Andreev's function [1, 16, 27, 14, 34]. In particular, [14, 34] implicitly prove that

$$\mathsf{L}(f \diamond \oplus_n) \geq \frac{\mathsf{L}(f) \cdot \mathsf{L}(\oplus_n)}{\mathrm{poly}(\log m, \log n)}.$$

However, our proof seems more likely to be generalized to other choices of g, and in addition, it gives a structural inverse result: not only is the naive way to compute $f \diamond \oplus_n$ optimal in terms of complexity, but it is essentially the *only* optimal way to compute $f \diamond \oplus_n$. More specifically, we show that any formula computing $f \diamond \oplus_n$ with near optimal complexity must incur a cost of $\approx \mathsf{L}(\oplus_n)$ before starting the computation of f. We discuss this result a bit more in Section 1.2 below, and a formal description is given in Section 3.

Bypassing a barrier for Karchmer-Wigderson relations

As all the previous works on the KRW conjecture, our proof is based on a method of Karchmer and Wigderson [21]. A particularly interesting feature of our proof of Theorem 1.1 is that it is the first proof of a super-quadratic formula lower bound that uses this method. In particular, this requires bypassing a known barrier for Karchmer-Wigderson relations, see Section 1.1 below for more detail.

Average-case lower bounds

A recent line of research [30, 18, 24, 8, 34] extended the aforementioned formula lower-bounds to the average-case setting. Our proof can be extended to the average-case setting as well, yielding the following results. In what follows, we say that a function F is (s,ε)-*hard* if every formula of size at most s computes F correctly on at most $\frac{1}{2}+\varepsilon$ fraction of the inputs.

▶ **Theorem 1.3.** *Let $f : \{0,1\}^m \to \{0,1\}$ be an (s,ε)-hard function. Then, $f \diamond \oplus_n$ is $(s', \varepsilon + 2^{-m})$-hard for*

$$s' \geq s \cdot \mathsf{L}(\oplus_n)/2^{\tilde{O}(\sqrt{m+\log n})}.$$

▶ **Corollary 1.4.** *For every $n, c \in \mathbb{N}$ there exists a function $F_{n,c} : \{0,1\}^n \to \{0,1\}$ bits that is (S, n^{-c})-hard for*

$$S \geq n^{3-\tilde{O}(\frac{1}{\sqrt{\log n}})}.$$

1.1 Background: Karchmer-Wigderson relations

Karchmer and Wigderson [21] observed an interesting connection between depth complexity and communication complexity: for every boolean function f, there exists a corresponding communication problem KW_f, such that any *deterministic* protocol for solving KW_f can be

syntactically converted to a formula computing f, and vice versa. In particular, the depth complexity of f is equal to the deterministic communication complexity of KW_f and the formula size of f equals the *protocol size* of KW_f, which is the smallest number of transcripts in a deterministic protocol that solves KW_f. The communication problem KW_f is often called the *Karchmer-Wigderson relation* of f, and we will refer to it as a *KW relation* for short.

The KW relation KW_f is defined as follows: Alice gets an input $x \in f^{-1}(0)$, and Bob gets as input $y \in f^{-1}(1)$. Clearly, it holds that $x \neq y$. The goal of Alice and Bob is to find a coordinate i such that $x_i \neq y_i$. Note that there may be more than one possible choice for i, which means that KW_f is a relation rather than a function. In what follows, we denote the communication complexity and protocol size of KW_f by $\mathsf{C}(KW_f)$ and $\mathsf{L}(KW_f)$ respectively.

The randomized-complexity barrier

KW relations allow us to translate questions about formula complexity to questions about communication complexity, thus giving us a different angle for attacking those questions. This method had great success in proving *monotone* formula lower-bounds [21, 13, 28, 20], culminating in exponential formula lower-bounds [28].

In contrast, in the non-monotone setting, this method has been stuck so far at proving quadratic lower-bounds. This is no coincidence: unlike the monotone setting, in the general setting it is known that every KW relation can be solved by a *randomized* protocol of quadratic size. Therefore, we cannot hope to prove better lower bounds using techniques that work against randomized protocols, and this fact severely restricts the techniques that we may employ. In particular, as noted by [12], this barrier implies that KW relations do not have "hard distributions", i.e., distributions over the inputs that are hard for every deterministic protocol. This fact makes it difficult to analyze those relations using information-theoretic techniques, and similar reasons prohibit the use of rectangle-based techniques [19].

As mentioned above, our proof of Theorem 1.1 is the first proof of a super-quadratic lower-bound using KW relations. In particular, our proof is the first to bypass the randomized-complexity barrier.

1.2 Proof outline

In order to prove Theorem 1.2, we analyze $KW_{f \diamond g}$ (for the case of $g = \oplus_n$) and show that

$$\mathsf{C}(KW_{f \diamond g}) \approx \mathsf{C}(KW_f) + \mathsf{C}(KW_g).$$

(We actually prove a similar but stronger statement, namely $\log \mathsf{L}(KW_{f \diamond g}) \approx \log \mathsf{L}(KW_f) + \log \mathsf{L}(KW_g)$ but for this outline we shall focus on the communication complexity.)

In the KW relation $KW_{f \diamond g}$, Alice and Bob's inputs are conveniently viewed as $m \times n$ matrices X, Y, respectively, such that $g(X) \in f^{-1}(0)$ and $g(Y) \in f^{-1}(1)$, where $g(X) \in \{0,1\}^m$ is obtained by applying g to each row of X and similarly $g(Y)$. Their goal is to find an entry (i, j) such that $X_{i,j} \neq Y_{i,j}$.

The naive protocol for Alice and Bob is as follows. Alice computes $a = g(X)$ and Bob computes $b = g(Y)$. In the first stage they solve KW_f on a, b and find an index $i \in [m]$ where $a_i \neq b_i$. Then, in the second stage, then solve KW_g on inputs X_i, Y_i to find j as required. This protocol shows that $\mathsf{C}(KW_{f \diamond g}) \leq \mathsf{C}(KW_f) + \mathsf{C}(KW_g)$. We remark that the naive strategy for $KW_{f \diamond g}$ corresponds to the naive formula for $f \diamond g$, but note that the order is reversed (top-down vs. bottom up).

The KRW conjecture asserts that the naive protocol for $KW_{f\diamond g}$ is essentially optimal. A natural approach for proving the KRW conjecture is to show that any optimal protocol that solves $KW_{f\diamond g}$ must behave approximately like the naive protocol. This approach potentially gives, in addition to a lower bound, a *structural result* about optimal protocols for $KW_{f\diamond g}$. This approach was first taken in [10] for the composition of two universal relations. In this work, we extend the argument of [10] to the case where f is an arbitrary function and $g = \oplus_n$ is the parity function.

Why should it be the case the any optimal behaves like the naive protocol? In order to gain intuition, consider the following thought experiment: Suppose that every message of Alice and Bob was either only "about" $g(X)$ and $g(Y)$, or only "about" X_i and Y_i for some $i \in [m]$. Intuitively, in the first case they are trying to solve KW_f on $g(X)$ and $g(Y)$, and in the second case they are trying to solve KW_g on some pair of rows X_i and Y_i. We now claim that if such a protocol was optimal, then Alice and Bob would have had to finish solving KW_f before solving KW_g on any pair of rows, or in other words, they would have had to behave as in the naive protocol.

More specifically, we claim that it only makes sense for Alice and Bob to communicate about a pair of rows X_i and Y_i if they already know that $g(X_i) \neq g(Y_i)$. To see why this is true, suppose that Alice and Bob communicate about some X_i and Y_i without knowing whether $g(X_i) \neq g(Y_i)$ or not. In such a case, Alice and Bob might send a lot of bits about X_i and Y_i, only to find out eventually that $X_i = Y_i$. This would mean that their effort has been in vain, since if $X_i = Y_i$ then the answer to $KW_{f\diamond g}$ cannot possibly lie in X_i and Y_i. Hence, if Alice and Bob do not wish to waste bits on rows where $X_i = Y_i$, they should first make sure that $g(X_i) \neq g(Y_i)$. However, finding $i \in [m]$ such that $g(X_i) \neq g(Y_i)$ requires solving KW_f on $g(X)$ and $g(Y)$. Therefore, Alice and Bob must solve KW_f before solving KW_g. We now discuss how to turn this intuitive argument into a formal proof.

We begin with an arbitrary optimal protocol Π for $KW_{f\diamond g}$, and show that it has an approximate two-stage structure similar to the naive protocol in the following sense. We split transcripts of Π into two parts π_1 and π_2, supposedly corresponding to the stages of solving KW_f and KW_g respectively. We identify a collection of partial transcripts π_1 that did not fully solve a certain random embedding of KW_f into $KW_{f\diamond g}$. We call these partial transcripts "alive" since the proof focuses only on them and shows that they lead to many distinct leaves of the protocol. We refer the reader to Section 3 for more details, and remark that this embedding is generic and allows embedding KW_f into $KW_{f\diamond g}$ for any choice of g. We then prove:

1. **The first stage is hard:** There are live partial transcripts π_1 whose length is almost about $\mathsf{C}(KW_f)$.
2. **The second stage is hard:** If π_1 is alive, then there is some π_2 whose length is about $\mathsf{C}(KW_g)$.

These two items together imply that Π has a transcript whose length is

$$|\pi_1| + |\pi_2| \approx \mathsf{C}(KW_f) + \mathsf{C}(KW_g).$$

In addition, observe that the second item implies a structural result on the optimal protocols for $KW_{f\diamond g}$: Essentially, this item says that as long as Alice and Bob have not solved KW_f on $g(X)$ and $g(Y)$, they must still incur a cost of $\mathsf{C}(KW_g)$. This roughly means that in any optimal protocol, Alice and Bob must first solve KW_f and then solve KW_g. Translating this result from the language of KW relations to the language of formulas, this means that any optimal formula for $f \diamond g$ must first compute g and then compute f (in the case of $g = \oplus_n$).

Our definition of π_1 being alive makes it not too difficult to prove the first item above (see the f-stage lemma in Section 4). However, the second item is much more technically difficult. Here we must prove that in order to solve KW_g on one of the m rows of X, Y, Alice and Bob must communicate $\mathsf{C}(KW_g)$ bits. The difficulty is that since Alice and Bob already spoke $|\pi_1|$ bits, they are not playing on all possible input pairs X, Y but rather on a residual rectangle that depends on π_1.

Nevertheless, since $|\pi_1| \leq m$, they only communicated about one bit on the average row. Intuitively, this means that on the typical row, the players should be quite far from solving KW_g. Hence, if they try to finish solving $KW_{f \diamond g}$ on one of those typical rows, they will have to communicate about $\mathsf{C}(KW_g)$ bits. However, there can be a few "revealed" rows on which π_1 reveals a lot, and on which it might be easier to solve $KW_{f \diamond g}$. We therefore take steps to *force* Alice and Bob to play on the typical "non-revealed" rows. In order to carry out our approach two ingredients are necessary:

- The first ingredient is a way to measure how much progress the players made on a given row, in a way that guarantees there will only be a few revealed rows. Luckily, for the parity function $g = \oplus_n$, this progress is directly related to the *information* that was communicated on the row. We then use an averaging argument which implies that on most rows, π_1 reveals at most one bit of information (and hence, only one bit of progress was made).
- The second ingredient is a way to force Alice and Bob to play only on the non-revealed rows. This is done by forcing X and Y to be identical on the revealed rows (so the final output (i, j) cannot be in these rows). Formally, this is done by focusing on a sub-rectangle of the residual rectangle of π_1, in which X and Y are identical on the revealed rows. However, one must do this without losing the complexity of the problem. Showing that this is possible is highly non-trivial, and is the most difficult part of our argument. The main difficulty comes from the fact that if, in the residual rectangle of π_1, it holds that $g(X_i) \neq g(Y_i)$ for some revealed row (X_i, Y_i), then we cannot force X_i and Y_i to be identical. The point is that such a situation cannot occur, because π_1 is alive, i.e., it has not fully solved KW_f yet. This implies that π_1 has could not find a small set of (revealed) rows in which the answer to KW_f lies. Thus, Alice and Bob cannot rule out that $g(X_i) = g(Y_i)$ in any revealed row i.

In implementing the two above ingredients, we develop two new tools that might be of use to future works:

- **Averaging argument for min-entropy:** In the discussion above, we argued that Alice and Bob gained only very little information on the *average* row of X and Y and therefore, by an averaging argument, this holds for *most* rows of X and Y. Such an averaging argument is easy to prove when we model information using Shannon entropy. Edmonds et al. [10], whose argument we extend, could not use Shannon entropy in their argument. Therefore, they defined another measure of information called "predictability" and proved an averaging argument for this measure.

  For our argument, neither Shannon entropy nor predictability are appropriate, and instead we model information using min-entropy. This requires us to prove a (non-trivial) averaging argument for min-entropy – see Section 6.1 for details.
- **Fortification lemma:** Throughout our proof, we often need to connect statements about information to statements about complexity. For example, we would like to say things like "Alice and Bob learned only little information, so the complexity of solving KW_f has not decreased by much". The reason is that in the implementation of the second ingredient, we restrict ourselves to a sub-rectangle. This restriction effectively gives information to

Alice and Bob, and we need to make sure that this information does not allow them to solve KW_f prematurely.
However, information is not always related to complexity. In particular, it is possible to come up with examples for relations KW_f in which Alice and Bob may get little information while reducing the complexity by much, or vice versa. In order to resolve this issue, we prove a general "fortification lemma", which shows that every relation KW_f has a sub-relation KW'_f for which the information and the complexity are related – see Section 6.2 for details.

1.3 Organization of the paper

We cover the required preliminaries in Section 2. Then, in Section 3, we prove our main theorem (Theorem 1.2), as well as our structural result and the resulting cubic lower bounds (Theorem 1.1). The proof of the main theorem uses three lemmas, which are proved in Sections 4, 5 and 7. We develop the new tools discussed above in Section 6. We extend our main theorem and the cubic lower bounds to the average-case setting in Section 8. Finally, in Section 9, we discuss some future directions and suggest some open problems whose solution might bring us closer to proving the KRW conjecture.

2 Preliminaries

We use $[n]$ to denote the set $\{1,\ldots,n\}$. Given two strings $x,y \in \{0,1\}^n$, the relative *(Hamming)* distance between x and y is the fraction of coordinates on which they disagree. For a function $t:\mathbb{N}\to\mathbb{N}$, we denote

$$\tilde{O}(t) \stackrel{\text{def}}{=} O(t\cdot \log^{O(1)} t)$$
$$\tilde{\Omega}(t) \stackrel{\text{def}}{=} \Omega(t/\log^{O(1)} t).$$

We denote the set of $m\times n$ binary matrices by $\{0,1\}^{m\times n}$. For every binary $m\times n$ matrix X, we denote by $X_j\in\{0,1\}^n$ the j-th row of X. Throughout the paper, we denote by $\oplus_n$ the parity function over n bits.

2.1 Formulas

▶ **Definition 2.1.** A *formula* ϕ is a binary tree, whose leaves are identified with literals of the forms x_i and $\neg x_i$, and whose internal vertices are labeled as AND ($\wedge$) or OR ($\vee$) gates. The *size* of a formula is the number of its *leaves* (which is the same as the number of its wires up to a factor of 2). We note that a single input coordinate x_i can be associated with many leaves.

▶ **Definition 2.2.** A formula ϕ computes a binary function $f:\{0,1\}^n\to\{0,1\}$ in the natural way. The *formula complexity* of a boolean function $f:\{0,1\}^n\to\{0,1\}$, denoted $\mathsf{L}(f)$, is the size of the smallest formula that computes f. The *depth complexity* of f, denoted $\mathsf{D}(f)$, is the smallest depth of a formula that computes f.

The following definition generalizes the above definitions from functions to promise problems, which will be useful when we discuss Karchmer-Wigderson relations.

▶ **Definition 2.3.** Let $\mathcal{X},\mathcal{Y}\subseteq\{0,1\}^n$ be disjoint sets. We say that a formula ϕ *separates* $\mathcal{X}$ and $\mathcal{Y}$ if $\phi(\mathcal{X})=0$ and $\phi(\mathcal{Y})=1$. The *formula complexity of the rectangle* $\mathcal{X}\times\mathcal{Y}$, denoted $\mathsf{L}(\mathcal{X}\times\mathcal{Y})$, is the size of the smallest formula that separates $\mathcal{X}$ and $\mathcal{Y}$. The *depth*

complexity of the rectangle $\mathcal{X} \times \mathcal{Y}$, denoted $\mathsf{D}(\mathcal{X} \times \mathcal{Y})$, is the smallest depth of a formula that separates $\mathcal{X}$ and $\mathcal{Y}$.

Note that Definition 2.2 is indeed a special case of Definition 2.3 where $\mathcal{X} = f^{-1}(0)$ and $\mathcal{Y} = f^{-1}(1)$. The following theorem establishes a tight connection between the formula complexity and the depth complexity of a function.

▶ **Theorem 2.4** ([4], following [31, 7]). *For every $\alpha > 0$ the following holds: For every formula ϕ of size s, there exists an equivalent formula ϕ' of depth at most $O(2^{\frac{1}{\alpha}} \cdot \log s)$ and size at most $s^{1+\alpha}$.*

2.2 Communication complexity

Let $\mathcal{X}$, $\mathcal{Y}$, and $\mathcal{O}$ be sets, and let $R \subseteq \mathcal{X} \times \mathcal{Y} \times \mathcal{O}$ be a relation. The communication problem [35] that corresponds to R is the following: two players, Alice and Bob, get inputs $x \in \mathcal{X}$ and $y \in \mathcal{Y}$, respectively. They would like to find $o \in \mathcal{O}$ such that $(x, y, o) \in R$. To this end, they send bits to each other until they find o, but they would like to send as few bits as possible. The communication complexity of R is the minimal number of bits that is transmitted by any protocol that solves R. More formally, we define a protocol as a binary tree, in which every vertex represents a possible state of the protocol, and every edge represents a message that moves the protocol from one state to another:

▶ **Definition 2.5.** A *(deterministic) protocol* that solves a relation $R \subseteq \mathcal{X} \times \mathcal{Y} \times \mathcal{O}$ is a rooted binary tree with the following structure:

- Every node of the tree is labeled by a rectangle $\mathcal{X}_v \times \mathcal{Y}_v$ where $\mathcal{X}_v \subseteq \mathcal{X}$ and $\mathcal{Y}_v \subseteq \mathcal{Y}$. The root is labeled by the rectangle $\mathcal{X} \times \mathcal{Y}$. Intuitively, the rectangle $\mathcal{X}_v \times \mathcal{Y}_v$ is the set of pairs of inputs that lead the players to the vertex v.
- Each internal vertex v is *owned* by Alice or by Bob. Intuitively, v is owned by Alice if it is Alice's turn to speak at state v, and same for Bob.
- Every edge of the tree is labeled by either 0 or 1.
- For every internal vertex v that is owned by Alice, the following holds: Let v_0 and v_1 be the children of v associated with the out-going edges labeled with 0 and 1, respectively. Then,
  - $\mathcal{X}_v = \mathcal{X}_{v_0} \cup \mathcal{X}_{v_1}$, and $\mathcal{X}_{v_0} \cap \mathcal{X}_{v_1} = \emptyset$.
  - $\mathcal{Y}_v = \mathcal{Y}_{v_0} = \mathcal{Y}_{v_1}$.

  Intuitively, when the players are at the vertex v, Alice sends 0 to Bob if her input is in $\mathcal{X}_{v_0}$ and 1 if her input is in $\mathcal{X}_{v_1}$. An analogous property holds for vertices owned by Bob, while changing the roles of $\mathcal{X}$ and $\mathcal{Y}$.
- For each leaf ℓ, there exists a value o such that $\mathcal{X}_\ell \times \mathcal{Y}_\ell \times \{o\} \subseteq R$. Intuitively, o is the output of the protocol at ℓ.

▶ **Definition 2.6.** Given a protocol Π and a vertex v of Π, the *transcript of* v is the string that is obtained by concatenating the labels of the edges on the path from the root to v. Intuitively, this string consists of the messages that Alice and Bob sent in their conversation until they got to v. Since the transcript determines v uniquely and vice versa, we will often identify the transcript with the vertex v. If v is a leaf of the protocol, we say that it is a *full transcript*, and otherwise we say that it is a *partial transcript*. Unless stated explicitly otherwise, whenever we say "transcript" we mean "full transcript".

Given a pair of inputs $(x, y) \in \mathcal{X} \times \mathcal{Y}$, we define the transcript of (x, y), denoted $\Pi(x, y)$, as the (full) transcript of the protocol when Alice and Bob get the inputs x and y respectively.

More formally, Let ℓ be the unique leaf ℓ such that $(x,y) \in \mathcal{X}_\ell \times \mathcal{Y}_\ell$, and define $\Pi(x,y)$ be the transcript of ℓ.

▶ **Definition 2.7.** The *communication complexity* of a protocol Π, denoted $\mathsf{C}(\Pi)$, is the the depth of the protocol tree. In other words, it is the maximum number of bits that can be sent in an execution of the protocol on any pair of inputs (x,y). For a relation R, we denote by $\mathsf{C}(R)$ the minimal communication complexity of a (deterministic) protocol that solves R.

▶ **Definition 2.8.** We define the *size* of a protocol Π to be its number of leaves. Note that this is also the number of distinct full transcripts of the protocol. We define the *protocol size*[3] of a relation R, denoted $\mathsf{L}(R)$, as the size of the smallest protocol that solves it.

2.3 Karchmer-Wigderson relations

In this section, we define KW relations formally, and state the correspondence between KW relations and formulas. We start by defining KW relations for general rectangles, and then specialize the definition to functions.

▶ **Definition 2.9.** Let $\mathcal{X},\mathcal{Y} \subseteq \{0,1\}^n$ be two disjoint sets. The *KW relation* $KW_{\mathcal{X}\times\mathcal{Y}} \subseteq \mathcal{X} \times \mathcal{Y} \times [n]$ is defined by

$$KW_{\mathcal{X}\times\mathcal{Y}} \stackrel{\text{def}}{=} \{(x,y,i) : x_i \neq y_i\}$$

Intuitively, $KW_{\mathcal{X}\times\mathcal{Y}}$ corresponds to the communication problem in which Alice gets $x \in \mathcal{X}$, Bob gets $y \in \mathcal{Y}$, and they would like to find a coordinate $i \in [n]$ such that $x_i \neq y_i$ (note that $x \neq y$ since $\mathcal{X} \cap \mathcal{Y} = \emptyset$).

▶ **Definition 2.10.** Let $f : \{0,1\}^n \to \{0,1\}$ be a non-constant function. The *KW relation of* f, denoted KW_f, is defined by $KW_f \stackrel{\text{def}}{=} KW_{f^{-1}(0)\times f^{-1}(1)}$.

We are now ready to state the connection between formulas and KW relations. We state the connection for general rectangles, and the specialization to functions is straightforward.

▶ **Theorem 2.11** (Implicit in [21][4])**.** *Let $\mathcal{X},\mathcal{Y} \subseteq \{0,1\}^n$ be two disjoint sets. Then, for every formula ϕ that separates $\mathcal{X}$ and $\mathcal{Y}$, there exists a protocol Π_ϕ that solves $KW_{\mathcal{X}\times\mathcal{Y}}$, whose underlying tree is the same as the underlying tree of ϕ. In the other direction, for every protocol Π that solves $KW_{\mathcal{X}\times\mathcal{Y}}$ there exists a formula ϕ_Π that separates $\mathcal{X}$ and $\mathcal{Y}$, whose underlying tree is the same as the underlying tree of Π.*

▶ **Corollary 2.12** ([21])**.** *For every two disjoints sets $\mathcal{X},\mathcal{Y} \subseteq \{0,1\}^n$ it holds that $\mathsf{D}(\mathcal{X}\times\mathcal{Y}) = \mathsf{C}(KW_{\mathcal{X}\times\mathcal{Y}})$, and $\mathsf{L}(\mathcal{X}\times\mathcal{Y}) = \mathsf{L}(KW_{\mathcal{X}\times\mathcal{Y}})$. In particular, for every non-constant $f : \{0,1\}^n \to \{0,1\}$, it holds that $\mathsf{D}(f) = \mathsf{C}(KW_f)$, and $\mathsf{L}(f) = \mathsf{L}(KW_f)$.*

Note that due to the connection between formula depth and formula size (Theorem 2.4), it holds that the communication complexity $\mathsf{C}(KW_f)$ and the logarithm of the protocol size $\log \mathsf{L}(KW_f)$ are always within constant factor of each other. In order to streamline the presentation, in many of the intuitive discussions in this paper we will identify those two measures: for example, we will say that "Alice and Bob must transmit t bits" and mean that

[3] This parameter is usually called the "protocol partition number" [25], but we prefer to use the term "protocol size" in order to streamline the presentation.

[4] This fact was discussed explicitly in [29, 19, 12].

the protocol size is at least 2^t. However, our formal results will always be about the protocol size.

Throughout this work, we will rely extensively on the following *sub-additivity property* of protocol size and formula complexity: for every $\mathcal{X}, \mathcal{Y} \subseteq \{0,1\}^n$ such that $\mathcal{X} = \mathcal{X}_0 \cup \mathcal{X}_1$ and $\mathcal{Y} = \mathcal{Y}_0 \cup \mathcal{Y}_1$, it holds that

$$\begin{aligned} \mathsf{L}(\mathcal{X} \times \mathcal{Y}) &\leq \mathsf{L}(\mathcal{X}_0 \times \mathcal{Y}) + \mathsf{L}(\mathcal{X}_1 \times \mathcal{Y}) \\ \mathsf{L}(\mathcal{X} \times \mathcal{Y}) &\leq \mathsf{L}(\mathcal{X} \times \mathcal{Y}_0) + \mathsf{L}(\mathcal{X} \times \mathcal{Y}_1). \end{aligned}$$

To see why the first inequality holds, consider the following protocol for $KW_{\mathcal{X}\times\mathcal{Y}}$: Alice starts by saying whether her input belongs to $\mathcal{X}_0$ or to $\mathcal{X}_1$. Then, the players proceed by invoking the optimal protocol for either $KW_{\mathcal{X}_0\times\mathcal{Y}}$ or $KW_{\mathcal{X}_1\times\mathcal{Y}}$. It is easy to see that the size of this protocol is at most $\mathsf{L}(\mathcal{X}_0 \times \mathcal{Y}) + \mathsf{L}(\mathcal{X}_1 \times \mathcal{Y})$. The proof of the second inequality is similar.

2.4 Information theory

We use basic concepts from information theory, see [9] for more details.

▶ **Definition 2.13** (Entropy)**.** The *entropy* of a random variable x is

$$H(x) \stackrel{\text{def}}{=} \mathbb{E}_{x_0 \leftarrow x}\left[\log \frac{1}{\Pr[x = x_0]}\right] = \sum_{x_0} \Pr[x = x_0] \cdot \log \frac{1}{\Pr[x = x_0]}.$$

The *conditional entropy* $H(x|y)$ is defined to be $\mathbb{E}_{y_0 \leftarrow y}[H(x|y = y_0)]$.

▶ **Fact 2.14.** *$H(x)$ is non-negative and is upper bounded by the logarithm of the size of the support of x. Equality is attained when x is uniformly distributed over its support.*

The notion of mutual information between two variables x and y, defined next, measures how much information x gives on y and vice versa. Intuitively, the information that x gives on y is captured by how much the uncertainty about y decreases when x becomes known.

▶ **Definition 2.15** (Mutual Information)**.** The *mutual information* between two random variables x, y, denoted $I(x : y)$ is defined as

$$I(x : y) \stackrel{\text{def}}{=} H(x) - H(x|y) = H(y) - H(y|x). \tag{1}$$

For a random variable z, the *conditional mutual information* $I(x; y|z)$ is defined as

$$I(x : y|z) \stackrel{\text{def}}{=} H(x|z) - H(x|y, z) = H(y|z) - H(y|x, z).$$

▶ **Fact 2.16.** *For all random variables x, y, z it holds that*

$$0 \leq I(x : y|z) \leq H(x|z) \leq H(x).$$

▶ **Definition 2.17.** The min-entropy of a random variable x is

$$H_\infty(x) = \min_{x_0}\left\{\log \frac{1}{\Pr[x = x_0]}\right\}.$$

In other words, $H_\infty(x)$ is the minimum number h such that $\Pr[x = x_0] = 2^{-h}$ for some x_0.

The following fact is an immediate consequence of the definitions of entropy and min-entropy.

▶ **Fact 2.18.** $H_\infty(x) \leq H(x)$.

2.5 The lower bound for parity

Since our main result is a lower bound on $KW_{f\diamond\oplus_n}$, it is helpful to recall a proof of the lower bound for $KW_{\oplus_n}$. We prove that every protocol that solves $KW_{\oplus_n}$ must transmit at least $2\log n$ bits, and more generally, must have at least n^2 distinct transcripts. We use the following fact from the field of interactive information complexity, which intuitively says that the information that Alice and Bob learn from the execution of a protocol is at most the information that an external observer learns.

▶ **Fact 2.19** ([6]). *Let Π be a protocol, and let $x$and y be random inputs to Alice and Bob in Π respectively. Let $\pi = \Pi(x,y)$ denote the transcript of Π when given x and y as inputs. Then*

$$I(\pi : x, y) \geq I(\pi : x|y) + I(\pi : y|x).$$

We also use the following definition of an edge of the boolean hypercube.

▶ **Definition 2.20.** An *edge (of the boolean hypercube)* is a pair of strings (x, y) in $\{0,1\}^n$ such that the parity of x is 0, and such that x and y differ on exactly one coordinate, which is called the *axis* of the edge.

We are now ready to prove the lower bound. The following proof is due to [12], and is based on the proof of [21].

▶ **Theorem 2.21** ([22]). *It holds that $\mathsf{L}(KW_{\oplus_n}) \geq n^2$.*

Proof. Fix a protocol Π that solves the $KW_{\oplus_n}$. Let (x, y) be a uniformly distributed edge of the hypercube, and let j denote its axis. The intuition for the proof is the following: At the end of the protocol, Alice and Bob must learn j, since it is the only valid output for (x, y). On the other hand, at the beginning of the protocol, Alice and Bob know nothing about j. Hence, throughout the protocol, each of them has to learn at least $\log n$ bits. In particular, this means that each of them has to send at least $\log n$ bits to the other, and therefore the protocol must send at least $2\log n$ bits in total.

Let $\pi = \Pi(x, y)$ be the transcript of the protocol when Alice and Bob get x and y as inputs. Since the entropy of a random variable is upper bounded by the logarithm of the size of its support, it holds that

$$\log \mathsf{L}(\Pi) \geq H(\pi) \geq I(\pi : x, y).$$

Hence, it suffices to prove that $I(\pi : x, y) \geq 2\log n$. By Fact 2.19, it holds that

$$I(\pi : x, y) \geq I(\pi : x|y) + I(\pi : y|x).$$

We prove that both terms on the right hand side the are equal to $\log n$, and this will imply the desired lower bound. For $I(\pi : y|x)$, observe that

$$I(\pi : y|x) = H(y|x) - H(y|x, \pi) = H(j|x) - H(j|x, \pi),$$

where the second equality holds because x and j together determine y, and x and y together determine j. Now, the term $H(j|x, \pi)$ is 0, because the transcript π reveals j (since it tells where x and y differ). As for the term $H(j|x)$, observe that j is uniformly distributed even conditioned on x, and therefore $H(j|x) = \log n$. It thus follows that $I(\pi : y|x) \geq \log n$. Similarly, it holds that $I(\pi : x|y) = \log n$. Those two equalities imply together than

$$\log \mathsf{L}(\Pi) \geq I(\pi : x, y) \geq I(\pi : x|y) + I(\pi : y|x) = 2\log n,$$

as required. ◀

2.6 Error-Correcting Codes

A *code* $C : \{0,1\}^n \to \{0,1\}^{n'}$ is an injective function. The images of the code C are called *codewords*, and we say that C has *relative distance* δ if the relative distance between every two distinct codewords c, c' is at least δ. The parameters n and n' are called the *message length* and the *block length* respectively. We use the following fact from coding theory:

▶ **Fact 2.22.** *Let $m, n \in \mathbb{N}$ be numbers such that $2^{m/2} \geq n$. Then, there exists a code $C : \{0,1\}^n \to \{0,1\}^{2^m}$ with relative distance at least $\frac{1}{2} - \frac{1}{2} \cdot \frac{n}{2^{m/2}}$. Furthermore, there exists a polynomial-time algorithm when given as input m, n, and $x \in \{0,1\}^n$, computes $C(x)$.*

Proof sketch. The code C is the concatenation of a Reed-Solomon code of block length $2^{m/2}$ and degree $n/(m/2)$ over $\mathbf{GF}(2^{m/2})$, and the Hadamard code of message length $m/2$. It is easy to see that the concatenated code has the required message length and block length. For the relative distance, observe that the Reed-Solomon code has relative distance $1 - \frac{n/(m/2)}{2^{m/2}} \geq 1 - \frac{n}{2^{m/2}}$, and that the Hadamard code has relative distance $\frac{1}{2}$. Hence, the concatenated code has relative distance at least $\frac{1}{2} - \frac{1}{2} \cdot \frac{n}{2^{m/2}}$, as required. ◀

We say that a code $C : \{0,1\}^n \to \{0,1\}^{n'}$ is (ρ, L)-*list decodable* if for every $w \in \{0,1\}^{n'}$, there are at most L codewords c whose relative distance to w is less than ρ. We use the following binary version of the Johnson bound, taken from [33].

▶ **Theorem 2.23** (Johnson bound). *A code $C : \{0,1\}^n \to \{0,1\}^{n'}$ with relative distance δ is (ρ, n')-list decodable for $\rho \stackrel{\text{def}}{=} \frac{1}{2} \cdot \left(1 - \sqrt{1 - 2 \cdot \delta}\right)$.*

By combining the Johnson bound with Fact 2.22, we get the following result.

▶ **Corollary 2.24.** *The code C of Fact 2.22 is $(\rho, 2^m)$-list decodable for $\rho \stackrel{\text{def}}{=} \frac{1}{2} - \frac{1}{2} \cdot \sqrt{\frac{n}{2^{m/2}}}$.*

3 Main theorem

In this section, we describe the proof of our main theorem, restated next.

▶ **Theorem 1.2** (restated). *Let $f : \{0,1\}^m \to \{0,1\}$ be a non-constant function. Then,*

$$\mathsf{L}(f \diamond \oplus_n) \geq \frac{\mathsf{L}(f) \cdot \mathsf{L}(\oplus_n)}{2^{\tilde{O}(\sqrt{m + \log n})}}.$$

We actually prove the equivalent statement that says that any protocol that solves $KW_{f \diamond \oplus_n}$ has at least $\mathsf{L}(f) \cdot \mathsf{L}(\oplus_n)/2^{\tilde{O}(\sqrt{m+\log n})}$ distinct transcripts.

The rest of this section is organized as follows: In Section 3.1, we state a structural result about protocols that solve $KW_{f \diamond \oplus_n}$. Then, in Section 3.2, we prove the structural result based on two lemmas that are proved in Sections 5 and 7 respectively. Next, in Section 3.3, we explain how to derive the main theorem from the structural result. Finally, in Section 3.4, we show how to derive the cubic lower bounds for Andreev's function from our main theorem.

3.1 The structural result

Let us recall how the communication problem $KW_{f \diamond \oplus_n}$ is defined. Alice and Bob get $m \times n$ boolean matrices X and Y and should find an entry (i, j) on which the matrices differ. Let $a, b \in \{0,1\}^m$ be the strings obtained by computing the parity of each row of X and Y respectively. Alice and Bob are guaranteed that $a \in f^{-1}(0)$ and $b \in f^{-1}(1)$. We would like

to prove that Alice and Bob must first solve KW_f on a and b, thus finding a row $i \in [m]$ such that $a_i \neq b_i$, and then solve $KW_{\oplus_n}$ on X_i and Y_i.

Fix a protocol Π for $KW_{f\diamond\oplus_n}$ and a partial transcript π_1 of Π. Intuitively, our structural result says that if Alice and Bob have not solved KW_f yet in π_1, then they have to send about $\mathsf{C}(KW_{\oplus_n})$ more bits before they finish solving $KW_{f\diamond\oplus_n}$ (actually, we will show the analogous lower bound on protocol size).

To make sense of the statement "Alice and Bob have not solved KW_f in π_1" we must first see how any protocol for $KW_{f\diamond\oplus_n}$ contains (many copies of) a protocol for KW_f. To this end, we define some notation, starting by recalling the definition of an edge.

▶ **Definition 2.20** (restated). An *edge (of the boolean hypercube)* is a pair of strings (z^0, z^1) in $\{0,1\}^n$ such that the parity of z^0 is 0 and such that z^0 and z^1 differ on exactly one coordinate, which is called the *axis* of the edge.

As we have seen in Section 2.5, the uniform distribution over edges of the boolean hypercube is a hard distribution for $KW_{\oplus_n}$. Therefore, we would like to use edges as inputs to $KW_{f\diamond\oplus_n}$. Now, an input to $KW_{f\diamond\oplus_n}$ contains m inputs to $KW_{\oplus_n}$, and this motivates the following definition.

▶ **Definition 3.1.** A *product of edges* is a pair of $m \times n$ boolean matrices $Z = (Z^0, Z^1)$ such that for every $i \in [m]$, the pair (Z_i^0, Z_i^1) is an edge. Let $\mathcal{Z} = \left\{(Z^0, Z^1)\right\}$ denote the set of all products of edges.

▶ **Definition 3.2.** Given $Z = (Z^0, Z^1) \in \mathcal{Z}$ and a string $w \in \{0,1\}^m$, we denote by Z^w the matrix defined by

$$Z_i^w \stackrel{\text{def}}{=} Z_i^{w_i}$$

for every $i \in [m]$.

Observe that for every $Z \in \mathcal{Z}$, there is a natural reduction from KW_f to $KW_{f\diamond\oplus_n}$: Given inputs $a \in f^{-1}(0)$ and $b \in f^{-1}(1)$ for Alice and Bob in KW_f, we define inputs for Alice and Bob in $KW_{f\diamond\oplus_n}$ by $X = Z^a$ and $Y = Z^b$. We now execute the protocol for $KW_{f\diamond\oplus_n}$ on X and Y, and it outputs an entry (i,j) such that $X_{i,j} \neq Y_{i,j}$. By the definition of X and Y, it follows that $a_i \neq b_i$, and therefore we obtained a solution for KW_f on a and b. The above reduction formalizes the idea that $KW_{f\diamond\oplus_n}$ contains a copy of KW_f for each Z.

Recall that we say that π_1 is *alive* if Alice and Bob have not solved KW_f in π_1. Intuitively, we will define the notion that "π_1 is alive" as follows. First, we will say π_1 is alive *with respect to a specific* Z if after the players sent π_1, they still have to send $\sqrt{m} \cdot \operatorname{poly}\log m$ bits in order to solve the copy of KW_f in $KW_{f\diamond\oplus_n}$that corresponds to Z. We will say that π_1 is alive if it is alive for many (at least 2^{-2m} fraction) of the Zs.

In order to formalize this intuitive definition, we generalize the reduction of KW_f to $KW_{f\diamond\oplus_n}$ to sub-relations of $KW_{f\diamond\oplus_n}$. Let $\mathcal{X} \subseteq (f\diamond\oplus_n)^{-1}(0)$ and $\mathcal{Y} \subseteq (f\diamond\oplus_n)^{-1}(1)$, and note that the rectangle $\mathcal{X}\times\mathcal{Y}$ defines a sub-relation $KW_{\mathcal{X}\times\mathcal{Y}}$ of $KW_{f\diamond\oplus_n}$. Now, given $Z = (Z^0, Z^1)$, we can define a corresponding sub-relation of KW_f by considering the rectangle $\mathcal{A}\times\mathcal{B}$ defined as follows:

$$\begin{aligned}\mathcal{A} &\stackrel{\text{def}}{=} \left\{a \in f^{-1}(0) | Z^a \in \mathcal{X}\right\}\\ B &\stackrel{\text{def}}{=} \left\{b \in f^{-1}(1) | Z^b \in \mathcal{Y}\right\}.\end{aligned}$$

We say that $\mathcal{A}\times\mathcal{B}$ is *the f-rectangle of $\mathcal{X}\times\mathcal{Y}$ with respect to Z*.

We are now ready to formalize what it means for a partial transcript π_1 of Π to be alive. Recall that the transcript π_1 is associated with a sub-rectangle $\mathcal{X}_{\pi_1} \times \mathcal{Y}_{\pi_1}$ of $KW_{f \diamond \oplus_n}$ in a natural way – $\mathcal{X}_{\pi_1} \times \mathcal{Y}_{\pi_1}$ contains all the pairs on inputs on which Alice and Bob transmit π_1. For every product of edges Z, we denote by $\mathcal{A}_{\pi_1,Z} \times \mathcal{B}_{\pi_1,Z}$ the f-rectangle of $\mathcal{X}_{\pi_1} \times \mathcal{Y}_{\pi_1}$ with respect to Z.

▶ **Definition 3.3.** Given a partial transcript π_1 of Π and $Z \in \mathcal{Z}$, we say that π_1 is ℓ-*alive with respect to* Z if $\mathsf{L}(\mathcal{A}_{\pi_1,Z} \times \mathcal{B}_{\pi_1,Z}) \geq 2^{\ell}$. We say that π_1 is (ℓ, α)-*alive* if it is ℓ-alive with respect to α fraction of the Z's, i.e., if

$$\Pr_{Z \in \mathcal{Z}} \left[\mathsf{L}(\mathcal{A}_{\pi_1,Z} \times \mathcal{B}_{\pi_1,Z}) \geq 2^{\ell} \right] \geq \alpha.$$

For our proof we will use $\alpha \stackrel{\text{def}}{=} 2^{-2m}$ and $\ell \stackrel{\text{def}}{=} C \cdot \sqrt{m} \cdot \log^C m$ for large enough constant $C > 0$. We say that π_1 is *alive* as short-hand for $(\ell, 2^{-2m})$-alive. We can finally state our structural result formally.

▶ **Theorem 3.4** (Structure theorem). *Let Π be a protocol for $KW_{f \diamond \oplus_n}$ and let π_1 be a live partial transcript of Π. Then, there exist at least $\mathsf{L}(\oplus_n)/2^{\tilde{O}(\sqrt{m})}$ distinct suffixes π_2 such that $\pi_1 \circ \pi_2$ is a (full) transcript of Π.*

3.2 Proof of the structure theorem

Let Π be a protocol for $KW_{f \diamond \oplus_n}$, and let π_1 be a live partial transcript of Π. Intuitively, we wish to prove that after Alice and Bob have transmitted the messages in π_1, they have to transmit another $\log \mathsf{L}(\oplus_n) - \tilde{O}(\sqrt{m})$ bits in order to solve $KW_{f \diamond \oplus_n}$. To this end, we will design a distribution over inputs $X \in \mathcal{X}_{\pi_1}$ and $Y \in \mathcal{Y}_{\pi_1}$ for Alice and Bob, and show that in order to solve $KW_{f \diamond \oplus_n}$ on inputs coming from this distribution, Alice and Bob must transmit $\log \mathsf{L}(\oplus_n) - \tilde{O}(\sqrt{m})$ bits (and thus must have $\mathsf{L}(KW_{\oplus_n})/2^{\tilde{O}(\sqrt{m})}$ distinct transcripts).

In order to design the latter distribution, we use the fact that the hardest distribution over inputs for $KW_{\oplus_n}$ is the uniform distribution over edges of the boolean hypercube (see Section 2.5). Our distribution for $KW_{f \diamond \oplus_n}$ will look roughly as follows. Let $\mathcal{Z}_{\pi_1} \subseteq \mathcal{Z}$ be the set of Zs that are "alive for π_1", i.e. for which $\mathsf{L}(\mathcal{A}_{\pi_1,Z} \times \mathcal{B}_{\pi_1,Z}) \geq 2^{\ell}$. We will choose a random $Z \in \mathcal{Z}_{\pi_1}$, then pick $a \in \mathcal{A}_{\pi_1,Z}$ and $b \in \mathcal{B}_{\pi_1,Z}$ at random, and then set $X = Z^a$ and $Y = Z^b$.

Observe that X and Y have the following property: for every $i \in [m]$, it either holds that $X_i = Y_i$ (when $a_i = b_i$) or that X_i and Y_i form an edge (when $a_i \neq b_i$). In particular, it is intuitively clear that when given inputs from this distribution, Alice and Bob must solve $KW_{\oplus_n}$ on some X_i and Y_i that form an edge. If we could show that this edge is always uniformly distributed, we could easily complete the argument by showing that Alice and Bob must send $\log \mathsf{L}(\oplus_n)$ bits. Indeed this would work if Z was uniform over *all* products of edges, i.e. if $\mathcal{Z}_{\pi_1} = \mathcal{Z}$.

Unfortunately, $\mathcal{Z}_{\pi_1}$ consists only of $\alpha = 2^{-2m}$ fraction of the products of edges, and therefore we cannot guarantee that X_i and Y_i form a *uniformly distributed* edge. However, intuitively, Alice and Bob "know" only $2m$ bits of information on Z, and therefore they know only two bits of information on the average row (X_i, Y_i). By an averaging argument, on most rows, Alice and Bob know very little information. Such rows are still hard for $KW_{\oplus_n}$, and therefore Alice and Bob must still transmit about $\log \mathsf{L}(\oplus_n)$ in order to solve $KW_{f \diamond \oplus_n}$ on one of those rows. If this is the case, we are done.

One must be careful because there could still be a few "revealed" rows on which Alice and Bob have a lot of information, and such rows might be easy for $KW_{\oplus_n}$. In order to prevent

Alice and Bob from solving $KW_{f\diamond\oplus_n}$ on those rows, we choose the distribution such that for every such row i it holds that $a_i = b_i$. This forces the equality $X_i = Y_i$, and therefore prevents Alice and Bob from solving $KW_{f\diamond\oplus_n}$ on the i-th row.

The following definition captures the essential properties of our distribution. The amount of information that Alice and Bob know about an edge is modeled using the min-entropy of the axis of the edge. The parameter t specifies the maximal amount of information that Alice and Bob may know on the axis of a row, and the set $\mathcal{R}$ consists of the rows on which Alice and Bob have too much information.

▶ **Definition 3.5.** Let X and Y be random $m \times n$ matrices. We say that (X, Y) is a *t-almost hard distribution* if there exists a set $\mathcal{R} \subseteq [m]$ such that the following properties hold:

- For every $i \in \mathcal{R}$, it holds that $X_i = Y_i$.
- For every $i \in [m] - \mathcal{R}$, there is a random coordinate j_i such that
  - Either $X_i = Y_i$, or $X_i = Y_i + \mathbf{e}_{j_i}$ i.e. X_i and Y_i form an edge with axis j_i.
  - For every specific choice X^* of X it holds that $H_\infty(j_i | X = X^*) \geq \log n - t$.
  - For every specific choice Y^* of Y it holds that $H_\infty(j_i | Y = Y^*) \geq \log n - t$.

The above argument is implemented in the following two lemmas, which are proved in Sections 7 and 5 respectively, and which together imply the structure theorem immediately. The first lemma says that there exists an almost-hard distribution over inputs that are consistent with π_1. Since this is the most involved part in our proof, we refer to it as the "main lemma".

▶ **Lemma 3.6** (Main Lemma). *Let Π be a protocol for $KW_{f\diamond\oplus_n}$, and let π_1 be a live partial transcript of Π. Then, there exists a t-almost hard distribution that is supported on $\mathcal{X}_{\pi_1} \times \mathcal{Y}_{\pi_1}$, where $t \stackrel{\text{def}}{=} c \cdot \sqrt{m} \cdot \log^c m$ for some absolute constant $c > 0$.*

The second lemma states that a hard distribution is indeed hard, i.e., that on inputs from this distribution, the players must transmit about $\log \mathsf{L}(\oplus_n)$ bits. We refer to this lemma as the "parity-stage lemma", since it analyzes the stage of the protocol Π in which the players solve $KW_{\oplus_n}$.

▶ **Lemma 3.7** (Parity-stage Lemma). *Let Π_2 be a protocol that solves $KW_{f\diamond\oplus_n}$ on a t-almost hard distribution (X, Y) with probability 1. Then, Π_2 has at least $\mathsf{L}(\oplus_n)/2^{2t}$ transcripts.*

3.3 Proof of the Main Theorem

We now explain how to prove our main theorem using the structure theorem. To this end, we use the following lemma, which says that there are many appropriate live partial transcripts π_1 to which the structure theorem can be applied. We refer to this lemma as the "f-stage lemma" since we view π_1 as the stage of the protocol in which the players solve KW_f.

▶ **Lemma 3.8** (f-stage lemma). *Let Π be a protocol for $KW_{f\diamond\oplus_n}$ of depth d. Then, there exist at least $\mathsf{L}(f)/\left(2^{\tilde{O}(\sqrt{m})} \cdot d^2\right)$ alive partial transcripts π_1 of Π, none of them is an ancestor of another.*

The intuition for the f-stage lemma is straightforward: if the players spoke less than

$$\log \mathsf{L}(f) - \tilde{O}(\sqrt{m}) \leq \mathsf{C}(KW_f) - \tilde{O}(\sqrt{m})$$

bits, then they could not have solved KW_f yet. The proof is is provided in Section 4.

We turn to the proof of the main theorem. Let Π be a protocol that solves $KW_{f\diamond\oplus_n}$. We would like to show that it has at least $\mathsf{L}(f)\cdot\mathsf{L}(\oplus_n)/2^{\tilde{O}(\sqrt{m+\log n})}$ distinct transcripts. The natural way to do so would be the following: first, we would apply the f-stage lemma to show that there are $\approx \mathsf{L}(f)$ alive partial transcripts π_1. Then, we would apply the structure theorem to those transcripts, thus showing that each of them has $\approx \mathsf{L}(g)$ suffixes. We would conclude that Π has $\approx \mathsf{L}(f)\cdot\mathsf{L}(g)$ distinct transcripts, as required.

This proof almost works, but has one issue: the f-stage lemma loses a factor that depends on the depth of Π. Thus, if Π has very large depth, the number of alive partial transcripts π_1 may be insufficient to prove the desired lower bound. In order to resolve this issue, we use a theorem that says that any protocol can be "balanced", i.e., every protocol can be transformed into an equivalent protocol whose depth is logarithmic in its size. We apply this theorem to Π to obtain a new balanced protocol Π', and then apply the foregoing proof to Π'. Specifically, we use the following theorem, which was stated in Section 2 for formulas, and which we now restate for protocols solving KW relations:

▶ **Theorem 2.4** (restated – [4], following [31, 7]). *For every $\alpha > 0$ the following holds: Let Π be a protocol of size s that solves a KW relation KW_f. Then, there exists a protocol Π' of depth at most $O(2^{\frac{1}{\alpha}}\cdot\log s)$ and size at most $s^{1+\alpha}$ that solves KW_f.*

Proof of the main theorem. Let Π be a protocol that solves $KW_{f\diamond\oplus_n}$, and let us denote its size by S. We wish to prove that $S \geq \mathsf{L}(f)\cdot\mathsf{L}(\oplus_n)/2^{\tilde{O}(\sqrt{m+\log n})}$. We may assume without loss of generality that

$$S \leq \mathsf{L}(f)\cdot\mathsf{L}(\oplus_n) \leq 2^m\cdot n^2,$$

since otherwise we are done. We apply Theorem 2.4 to Π with $\alpha = \frac{1}{\sqrt{m+\log n}}$, thus obtaining a new protocol Π' whose depth and size are

$$\begin{aligned} d' &\leq O\left(2^{\sqrt{m+\log n}}\cdot(m+2\log n)\right) = 2^{O(\sqrt{m+\log n})} \\ S' &\leq S^{1+\frac{1}{\sqrt{m+\log n}}}, \end{aligned}$$

respectively. We will prove that $S' \geq \mathsf{L}(f)\cdot\mathsf{L}(\oplus_n)/2^{\tilde{O}(\sqrt{m+\log n})}$, and this will imply the same lower bound for S as follows:

$$\begin{aligned} S &\geq (S')^{1/(1+\frac{1}{\sqrt{m+\log n}})} \\ &\geq (S')^{1-\frac{1}{\sqrt{m+\log n}}} \\ &= S'/(S')^{\frac{1}{\sqrt{m+\log n}}} \\ (\text{Since } S' \leq S^2) \quad &\geq S'/S^{\frac{2}{\sqrt{m+\log n}}} \\ (\text{Since } S \leq 2^m\cdot n^2) \quad &\geq S'/\left(2^m\cdot n^2\right)^{\frac{2}{\sqrt{m+\log n}}} \\ &= S'/2^{O(\sqrt{m+\log n})} \\ &\geq \mathsf{L}(f)\cdot\mathsf{L}(\oplus_n)/2^{\tilde{O}(\sqrt{m+\log n})}. \end{aligned}$$

In order to prove that $S' \geq \mathsf{L}(f)\cdot\mathsf{L}(\oplus_n)/2^{\tilde{O}(\sqrt{m+\log n})}$, we apply the f-stage lemma to Π', thus obtaining a collection of $\mathsf{L}(f)/2^{\tilde{O}(\sqrt{m+\log n})}$ alive partial transcripts π_1, none of which is an ancestor of another. For each such π_1, we apply the structure theorem and obtain $\mathsf{L}(\oplus_n)/2^{\tilde{O}(\sqrt{m})}$ distinct suffixes π_2 such that $\pi_1\circ\pi_2$ is a transcript of Π'. Since none of the

π_1's is an ancestor of another, all the transcripts $\pi_1 \circ \pi_2$ obtained in this way are distinct. It follows that the number of distinct transcripts $\pi_1 \circ \pi_2$ constructed in this way is at least

$$\mathsf{L}(f) \cdot \mathsf{L}(\oplus_n)/2^{\tilde{O}(\sqrt{m+\log n})},$$

as required. ◀

3.4 Cubic Lower Bounds for Andreev's Function

In this section, we derive the cubic lower bounds for Andreev's function from our main theorem. The following argument is due to Andreev [1], and was used in all the works on Andreev's function.

▶ **Theorem 1.1** (restated). *Let $And_n : \{0,1\}^n \to \{0,1\}$ be Andreev's function [1] over n bits. Then,*

$$\mathsf{L}(And_n) \geq n^{3-o(1)}.$$

Andreev's function is defined as follows: the input consists of two parts, each of length $n/2$. The first part is the truth table of a function $f : \{0,1\}^m \to \{0,1\}$ over $m \stackrel{\text{def}}{=} \log(n/2)$ bits. The second part is a sequence $x_1, \ldots, x_m$ of strings in $\{0,1\}^{n/2m}$. Andreev's function is now defined by

$$And_N(f, x_1, \ldots, x_m) \stackrel{\text{def}}{=} (f \diamond \oplus_{\frac{n}{2m}})(x_1, \ldots, x_m).$$

Proof of Theorem 1.1. It is well known that there are functions over m bits whose formula complexity is at least $2^m/\log m$ (see, e.g., [17, Theorem 1.23]). We fix the input $f : \{0,1\}^m \to \{0,1\}$ of And_n to be such a function. Clearly, the formula complexity of And_n can only be decreased by such a fixing. After the fixing, the function And_n is exactly the function $f \diamond \oplus_{\frac{n}{2m}}$. By our main theorem, the formula complexity of the latter function is at least

$$2^{m-\tilde{O}(\sqrt{m+\log n})} \cdot \left(\frac{n}{2m}\right)^2 = n^{3-\tilde{O}(\sqrt{\log n})}.$$

Therefore, the formula complexity of And_n is at least $n^{3-o(1)}$, as required. ◀

4 The f-Stage Lemma

In this section we prove the f-stage lemma. Before we restate the lemma, let us restate the definition of a *alive* partial transcript.

▶ **Definition 3.3** (restated). Given a partial transcript π_1 of Π and $Z \in \mathcal{Z}$, we say that π_1 is *ℓ-alive with respect to Z* if $\mathsf{L}(\mathcal{A}_{\pi_1,Z} \times \mathcal{B}_{\pi_1,Z}) \geq 2^\ell$. We say that π_1 is *(ℓ,α)-alive* if it is ℓ-alive with respect to α fraction of the Z's, i.e., if

$$\Pr_{Z \in \mathcal{Z}}\left[\mathsf{L}(\mathcal{A}_{\pi_1,Z} \times \mathcal{B}_{\pi_1,Z}) \geq 2^\ell\right] \geq \alpha.$$

We say that π_1 is *alive* if it is $(\ell = C \cdot \sqrt{m} \cdot \log^C m, \alpha = 2^{-2m})$-alive (where C is some large constant to be fixed later).

▶ **Lemma 3.8** (restated – f-stage lemma). *Let Π be a protocol for $KW_{f \diamond \oplus_n}$ of depth d. Then, there exist at least $\mathsf{L}(f)/\left(2^{\tilde{O}(\sqrt{m})} \cdot d^2\right)$ alive partial transcripts π_1 of Π, none of them is an ancestor of another.*

For the rest of this section, we fix Π to be a protocol for $KW_{f\diamond\oplus_n}$. Let $\ell \stackrel{\text{def}}{=} C\cdot\sqrt{m}\cdot\log^C m$ be the parameter from the definition of "alive". We will prove that Π has at least $\mathsf{L}(f)/O(2^\ell\cdot d^2)$ partial transcripts π_1 are alive, none of them is an ancestor of another.

This section is organized as follows: We start with a motivating discussion for the proof in Section 4.1. Next, in Section 4.2, we prove the f-stage lemma based on a combinatorial lemma, which is then proved in Section 4.3. Finally, in Section 4.4, we state and prove a generalization of the f-stage lemma, which will be used in Section 8 below to prove the average-case version of the main theorem.

4.1 Motivation

The basic intuition for the f-stage lemma is the following: Recall that for every product of edges $Z \in \mathcal{Z}$, there is copy of KW_f that is embedded in $KW_{f\diamond\oplus_n}$, obtained by mapping inputs a and b for KW_f into the inputs $X \stackrel{\text{def}}{=} Z^a$ and $Y \stackrel{\text{def}}{=} Z^b$ for $KW_{f\diamond\oplus_n}$.

Now, suppose that we choose a uniformly distributed $Z \in \mathcal{Z}$ and some inputs a and b according to some (unspecified) distribution, and then we run the protocol Π on inputs $X \stackrel{\text{def}}{=} Z^a$ and $Y \stackrel{\text{def}}{=} Z^b$ until it transmits $\log\mathsf{L}(f)-\ell$ bits. Let π_1 be the resulting transcript. Intuitively, since Π only transmitted $\log\mathsf{L}(f)-\ell$ bits in π_1, the players must still transmit at least ℓ bits in order to solve KW_f. On the other hand, since $\log\mathsf{L}(f)-\ell \le 2m$, the protocol has revealed at most $2m$ bits of information on Z. Therefore, we expect that after transmitting π_1, the players will still be "ℓ bits far" from solving the copy KW_f for at least 2^{-2m} fraction of the Z's – and this is roughly the definition of π_1 being alive.

The above intuitive argument can be formalized, and it shows that there exists at least one alive transcript π_1 of length $\log\mathsf{L}(f)-\ell$. However, we want to prove something stronger: we want to prove that there exist *many* alive transcripts π_1 – specifically, we wish to prove that there are about $\mathsf{L}(f)/2^\ell$ such transcripts. It turns out that this claim is more difficult to prove. To see why, it is useful to consider the following simpler version of the f-stage lemma, which refers to KW_f rather than $KW_{f\diamond\oplus_n}$:

▶ **Lemma.** *Let Π_f be a protocol that solves KW_f. Then, there exist $\mathsf{L}(f)/2^\ell$ partial transcripts π_f of Π_f whose corresponding rectangle $\mathcal{A}_{\pi_f}\times\mathcal{B}_{\pi_f}$ satisfies $\mathsf{L}(\mathcal{A}_{\pi_f}\times\mathcal{B}_{\pi_f})\ge 2^\ell$, none of them is an ancestor of another.*

It turns out that this "lemma" is false. To see why, consider the protocol Π_f for KW_f in which Alice sends Bob the unary representation of her input a – in other words, Alice views a as a number and sends the string 1^a0 to Bob. After receiving Alice's message, Bob knows a coordinate i such that $a_i \ne b_i$ and sends it to Alice using $\log m$ bits. It is now easy to see that every partial transcript π of the form 1^t0 satisfies

$$\mathsf{L}(\mathcal{A}_\pi\times\mathcal{B}_\pi)\le m \ll 2^\ell.$$

Therefore, the only partial transcripts π_f for which $\mathsf{L}(\mathcal{A}_{\pi_f}\times\mathcal{B}_{\pi_f})\ge 2^\ell$ are those of the form 1^t for some $t\in\mathbb{N}$. However, it is obvious that we cannot find even two such transcripts such that neither of them is an ancestor of the other, and therefore the claim is false.

A notable feature of the counterexample Π_f above is that it is very unbalanced; in particular, its depth is more than 2^m. It turns out that a variant of the above lemma holds if we consider only protocols Π_f that are not too deep. Specifically, we have the following result.

▶ **Lemma 4.1.** *Let Π_f be a protocol of depth d that solves KW_f. Then, there exist*

$$\frac{\mathsf{L}(f)}{(d+1)\cdot d\cdot 2^{\ell}}$$

partial transcripts π_f of Π_f whose corresponding rectangle $\mathcal{A}_{\pi_f}\times\mathcal{B}_{\pi_f}$ satisfies $\mathsf{L}(\mathcal{A}_{\pi_f}\times\mathcal{B}_{\pi_f})\geq 2^{\ell}$, none of them is an ancestor of another.

As far as we know, Lemma 4.1 is new, and we believe that it is interesting in its own right. In order to go from Lemma 4.1 to the f-stage lemma, we first observe that the only property of protocols that Lemma 4.1 uses is the fact that the protocol size $\mathsf{L}(\cdot)$ is a sub-additive measure. We therefore generalize Lemma 4.1 to a general lemma about sub-additive measures on trees:

▶ **Definition 4.2.** Given a rooted binary tree $T=(V,E)$, we say that $\phi : V\to\mathbb{N}$ is a *sub-additive measure on T* if for every vertex u with children v and w in T it holds that $\phi(u)\leq\phi(v)+\phi(w)$.

▶ **Lemma 4.3.** *Let $T=(V,E)$ be a rooted binary tree with root r and depth d, and let ϕ be a sub-additive measure on T. Suppose that there is some $t_0\in\mathbb{N}$ such that $\phi(l)\leq t_0$ for every leaf l of T. Then, for every $t\in\mathbb{N}$ such that $t\geq t_0$ there are at least*

$$\left\lfloor\frac{\phi(r)}{(d+1)\cdot d\cdot t}\right\rfloor$$

vertices v with $\phi(v)\geq t$, none of which is the ancestor of another.

Lemma 4.1 is a special case of Lemma 4.3 where the tree T is the protocol Π_f, and where the sub-additive measure ϕ is defined by

$$\phi(\pi)\stackrel{\text{def}}{=}\mathsf{L}(\mathcal{A}_\pi\times\mathcal{B}_\pi).$$

Now, in order to prove the f-stage lemma, we apply Lemma 4.3 to the protocol Π with a different sub-additive measure. This measure takes into account both the complexity of rectangles of the form $\mathcal{A}_{\pi_1,Z}\times\mathcal{B}_{\pi_1,Z}$ and the number of Z's. We can therefore obtain many transcripts π_1 for which $\mathsf{L}(\mathcal{A}_{\pi_1,Z}\times\mathcal{B}_{\pi_1,Z})$ is large for many of the Z's, as required by the f-stage lemma.

We prove the f-stage lemma from Lemma 4.3 in Section 4.2, and then prove Lemma 4.3 in Section 4.3.

4.2 Proof of the f-stage lemma

Let us view Π as a tree, and its partial transcripts as vertices. We define the following measure on Π:

$$\phi(\pi)\stackrel{\text{def}}{=}\mathbb{E}_Z\left[\mathsf{L}(\mathcal{A}_{\pi,Z}\times\mathcal{B}_{\pi,Z})\right].$$

where the expectation is with respect to a uniformly chosen $Z\in\mathcal{Z}$. This measure is sub-additive since for every fixed Z, the measure $\mathsf{L}(\mathcal{A}_{\pi,Z}\times\mathcal{B}_{\pi,Z})$ is sub-additive. Furthermore, it holds that:

- ϕ assigns $\mathsf{L}(f)$ to the root of Π.
- For every leaf π of Π, it holds that $\phi(\pi)\leq 1$. The reason is that a leaf π must solve $KW_{f\diamond\oplus_n}$, and in particular must solve KW_f with respect to any Z that can reach it. Hence, $\mathsf{L}(\mathcal{A}_{\pi,Z}\times\mathcal{B}_{\pi,Z})\leq 1$.

We now apply Lemma 4.3 to Π and ϕ with $t = 2 \cdot 2^\ell$, and we get that there are at least

$$\frac{\mathsf{L}(f)}{(d+1) \cdot d \cdot 2 \cdot 2^\ell} = \frac{\mathsf{L}(f)}{O(d^2 \cdot 2^\ell)}$$

partial transcripts π_1 such that $\phi(\pi_1) \geq 2 \cdot 2^\ell$, none of them is an ancestor of another. We show that every such transcript π_1 is alive, and this will conclude the proof.

Let π_1 be partial transcript such that $\phi(\pi_1) \geq 2 \cdot 2^\ell$. In other words, it holds that

$$\mathbb{E}_{Z \in \mathcal{Z}} \left[\mathsf{L}(\mathcal{A}_{\pi,Z} \times \mathcal{B}_{\pi,Z}) \right] \geq 2 \cdot 2^\ell.$$

We now apply a standard averaging argument as follows. Since for any Z it holds that $\mathsf{L}(\mathcal{A}_{\pi,Z} \times \mathcal{B}_{\pi,Z}) \leq \mathsf{L}(f)$, there must be at least $2^\ell/\mathsf{L}(f)$ fraction of Z's for which $\mathsf{L}(\mathcal{A}_{\pi,Z} \times \mathcal{B}_{\pi,Z}) \geq 2^\ell$ (otherwise the expectation cannot reach $2 \cdot 2^\ell$). Since $2^\ell/\mathsf{L}(f) > 2^{-2m}$ we conclude that π_1 is $(\ell, 2^{-2m})$-alive, as required.

4.3 Proof of Lemma 4.3

Proof. Fix $t \in \mathbb{N}$. We can assume that $\phi(r) \geq (d+1) \cdot d \cdot t$ since otherwise there is nothing to prove. We say that a vertex v is a *maximal vertex* if $\phi(v) \leq d \cdot t$, and ϕ assigns to its parent a number that is greater than $d \cdot t$. We claim that T has at least $\phi(r)/d \cdot t$ maximal vertices: To see it, observe that there is a maximal vertex on every path from the root r to a leaf (since ϕ takes value t_0 at the leaves and at least $(d+1)dt > dt$ at the root). Hence, by the sub-additivity, if we denote by M the set of maximal vertices, we get that

$$\phi(r) \leq \sum_{v \in M} \phi(v) \leq d \cdot t \cdot |M| \,.$$

This implies that $|M| \geq \phi(r)/d \cdot t$, as required. We say that a maximal vertex v is *good* if $\phi(v) \geq t$, otherwise we say it is *bad*. We will prove that at least $1/(d+1)$ fraction of the maximal vertices are good, and this will imply the required result.

Let T' be the tree obtained by trimming T at maximal vertices – that is, for every maximal vertex v, we remove all the descendants of v and leave v as a leaf of T'. From now on, we refer to maximal vertices as *leaves* (since they are leaves of T'). In the new terminology, we wish to prove that at least $1/(d+1)$ fraction of the leaves of T' are good. We will prove it by constructing a d-to-1 mapping from the bad leaves to the good leaves. In order to construct this mapping, we use the following claim.

▶ **Claim 4.4.** *Every internal node of T' has at least one good leaf as a descendant.*

Proof. It suffices to prove the claim for internal nodes u such that $\phi(u) \leq 2 \cdot d \cdot t$, since every other internal node clearly has an internal descendant that satisfies this property.

Fix an internal node u such that $\phi(u) \leq 2 \cdot d \cdot t$. Now, observe that in the sub-tree rooted at u, every internal node has at least one child that is a leaf. In other words, this sub-tree looks like a path, with a leaf hanging from each vertex in the path. Therefore, this sub-tree contains at most d leaves. If all of those leaves were bad, then $\phi(u)$ would have been less than $d \cdot t$ (since by the sub-additivity, $\phi(u)$ is at most the sum of $\phi(v)$ for every leaf v in the sub-tree). However, we assumed that u is an internal node, and therefore $\phi(u)$ is greater than $d \cdot t$. Hence, at least one of the leaves must be good. ◀

Now, we define the mapping from the bad leaves to the good leaves as follows: Let v_{bad} be a bad leaf, and let u be the parent of v_{bad}. Then, we map v_{bad} to some arbitrarily chosen good leaf v_{good} that is a descendant of u – such a leaf v_{good} exists by the above claim.

We conclude the proof by showing that this mapping is d-to-1. Fix a good leaf v_{good}. Then, all the bad leaves that are mapped to v_{good} are direct children of the ancestors of v_{good}. Since T is of depth d, it follows that v_{good} has at most d ancestors, and therefore there are at most d bad leaves that are mapped to v_{good}. It follows that at least $1/(d+1)$ fraction of the leaves are good, as required. ◀

4.4 Generalized f-stage lemma

In this section, we prove a generalization of the f-stage lemma, that will be used in Section 8 below to prove the average-case version of the main theorem. While the f-stage lemma applies to protocols Π that solve $KW_{f\diamond\oplus_n}$, the generalization applies to protocols that only solve a sub-rectangle $\mathcal{X}\times\mathcal{Y}$ of $KW_{f\diamond\oplus_n}$, provided that $\mathcal{X}\times\mathcal{Y}$ has many "hard" f-rectangles. Recall that given a sub-rectangle $\mathcal{X}\times\mathcal{Y}$ of $KW_{f\diamond\oplus_n}$ and a product of edges Z, the f-rectangle of $\mathcal{X}\times\mathcal{Y}$ with respect to Z is the rectangle $\mathcal{A}\times\mathcal{B}$ defined by:

$$\begin{aligned}\mathcal{A} &\stackrel{\text{def}}{=} \left\{a\in f^{-1}(0) | Z^a\in\mathcal{X}\right\}\\ B &\stackrel{\text{def}}{=} \left\{b\in f^{-1}(1) | Z^b\in\mathcal{Y}\right\}.\end{aligned}$$

We have the following result.

▶ **Lemma 4.5** (generalized f-stage lemma). *Let $s\in\mathbb{N}$. Let $\mathcal{X}\times\mathcal{Y}$ be a sub-rectangle of $KW_{f\diamond\oplus_n}$ such that for at least 2^{-m} fraction of the Z's, the f-rectangle $\mathcal{A}\times\mathcal{B}$ of $\mathcal{X}\times\mathcal{Y}$ with respect to Z satisfies $\mathsf{L}(\mathcal{A}\times\mathcal{B})\geq s$. Let Π be a protocol for $KW_{\mathcal{X}\times\mathcal{Y}}$ of depth d. Then, there exist at least $s/O(2^\ell\cdot d^2)$ alive partial transcripts π_1 of Π, none of them is an ancestor of another.*

Proof. Let $\mathcal{Z}'$ be the set of Z's for which the f-rectangle $\mathcal{A}\times\mathcal{B}$ of $\mathcal{X}\times\mathcal{Y}$ with respect to Z satisfies $\mathsf{L}(\mathcal{A}\times\mathcal{B})\geq s$. As in the proof of the f-stage lemma in Section 4.2, we apply Lemma 4.3 to Π and ϕ where

$$\phi(\pi)\stackrel{\text{def}}{=}\mathbb{E}_{Z\in\mathcal{Z}'}\left[\mathsf{L}(\mathcal{A}_{\pi,Z}\times\mathcal{B}_{\pi,Z})\right].$$

We can lower bound the value that ϕ assigns to the root of Π by s, and upper bound each leaf by 1. We apply the lemma with $t=2\cdot 2^\ell$. We thus obtain that there are at least

$$\frac{s}{O(d^2\cdot 2^\ell)}$$

partial transcripts π_1 with $\phi(\pi_1)\geq 2\cdot 2^\ell$.

We now apply an averaging argument as before. Since for any $Z\in\mathcal{Z}'$ it holds that $\mathsf{L}(\mathcal{A}_{\pi,Z}\times\mathcal{B}_{\pi,Z})\leq\mathsf{L}(f)$, there must be at least $2^\ell/\mathsf{L}(f)$ fraction of Z's in $\mathcal{Z}'$ for which $\mathsf{L}(\mathcal{A}_{\pi,Z}\times\mathcal{B}_{\pi,Z})\geq 2^\ell$ (otherwise the expectation cannot reach 2ℓ). Since $2^\ell/\mathsf{L}(f)>2^{-m}$ we conclude that $\mathsf{L}(\mathcal{A}_{\pi,Z}\times\mathcal{B}_{\pi,Z})\geq 2^\ell$ for 2^{-m} fraction of $Z\in\mathcal{Z}'$ which is at least 2^{-2m} fraction of $\mathcal{Z}$. So π_1 is $(2^\ell,2^{-2m})$-alive as required. ◀

5 The Parity-Stage Lemma

In this section, we prove the parity-stage lemma, restated next. It is instructive to compare this proof to the proof of the lower bound for $KW_{\oplus_n}$ in Section 2.5.

▶ **Lemma 3.7** (restated). *Let Π_2 be a protocol that solves $KW_{f\diamond\oplus_n}$ on a t-almost hard distribution (X,Y) with probability 1. Then, Π_2 has at least $\mathsf{L}(\oplus_n)/2^{2t}$ transcripts.*

Proof. The basic idea of the proof is similar to that of the lower bound for $KW_{\oplus_n}$ in Section 2.5: At the end of the protocol, Alice and Bob must learn an axis j_i for some $i \in [m] - \mathcal{R}$, since the matrices X and Y differ only such axes. On the other hand, at the beginning of the protocol each of them knows at most t bits on each axis j_i, due to the definition of an almost hard distribution. Therefore, by the end of the protocol, each of the players has to learn at least $\log n - t$ bits of information, and the protocol must transmit at least $2\log n - 2t$ bits in total. The difference between this proof an the proof in Section 2.5 is that in the current proof, the players may choose which axis j_i they will learn among multiple options, and this complicates the argument a bit. Details follow.

Assume, for the sake of contradiction, that there is a protocol Π_2 that solves $KW_{f\diamond\oplus_n}$ on a t-almost hard distribution (X, Y) and is too efficient, i.e., has less than $\mathsf{L}(\oplus_n)/2^{2t}$ transcripts. Let $\pi_2 = \Pi_2(X, Y)$ be the (random) transcript of Π_2 when Alice and Bob get X and Y as inputs. By assumption, the support of π_2 is of size less than $\mathsf{L}(\oplus_n)/2^{2t} = n^2/2^{2t}$, and therefore

$$I(\pi_2 : X, Y) \leq H(\pi_2) < 2\log n - 2t.$$

On the other hand, by Fact 2.19 it holds that

$$I(\pi_2 : X, Y) \geq I(\pi_2 : X|Y) + I(\pi_2 : Y|X).$$

Hence, at least one of the terms on the right-hand side is smaller than $\log n - t$. Without loss of generality, assume that it is $I(\pi_2 : Y|X)$. It thus holds that

$$\log n - t > I(\pi_2 : Y|X) = H(\pi_2|X) - H(\pi_2|X, Y) = H(\pi_2|X),$$

where the last equality holds since X and Y determine π_2. Hence, there exists some specific X^* such that $H(\pi_2|X^*) < \log n - t$. Furthermore, since entropy is an upper-bound on min-entropy (Fact 2.18), it follows that $H_\infty(\pi_2|X^*) < \log n - t$. Therefore, there exists a specific transcript π_2^* such that $\log \frac{1}{\Pr[\pi_2^*|X^*]} < \log n - t$ or in other words,

$$\Pr[\pi_2^*|X^*] > \frac{2^t}{n}. \tag{2}$$

Suppose that this transcript π_2^* ends by outputting (i^*, j^*). Assuming the protocol solves $KW_{f\diamond\oplus_n}$, this means that for all X, Y's consistent with π_2^*, it holds that $X_{i^*,j^*} \neq Y_{i^*,j^*}$.

Now, let $\mathcal{R} \subseteq [m]$ be the set whose existence is guaranteed by the definition of an almost-hard distribution. We consider two cases, $i^* \in \mathcal{R}$ and $i^* \notin \mathcal{R}$, and show that in both cases there is a non-zero probability that $X_{i^*,j^*} = Y_{i^*,j^*}$ conditioned on π_2^*, thus obtaining a contradiction to the correctness of the protocol. Suppose first that $i^* \in \mathcal{R}$. In this case, it holds that $X_{i^*} = Y_{i^*}$ with probability 1. In particular, it follows that $X_{i^*,j^*} = Y_{i^*,j^*}$, as required.

Next, suppose that $i^* \notin \mathcal{R}$. This means that there is a random coordinate j_{i^*} such that either $X_{i^*} = Y_{i^*}$, or X_{i^*} and Y_{i^*} differ only on j_{i^*}. Moreover, it holds that

$$H_\infty(j_{i^*}|X^*) \geq \log n - t,$$

and in particular,

$$\Pr[j_{i^*} = j^*|X^*] \leq \frac{2^t}{n}. \tag{3}$$

By combining Inequalities 2 and 3, it holds that

$$\Pr\left[j_{i^*} = j^* | X^*, \pi_2^*\right] < 1.$$

The latter inequality implies that conditioned on X^* and π_2^*, the event "$j_{i^*} \neq j^*$" has non-zero probability. Now, observe that in this event it must hold that $X_{i^*,j^*} = Y_{i^*,j^*}$, since j_{i^*} is the only coordinate on which X_{i^*} and Y_{i^*} may differ. We conclude $X^*_{i^*j^*} = Y^*_{i^*j^*}$ with non-zero probability and this contradicts the correctness of the protocol. ◀

6 Technical tools

In this section we describe two technical tools that may be of independent interest.

The first is an averaging argument for min-entropy. Basically, it says that if we reveal $t \ll m$ bits of information on an m-tuple, then on most elements almost no information was revealed.

The second tool, which we call fortification, is a way to relate the information transmitted between Alice and Bob to the communication complexity of the residual problem. This is important because some of the steps we take in the proof of the main lemma reveal information to Alice and Bob, and we need to make sure that this does not decrease the complexity of the problem by too much.

6.1 An averaging argument for min-entropy

In our proof of the main lemma, we would like to say that if Alice and Bob communicated a small amount of information on the average row, then they communicated a small amount of information on *most* rows. This requires some sort of an averaging argument for information. Such an averaging argument is easy to prove for entropy, and was proved by [10] for an information measure called "predictability". In this section, we prove such an averaging argument for min-entropy.

On the high level, the averaging argument says that if at most r bits of information were communicated on a tuple $(u_1, \ldots, u_m)$ of random variables, then for every $k \geq 1$, at most $\frac{r}{k}$ bits of information were communicated on all but k of the random variables. As a warm-up, we first prove the following weak version of our averaging argument. We note that the following proof is similar to the proof of [10], and is also in the spirit of standard arguments from the literature on extractors.

▶ **Lemma 6.1** (Weak averaging argument for min-entropy). *Let $\mathcal{U}$ be some finite universe, and let $\overline{u} = (u_1, \ldots, u_m)$ be a tuple of random variables taking values in $\mathcal{U}$ such that $H_\infty(\overline{u}) \geq m \log|\mathcal{U}| - r$. Then, for every $k \geq 1$, there exists a set $\mathcal{R} \subseteq [m]$ of size at most k, and an event $E \subseteq \mathcal{U}^m$ of probability at least $|\mathcal{U}|^{-k}$, such that for every $i \in [m] - \mathcal{R}$ it holds that*

$$H_\infty(u_i | E) \geq \log|\mathcal{U}| - \frac{r}{k}.$$

Proof. In what follows, for every $\mathcal{S} \subseteq [m]$ we denote by $\overline{u}_\mathcal{S}$ the tuple of u_i's that belong to $\mathcal{S}$. We construct the set $\mathcal{R}$ and the event E iteratively. We start with $\mathcal{R} = \emptyset$ and $E = \mathcal{U}^m$. Then, in each iteration, if there is some $i \in [m] - \mathcal{R}$ that violates the above requirement, we add it to the set $\mathcal{R}$. More specifically, if i violates the requirement, then there is some specific value u_i^* such that

$$\Pr\left[u_i^* | E\right] \geq \frac{2^{r/k}}{|\mathcal{U}|}.$$

Then, we add i to $\mathcal{R}$, and add the condition $u_i = u_i^*$ to the event E (i.e., we set E to $E \cap \{\overline{u}' : u'_i = u_i^*\}$). The process stops when there is no $i \in [m] - \mathcal{R}$ that violates the requirement.

It remains to prove that $|\mathcal{R}| \leq k$. To this end, we prove that the following invariant is maintained throughout the iterations:

$$H_\infty(\overline{u}_{[m]-\mathcal{R}}|E) \geq (m - |\mathcal{R}|) \cdot \log|\mathcal{U}| - r + \frac{r}{k} \cdot |\mathcal{R}| \,.$$

This will imply the required upper bound on $|\mathcal{R}|$, since clearly the left-hand side cannot exceed $(m - |\mathcal{R}|) \cdot \log|\mathcal{U}|$.

We prove that the invariant is maintained by induction. First, note that it holds trivially when the process starts, i.e., when $\mathcal{R} = \emptyset$ and $E = \mathcal{U}^m$. Next, suppose that the invariant holds at the beginning of some iteration, and that in this iteration we add a coordinate i to $\mathcal{R}$. Then, for every assignment $\overline{u}^*_{[m]-(\mathcal{R}\cup\{i\})} \in \mathcal{U}^{[m]-(\mathcal{R}\cup\{i\})}$, it holds that

$$\begin{aligned}
& \Pr\left[\overline{u}_{[m]-(\mathcal{R}\cup\{i\})} = \overline{u}^*_{[m]-(\mathcal{R}\cup\{i\})} | E, u_i = u_i^*\right] \\
= \; & \frac{\Pr\left[\overline{u}_{[m]-(\mathcal{R}\cup\{i\})} = \overline{u}^*_{[m]-(\mathcal{R}\cup\{i\})} \text{ and } u_i = u_i^* | E\right]}{\Pr\left[u_i = u_i^* | E\right]} \\
\leq \; & 2^{-\left[(m-|\mathcal{R}|)\cdot\log|\mathcal{U}| - r + \frac{r}{k}\cdot|\mathcal{R}|\right]} / \Pr\left[u_i = u_i^* | E\right] \qquad (4) \\
\leq \; & 2^{-\left[(m-|\mathcal{R}|)\cdot\log|\mathcal{U}| - r + \frac{r}{k}\cdot|\mathcal{R}|\right]} / 2^{-(\log|\mathcal{U}| - r/k)} \qquad (5) \\
= \; & 2^{-\left[(m-|\mathcal{R}|-1)\cdot\log|\mathcal{U}| - r + \frac{r}{k}\cdot(|\mathcal{R}|+1)\right]},
\end{aligned}$$

where Inequality 4 holds due to the induction hypothesis, and Inequality 5 holds since i violates the requirement. This implies that

$$H_\infty(\overline{u}_{[m]-(\mathcal{R}\cup\{i\})}|E, u_i^*) \geq (m - |\mathcal{R}\cup\{i\}|) \cdot \log|\mathcal{U}| - r + \frac{r}{k} \cdot |\mathcal{R}\cup\{i\}| \,,$$

as required. Hence, it holds that $|\mathcal{R}| \leq k$ when the process ends. It is now easy to see that when the process ends, the probability of E is at least $\left(\frac{2^{r/k}}{|\mathcal{U}|}\right)^k \geq |\mathcal{U}|^{-k}$. The result follows. ◀

The reason we say that the above lemma is weak is because it only provides a lower bound of $|\mathcal{U}|^{-k}$ on the probability of the event E, which is very small when $\mathcal{U}$ is large. Intuitively, this means that in order to use this lemma, we need to reveal a lot of information to Alice and Bob. We therefore prove the following stronger version of the lemma that gives a lower bound of $m^{-O(k)}$, which is better when m is much smaller than $\mathcal{U}$, as is the case in our application. To the best of our knowledge, this stronger version of the lemma is new.

The basic idea of the proof is the following: whenever a coordinate i violates the requirement, it is because there was some "heavy" value u_i^*. In the above proof, we resolved this situation by conditioning on $u_i = u_i^*$, but this event may have a very low probability. In order to condition on an event with a higher probability, we consider two cases: If the heavy values, taken together, have relatively high probability, then we condition on the event that u_i takes a heavy value and add i to $\mathcal{R}$. If, on the other hand, the heavy values, taken together, have low probability, then we condition on u_i *not* taking a heavy value, and do *not* add i to $\mathcal{R}$ – hopefully, this will resolve the issue, because after discarding the heavy values, u_i will satisfy the requirement. This idea works, except for two minor issues:

- When we condition on i not taking a heavy value, this conditioning may cause new values to become heavy, even if they were not heavy before. This may get us into a "vicious cycle" of discarding values. In order to resolve this issue, whenever we discard heavy values, we increase the threshold that determines which values are considered heavy, so no new heavy values can be created immediately.
- When we condition u_i on any event – whether it is taking a heavy value or not taking a heavy value – it may cause new values to become heavy *for another random variable* $u_{i'}$. This may get us into a different "vicious cycle", in which we condition u_i, then condition $u_{i'}$, then condition u_i again, etc. In order to resolve this issue, we choose the different parameters such that u_i may cause new heavy values for another coordinate $u_{i'}$ only if u_i was conditioned on taking a heavy value. However, when u_i is conditioned on taking a heavy value, it is added to $\mathcal{R}$, and thus will not be selected again. Thus, the "vicious cycle" cannot happen.

We turn to provide the formal lemma and proof.

▶ **Lemma 6.2** (Averaging argument for min-entropy). *Let $\mathcal{U}$ be some finite universe, and let $\overline{u} = (u_1, \ldots, u_m)$ be a tuple of random variables taking values in $\mathcal{U}$ such that $H_\infty(\overline{u}) \geq m \log |\mathcal{U}| - r$. Then, for every $k \geq 1$, there exists a set $\mathcal{R} \subseteq [m]$ of size at most k, and an event E of probability at least $\frac{1}{4} \cdot m^{-2k}$, such that for every $i \in [m] - \mathcal{R}$ it holds that*

$$H_\infty(u_i|E) \geq \log |\mathcal{U}| - \frac{r+4}{k} - 2 \cdot \log m - 2.$$

Proof. For convenience, we denote

$$\tau \stackrel{\text{def}}{=} \log |\mathcal{U}| - \frac{r+4}{k} - 2 \cdot \log m,$$

that is, τ is the threshold of the lemma except for the additive term of -2.

We construct the set $\mathcal{R}$ and the event E iteratively. We start with $\mathcal{R} = \emptyset$ and $E = \mathcal{U}^m$. In each iteration, we select a coordinate $i \in [m]$ and do something with it. We describe a single iteration: Suppose that there is a coordinate $i \in [m] - \mathcal{R}$ that has been chosen in p previous iterations and that satisfies

$$H_\infty(u_i|E) \leq \tau + \log(1 - \frac{1}{m})^p.$$

Then, we select the coordinate i (the right-hand side is going to be the threshold that controls which values are considered "heavy"). By assumption, there exist values u_i^* such that

$$\Pr[u_i^*|E] \geq 2^{-\tau} / \left(1 - \frac{1}{m}\right)^p.$$

We define those values to be our "heavy values". Let E' be the event that u_i takes a heavy value. We consider two cases:

- If $\Pr[E'|E] \geq \frac{1}{m^2}$, then we set $E = E \cap E'$ and add i to $\mathcal{R}$.
- Otherwise, we set $E = E - E'$.

The following claim deals with the issues from the discussion above. Specifically, it shows that we chose the parameters in a way such that "new heavy values" can be created only by the first case above but not by the second case.

▶ **Claim 6.3.** *The second case above cannot occur twice for the same coordinate i without the first case occurring in between (for some index).*

Proof. Suppose otherwise. This means that there are some coordinate i and numbers $h_1 \leq h_2$ such that the second case occurred for i in both the h_1-th and h_2-th iterations, and the first case did not occur for any coordinate between those two iterations. Without loss of generality, we choose i, h_1, h_2 such that $h_2 - h_1$ is minimal among all the triplets (i, h_1, h_2) that satisfy those conditions.

Let p be the number of iterations in which i has been chosen before the h_1-th iteration. Let E_1 and E_2 be the event E at the h_1-th and the h_2-th iterations respectively. By assumption, only the second case happened for all the coordinates between those two iterations, and all those coordinates have been chosen at most once (because we assumed $h_2 - h_1$ is minimal). This implies that there have been at most m iterations between the h_1-th and h_2-th iterations, and in each of those iterations, the second case occurred. Now, observe that every time the second case occurs, the probability of E is multiplied by a factor that is at least $1 - \frac{1}{m^2}$. Therefore,

$$\Pr\left[E_2|E_1\right] \geq \left(1 - \frac{1}{m^2}\right)^m \geq 1 - \frac{1}{m}.$$

Let E_1' be the event E' at the h_1-th iteration, and observe that $E_2 \subseteq E_1 - E_1'$. Now, for every specific choice u_i^* of u_i in $E_1 - E_1'$, it holds that

$$\Pr\left[u_i^*|E_1\right] < 2^{-\tau} / \left(1 - \frac{1}{m}\right)^p.$$

Therefore, for every u_i^* it holds that

$$\begin{aligned}
\Pr\left[u_i^*|E_2\right] &= \frac{\Pr\left[u_i^* \wedge E_2|E_1\right]}{\Pr\left[E_2|E_1\right]} \\
&\leq \frac{\Pr\left[u_i^*|E_1\right]}{\Pr\left[E_2|E_1\right]} \\
&< \frac{2^{-\tau} / \left(1 - \frac{1}{m}\right)^p}{1 - \frac{1}{m}} \\
&= 2^{-\tau} / \left(1 - \frac{1}{m}\right)^{p+1}.
\end{aligned}$$

But this means that i could not have been selected at the h_2-th iteration, which is a contradiction. ◀

Observe that the first case cannot occur more than m times, and thus, combined with the latter claim, we get that the total number of iterations is at most m^2. In particular, the second case cannot happen for a coordinate i more than m times. Therefore, when the process terminates, every $i \in [m] - \mathcal{R}$ has been selected at most m times (since if $i \notin \mathcal{R}$, the first case never occurred for it). This implies that, when the process terminates, it holds for every $i \in [m] - \mathcal{R}$ that

$$\begin{aligned}
H_\infty(u_i|E) &\geq \tau + \log(1 - \frac{1}{m})^m \\
&\geq \tau - 2,
\end{aligned}$$

where the second inequality holds for sufficiently large m. This means that every $i \in [m] - \mathcal{R}$ satisfies the requirement of the lemma.

We turn to upper bounding the size of the set $\mathcal{R}$. Again, we do it by proving that an invariant is maintained throughout the iterations. Formally, we prove the following.

▶ **Claim 6.4.** *At each iteration, the following invariant is maintained:*

$$H_\infty(\overline{u}_{[m]-\mathcal{R}}|E) \geq (m-|\mathcal{R}|)\cdot\log|\mathcal{U}| - r + \frac{r+4}{k}\cdot|\mathcal{R}| - \frac{4\cdot s}{m^2},$$

where s is the total number of times the second case has occurred before this iteration.

Proof. We prove the claim by induction. Before the first iteration, when $|\mathcal{R}|=\emptyset$, the claim holds by the assumption of the lemma. Fix an iteration, let i be the coordinate that is selected in this iteration, and let s be the number of times the second case occurred before this iteration. We consider each of the two cases that may occur separately.

Suppose that the first case occurred, so $\Pr[E'|E] \geq \frac{1}{m^2}$. Then, for every assignment $\overline{u}^*_{[m]-(\mathcal{R}\cup\{i\})} \in \mathcal{U}^{[m]-(\mathcal{R}\cup\{i\})}$ the following holds:

$$\begin{aligned}
&\Pr\left[\overline{u}^*_{[m]-(\mathcal{R}\cup\{i\})}|E\cap E'\right]\\
=\;& \frac{\Pr\left[\overline{u}^*_{[m]-(\mathcal{R}\cup\{i\})} \text{ and } E'|E\right]}{\Pr[E'|E]}\\
\leq\;& m^2\cdot\Pr\left[\overline{u}^*_{[m]-(\mathcal{R}\cup\{i\})} \text{ and } E'|E\right]\\
=\;& m^2\cdot\sum_{u^*_i \text{ is a heavy value}}\Pr\left[\overline{u}^*_{[m]-(\mathcal{R}\cup\{i\})} \text{ and } u^*_i|E\right]\\
\leq\;& m^2\cdot\sum_{u^*_i \text{ is a heavy value}}2^{-H_\infty(\overline{u}_{[m]-\mathcal{R}}|E)}\\
\leq\;& m^2\cdot 2^{\tau}\cdot 2^{-H_\infty(\overline{u}_{[m]-\mathcal{R}}|E)} \qquad (6)\\
=\;& m^2\cdot\frac{2^{\log|\mathcal{U}|-(r+4)/k}}{m^2}\cdot 2^{-H_\infty(\overline{u}_{[m]-\mathcal{R}}|E)}\\
\leq\;& 2^{\log|\mathcal{U}|-(r+4)/k}\cdot 2^{-\left[(m-|\mathcal{R}|)\cdot\log|\mathcal{U}|-r+\frac{r+4}{k}\cdot|\mathcal{R}|-\frac{4\cdot s}{m^2}\right]} \qquad (7)\\
=\;& 2^{-\left[(m-|\mathcal{R}|-1)\cdot\log|\mathcal{U}|-r+\frac{r+4}{k}\cdot(|\mathcal{R}|+1)-\frac{4\cdot s}{m^2}\right]},
\end{aligned}$$

where Inequality 6 follows from the fact that there can be at most 2^τ heavy values, and Inequality 7 follows from the induction assumption. It follows that

$$H_\infty(\overline{u}_{[m]-(\mathcal{R}-\{i\})}) \geq (m-|\mathcal{R}\cup\{i\}|)\cdot\log|\mathcal{U}| - r + \frac{r+4}{k}\cdot|\mathcal{R}\cup\{i\}| - \frac{4\cdot s}{m^2},$$

as required.

Suppose now that the second case occurred. Then, for every assignment $\overline{u}^*_{[m]-\mathcal{R}} \in \mathcal{U}^{[m]-\mathcal{R}}$ it holds that

$$\begin{aligned}
\Pr\left[\overline{u}^*_{[m]-\mathcal{R}}|E-E'\right] &\leq \frac{\Pr\left[\overline{u}^*_{[m]-\mathcal{R}}|E\right]}{\Pr[E-E'|E]}\\
&\leq \frac{1}{1-\frac{1}{m^2}\cdot}\Pr\left[\overline{u}^*_{[m]-\mathcal{R}}|E\right]\\
&\leq \left(1+\frac{2}{m^2}\right)\cdot\Pr\left[\overline{u}^*_{[m]-\mathcal{R}}|E\right]\\
&\leq \exp(\frac{2}{m^2})\cdot\Pr\left[\overline{u}^*_{[m]-\mathcal{R}}|E\right]\\
&\leq 2^{-H_\infty(\overline{u}_{[m]-\mathcal{R}}|E)+\frac{4}{m^2}}\\
&\leq 2^{-\left[(m-|\mathcal{R}|)\cdot\log n-r+\frac{r+1}{k}\cdot|\mathcal{R}|-\frac{4\cdot(s+1)}{m^2}\right]},
\end{aligned}$$

where the last inequality follows from the induction assumption. It follows that

$$H_\infty(\overline{u}_{[m]-\mathcal{R}}) \geq (m - |\mathcal{R}|) \cdot \log n - r + \frac{r+4}{k} \cdot |\mathcal{R}| - \frac{4 \cdot (s+1)}{m^2},$$

as required. ◀

We can now bound the size of the set $\mathcal{R}$: since it must hold that $H_\infty(\overline{u}_{[m]-\mathcal{R}}|E) \leq (m - |\mathcal{R}|) \cdot \log n$, and since $s \leq m^2$, it follows from the last claim that $|\mathcal{R}| \leq k$, as required. It remains to lower bound the probability of the event E. The probability of E decreases by a factor of $\frac{1}{m^2}$ whenever the first case occurs, and by a factor of $1 - \frac{1}{m^2}$ whenever the second case occurs. The first case occurs at most k times, and the second case occurs at most m^2 times. Hence, the probability of E is at least

$$\left(\frac{1}{m^2}\right)^k \cdot (1 - \frac{1}{m^2})^{m^2} \geq \frac{1}{4} \cdot m^{2k},$$

as required. ◀

6.2 Fortification

In the proof of the main lemma, we will want to relate the information that Alice and Bob transmit about their inputs to the reduction in the complexity of the communication problem. For example, we will want to argue that if Alice and Bob transmitted only one bit of information, then the communication complexity of the problem was decreased by at most one bit (or, alternatively, that the protocol size of the problem was decreased by a factor of at most two).

However, this is not always true. For example, consider a KW relation $KW_{\mathcal{A}\times\mathcal{B}}$ (where $\mathcal{A}, B \subseteq \{0,1\}^m$ are disjoint), and suppose that the first bit of all the strings in $\mathcal{B}$ is 0, while in $\mathcal{A}$, the first bit is 0 for exactly half of the strings. In this case, if Alice tells Bob that the first bit of her input is 1, she only tells him only one bit of information, but the communication complexity of the problem drops to zero – since now Alice and Bob know that they differ on the first bit.

We say that a rectangle $\mathcal{A} \times \mathcal{B}$ is *fortified* [5] (with respect to a given protocol Π) if when Alice and Bob speak, the complexity is decreased in proportion to the information transmitted. More formally, we define fortified rectangles as follows.

▶ **Definition 6.5.** We say that a rectangle $\mathcal{A} \times \mathcal{B}$ is *ρ-fortified on Alice's side* if for every $\tilde{\mathcal{A}} \subseteq \mathcal{A}$ it holds that

$$\frac{\mathsf{L}(\tilde{\mathcal{A}} \times \mathcal{B})}{\mathsf{L}(\mathcal{A} \times \mathcal{B})} \geq \rho \cdot \frac{|\tilde{\mathcal{A}}|}{|\mathcal{A}|}.$$

Similarly, we say that $\mathcal{A} \times \mathcal{B}$ is *ρ-fortified on Bob's side* if the same holds for subsets $\tilde{\mathcal{B}} \subseteq \mathcal{B}$.

In this section, we show that even though there are rectangles $\mathcal{A} \times \mathcal{B}$ that are not fortified, every rectangle has a fortified sub-rectangle with similar complexity. For example, in the non-fortified rectangle $\mathcal{A} \times \mathcal{B}$ described above, we could take the sub-rectangle $\mathcal{A}' \times \mathcal{B}$ where $\mathcal{A}' \stackrel{\text{def}}{=} \{a \in \mathcal{A} : a_1 = 0\}$. More generally, we have the following result.

[5] *The term "fortified" was coined by Moshkovitz [26] in order to denote two-prover games that remain hard when restricted to sub-rectangles. She also proved a fortification lemma that transforms two-prover games into fortified ones. While our notion of fortification is very different from hers on the technical level, there is a conceptual similarity between the two notions.*

▶ **Lemma 6.6** (Fortification lemma). *Let $\mathcal{A}, \mathcal{B} \subseteq \{0,1\}^m$ be disjoint sets. There exists a subset $\mathcal{A}' \subseteq \mathcal{A}$ such that $\mathcal{A}' \times \mathcal{B}$ is $\frac{1}{4m}$-fortified on Alice's side, and such that $\mathsf{L}(\mathcal{A}' \times \mathcal{B}) \geq \frac{1}{4} \cdot \mathsf{L}(\mathcal{A} \times \mathcal{B})$. An analogous statement holds for Bob's side.*

▶ Remark. Although Definition 6.5 and Lemma 6.6 are phrased in terms of Karchmer-Wigderson relations, they work equally well for any communication problem.

We begin our proof of the fortification lemma by proving the following proposition, which is almost what we want.

▶ **Proposition 6.7.** *Let $\mathcal{A}, \mathcal{B} \subseteq \{0,1\}^m$ be disjoint sets. For every $0 < \rho < 1$, there exists $\mathcal{A}_1 \subseteq \mathcal{A}$ such that*

- *for every $\tilde{\mathcal{A}} \subseteq \mathcal{A}_1$ it holds that $\frac{\mathsf{L}(\tilde{\mathcal{A}} \times \mathcal{B})}{\mathsf{L}(\mathcal{A} \times \mathcal{B})} \geq \rho \cdot \frac{|\tilde{\mathcal{A}}|}{|\mathcal{A}|}$.*
- *$\mathsf{L}(\mathcal{A}_1 \times \mathcal{B}) \geq (1-\rho) \cdot \mathsf{L}(\mathcal{A} \times \mathcal{B})$.*

The same holds for $\mathcal{B}_1 \subseteq \mathcal{B}$.

The reason that Proposition 6.7 is not exactly what we want is that in the first item, the denominator on the right hand side is $|\mathcal{A}|$, while it should be $|\mathcal{A}_1|$ in order to satisfy the definition of a fortified rectangle. This is problematic, since in our application we will be able to control the ratio $\frac{|\tilde{\mathcal{A}}|}{|\mathcal{A}_1|}$, but we will have no way to control the ratio $\frac{|\tilde{\mathcal{A}}|}{|\mathcal{A}|}$.

Proof of Proposition 6.7. We prove the proposition for $\mathcal{A}_1 \subseteq \mathcal{A}$, and the proof for $\mathcal{B}_1 \subseteq \mathcal{B}$ is analogous. Let $\mathcal{A}_{\max} \subseteq \mathcal{A}$ be a maximal set that satisfies

$$\mathsf{L}(\mathcal{A}_{\max} \times \mathcal{B}) < \rho \cdot \frac{|\mathcal{A}_{\max}|}{|\mathcal{A}|} \cdot \mathsf{L}(\mathcal{A} \times \mathcal{B}). \tag{8}$$

We choose $\mathcal{A}_1 \stackrel{\text{def}}{=} \mathcal{A} - \mathcal{A}_{\max}$. Observe that it indeed holds that $\mathsf{L}(\mathcal{A}_1 \times \mathcal{B}) \geq (1-\rho) \cdot \mathsf{L}(\mathcal{A} \times \mathcal{B})$ by the sub-additivity of formula complexity. Now, suppose for the sake of contradiction that there is a set $\tilde{\mathcal{A}} \subseteq \mathcal{A}_1$ such that $\mathsf{L}(\tilde{\mathcal{A}}, \mathcal{B}) < \rho \cdot \frac{|\tilde{\mathcal{A}}|}{|\mathcal{A}|} \cdot \mathsf{L}(\mathcal{A}, \mathcal{B})$. Then, this would imply that

$$\begin{aligned}
\mathsf{L}\left((\tilde{\mathcal{A}} \cup \mathcal{A}_{\max}) \times \mathcal{B}\right) &\leq \mathsf{L}(\tilde{\mathcal{A}} \times \mathcal{B}) + \mathsf{L}(\mathcal{A}_{\max} \times \mathcal{B}) \\
&< \rho \cdot \frac{|\tilde{\mathcal{A}}|}{|\mathcal{A}|} \cdot \mathsf{L}(\mathcal{A} \times \mathcal{B}) + \rho \cdot \frac{|\mathcal{A}_{\max}|}{|\mathcal{A}|} \cdot \mathsf{L}(\mathcal{A} \times \mathcal{B}) \\
&= \rho \cdot \frac{|\tilde{\mathcal{A}} \cup \mathcal{A}_{\max}|}{|\mathcal{A}|} \cdot \mathsf{L}(\mathcal{A} \times \mathcal{B}),
\end{aligned}$$

where the first inequality holds by the sub-additivity of protocol size, and the second inequality holds by our assumptions on $\tilde{\mathcal{A}}$ and $\mathcal{A}_{\max}$. It follows that $\tilde{\mathcal{A}} \cup \mathcal{A}_{\max}$ is a set that satisfies Inequality 8 and that strictly contains $\mathcal{A}_{\max}$, thus contradicting the maximality of $\mathcal{A}_{\max}$. Hence, no such set $\tilde{\mathcal{A}}$ exists. ◀

▶ Remark. Consider again the example of a non-fortified rectangle $\mathcal{A} \times \mathcal{B}$ from the beginning of this section. For this rectangle, the above proof would take $\mathcal{A}_{\max}$ to be the set of strings a such that $a_1 = 1$. Thus, the set $\mathcal{A}'$ would be the set of strings a in which $a_1 = 0$, as required.

In order to prove the fortification lemma from Proposition 6.7, we need to replace the ratio $\frac{|\tilde{\mathcal{A}}|}{|\mathcal{A}|}$ with the ratio $\frac{|\tilde{\mathcal{A}}|}{|\mathcal{A}_1|}$. To this end, observe that

$$\frac{|\tilde{\mathcal{A}}|}{|\mathcal{A}_1|} = \frac{|\tilde{\mathcal{A}}|}{|\mathcal{A}|} / \frac{|\mathcal{A}_1|}{|\mathcal{A}|}.$$

Hence, we could achieve our goal by controlling the ratio $|\mathcal{A}_1|/|\mathcal{A}|$. The following proposition provides us the means to do so. Intuitively, this proposition is a form of "inverse fortification" – it allows us to lower bound the density of a subset $\tilde{\mathcal{A}}$ in terms of its complexity.

▶ **Proposition 6.8.** *Let $\mathcal{A}, \mathcal{B} \subseteq \{0,1\}^m$ be disjoint sets. For every $c \geq 1$, there exists a subset $\mathcal{A}_0 \subseteq \mathcal{A}$ such that for every $\tilde{\mathcal{A}} \subseteq \mathcal{A}_0$ it holds that*

$$\frac{|\tilde{\mathcal{A}}|}{|\mathcal{A}_0|} \geq \left(\frac{\mathsf{L}(\tilde{\mathcal{A}} \times \mathcal{B})}{\mathsf{L}(\mathcal{A}_0 \times \mathcal{B})}\right)^c, \tag{9}$$

and such that

$$\mathsf{L}(\mathcal{A}_0 \times \mathcal{B}) \geq 2^{-m/c} \cdot \mathsf{L}(\mathcal{A} \times \mathcal{B}). \tag{10}$$

Proof. We set $\mathcal{A}_0$ to be a minimal set that satisfies

$$\frac{|\mathcal{A}_0|}{|\mathcal{A}|} \leq \left(\frac{\mathsf{L}(\mathcal{A}_0 \times \mathcal{B})}{\mathsf{L}(\mathcal{A} \times \mathcal{B})}\right)^c.$$

Observe that $\mathcal{A}_0$ indeed satisfies Inequality 9: otherwise, there would have been $\tilde{\mathcal{A}} \subset \mathcal{A}_0$ that satisfied

$$\frac{|\tilde{\mathcal{A}}|}{|\mathcal{A}_0|} < \left(\frac{\mathsf{L}(\tilde{\mathcal{A}} \times \mathcal{B})}{\mathsf{L}(\mathcal{A}_0 \times \mathcal{B})}\right)^c,$$

and this would have implied that

$$\begin{aligned}
\frac{|\tilde{\mathcal{A}}|}{|\mathcal{A}|} &= \frac{|\tilde{\mathcal{A}}|}{|\mathcal{A}_0|} \cdot \frac{|\mathcal{A}_0|}{|\mathcal{A}|} \\
&< \left(\frac{\mathsf{L}(\tilde{\mathcal{A}} \times \mathcal{B})}{\mathsf{L}(\mathcal{A}_0 \times \mathcal{B})}\right)^c \cdot \left(\frac{\mathsf{L}(\mathcal{A}_0 \times \mathcal{B})}{\mathsf{L}(\mathcal{A} \times \mathcal{B})}\right)^c \\
&= \left(\frac{\mathsf{L}(\tilde{\mathcal{A}} \times \mathcal{B})}{\mathsf{L}(\mathcal{A} \times \mathcal{B})}\right)^c,
\end{aligned}$$

thus contradicting the minimality of $\mathcal{A}_0$. It remains to show that $\mathcal{A}_0 \times \mathcal{B}$ satisfies Inequality 10. It holds that

$$\frac{|\mathcal{A}_0|}{|\mathcal{A}|} \leq \left(\frac{\mathsf{L}(\mathcal{A}_0 \times \mathcal{B})}{\mathsf{L}(\mathcal{A} \times \mathcal{B})}\right)^c,$$

or in other words

$$\begin{aligned}
\mathsf{L}(\mathcal{A}_0 \times \mathcal{B}) &\geq \left(\frac{|\mathcal{A}_0|}{|\mathcal{A}|}\right)^{\frac{1}{c}} \cdot \mathsf{L}(\mathcal{A} \times \mathcal{B}) \\
&\geq \left(\frac{1}{2^m}\right)^{\frac{1}{c}} \cdot \mathsf{L}(\mathcal{A} \times \mathcal{B}) \\
&= 2^{-m/c} \cdot \mathsf{L}(\mathcal{A} \times \mathcal{B}),
\end{aligned}$$

as required. ◀

We are now ready to prove the fortification lemma.

Proof of Lemma 6.6. Let $\mathcal{A}, \mathcal{B} \subseteq \{0,1\}^m$ be disjoint sets. Our goal is to find a subset $\mathcal{A}' \subseteq \mathcal{A}$ such that $\mathcal{A}' \times \mathcal{B}$ is $\frac{1}{4m}$-fortified on Alice's side, and such that $\mathsf{L}(\mathcal{A}' \times \mathcal{B}) \geq \frac{1}{3} \cdot \mathsf{L}(\mathcal{A} \times \mathcal{B})$ (the proof for Bob's side is analogous). We start by applying Proposition 6.8 to $\mathcal{A} \times \mathcal{B}$ with $c = m$, thus obtaining a subset $\mathcal{A}_0 \subseteq \mathcal{A}$. Then, we apply Proposition 6.7 to $\mathcal{A}_0 \times \mathcal{B}$ with $\rho = \frac{1}{2m}$, thus obtaining a subset $\mathcal{A}_1 \subseteq \mathcal{A}_0$. Finally, we choose $\mathcal{A}'$ to be $\mathcal{A}_1$.

We prove that $\mathcal{A}'$ has the required properties. Observe that by Proposition 6.7, it holds that

$$\mathsf{L}(\mathcal{A}' \times \mathcal{B}) \geq (1 - \frac{1}{2m}) \cdot \mathsf{L}(\mathcal{A}_0 \times \mathcal{B}) \geq \frac{1}{2} \cdot \mathsf{L}(\mathcal{A}_0 \times \mathcal{B}),$$

and that by Proposition 6.8, it holds that

$$\mathsf{L}(\mathcal{A}_0 \times \mathcal{B}) \geq \frac{1}{2} \cdot \mathsf{L}(\mathcal{A} \times \mathcal{B}).$$

Therefore,

$$\mathsf{L}(\mathcal{A}' \times \mathcal{B}) \geq \frac{1}{4} \cdot \mathsf{L}(\mathcal{A} \times \mathcal{B}),$$

as required.

It remains to prove that $\mathcal{A}'$ is $\frac{1}{4m}$-fortified on Alice's side. Let $\tilde{\mathcal{A}} \subseteq \mathcal{A}'$. By Proposition 6.7, it holds that

$$\mathsf{L}(\tilde{\mathcal{A}} \times \mathcal{B}) \geq \frac{1}{2m} \cdot \frac{|\tilde{\mathcal{A}}|}{|\mathcal{A}_0|} \cdot \mathsf{L}(\mathcal{A}_0 \times \mathcal{B}) \geq \frac{1}{2m} \cdot \frac{|\mathcal{A}'|}{|\mathcal{A}_0|} \cdot \frac{|\tilde{\mathcal{A}}|}{|\mathcal{A}'|} \cdot \mathsf{L}(\mathcal{A}' \times \mathcal{B}).$$

Next, by Proposition 6.8, it holds that

$$\frac{|\mathcal{A}'|}{|\mathcal{A}_0|} \geq \left(\frac{\mathsf{L}(\mathcal{A}' \times \mathcal{B})}{\mathsf{L}(\mathcal{A}_0 \times \mathcal{B})}\right)^m \geq \left(1 - \frac{1}{2m}\right)^m \geq \frac{1}{2}.$$

Thus,

$$\mathsf{L}(\tilde{\mathcal{A}} \times \mathcal{B}) \geq \frac{1}{4m} \cdot \frac{|\tilde{\mathcal{A}}|}{|\mathcal{A}'|} \cdot \mathsf{L}(\mathcal{A}' \times \mathcal{B}).$$

The required result follows. ◀

7 Restating Lemma Proof of the Main Lemma

In this section, we prove the main lemma, restated next.

▶ **Lemma 3.6** (restated). *Let Π be a protocol for $KW_{f \diamond \oplus_n}$, and let π_1 be a live partial transcript of Π. Then, there exists a $\tilde{O}(\sqrt{m})$-almost hard distribution that is distributed over $\mathcal{X}_{\pi_1} \times \mathcal{Y}_{\pi_1}$.*

Fix a protocol Π for $KW_{f \diamond \oplus_n}$, and let π_1 be a live partial transcript of Π. Let $\mathcal{X}_{\pi_1} \times \mathcal{Y}_{\pi_1}$ be the rectangle associated with π_1. We would like to construct a t-almost hard distribution over $\mathcal{X}_{\pi_1} \times \mathcal{Y}_{\pi_1}$, where $t \stackrel{\text{def}}{=} \tilde{O}(\sqrt{m})$ and where the constant in the Big-O notation will be chosen to be sufficiently large as to make our argument hold.

7.1 Basic idea

By the definition of π_1 being alive, there is a set $\mathcal{Z}_{\pi_1} \subset \mathcal{Z}$, with $|\mathcal{Z}_{\pi_1}| \geq 2^{-2m} \cdot |\mathcal{Z}|$, such that for each $Z \in \mathcal{Z}_{\pi_1}$, it holds that $\mathsf{L}(\mathcal{A}_{\pi_1,Z} \times \mathcal{B}_{\pi_1,Z}) \geq 2^\ell$ where $\ell = C \cdot \sqrt{m} \cdot \log^C m$ for some large constant C to be determined later, and where

$$\begin{aligned} \mathcal{A}_{\pi_1,Z} &\stackrel{\text{def}}{=} \left\{a \in f^{-1}(0) | Z^a \in \mathcal{X}_{\pi_1}\right\} \\ B_{\pi_1,Z} &\stackrel{\text{def}}{=} \left\{b \in f^{-1}(1) | Z^b \in \mathcal{Y}_{\pi_1}\right\}. \end{aligned}$$

Consider the following graph G: the graph G is a layered graph, and the three layers are $\mathcal{X}_{\pi_1}$, $\mathcal{Z}_{\pi_1}$, and $\mathcal{Y}_{\pi_1}$. A vertex $X \in \mathcal{X}_{\pi_1}$ (respectively, $Y \in \mathcal{Y}_{\pi_1}$) is a neighbor of a vertex $Z = (Z^0, Z^1) \in \mathcal{Z}_{\pi_1}$ if and only if $X = Z^a$ for some $a \in f^{-1}(0)$ (respectively, $Y = Z^b$ for some $b \in f^{-1}(1)$). We define *the distribution* (X, Y) *of* G as the distribution that is sampled by picking a uniformly distributed path $X - Z - Y$ in G. While this distribution is not an almost-hard distribution, we will show that there is a subgraph G' of G such that the distribution *of* G' is an almost-hard distribution.

Let us examine the properties of the distribution (X, Y) of G more closely. Let Z denote the vertex that is sampled by this distribution, and let $j_1, \ldots, j_m \in [n]$ be the axes of Z (i.e., j_i is the unique coordinate on which Z_i^0 and Z_i^1 differ). Then, for every $i \in [m]$, it always holds that either $X_i = Y_i$, or that X_i and Y_i disagree exactly on one coordinate, which is j_i. Hence, in order for (X, Y) to be a t-almost hard distribution, it only needs to satisfy the property that for every $i \in [m]$, either that $X_i = Y_i$ with probability 1, or that for all specific choice $X^*$and Y^* of X and Y respectively, it holds that

$$\begin{aligned} H_\infty(j_i | X = X^*) &\geq \log n - t \\ H_\infty(j_i | Y = Y^*) &\geq \log n - t. \end{aligned}$$

This property would have been satisfied if $\mathcal{Z}_{\pi_1} = \mathcal{Z}$. In this case, $j_1, \ldots, j_m$ would have been uniformly distributed over $[n]$, and therefore all of them would have had min-entropy $\log n$ (conditioned on either X or Y). However, $\mathcal{Z}_{\pi_1}$ only constitutes 2^{-2m} fraction of $\mathcal{Z}$, and therefore the min-entropy of some j_i's may be as low as $\log n - 2m$. In order to resolve this issue, we apply the averaging argument for min-entropy (Lemma 6.2), and conclude the min-entropy of all but $\sqrt{m}$ of the j_i's is at least about $\log n - O(\sqrt{m})$. We refer to the $\sqrt{m}$ rows in which the min-entropy of j_i is lower than $\log n - O(\sqrt{m})$ as the *revealed rows*, and to the other rows as the *non-revealed rows*.

The non-revealed rows already satisfy what we need, so it remains to deal with the revealed rows. Let $\mathcal{R} \subseteq [m]$ denote the set of revealed rows. We will make sure that $X_i = Y_i$ for every $i \in \mathcal{R}$. To this end, recall that $X = Z^a$ and $Y = Z^b$ for some $a \in f^{-1}(0)$ and $b \in f^{-1}(1)$. We will construct the graph G' such that a and b always agree on the coordinates in $\mathcal{R}$.

To see why this is possible, recall that conditioned on the choice of Z, the strings a and b are taken from the rectangle $\mathcal{A}_{\pi_1,Z} \times \mathcal{B}_{\pi_1,Z}$. Since π_1 is alive with respect to Z, this means that $\mathsf{L}(\mathcal{A}_{\pi_1,Z} \times \mathcal{B}_{\pi_1,Z}) \geq 2^\ell$. We claim that this means that a and b can be chosen such that $a|_{\mathcal{R}} = b|_{\mathcal{R}}$. If this was not the case, i.e., if it was the case that $a|_{\mathcal{R}} \neq b|_{\mathcal{R}}$ for all $a \in \mathcal{A}_{\pi_1,Z}$ and $b \in \mathcal{B}_{\pi_1,Z}$, then the complexity of $\mathcal{A}_{\pi_1,Z} \times \mathcal{B}_{\pi_1,Z}$ would have been lower: Alice and Bob could have solved the game by sending each other their values at $\mathcal{R}$, and this protocol is of size at most $2^{O(|\mathcal{R}|)} < 2^\ell$ (for an appropriate choice of ℓ). There are two more complications:

- It is not sufficient to show that there exists at least one choice of a and b such that $a|_{\mathcal{R}} = b|_{\mathcal{R}}$. Rather, we need to show that there are many such choices – otherwise, forcing a and b to agree on $\mathcal{R}$ would reveal too much information to Alice and Bob.

  To this end, we process the graph as follows: we partition the strings $a \in \mathcal{A}_{\pi_1,Z}$ according

to $a|_{\mathcal{R}}$, and remove the classes that are too small. We do the same for $\mathcal{B}_{\pi_1,Z}$. By choosing the parameters appropriately, we can make sure that at most half of the strings in $\mathcal{A}_{\pi_1,Z}$ and $\mathcal{B}_{\pi_1,Z}$ are removed in the latter process. We then use the fortification lemma (Lemma 6.6) to show that the latter removal of strings did not decrease the complexity of $\mathcal{A}_{\pi_1,Z} \times \mathcal{B}_{\pi_1,Z}$ by too much, and hence this complexity is still large. We now argue as before that since $\mathcal{A}_{\pi_1,Z} \times \mathcal{B}_{\pi_1,Z}$ has large complexity, we can choose a class of $\mathcal{A}_{\pi_1,Z}$ and a class of $\mathcal{B}_{\pi_1,Z}$ that agree on $\mathcal{R}$.

Finally, we observe that the classes we chose must be large, since all the small classes were already removed. Hence, there are indeed many choices of a and b that agree on $\mathcal{R}$.

- The above discussion assumed implicitly that conditioned on X, the vertex Z is distributed uniformly over neighbors of X, and similarly for Y. This is may not always hold, but it does hold if all the vertices Z have the same degree. Throughout the proof, we take steps to ensure that the vertices Z have roughly equal degrees, and this will be good enough for our purposes.

7.2 A technical road-map

In the rest of this section, we describe the proof in detail. The proof follows the basic idea described above, but along the way there are some technical issues that need to be resolved and careful accounting that needs to be done. Therefore, we start by giving a "technical road-map" of the proof, which explains the main steps, the issues that we deal with, and the considerations that underly the accounting.

Notation

In what follows, we will consider subgraphs of the graph G described above. Given such a subgraph G_0 and a vertex Z, we define the *rectangle of Z in G_0* by $\mathcal{A}_{0,Z} \times B_{0,Z}$ where

$$\begin{aligned} \mathcal{A}_{0,Z} &\stackrel{\text{def}}{=} \left\{a \in f^{-1}(0) | Z^a \text{ is a neighbor of } Z \text{ in } G_0\right\} \\ B_{0,Z} &\stackrel{\text{def}}{=} \left\{b \in f^{-1}(1) | Z^b \text{ is a neighbor of } Z \text{ in } G_0\right\}. \end{aligned}$$

In general, we will identify the edges that come out of Z with the elements of $A_{0,Z}$ and $B_{0,Z}$. For example, we may say that we remove a string from $A_{0,Z}$ and mean that we remove the corresponding edge. We define the *complexity of Z in G_0* to be the protocol size of its rectangle, i.e., $\mathsf{L}(\mathcal{A}_{0,Z} \times B_{0,Z})$. Observe that in G, all the Z's have complexity at least 2^ℓ by the assumption that π_1 is alive.

Throughout the proof, we refer to the edges between $\mathcal{X}_{\pi_1}$ and $\mathcal{Z}$ as *the $\mathcal{X}$-side* of the graph or as *Alice's side* of the graph. Similarly, we refer to the edges between $\mathcal{Y}_{\pi_1}$ and $\mathcal{Z}$ as *the $\mathcal{Y}$-side* or as *Bob's side*.

7.2.1 The main steps

The proof consists of five main parts:

1. We process Alice's side, which means we remove vertices and edges in order to obtain some desired properties. Along the way, we construct the set $\mathcal{R}_{\mathcal{X}}$ of *revealed rows for the $\mathcal{X}$-side*, i.e., the set of indices $i \in [m]$ such that $H_\infty(j_i|X)$ is too small.
2. We process Bob's side in a similar manner, thus obtaining the set $\mathcal{R}_{\mathcal{Y}}$ of revealed rows for the $\mathcal{Y}$-side.
3. We force the X's and Y's in the graph to agree on the set of revealed rows $\mathcal{R} \stackrel{\text{def}}{=} \mathcal{R}_{\mathcal{X}} \cup \mathcal{R}_{\mathcal{Y}}$.

4. We perform a clean-up step that removes all the vertices whose degree became too small, and denote the resulting graph by G'.
5. We conclude by proving that the distribution of G' is an almost-hard distribution, as required.

The most technical part is the processing of Alice's side. It consists of four steps:

- **Fortification:** We fortify the the rectangle of each Z. This is done to make sure that the following steps, which remove edges on Alice's side, do not reduce the complexity of the Z's by too much. This results in a subgraph of G that we denote by G_{A1} (here, "$A1$" denotes "first step on Alice's side").
- **Regularization:** As discussed above, throughout the proof we will need to guarantee that the Z's are *roughly regular*, i.e., that all the Z's have roughly the same degree. We create this property in this step, by taking a subset of the Z's that have roughly the same degree on the $\mathcal{X}$-side in G_{A1}, and discarding all the rest. We denote the resulting subgraph by G_{A2}.
- **Finding the revealed rows:** For each X in G_{A2}, we consider the distribution on axes $j_1, \ldots, j_m$ that is induced by choosing a random neighbor Z of X. We observe that this distribution has min-entropy which is at least $m \cdot \log n - O(m)$, and apply the averaging argument for min-entropy to this distribution (Lemma 6.2). This yields a set of revealed axes $\mathcal{R}_X$ of size $\sqrt{m}$, such that the min-entropy of each j_i for $i \in [m] - \mathcal{R}_X$ is at least $\log n - \tilde{O}(\sqrt{m})$.

  Note that the averaging argument only says that the min-entropy of j_i is large conditioned on some event E_X. We therefore remove from the graph all the edges that are not consistent with E_X, for each X. In addition, note that the set $\mathcal{R}_X$ may be different for each X. We now choose the most popular set $\mathcal{R}_X$, denote it by $\mathcal{R}_{\mathcal{X}}$, and discard all the X's with a different set. We denote the resulting subgraph by G_{A3}.
- **Removing the small classes:** For each Z, consider its rectangle $\mathcal{A}_{A3,Z} \times \mathcal{B}_{A3,Z}$ in G_{A3}. We would like to partition the strings $a \in \mathcal{A}_{A3}$ into classes according to $a|_{\mathcal{R}}$, and remove the small classes – as discussed above, this is done in order to make sure that when we force the a's and the b's to agree on $\mathcal{R}$, we will retain many a's.

  However, there is a small issue here that needs to be dealt with: at this point we do not know yet the set $\mathcal{R}$ of revealed rows – we know the set $\mathcal{R}_{\mathcal{X}}$ of revealed rows for the $\mathcal{X}$-side, but we do not know yet the set $\mathcal{R}_{\mathcal{Y}}$ of revealed rows for the $\mathcal{Y}$-side. Therefore, we perform this step of "removing the small classes" for every possible candidate for $\mathcal{R}$. By choosing the parameters appropriately, we can ensure that doing so does not remove too many a's. We denote the resulting subgraph by G_{A4}.

The processing of Bob's side is similar to that of Alice's side, except that the step of removing the small classes is a little simpler since at this point we know $\mathcal{R}$. This processing creates corresponding subgraphs $G_{B1}, G_{B2}, G_{B3}, G_{B4}$.

Next, we force the X's and Y's to agree on the revealed rows as follows: For every Z in G_{B4}, we consider the rectangle $\mathcal{A}_{B4,Z} \times \mathcal{B}_{B4,Z}$. We claim that there must be $a \in \mathcal{A}_{B4,Z}$ and $b \in \mathcal{B}_{B4,Z}$ such that $a|_{\mathcal{R}} = b|_{\mathcal{R}}$, or otherwise the complexity of $\mathcal{A}_{B4,Z} \times \mathcal{B}_{B4,Z}$ would have been too small. We then claim that a and b must belong to large classes of $\mathcal{A}_{B4,Z}$ and $\mathcal{B}_{B4,Z}$ respectively, since the small classes have already been removed, and therefore there are *many* a's and b's such that $a|_{\mathcal{R}} = b|_{\mathcal{R}}$. We now discard all the other a's and b's for every Z, thus creating a new subgraph G_{agr}.

The final step is the clean-up step. The reason that this step is needed is that each of the previous steps removed some edges. This is problematic for two reasons: First, the degree of some X's may have become too small, in which case the min-entropy $H_\infty(j_i|X)$ may also

become too small, and the same goes for the Y's. Second, the degree of some Z's may have become too small, thus violating the rough regularity of the Z's. In order to rectify those violations, we remove the vertices whose degrees are too small. However, this removal may decrease the degrees of other vertices, so we continue removing vertices until there are no more vertices whose degrees are too small. By choosing the parameters appropriately, we can make sure that the process terminates before the whole graph is deleted.

7.2.2 Issues and accounting

Retaining a large number of edges

Recall that at the end of the step of "finding the revealed rows" on Alice's side, we have for each X the property that for every $i \in [m] - \mathcal{R}_{\mathcal{X}}$, it holds that

$$H_\infty(j_i|X) \geq \log n - \tilde{O}(\sqrt{m}).$$

However, in the following steps, we remove vertices and edges from the graph, and this may destroy this property. More specifically, after we remove edges from the graph, this property will continue to hold for every X whose degree was reduced by a factor of at most $2^{\tilde{O}(\sqrt{m})}$, but may cease to hold for X's whose degree was reduced by more than that.

As explained above, we deal with this issue in the clean-up step by removing all the X's whose degree is too small, i.e., whose degree was reduced by a factor of more than $2^{\tilde{O}(\sqrt{m})}$. However, in order for this solution to be effective, we need to make sure that *the degree of most X's is not too small* (or otherwise the clean-up may remove too many X's).

To this end, it suffices to show that the number of edges of G_{agr} on the $\mathcal{X}$-side is at least $2^{-\tilde{O}(\sqrt{m})}$ times the number of edges of G_{A3} on the $\mathcal{X}$-side. In order to do so, we keep track of the number of edges on the $\mathcal{X}$-side throughout the proof and make sure that it does not decrease too much. The same goes for the $\mathcal{Y}$-side.

Retaining a large number of Z's

When we perform the step of "finding the revealed rows" on Alice's side, we use the fact that in G_{A2}, the distribution $j_1, \ldots, j_m$ has min-entropy at least

$$m \cdot \log(n) - O(m).$$

In order to show this lower bound on the min-entropy, we use the fact that the number of Z's in G_{A2} is at least $2^{-O(m)}$ fraction of all the possible Z's. The latter fact follows from the assumption that π_1 is alive, but we also need to make sure that it is not invalidated by the regularization step. Therefore, when performing the regularization, we make sure that we did not remove too many Z's.

Furthermore, since we also perform the step of "finding the revealed rows" on Bob's side, we also need to make sure that the number of Z's in the graph G_{B2} is sufficiently large. To this end, we keep track of the number of Z's throughout the processing on Alice's side and make sure that we do not remove too many Z's.

Interaction between the two sides of the graph

When we process Bob's side, we remove some of the Z's in the regularization step and in the step of "removing the small classes". However, when we remove Z's, it also causes the removal of edges on the $\mathcal{X}$-side. Hence, we have to make sure that those steps do not remove too many edges on the $\mathcal{X}$-side.

To this end, we first make sure that those steps do not remove too many Z's: in particular, we make sure that after each step, we retain at least $2^{-\tilde{O}(\sqrt{m})}$ fraction of the Z's. Then, we use the fact that the Z's are roughly regular on the $\mathcal{X}$-side to deduce that we retained at least $2^{-\tilde{O}(\sqrt{m})}$ fraction of the edges on the $\mathcal{X}$-side.

Average degree vs. minimum degree

In many places throughout the proof, we will have a lower bound on the *average degree* of vertices, but we will want this lower bound to hold for the *minimum degree*, i.e., we will want it to hold for every vertex. For example, at the beginning of the step of "finding the revealed rows" on Alice's side, we know that *the average* X is connected to at least $2^{-O(m)}$ fraction of all the possible Z's, but we will want it to hold *for every* X. Whenever we encounter such a situation, we resolve the issue by removing from the graph all the vertices whose degree is too small compared to the average degree. We will use the following fact to show that this removal does not discard too many edges.

▶ **Fact 7.1.** *Let $G_0 = (\mathcal{U}_0 \cup \mathcal{V}_0, \mathcal{E}_0)$ be a bipartite graph, and denote the average degree of $\mathcal{U}_0$ by $d_{\mathcal{U}}$. If we remove all the vertices of $\mathcal{U}_0$ whose degree is less than $\varepsilon \cdot d_{\mathcal{U}}$, then we remove at most ε fraction of the total number of edges.*

Proof. By the definition of average degree, it holds that $|\mathcal{E}_0| = d_{\mathcal{U}} \cdot |\mathcal{U}_0|$. The number of vertices that we remove is at most $|\mathcal{U}_0|$, and each of them is connected to at most $\varepsilon \cdot d_{\mathcal{U}}$ edges. Hence, the total number of edges we removed is at most $\varepsilon \cdot d_{\mathcal{U}} \cdot |\mathcal{U}_0| = \varepsilon \cdot |\mathcal{E}_0|$, as required. ◀

We finally turn to present the full proof.

7.3 Processing Alice's side

Fortification

The first step we take in processing the graph on Alice's side is fortifying the Z's on Alice's side. For each Z, we apply the fortification lemma (Lemma 6.6) to the rectangle of Z in G, namely $\mathcal{A}_{\pi_1,Z} \times \mathcal{B}_{\pi_1,Z}$, thus obtaining a sub-rectangle $\mathcal{A}_{A1,Z} \times \mathcal{B}_{A1,Z}$ that is $\frac{1}{4m}$-fortified on Alice's side (where $\mathcal{B}_{A1,Z} = \mathcal{B}_{\pi_1,Z}$). We then replace $A_{\pi_1,Z} \times B_{\pi_1,Z}$ with $\mathcal{A}_{A1,Z} \times \mathcal{B}_{A1,Z}$ by removing from G all the edges that correspond to strings in $\mathcal{A}_{\pi_1,Z} - \mathcal{A}_{A1,Z}$. We denote the resulting graph by G_{A1}.

Regularization

Next, we make sure that all the vertices Z have roughly the same degree on the $\mathcal{X}$-side (i.e., have the same number of neighbors X). To this end, we partition the Z's to $m+1$ classes, such that the Z's in the i-th class has degree at least 2^{i-1} and less than 2^i (for $1 \leq i \leq m+1$). Let i be such that the i-th class is the class that contains a largest number of Z's. We remove from G_{A1} all the Z's outside the i-th class, and denote the resulting graph by G_{A2} and the resulting set of Z's by $\mathcal{Z}_{A2}$.

Let $d_{\mathcal{Z},\mathcal{X}} \stackrel{\text{def}}{=} 2^i$. By definition, all the vertices Z in $\mathcal{Z}_{A2}$ have degrees between $\frac{1}{2} \cdot d_{\mathcal{Z},\mathcal{X}}$ and $d_{\mathcal{Z},\mathcal{X}}$. Moreover, observe that G_{A2} retains at least $\frac{1}{m+1}$ fraction of the Z's. Since G originally had at least $2^{-2m} \cdot |\mathcal{Z}|$ vertices Z (and so did G_{A1}), it follows that G_{A2} has at least $2^{-2m-\log(m+1)} \cdot |\mathcal{Z}|$ vertices Z.

Finding the revealed rows

We turn to applying the averaging argument to the X's in order to find the revealed rows. However, we can only do so for X's with sufficiently large degree. To compute the average degree of the X's we observe that each Z must be connected to at least one vertex X, and therefore the average degree of the X's is at least

$$\frac{|\mathcal{Z}_{A2}|}{|\mathcal{X}_{\pi_1}|} \geq \frac{2^{-2m-\log(m+1)} \cdot |\mathcal{Z}|}{2^{m \cdot n}} = \frac{2^{-2m-\log(m+1)} \cdot \left(2^{m \cdot (n-1)} \cdot n^m\right)}{2^{m \cdot n}} = 2^{-3m-\log(m+1)} \cdot n^m.$$

We remove from the graph all the X's with degree less than $2^{-4m} \cdot n^m$. By Fact 7.1, we removed less than half of the edges of the graph on the $\mathcal{X}$-side.

Now, for each of the remaining $X's$, we perform the following steps. Let Z be a uniformly distributed neighbor of X, and let $j_1, \ldots, j_m$ be the axes of Z. Observe that given X, there is a one-to-one correspondence between Z and the sequence $j_1, \ldots, j_m$. Thus, the fact that the degree of X is at least $2^{-4m} \cdot n^m$ implies that

$$H_\infty(j_1, \ldots, j_m) \geq m \cdot \log n - 4 \cdot m.$$

We apply the averaging argument for min-entropy (Lemma 6.2) to $j_1, \ldots, j_m$ with parameters $r = 4m$ and $k = \sqrt{m}$, thus obtaining a set $\mathcal{R}_X$ of size $\sqrt{m}$ and an event $E_X \subseteq [n]^m$ of probability at least $2^{-O(\sqrt{m} \log m)} = 2^{-\tilde{O}(\sqrt{m})}$ such that for every $i \in [m] - \mathcal{R}_X$ it holds that

$$H_\infty(j_i | E_X) \geq \log n - O(\sqrt{m}).$$

Observe that the event E_X is a set of tuples $(j_1, \ldots, j_m)$, each of which corresponds to an edge going out of X. We remove all the edges of X that do not belong to E_X. Note that we retain at least $2^{-\tilde{O}(\sqrt{m})}$ fraction of the edges since the probability of E_X is at least $2^{-\tilde{O}(\sqrt{m})}$.

Next, we partition the X's according to their set $\mathcal{R}_X$, pick the class that is connected to the largest number of edges, and remove all the X's outside of this class. Let $\mathcal{R}_\mathcal{X}$ be the set R_X of the class that was picked, and denote by G_{A3} the resulting graph. There are $\binom{m}{\sqrt{m}} = 2^{\tilde{O}(\sqrt{m})}$ classes so it is easy to see that after the removal we retain at least $2^{-\tilde{O}(\sqrt{m})}$ fraction of the edges, and therefore G_{A3} retains at least $2^{-\tilde{O}(\sqrt{m})}$ fraction of the edges of G_{A2}.

Summing up the discussion so far, the graph G_{A3} has the following property: Let X be a vertex in G_{A3}, let Z be a uniformly distributed neighbor of X in G_{A3}, and let $j_1, \ldots, j_m$ be the axes of the edges in Z. Then, for every $i \in [m] - \mathcal{R}_\mathcal{X}$ it holds that

$$H_\infty(j_i) \geq \log n - O(\sqrt{m}). \tag{11}$$

Removing small classes from the rectangles of the Z's

The last step we perform is a preparation toward forcing the a's and the b's of each Z to agree on the revealed rows – see the discussion in Section 7.1 about the first complication. As explained there, for each Z, we would like to partition its set of a's according to their values at the revealed rows $\mathcal{R}$, and remove the classes of the partition that are too small.

However, we do not know yet what is the set $\mathcal{R}$ of revealed rows. Indeed, we know the set $\mathcal{R}_\mathcal{X}$ of the revealed rows on Alice's side, but we do not know yet the revealed rows on Bob's side. In order to resolve this issue, we define classes of edges *for all the possible candidates for* $\mathcal{R}$, and remove the small classes. Note that now the classes no longer form a partition of the a's of Z, but it does not matter for our argument.

Formally, we define a *label* to be a pair $(\mathcal{R}, \lambda)$ where $\mathcal{R} \subseteq [m]$ is a set of size $2\sqrt{m}$ that contains $\mathcal{R}_{\mathcal{X}}$, and $\lambda \in \{0,1\}^{\mathcal{R}}$ is an assignment of bits to $\mathcal{R}$. There are only $2^{\tilde{O}(\sqrt{m})}$ possible labels. We say that a string $a \in \{0,1\}^m$ is *consistent with the label* $(\mathcal{R}, \lambda)$ if $a|_{\mathcal{R}} = \lambda$.

Next, we perform the following for each vertex Z in G: Let $\mathcal{A}_{A3,Z} \times \mathcal{B}_{A3,Z}$ be the rectangle of Z in G_3. For every possible label $(\mathcal{R}, r)$, define *the class of* $(\mathcal{R}, \lambda)$ to be the subset of all strings $a \in \mathcal{A}_{A3,Z}$ that are consistent with $(\mathcal{R}, \lambda)$. We say that a class is *small* if it contains less than $2^{-3 \cdot \sqrt{m} \cdot \log m}$ fraction of the strings in $\mathcal{A}_{A3,Z}$. We now remove from $\mathcal{A}_{A3,Z}$ every string a that belongs to some small class. If new small classes are created by the latter removal, we remove them as well, and repeat this process until no small classes remain. By the union bound, it is not hard to see that this removes at most half of the strings in $\mathcal{A}_{A3,Z}$. Denote the resulting set by $\mathcal{A}_{A4,Z}$, and let $\mathcal{B}_{A4,Z} \stackrel{\text{def}}{=} \mathcal{B}_{A3,Z}$.

Finally, observe that the average degree of the Z's on the $\mathcal{X}$-side is at least $2^{-\tilde{O}(\sqrt{m})} \cdot d_{\mathcal{Z},\mathcal{X}}$: After the regularization, the average degree was at least $\frac{1}{2} \cdot d_{\mathcal{Z},\mathcal{X}}$, and after finding the revealed rows and removing the small classes we retained at least $2^{-\tilde{O}(\sqrt{m})}$ fraction of the edges. We now remove all the $Z's$ whose degree is less than half the average degree in order to maintain the property that all the Z's have roughly the same degree – in particular, after the removal, all Z's will have degree between $2^{-\tilde{O}(\sqrt{m})} \cdot d_{\mathcal{Z},\mathcal{X}}$ and $d_{\mathcal{Z},\mathcal{X}}$. We denote the resulting set of Z's by $\mathcal{Z}_{A4}$, and the resulting graph by G_{A4}.

Observe that G_{A4} retains quarter of the edges of G_{A3} on the $\mathcal{X}$-side: The removal of the small classes removed at most half of the edges of each Z, and hence at most half of the edges of G_{A3}. Then, the removal of low-degree Z's removed at most half of the remaining edges by Fact 7.1. Since G_{A3} retained $2^{-\tilde{O}(\sqrt{m})}$ fraction of the edges of G_{A2}, it follows that G_{A4} retains $2^{-\tilde{O}(\sqrt{m})}$ fraction of the edges of G_{A2}.

Furthermore, we claim that the set $\mathcal{Z}_{A4}$ of Z's in G_{A4} contains at least $2^{-\tilde{O}(\sqrt{m})}$ fraction of the Z's in $\mathcal{Z}_{A2}$. To see why this is the case, recall that the number of edges on the $\mathcal{X}$-side in G_{A2} is at least $\frac{1}{2} \cdot d_{\mathcal{Z},\mathcal{X}} \cdot |\mathcal{Z}_{A,2}|$ (since the minimal degree of a Z in G_{A2} is $\frac{1}{2} \cdot d_{\mathcal{Z},\mathcal{X}}$). On the other hand, the number of edges on the $\mathcal{X}$-side in G_{A4} is at most $d_{\mathcal{Z},\mathcal{X}} \cdot |\mathcal{Z}_{A4}|$, and we know that this number is at least $2^{-\tilde{O}(\sqrt{m})}$ fraction of the number of edges in G_{A2}. Therefore,

$$\begin{aligned}
d_{\mathcal{Z},\mathcal{X}} \cdot |\mathcal{Z}_{A4}| &\geq 2^{-\tilde{O}(\sqrt{m})} \cdot \frac{1}{2} \cdot d_{\mathcal{Z},\mathcal{X}} \cdot |\mathcal{Z}_{A,2}| \\
|\mathcal{Z}_{A4}| &\geq 2^{-\tilde{O}(\sqrt{m})} \cdot |\mathcal{Z}_{A,2}| \\
&\geq 2^{-\tilde{O}(\sqrt{m})} \cdot |\mathcal{Z}_{A,1}| \\
&\geq 2^{-O(m)} \cdot |\mathcal{Z}| \,.
\end{aligned}$$

Moreover, observe that the complexity of every $Z \in \mathcal{Z}_{A4}$ is at least $2^{-\tilde{O}(\sqrt{m})}$ fraction of its original complexity in G: First, recall that the complexity of the fortified rectangles $\mathcal{A}_{A1,Z} \times \mathcal{B}_{A1,Z}$ was $\frac{1}{3}$ fraction of the original complexity. By the fortification, the complexity of each Z in G_4 is

$$\begin{aligned}
\mathsf{L}(\mathcal{A}_{A4,Z} \times \mathcal{B}_{A4,Z}) &\geq \frac{1}{4m} \cdot \frac{|\mathcal{A}_{A4,Z}|}{|\mathcal{A}_{A1,Z}|} \cdot \mathsf{L}(\mathcal{A}_{A1,Z} \times \mathcal{B}_{A1,Z}) \\
&\geq 2^{-\tilde{O}(\sqrt{m})} \cdot \mathsf{L}(\mathcal{A}_{A1,Z} \times \mathcal{B}_{A1,Z}) \\
&\geq 2^{\ell - \tilde{O}(\sqrt{m})}.
\end{aligned}$$

7.4 Processing Bob's side

We now take the same steps as in Section 7.3 in the $\mathcal{Y}$-side of the graph: We apply the fortification on Bob's side to the vertices Z in G_{A4}, thus obtaining a new graph G_{B1}. We

then apply regularization, thus obtaining a new graph G_{B2} such that the degrees of the Z's on the $\mathcal{Y}$-side are are between $\frac{1}{2} \cdot d_{\mathcal{Z},\mathcal{Y}}$ and $d_{\mathcal{Z},\mathcal{Y}}$ for some degree $d_{\mathcal{Z},\mathcal{Y}}$. Next, we find the revealed rows for the Y's, thus obtaining a new graph G_{B3} and a set $\mathcal{R}_{\mathcal{Y}}$ such that the following holds for every Y: Let Z be a uniformly distributed neighbor of Y, and let $j_1, \ldots, j_m$ be the axes of Z. Then, for every $i \in [m] - \mathcal{R}_{\mathcal{Y}}$ it holds that

$$H_\infty(j_i) \geq \log n - \tilde{O}(\sqrt{m}). \tag{12}$$

Let $\mathcal{R} \stackrel{\text{def}}{=} \mathcal{R}_{\mathcal{X}} \cup \mathcal{R}_{\mathcal{Y}}$.

There is a small difference in the step of "removing the small classes": Now, we know the set of revealed rows $\mathcal{R}$, so we do not need the labels to contain a candidate for $\mathcal{R}$. Instead, for each Z, we simply partition the strings $b \in \mathcal{B}_{B3,Z}$ according to $b|_{\mathcal{R}}$, and remove all the classes that contain only $2^{-3\cdot\sqrt{m}}$ fraction of the strings in $\mathcal{B}_{B3,Z}$. The rest of this step proceeds as before, and we denote the resulting graph by G_{B4}.

Again, we note that the following points:

- The graph G_{B4} retains at least $2^{-\tilde{O}(\sqrt{m})}$ fraction of the vertices Z of G_{A4}.
- The degree of every Z in G_{B4} on the $\mathcal{Y}$-side is at least $2^{-\tilde{O}(\sqrt{m})} \cdot d_{\mathcal{Z},\mathcal{Y}}$.
- The complexity of each Z in G_{B4} is at least $2^{\ell - \tilde{O}(\sqrt{m})}$.
- The graph G_{B4} retains at least $2^{-\tilde{O}(\sqrt{m})}$ fraction of the edges of G_{B3} on the $\mathcal{Y}$-side.

It is also important to note that G_{B4} does not lose too many edges on the $\mathcal{X}$-side: We lose edges on the $\mathcal{X}$-side when we remove Z's. However, since $|\mathcal{Z}_{B4}| \geq 2^{-\tilde{O}(\sqrt{m})} \cdot |\mathcal{Z}_{A4}|$, and since all the degrees of Z's on the $\mathcal{X}$-side are between $2^{-\tilde{O}(\sqrt{m})} \cdot d_{\mathcal{Z},\mathcal{X}}$ and $d_{\mathcal{Z},\mathcal{X}}$, the graph G_{B4} retains at least $2^{-\tilde{O}(\sqrt{m})}$ fraction of the edges of G_{A4} on the $\mathcal{X}$-side.

7.5 Forcing agreement on the revealed rows

We are now ready to force the a's and b's of each Z to agree on $\mathcal{R}$. Fix a vertex Z. We show that there exists an assignment $\lambda_z \in \{0,1\}^{\mathcal{R}}$, and strings $a \in \mathcal{A}_{B4,Z}$ and $b \in \mathcal{B}_{B4,Z}$ such that

$$a|_{\mathcal{R}} = b|_{\mathcal{R}} = \lambda_z.$$

To this end, we show that if this was not the case, the formula complexity $\mathsf{L}(\mathcal{A}_{B4,Z} \times \mathcal{B}_{B4,Z})$ was at most $2^{2\sqrt{m}} \cdot m$ – thus contradicting the lower bound of $2^{\ell - \tilde{O}(\sqrt{m})}$ we have on $\mathsf{L}(\mathcal{A}_{B4,Z} \times \mathcal{B}_{B4,Z})$ (for an appropriate choice of the constant C in the definition of ℓ). The upper bound of $2^{2\sqrt{m}} \cdot m$ is derived by considering the following protocol for $KW_{\mathcal{A}_{B4,Z} \times \mathcal{B}_{B4,Z}}$: Alice sends to Bob $a|_{\mathcal{R}}$. By assumption, $a|_{\mathcal{R}} \neq b|_{\mathcal{R}}$, so now Bob knows a coordinate i such that $a_i \neq b_i$ and sends it to Alice. At this point, they solved $KW_{\mathcal{A}_{B4,Z} \times \mathcal{B}_{B4,Z}}$. It is not hard to see that the size of this protocol is at most $2^{2\sqrt{m}} \cdot m$. Hence, there exist a, b, λ_z as above.

Due to the step of "removing the small classes" on Alice's side, we know that the fraction of the strings $a' \in \mathcal{A}_{B4,Z}$ that satisfy $a'|_{\mathcal{R}} = \lambda_z$ is at least $2^{-3\cdot\sqrt{m}\cdot\log m}$: To see why, first observe that $\mathcal{A}_{B4} = \mathcal{A}_{A4} \subseteq \mathcal{A}_{A3}$. Then, recall that in the step of "removing the small classes", we partitioned $\mathcal{A}_{A3}$ to classes which were labeled by pairs $(\mathcal{R}', \lambda')$, and we obtained $\mathcal{A}_{A4}$ by removing the classes that consisted of less than $2^{-3\cdot\sqrt{m}\cdot\log m}$ fraction of the strings in $\mathcal{A}_{A3}$. Now, we know that there is a string $a \in \mathcal{A}_{B4}$ and r_Z such that $a|_{\mathcal{R}} = \lambda_Z$, and this implies that the class labeled by $(\mathcal{R}, \lambda_Z)$ was not removed. Hence, this class, consists of at least $2^{-3\cdot\sqrt{m}\cdot\log m}$ fraction of the strings in $\mathcal{A}_{A3}$, and in particular consists of at least $2^{-3\cdot\sqrt{m}\cdot\log m}$ fraction of the strings in $\mathcal{A}_{B4}$. A similar argument shows that at least $2^{-3\cdot\sqrt{m}}$ fraction of the strings $b' \in \mathcal{B}_{B4,Z}$ satisfy $b|_{\mathcal{R}} = \lambda_z$.

We now define for every Z the sets

$$\begin{aligned}\mathcal{A}_{\mathrm{agr},Z} &= \{a \in \mathcal{A}_{B4} : a|_{\mathcal{R}} = \lambda_Z\}\\ \mathcal{B}_{\mathrm{agr},Z} &= \{b \in \mathcal{B}_{B4} : b|_{\mathcal{R}} = \lambda_Z\}\end{aligned}$$

and remove all the edges of Z that correspond to strings outside $\mathcal{A}_{\mathrm{agr},Z}$ and $\mathcal{B}_{\mathrm{agr},Z}$. We denote the resulting graph by G_{agr}. We summarize the properties of G_{agr}:

- For every Z, it holds that $a|_{\mathcal{R}} = b|_{\mathcal{R}}$ for all $a \in \mathcal{A}_{\mathrm{agr},Z}$ and $b \in \mathcal{B}_{\mathrm{agr},Z}$.
- The graph G_{agr} retains at least $2^{-\tilde{O}(\sqrt{m})}$ fraction of the edges of G_{B4} on the $\mathcal{X}$-side, and hence at least $2^{-\tilde{O}(\sqrt{m})}$ fraction of the edges of G_{A3} on the $\mathcal{X}$-side. Similarly, G_{agr} contains at least $2^{-\tilde{O}(\sqrt{m})}$ fraction of the edges of G_{B3} on the $\mathcal{Y}$-side.
- The Z's are "roughly regular": For every Z, its degree on the $\mathcal{X}$-side is between $2^{-\tilde{O}(\sqrt{m})} \cdot d_{\mathcal{Z},\mathcal{X}}$ and $d_{\mathcal{Z},\mathcal{X}}$. The same holds for the $\mathcal{Y}$-side and $d_{\mathcal{Z},\mathcal{Y}}$.

7.6 Clean-up

We are almost ready to define our almost-hard distribution. Recall that this distribution is going to be defined by sampling a uniformly distributed path $X - Z - Y$ on a graph G', and that we denote by $j_1, \ldots, j_m$ the axes of the edges in Z. We would like this distribution to satisfy the following properties:

- For every $i \in \mathcal{R}$, it holds that $X_i = Y_i$ with probability 1.
- For every $i \in [m] - \mathcal{R}$ and every specific choice X^*, the min-entropy $H_\infty(j_i|X = X^*)$ is at least $\log n - \tilde{O}(\sqrt{m})$. The same holds for Y^*'s.

The first property holds for the distribution of G_{agr}. The second property basically follows from our step of "finding the revealed rows" in Alice's and Bob's sides, that is, Inequalities 11 and 12 above. However, the latter inequalities were proved for G_{A3} and G_{B3} respectively, and they do not imply similar inequalities for G_{agr} because of two issues:

- G_{agr} contains only some of the edges of G_{A3}, and this may cause the min-entropy $H(j_i|X^*)$ in G_{agr} to be much smaller than in G_{A3}.
  We note that this is an issue only for a minority of the vertices X^*: since G_{agr} retains $2^{-\tilde{O}(\sqrt{m})}$ fraction of the edges of G_{A3}, it holds that the degree of the average X^* is at least $2^{-\tilde{O}(\sqrt{m})}$ fraction of its degree in G_{A3}. For such vertices X^*, the min-entropy $H(j_i|X^*)$ is still sufficiently large. However, in order for the above second property to hold, we need the min-entropy to be large for *every* X^*. Similar considerations apply for Y^* and G_{B3}.
- When we proved the lower bound on the min-entropy $H_\infty(j_i|X^*)$ for G_{A3}, we assumed that Z is a uniformly distributed neighbor of X^*. However, in a uniformly distributed path $X - Z - Y$, this is not necessarily the case.
  It turns out that this is not a problem: It can be shown that the probability of each specific choice Z^* of Z is at most $2^{\tilde{O}(\sqrt{m})}$ times the probability of any other specific choice, and this is sufficiently good for our purposes. This follows from the "rough regularity" of the Z's, i.e., the fact that the degree of each specific choice Z^* on the $\mathcal{X}$-side is at most $2^{\tilde{O}(\sqrt{m})}$ larger than the degree of any other specific choice. The same argument works for the Y^*'s.

We could try to resolve the first issue by removing from G_{agr} the X's and Y's whose degree is too low. However, this might harm the rough regularity of the Z's, since it may cause some of the Z's to lose too many edges. We could fix the rough regularity by removing the Z's whose degree is too small, but then we will have X's and Y's with low degrees again. Fortunately, it turns out that if we repeat this process sufficiently many times, we end up with a graph in which all X's, Y's, and Z's have sufficiently large degrees. We choose the latter graph to be G'.

We turn to describing G' formally. Let $\varepsilon > 0$ be a number such that

- G_{agr} retains at least ε fraction of the edges of G_{A3} (respectively, G_{B3}) on the $\mathcal{X}$-side (respectively, on the $\mathcal{Y}$-side).
- Every Z in G_{agr} has degree at least $\varepsilon \cdot d_{\mathcal{Z},\mathcal{X}}$ (respectively, $\varepsilon \cdot d_{\mathcal{Z},\mathcal{Y}}$) on the $\mathcal{X}$-side (respectively, on the $\mathcal{Y}$-side).

It holds that $\varepsilon = 2^{-\tilde{O}(\sqrt{m})}$. We define the graph G' to be the graph obtained from G_{agr} by performing the following steps iteratively, until there are no more vertices to remove:

1. Remove all the vertices X whose degree is less than $\frac{1}{4} \cdot \varepsilon^3$ fraction of their degree in G_{A3}.
2. Remove all the vertices Y whose degree is less than $\frac{1}{4} \cdot \varepsilon^3$ fraction of their degree in G_{B3}.
3. Remove all the vertices Z whose degree on the $\mathcal{X}$-side is less than $\frac{1}{4} \cdot \varepsilon \cdot d_{\mathcal{Z},\mathcal{X}}$.
4. Remove all the vertices Z whose degree on the $\mathcal{Y}$-side is less than $\frac{1}{4} \cdot \varepsilon \cdot d_{\mathcal{Z},\mathcal{Y}}$.

When the process ends, we define the resulting graph to be G'. Our almost-hard distribution will be the distribution of G'. However, in order for this distribution to be well defined, we need to prove that G' is not empty. The basic idea of the proof is the following: First, we observe that Steps 1 and 2 cannot remove too many edges, since they only remove vertices whose degree is much lower than the average degree. Then, we observe that Steps 3 and 4 cannot remove too many Z's – the reason is that a vertex Z is only removed if many of its edges were removed in Steps 1 and 2. Finally, we observe that since only a few Z's are removed in Steps 3 and 4, and since the Z's are roughly regular, then those steps also cannot remove too many edges. We conclude that the process has not removed too many edges in all of the steps, and hence some edges must have remained. Details follow.

In order to prove that G' is not empty, we upper bound the number of edges that are removed by the foregoing process, and show that this number is less than the total number of edges of G_{agr}. First, we define some notation:

- We denote by $e_{A3,\mathcal{X}}$ and $e_{A3,\mathcal{Y}}$, the numbers of edges of G_{A3} on the $\mathcal{X}$-side and $\mathcal{Y}$-side respectively. We similarly denote $e_{B3,\mathcal{X}}$, $e_{B3,\mathcal{Y}}$, $e_{\text{agr},\mathcal{X}}$ and $e_{\text{agr},\mathcal{Y}}$ for G_B and G_{agr}.
- We denote $\mathcal{Z}_{\text{agr}}$ the set of Z's of G_{agr}.
- We denote by $\mathcal{X}$ and $\mathcal{Y}$ the sets of X's and Y's in G_{agr}. Observe that $\mathcal{X}$ is equal to the set of X's in G_{A3}, and $\mathcal{Y}$ is equal to the set of Y's in G_{B3}.
- For every $X \in \mathcal{X}$, we denote by d_X the degree of X in G_{A3}. Note that this is the degree in G_{A3}, and *may be different than the degree in G_{B3} or G_{agr}*.
- With some abuse of notation, for every $Y \in \mathcal{Y}$, we denote by d_Y the degree of Y in G_{B3}. *Note that this is the degree in G_{B3} and not in G_{A3}.*

We now prove that the $\mathcal{X}$-side of G' is not empty, and a similar proof holds for the $\mathcal{Y}$-side. To this end, we upper bound the number of edges on the $\mathcal{X}$-side that are removed in each step of the iterative construction above, and show that the total number of edges removed is less than $e_{\text{agr},\mathcal{X}}$. We start our proof by upper bounding the total number of edges that are removed in Step 1 above (in all iterations combined): Whenever we remove a vertex X, we remove at most $\frac{1}{4} \cdot \varepsilon^3 \cdot d_X$ edges. Hence, the total number of edges that are removed in Step 1 is at most

$$\sum_{X \in \mathcal{X}} \frac{1}{4} \cdot \varepsilon^3 \cdot d_X = \frac{1}{4} \cdot \varepsilon^3 \cdot \sum_{X \in \mathcal{X}} d_X = \frac{1}{4} \cdot \varepsilon^3 \cdot e_{A3,\mathcal{X}} \leq \frac{1}{4} \cdot \varepsilon^2 \cdot e_{\text{agr},\mathcal{X}}, \tag{13}$$

where the inequality holds since $e_{\text{agr},\mathcal{X}} \geq \varepsilon \cdot e_{A3,\mathcal{X}}$ by the definition of ε. Next, observe that the number of edges on the $\mathcal{X}$-side that are removed in Step 3 (in all iterations combined) is at most

$$\frac{1}{4} \cdot \varepsilon \cdot d_{\mathcal{Z},\mathcal{X}} \cdot |\mathcal{Z}_{\text{agr}}| \leq \frac{1}{4} \cdot e_{\text{agr},\mathcal{X}},$$

where the inequality follows from the fact that every $Z \in \mathcal{Z}_{\text{agr}}$ has at least $\varepsilon \cdot d_{\mathcal{Z},\mathcal{X}}$ edges on the $\mathcal{X}$-side in G_{agr}. Finally, we upper bound the number of edges that are removed on the $\mathcal{X}$-side in Step 4 (again, in all iterations combined): In order for a vertex Z to be removed in Step 4, we must have removed at least $\frac{3}{4} \cdot \varepsilon \cdot d_{\mathcal{Z},\mathcal{Y}}$ of its edges on the $\mathcal{Y}$-side previously. Those edges could only be removed in Step 2. On the other hand, it can be shown that the total number of edges removed on the $\mathcal{Y}$-side in Step 2 is at most $\frac{1}{4} \cdot \varepsilon^2 \cdot e_{\text{agr},\mathcal{Y}}$ using the same argument as in Inequality 13. Therefore the total number of $\mathcal{Z}$'s that are removed in Step 4 is at most

$$\frac{\frac{1}{4} \cdot \varepsilon^2 \cdot e_{\text{agr},\mathcal{Y}}}{\frac{3}{4} \cdot \varepsilon \cdot d_{\mathcal{Z},\mathcal{Y}}} \leq \frac{\frac{1}{4} \cdot \varepsilon^2 \cdot d_{\mathcal{Z},\mathcal{Y}} \cdot |\mathcal{Z}_{\text{agr}}|}{\frac{3}{4} \cdot \varepsilon \cdot d_{\mathcal{Z},\mathcal{Y}}} = \frac{1}{3} \cdot \varepsilon \cdot |\mathcal{Z}_{\text{agr}}| \,.$$

where the inequality again follows from the fact that every $Z \in \mathcal{Z}_{\text{agr}}$ has at least $\varepsilon \cdot d_{\mathcal{Z},\mathcal{Y}}$ edges on the $\mathcal{Y}$-side in G_{agr}. Now, note that each of those Z's can have at most $d_{\mathcal{Z},\mathcal{X}}$ edges on the $\mathcal{X}$-side, so the total number of edges that are removed in Step 4 on the $\mathcal{X}$-side is at most

$$\frac{1}{3} \cdot \varepsilon \cdot d_{\mathcal{Z},\mathcal{X}} \cdot |\mathcal{Z}_{\text{agr}}| \leq \frac{1}{3} \cdot e_{\text{agr},\mathcal{X}}.$$

Summing up, the total number of edges that are removed on the $\mathcal{X}$-side is at most

$$\frac{1}{4} \cdot \varepsilon^2 \cdot e_{\text{agr},\mathcal{X}} + \frac{1}{4} \cdot e_{\text{agr},\mathcal{X}} + \frac{1}{3} \cdot e_{\text{agr},\mathcal{X}} < e_{\text{agr},\mathcal{X}},$$

and therefore G' is non-empty on the $\mathcal{X}$-side. Similarly, it can be shown that G' is non-empty on the $\mathcal{Y}$-side, as required.

7.7 The almost-hard distribution

As mentioned above, our almost-hard distribution is the distribution of G': choose a uniformly distributed path $X - Z - Y$ in G', and output (X, Y). We now prove that this is indeed an $\tilde{O}(\sqrt{m})$-almost hard distribution. Clearly, for every $i \in \mathcal{R}$ it holds that $X_i = Y_i$ with probability 1. For every $i \in [m] - \mathcal{R}$ it either holds that $X_i = Y_i$ or it holds that X_i and Y_i disagree on exactly one coordinate, which is j_i, the i-th axis of Z. It remains to prove that for every X^* or Y^*, it holds that

$$H_\infty(j_i | X = X^*) \geq \log n - \tilde{O}(\sqrt{m}) \tag{14}$$

$$H_\infty(j_i | Y = Y^*) \geq \log n - \tilde{O}(\sqrt{m}). \tag{15}$$

We use the following claim, whose proof is deferred to the end of this section.

▶ **Claim 7.2.** *Fix a specific choice X^* of X. The probability of each specific choice Z^* of Z to be chosen conditioned on $X = X^*$ is at most $2^{\tilde{O}(\sqrt{m})}$ times larger than the probability of any other specific choice. The same holds for Y^*.*

We now prove Inequality 14, and Inequality 15 can be proved similarly. Basically, Inequality 14 follows from the corresponding inequality for G_{A3} (Inequality 11). As discussed in Section 7.6, there are two issues to deal with: First, the latter inequality assumes that Z is uniformly distributed, while in Inequality 14 the vertex Z is not uniformly distributed – this issue is resolved using Claim 7.2. Second, the degree of X^* in G' is smaller than its degree in G_{A3} – however, it is only smaller by a factor of $2^{\tilde{O}(\sqrt{m})}$, so this does not decrease the min-entropy of j_i by too much. We now provide the formal argument, which is a straightforward calculation.

Fix $i \in [m] - \mathcal{R}$, and fix a specific choice X^* of X. Fix a specific choice j^* for j_i, and let $\mathcal{Z}_{i,j^*}$ be the set of neighbors Z^* of X^* in G' whose axis on the i-th row is j^*. Recall that we proved that in G_{A3}, if Z is a *uniformly distributed* neighbor of X^*, then

$$H_\infty(j_i|X = X^*) \geq \log n - \tilde{O}(\sqrt{m}).$$

This implies in particular that under this distribution it holds that

$$\Pr[j_i = j^*|X = X^*] \leq \frac{2^{\tilde{O}(\sqrt{m})}}{n}.$$

In other words, this means that $\mathcal{Z}_{i,j^*}$ constitutes at most $2^{\tilde{O}(\sqrt{m})}/n$ fraction of the neighbors of X^* in G_{A3}. Next, observe that by our construction of G', the degree of X^* in G' is at least $2^{-\tilde{O}(\sqrt{m})}$ fraction of its degree in G_{A3}. Therefore $\mathcal{Z}_{i,j^*}$ constitutes at most $2^{\tilde{O}(\sqrt{m})}/n$ fraction of *the neighbors of* X^* *in* G'. Finally, the latter fact together with Claim 7.2 implies that the probability that $Z \in \mathcal{Z}_{i,j^*}$ is at most $2^{\tilde{O}(\sqrt{m})}/n$, as required. This concludes the proof of the main lemma.

Proof of Claim 7.2. Fix choices X^* and Z^*. For every specific choice Z' of Z, it holds that

$$\frac{\Pr[Z^*|X^*]}{\Pr[Z'|X^*]} = \frac{\Pr[Z^* \text{ and } X^*]}{\Pr[Z' \text{ and } X^*]}.$$

Now, $\Pr[Z^* \text{ and } X^*]$ is the probability of the edge (X^*, Z^*) to be selected, which is proportional to the number of paths $X - Z - Y$ in which it participates. The latter number is exactly the degree of Z^* on the $\mathcal{Y}$-side, which is between $2^{-\tilde{O}(\sqrt{m})} \cdot d_{\mathcal{Z},\mathcal{Y}}$ and $d_{\mathcal{Z},\mathcal{Y}}$. The same holds for the probability $\Pr[Z' \text{ and } X^*]$. It thus follows that

$$\frac{\Pr[Z^* \text{ and } X^*]}{\Pr[Z' \text{ and } X^*]} \leq \frac{d_{\mathcal{Z},\mathcal{Y}}}{2^{-\tilde{O}(\sqrt{m})} \cdot d_{\mathcal{Z},\mathcal{Y}}} \leq 2^{\tilde{O}(\sqrt{m})},$$

as required. ◀

8 Average-Case Lower Bounds

In this section, we prove average-case analogues of our main theorem (in Section 8.1) and of the cubic lower bound for Andreev's function (in Section 8.2). Hardness on-average is defined as follows.

▶ **Definition 8.1.** A function $F : \{0,1\}^N \to \{0,1\}$ is said to be (s, ε)-*hard* if every formula of size at most s computes F correctly on at most $\frac{1}{2} + \varepsilon$ fraction of the inputs.

8.1 Average-case lower bound for composition

We prove the following theorem, which is an average-case analogue of our main theorem.

▶ **Theorem 1.3** (restated). *Let $f : \{0,1\}^m \to \{0,1\}$ be an (s, ε)-hard function. Then, $f \diamond \oplus_n$ is $(s', \varepsilon + 2^{-m})$-hard for*

$$s' \geq s \cdot \mathsf{L}(\oplus_n)/2^{\tilde{O}(\sqrt{m+\log n})}.$$

To this end, we use the following immediate corollary of the Karchmer-Wigderson connection (Theorem 2.11).

▶ **Corollary 8.2.** *A function $F : \{0,1\}^N \to \{0,1\}$ is (s,ε)-hard if and only if for every two sets $\mathcal{X} \subseteq F^{-1}(0)$ and $\mathcal{Y} \subseteq F^{-1}(1)$ such that $|\mathcal{X}| + |\mathcal{Y}| > (\frac{1}{2} + \varepsilon) \cdot 2^N$, it holds that $\mathsf{L}(KW_{\mathcal{X}\times\mathcal{Y}}) \geq s$.*

Let $f : \{0,1\}^m \to \{0,1\}$ be an (s,ε)-hard function and let $\mathcal{X} \subseteq (f \diamond \oplus_n)^{-1}(0)$ and $\mathcal{Y} \subseteq (f \diamond \oplus_n)^{-1}(1)$ be such that $(\frac{1}{2} + \varepsilon + 2^{-m}) \cdot 2^{m \cdot n}$. Our goal is to prove that

$$\mathsf{L}(KW_{\mathcal{X}\times\mathcal{Y}}) \geq s \cdot n^2 / 2^{\tilde{O}(\sqrt{m+\log n})}.$$

In order to do so, we prove that the rectangle $\mathcal{X}\times\mathcal{Y}$ satisfies the requirement of the generalized f-stage lemma (Lemma 4.5). We then derive the lower bound by plugging the latter lemma into the proof of the main theorem in Section 3.3.

Recall that for every product of edges $Z = (Z^0, Z^1)$, we define the f-rectangle of $\mathcal{X} \times \mathcal{Y}$ with respect to Z as the rectangle $\mathcal{A}_Z \times \mathcal{B}_Z$ where

$$\begin{aligned} \mathcal{A}_Z &\stackrel{\text{def}}{=} \left\{a \in f^{-1}(0) | Z^a \in \mathcal{X}\right\} \\ B_Z &\stackrel{\text{def}}{=} \left\{b \in f^{-1}(1) | Z^b \in \mathcal{Y}\right\}. \end{aligned}$$

In order to show that the rectangle $\mathcal{X} \times \mathcal{Y}$ satisfies the requirement of the generalized f-stage lemma, we need to show that for at least 2^{-m} fraction of the Z's it holds that $\mathsf{L}(KW_{\mathcal{A}_Z\times\mathcal{B}_Z}) \geq s$. To this end, it suffices to prove that at least 2^{-m} fraction of the Z's satisfy that $|\mathcal{A}_Z| + |\mathcal{B}_Z| \geq (\frac{1}{2} + \varepsilon) \cdot 2^m$, and this will imply the required lower bound on $\mathsf{L}(KW_{\mathcal{A}_Z\times\mathcal{B}_Z})$ by the average-case hardness of f. We prove this via a straightforward averaging argument.

More specifically, consider the following bipartite graph G: One side of the graph is the set $\mathcal{X} \cup \mathcal{Y}$, and other side is the set $\mathcal{Z}$ of all Z's. A matrix $W \in \mathcal{X} \cup \mathcal{Y}$ is connected to $Z \in \mathcal{Z}$ if and only if $W = Z^w$ for some $w \in \{0,1\}^m$. It is easy to see that the degree of every $W \in \mathcal{X} \cup \mathcal{Y}$ is exactly n^m, so the total number of edges in the graph is

$$|\mathcal{X} \cup \mathcal{Y}| \cdot n^m \geq (\frac{1}{2} + \varepsilon + 2^{-m}) \cdot 2^{m \cdot n} \cdot n^m = (\frac{1}{2} + \varepsilon + 2^{-m}) \cdot 2^m \cdot |\mathcal{Z}|,$$

where the equality holds since $|\mathcal{Z}| = 2^{m\cdot(n-1)} \cdot n^m$. On the other hand, the degree of each Z in this graph is exactly $|\mathcal{A}_Z| + |\mathcal{B}_Z|$. Now, the Z's whose degree is less than $\left(\frac{1}{2} + \varepsilon\right) \cdot 2^m$ contribute less than

$$\left(\frac{1}{2} + \varepsilon\right) \cdot 2^m \cdot |\mathcal{Z}|$$

edges. Therefore, at least $2^{-m} \cdot 2^m \cdot |\mathcal{Z}|$ edges are connected to Z's whose degree is at least $\left(\frac{1}{2} + \varepsilon\right) \cdot 2^m$. The degree of every such Z is at most 2^m, and therefore the number of such Z's must be at least:

$$2^{-m} \cdot 2^m \cdot |\mathcal{Z}| / 2^m = 2^{-m} \cdot |\mathcal{Z}|.$$

It thus follows that $|\mathcal{A}_Z| + |\mathcal{B}_Z| \geq (\frac{1}{2} + \varepsilon) \cdot 2^m$ for at least 2^{-m} fraction of the Z's, and therefore the rectangle $\mathcal{X} \times \mathcal{Y}$ satisfies the requirement of the generalized f-stage lemma.

We finally turn to prove the lower bound. Fix a protocol Π that solves $KW_{\mathcal{X}\times\mathcal{Y}}$, and let us denote its size by S. Without loss of generality, we may assume that $S \leq 2^m \cdot n^2$, or otherwise we are done. We apply Theorem [6] 2.4 to Π with $\alpha = \frac{1}{\sqrt{m+\log n}}$, thus obtaining a

[6] See also the restatement of this theorem in Section 3.3

new protocol Π' of depth at most $2^{\tilde{O}(\sqrt{m+\log n})}$ and size $S' \leq S^{1+\frac{1}{\sqrt{m+\log n}}}$. We prove that $S' \geq s \cdot \mathsf{L}(\oplus_n)/2^{\tilde{O}(\sqrt{m+\log n})}$ and this will imply the same lower bound for S, as required (see Section 3.3 for details).

By the generalized f-stage lemma (Lemma 4.5), it follows that Π' has at least $s/2^{\tilde{O}(\sqrt{m+\log n})}$ partial transcripts π_1 that are alive, where none of them is an ancestor of another. By the structure theorem (Theorem 3.4), for each such partial transcript π_1 there are at least $\mathsf{L}(\oplus_n)/2^{\tilde{O}(\sqrt{m})}$ suffixes π_2 such that $\pi_1 \circ \pi_2$. Summing over all the possible choices for π_1 and π_2, it follows that Π' has at least $s \cdot \mathsf{L}(\oplus_n)/2^{\tilde{O}(\sqrt{m+\log n})}$ distinct transcripts, which is what we wanted to prove.

8.2 Average-case cubic lower bound

In the rest of this section, we prove Corollary 1.4, which gives average-case cubic lower bounds for a variant of the Andreev function due to Komargodski and Raz [23]. Our proof is essentially the same as that of [23], modulo the proof of Theorem 1.3, and some different choices of the parameters.

▶ **Corollary 1.4** (restated). *For every $n, c \in \mathbb{N}$ there exists a function $F_{n,c} : \{0,1\}^n \to \{0,1\}$ bits that is (S, n^{-c})-hard for*

$$S \geq n^{3-\tilde{O}(\frac{1}{\sqrt{\log n}})}.$$

Let $n, c \in \mathbb{N}$ be as in the theorem. Let $m \stackrel{\text{def}}{=} 10 \cdot c \cdot \log n$. Let $C : \{0,1\}^{n/2} \to \{0,1\}^{2^m}$ be the list-decodable code of Fact 2.22, and recall that the list-decodability means that for every string $w \in \{0,1\}^{2^m}$, there are at most 2^m codewords of C that are $(\frac{1}{2} - \frac{1}{2} \cdot \sqrt{\frac{n}{2^{m/2}}})$-close to w. The function $F_{n,c}$ is defined as follows: The input of $F_{n,c}$ consists of two parts, each of length $n/2$. The first part of the input is denoted f. Recall that $C(f)$ is a string of length 2^m, and we view it as a truth table of a function from $\{0,1\}^m$ to $\{0,1\}$. The second part of the input is a sequence $x_1, \ldots, x_m$ of strings in $\{0,1\}^{n/2m}$. The function $F_{n,c}$ is now defined by

$$F_{n,c}(f, x_1, \ldots, x_m) \stackrel{\text{def}}{=} \left(C(f) \diamond \oplus_{\frac{n}{2m}}\right)(x_1, \ldots, x_m).$$

We use the following claim, which is proved by a straightforward counting argument.

▶ **Claim 8.3.** *Let f be a uniformly distributed string in $\{0,1\}^{n/2}$. Then, the function $C(f)$ is (s, n^{-2c})-hard for $s \stackrel{\text{def}}{=} n/16 \cdot \log m$ with probability at least $1 - 2^{-n/5}$.*

Proof. We count the number of functions from $\{0,1\}^m$ to $\{0,1\}$ that can be approximated by formulas of size $s \stackrel{\text{def}}{=} n/16 \cdot \log m$. Following a calculation in [17] (see the proof of Theorem 1.23), the number of formulas of size s over m variables is at most $(9m)^s \leq 2^{n/4}$. By the list-decodability of C, for each such formula ϕ there are at most 2^m strings $h \in \{0,1\}^{n/2}$ such that

$$\Pr_{x \leftarrow \{0,1\}^{n/2}}[C(h)(x) = \phi(x)] > \frac{1}{2} + n^{-2c} \geq \frac{1}{2} + \frac{1}{2} \cdot \sqrt{\frac{n}{2^{m/2}}}. \tag{16}$$

It follows that the total number of strings h that satisfy Inequality 16 for *any* formula of size s is at most $2^{n/4} \cdot 2^m$. Therefore, if f is chosen uniformly at random, the probability that $C(f)$ is (s, n^{-2c})-hard is at least

$$1 - \frac{2^{n/4} \cdot 2^m}{2^{n/2}} \geq 1 - 2^{-n/5},$$

as required. ◀

By Theorem 1.3, for every fixed choice of f for which $C(f)$ is (s, n^{-2c})-hard, it holds that $C(f) \diamond \oplus_{\frac{n}{2m}}$ is $(S, n^{-2c} + 2^{-m})$-hard for

$$S \stackrel{\text{def}}{=} s \cdot n^2 / 2^{\tilde{O}(\sqrt{m+\log n})} = n^{3-\tilde{O}(\frac{1}{\sqrt{\log n}})}.$$

Therefore, for every such fixed choice of f and every fixed formula ϕ of size S, it holds that

$$\Pr_{x_1,\ldots,x_m \leftarrow \{0,1\}^{n/2m}} [F_{n,c}(f, x_1, \ldots, x_m) = \phi(f, x_1, \ldots, x_m)] \leq \frac{1}{2} + n^{-2c} + 2^{-m}.$$

Now, let f be uniformly distributed, and let H_f denote the event in which $C(f)$ is (s, n^{-2c})-hard. It follows that for every formula ϕ of size at most S and for uniformly distributed f and $x_1, \ldots, x_m$ it holds that

$$\begin{aligned}
& \Pr [F_{n,c}(f, x_1, \ldots, x_m) = \phi(f, x_1, \ldots, x_m)] \\
\leq\ & \Pr [F_{n,c}(f, x_1, \ldots, x_m) = \phi(f, x_1, \ldots, x_m) | H_f] + \Pr [\neg H_f] \\
\leq\ & \frac{1}{2} + n^{-2c} + 2^{-m} + 2^{-n/5} \\
\leq\ & \frac{1}{2} + n^{-c}.
\end{aligned}$$

Hence, $F_{n,c}$ is (s, n^{-c})-hard for $S \geq n^{3-\tilde{O}(\frac{1}{\sqrt{\log n}})}$, as required.

9 Future Directions and Open Problems

In order to prove the KRW conjecture, one should replace the parity function in our result with a general function $g : \{0,1\}^n \to \{0,1\}$. It seems to us that a good starting point would be to prove the KRW conjecture for the composition of a universal relation and a function g, denoted $U \diamond g$. We now explain what this composition is, and then discuss how one might prove the KRW composition for it.

The composition $U \diamond g$

The *universal relation* is the following communication problem: Alice and Bob get two distinct strings $x, y \in \{0,1\}^m$, and should find a coordinate on which x and y disagree. The difference between the universal relation and KW relations is that x and y are not required to be a 0-preimage and 1-preimage of some function f. This makes the universal relation much simpler and easier to analyze, and therefore the universal relation is often a good starting point for studying KW relations. For convenience, we denote the universal relation by U.

As was observed by [15], it is often useful to relax the requirement that x and y are distinct as follows: We allow x and y to be equal, but in this case, we also allow Alice and Bob to reject the inputs instead of outputting a coordinate. It is not hard to show that this relaxation does not increase the complexity of the problem by much. It is well-known that the communication complexity of the (relaxed) universal relation is at least m, and that the "hardest inputs" are those in which $x = y$ [20, 10, 15, 12].

The composition $U \diamond g$ is the following communication problem: Alice and Bob get as inputs $m \times n$ matrices X and Y respectively such that $g(X) \neq g(Y)$, and their goal is to find an entry (i, j) such that $X_{i,j} \neq Y_{i,j}$. Again, we relax the requirement that $g(X) \neq g(Y)$ as follows: We allow X and Y to satisfy $g(X) = g(Y)$, but in this case, we also allow Alice and Bob to reject the inputs and not output an entry (i, j). Here, too, the relaxation does not increase the complexity of the problem by much.

The KRW conjecture for $U \diamond g$

The analogue of the KRW conjecture for $U \diamond g$ would be to prove that

$$\mathsf{C}(U \diamond g) \approx \mathsf{C}(U) + \mathsf{C}(KW_g) \approx m + \mathsf{C}(KW_g)$$

(for simplicity, we focus on the communication complexity rather than on the protocol size). We could try to to prove it using the approach of this paper as follows. Suppose that there is a protocol Π that solves $U \diamond g$. Then, we would have liked to prove the following claims:

- **An analogue of the f-stage lemma:** There is a partial transcript of π_1 of length $m - \tilde{O}(\sqrt{m})$ that is alive, i.e., that has not solved the universal relation on $g(X)$ and $g(Y)$.
- **An analogue of the structure theorem:** Any live partial transcript π_1 has a suffix of length $\mathsf{C}(KW_g) - \tilde{O}(\sqrt{m})$.

If we could prove those two claims, they would have implied the lower bound

$$\mathsf{C}(U \diamond g) \geq m + \mathsf{C}(KW_g) - \tilde{O}(\sqrt{m}), \tag{17}$$

which would have been sufficiently good for our purposes.

An analogue of the f-stage lemma

Recall that in Section 3, we implemented the above approach by defining products of edges $Z = (Z^0, Z^1)$. We then invoked the protocol on inputs X and Y of the form $X = Z^a$, $Y = Z^b$ for $a \in f^{-1}(0)$ and $b \in f^{-1}(1)$. In particular, we proved the f-stage lemma by considering the invocation of the protocol on such inputs for different Z's.

We would like to prove an analogue of the f-stage lemma for $U \diamond g$ using a similar strategy. To this end, we would like to invoke the protocol Π on inputs of the form $X = Z^a$, $Y = Z^b$, and show that it cannot solve the universal relation on a and b using $m - \tilde{O}(\sqrt{m})$ bits. A natural way to do so would be to choose the pair (a, b) to be a hard input for the universal relation.

As we noted above, the hard inputs to the universal relation are those in which $a = b$. Now, observe that whenever $a = b$, it also holds that $X = Y$. Thus, it seems that for an analogue of the f-stage lemma for $U \diamond g$, we should invoke the protocol Π on inputs of the form (X, X). This leads to the following natural definition for what it means that "π_1 is alive".

▶ **Definition 9.1.** We say that a partial transcript π_1 *is alive* if for at least $2^{-(m-\tilde{O}(\sqrt{m}))}$ fraction of the matrices $X \in \{0,1\}^{m \times n}$, the input (X, X) is consistent with π_1. In other words, if we denote by $\mathcal{X}_{\pi_1} \times \mathcal{Y}_{\pi_1}$ the rectangle of π_1, then $X \in \mathcal{X}_{\pi_1} \cap \mathcal{Y}_{\pi_1}$ for at least $2^{-(m-\tilde{O}(\sqrt{m}))}$ fraction of the matrices $X \in \{0,1\}^{m \times n}$.

Intuitively, this definition says that π_1 has gives at most $m - \tilde{O}(\sqrt{m})$ bits of information about the inputs of the players. In particular, π_1 gives at most $m - \tilde{O}(\sqrt{m})$ bits about a and b, and therefore it is still far from solving the universal relation on a and b. This intuition can be formalized using the ideas of [10, 15, 12], but it is not necessary for our discussion. The following analogue of the f-stage lemma can now be proved using a straightforward averaging argument.

▶ **Lemma 9.2** (Universal-stage lemma). *There is a live partial transcript π_1 of length $m - \tilde{O}(\sqrt{m})$ that has not solved the universal relation.*

An analogue of the structure theorem

The difficult part in proving the lower bound on $\mathsf{C}(U \diamond g)$ would be proving an analogue of the structure theorem. Such an analogue would say that if Alice and Bob have not solved the universal relation yet, then they must transmit $\mathsf{C}(KW_g) - \tilde{O}(\sqrt{m})$ more bits. Given Definition 9.1, this can be formalized as follows.

▶ **Conjecture 9.3.** *Let $\mathcal{X} \subseteq \{0,1\}^{m \times n}$ be a set of matrices of density at least $2^{-(m-\tilde{O}(\sqrt{m}))}$. Then, the restriction of $U \diamond g$ to the rectangle $\mathcal{X} \times \mathcal{X}$ has communication complexity at least $\mathsf{C}(KW_g) - \tilde{O}(\sqrt{m})$.*

We note that it is possible to construct artificial examples of functions g for which Conjecture 9.3 does not hold: in particular, if g is easy on $(1-\varepsilon)$-fraction of its inputs, it is possible that all the matrices in $\mathcal{X}$ contain only easy inputs as rows[7]. However, it might be possible to prove it for some "reasonable" class of functions, and that might be sufficient for proving formula lower bounds. For example, it might be possible to prove this conjecture for the case where g is a random function. We also note there is a simple (but non-trivial) proof of the conjecture for the case where $g = \oplus_n$ – in fact, this observation was the trigger to this work.

Another way to deal with the aforementioned artificial examples is to change the conjecture such that it allows us to get rid of the easy inputs of g. This is done by replacing $\{0,1\}^{m \times n}$ with some subset $\mathcal{X}_0$ that depends on g and should consist of the hard inputs:

▶ **Conjecture 9.4.** *For every non-constant function $g : \{0,1\}^n \to \{0,1\}$ there exists $\mathcal{X}_0 \subseteq \{0,1\}^{m \times n}$ such that the following holds: Let $\mathcal{X} \subseteq \mathcal{X}_0$ be a set of matrices of density at least $2^{-(m-\tilde{O}(\sqrt{m}))}$ in $\mathcal{X}_0$. Then, the restriction of $U \diamond g$ to the rectangle $\mathcal{X} \times \mathcal{X}$ has communication complexity at least $\mathsf{C}(KW_g) - \tilde{O}(\sqrt{m})$.*

It is not hard to see that Conjecture 9.4 is sufficient for proving the lower bound on $\mathsf{C}(U \diamond g)$: this can be done by replacing $\{0,1\}^{m \times n}$ with $\mathcal{X}_0$ in Definition 9.1 and Lemma 9.2 above.

Conjecture 9.4 could serve as the next intermediate goal toward proving the KRW conjecture, and we suggest it as an open problem. In fact, we do not know how to prove this conjecture even if the density of $\mathcal{X}$ in $\mathcal{X}_0$ is allowed to be as high as $\frac{1}{2}$, and the desired lower bound is allowed to be as small as $\mathsf{C}(KW_g) - 0.99 \cdot m$.

The 1-out-of-k problem

We now discuss a special case of Conjecture 9.3 which seems to be interesting in its own right. First, we define the following communication problem.

▶ **Definition 9.5** (The 1-out-of-k problem)**.** Let $g : \{0,1\}^n \to \{0,1\}$ be a non-constant function, and let $k \in \mathbb{N}$. The 1-*out-of-k version of* KW_g is the following communication problem: Alice and Bob get matrices $X, Y \in \{0,1\}^{k \times n}$ respectively such that

- $g(X)$ and $g(Y)$ are the all-zeroes and all-ones strings respectively.
- All the rows of X and Y are all distinct.

The goal of Alice and Bob is to find an entry (i,j) such that $X_{i,j} \neq Y_{i,j}$.

[7] Consider a function $g : \{0,1\}^n \to \{0,1\}$ that is defined as follows: given an input x, if the first five bits of x are all zeroes, then $g(x)$ is some hard function of the remaining bits, and otherwise $g(x) = x_6$. Now, consider the set $\mathcal{X} \subseteq \{0,1\}^{m \times n}$ that consists of all the matrices X in which there is no row with the first five bits all being zeroes. It is not hard to see that the communication complexity of $U \diamond g$ restricted to $\mathcal{X} \times \mathcal{X}$ is at most $m + O(1)$, and this might be much smaller than $\mathsf{C}(KW_g)$ if $m \ll n$.

Clearly, the communication complexity of the 1-out-of-k version of KW_g is at most $\mathsf{C}(KW_g)$, since Alice and Bob can run the optimal protocol for KW_g on the first rows of X and Y. The question is whether the communication complexity of the 1-out-of-k version of KW_g can be much smaller? We suggest proving the following conjecture as another open problem.

▶ **Conjecture 9.6.** *For every non-constant $g : \{0,1\}^n \to \{0,1\}$ and $k \in \mathbb{N}$, the communication complexity of the 1-out-of-k version of KW_g is at least $\mathsf{C}(KW_g) - \operatorname{poly}\log n$.*

Observe that Conjecture 9.6 is indeed a special case of Conjecture 9.3: the reason is that we can always choose the subset $\mathcal{X}$ to be the set of matrices X such that the first k bits of $g(X)$ are equal, and all the rows of X are distinct. The density of this set $\mathcal{X}$ is slightly less than 2^{-k}, and the communication complexity of the restriction of $U \diamond g$ to the rectangle $\mathcal{X} \times \mathcal{X}$ is at most the communication complexity of the 1-out-of-k version of KW_g.

We note that although we defined the 1-out-of-k problem only for KW relations, it could be generalized to other models of computation. For those models, one could state analogues of Conjecture 9.6 that are interesting in their own right. For example, consider the following analogues for communication complexity and circuit complexity:

▶ **Conjecture 9.7** (The 1-out-of-k problem for communication complexity)**.** *Let $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, and consider the communication problem of computing f. The 1-out-of-k version of f is defined as follows: Alice gets distinct $x_1, \ldots, x_k \in \{0,1\}^n$, Bob gets distinct $y_1, \ldots, y_k \in \{0,1\}^n$, and their goal is to output an index $i \in [k]$ and the bit $f(x_i, y_i)$. The conjecture is that the communication complexity of this problem is at least $\mathsf{C}(f) - \operatorname{poly}\log n$.*

▶ **Conjecture 9.8** (The 1-out-of-k problem for circuit complexity)**.** *Let $f : \{0,1\}^n \to \{0,1\}$, and consider the problem of computing f using a boolean circuit. The 1-out-of-k version of f is defined as follows: a circuit gets as input distinct $x_1, \ldots, x_k$, and it should output an index $i \in [k]$ and the bit $f(x_i)$. The conjecture is that the circuit complexity of this problem is at least the circuit complexity of f up to a polynomial factor.*

The 1-out-of-k problem is a close variant of the "choose" problem introduced by Beimel et al. [3], who also posed conjectures that correspond to Conjectures 9.7 and 9.8. The difference between the 1-out-of-k problem defined above and the "choose" problem of [3] is that in the "choose" problem, the inputs are not required to be distinct, and on the other hand, we have k functions $f_1, \ldots, f_k$ instead of a single function f. The question is whether choosing one of the functions f_i and computing it on its corresponding input is easier than computing the easiest function among $f_1, \ldots, f_k$ in isolation.

[3] made an interesting observation, which also translates to the 1-out-of-k problem as follows: 1-out-of-k conjectures of the above form are implied by *direct-sum conjectures*. For concreteness, we explain this claim for the example of communication complexity. A direct-sum conjecture for communication complexity says that the complexity of computing k independent instances of f is $k \cdot \mathsf{C}(f)$. The observation of [3] is that the latter direct-sum conjecture implies that the communication complexity of the 1-out-of-k version of f is $\mathsf{C}(f)$.

To see why this is true, suppose there was a protocol that solved the 1-out-of-k version of f using less than $\mathsf{C}(f)$ bits. If this was the case, it would have been possible to compute k independent instances of f using less than $k \cdot \mathsf{C}(f)$ as follows: Alice and Bob first use the protocol for the 1-out-of-k version of f on the k instances, thus computing f on one instance. Then, they would compute f independently on each of the remaining instances. The complexity of this protocol would be $(k-1) \cdot \mathsf{C}(f)$ plus the complexity of the 1-out-of-k version of f, which is less than $k \cdot \mathsf{C}(f)$ by assumption.

Direct-sum conjectures have been studied in many different areas. In particular, the direct-sum conjecture for communication complexity has been proposed in [20], and partial results were obtained in [11, 19, 2, 6, 5] . In particular, the result of [11] implies that the complexity of the 1-out-of-k problem of Conjecture 9.7 above is at least $\sqrt{\mathsf{C}(f)}$. Unfortunately, the known results are insufficient for proving Conjecture 9.6. It is interesting question whether proving Conjectures 9.6 and 9.7 is easier than proving the corresponding direct-sum conjectures, or alternatively, whether 1-out-of-k conjectures imply direct-sum conjectures.

Acknowledgements. We would like to thank Avi Wigderson, Avishay Tal, Gillat Kol, Oded Goldreich, and Ilan Komargodski for useful discussions and ideas. We would also like to thank anonymous referees for comments that improved the presentation of this work.

References

1 Alexander E. Andreev. On a method for obtaining more than quadratic effective lower bounds for the complexity of π-schemes. *Moscow University Mathematics Bulletin*, 42(1):24–29, 1987.

2 Boaz Barak, Mark Braverman, Xi Chen, and Anup Rao. How to compress interactive communication. In *STOC*, pages 67–76, 2010.

3 Amos Beimel, Sebastian Ben Daniel, Eyal Kushilevitz, and Enav Weinreb. Choosing, agreeing, and eliminating in communication complexity. *Computational Complexity*, 23(1):1–42, 2014.

4 Maria Luisa Bonet and Samuel R. Buss. Size-depth tradeoffs for boolean formulae. *Inf. Process. Lett.*, 49(3):151–155, 1994.

5 Mark Braverman. Interactive information complexity. In *STOC*, pages 505–524, 2012.

6 Mark Braverman and Anup Rao. Information equals amortized communication. In *FOCS*, pages 748–757, 2011.

7 Richard P. Brent. The parallel evaluation of general arithmetic expressions. *J. ACM*, 21(2):201–206, 1974.

8 Ruiwen Chen, Valentine Kabanets, Antonina Kolokolova, Ronen Shaltiel, and David Zuckerman. Mining circuit lower bound proofs for meta-algorithms. *Computational Complexity*, 24(2):333–392, 2015.

9 Thomas M. Cover and Joy A. Thomas. *Elements of information theory*. Wiley-Interscience, 1991.

10 Jeff Edmonds, Russell Impagliazzo, Steven Rudich, and Jiri Sgall. Communication complexity towards lower bounds on circuit depth. *Computational Complexity*, 10(3):210–246, 2001.

11 Tomás Feder, Eyal Kushilevitz, Moni Naor, and Noam Nisan. Amortized communication complexity. *SIAM J. Comput.*, 24(4):736–750, 1995.

12 Dmitry Gavinsky, Or Meir, Omri Weinstein, and Avi Wigderson. Toward better formula lower bounds: an information complexity approach to the KRW composition conjecture. In *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 – June 03, 2014*, pages 213–222, 2014.

13 Michelangelo Grigni and Michael Sipser. Monotone separation of Logspace from NC. In *Structure in Complexity Theory Conference*, pages 294–298, 1991.

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

15 Johan Håstad and Avi Wigderson. Composition of the universal relation. In *Advances in computational complexity theory, AMS-DIMACS*, 1993.

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

17 Stasys Jukna. *Boolean Function Complexity – Advances and Frontiers*, volume 27 of *Algorithms and combinatorics*. Springer, 2012.

18 Yael Tauman Kalai and Ran Raz. Interactive PCP. In *ICALP (2)*, pages 536–547, 2008.

19 Mauricio Karchmer, Eyal Kushilevitz, and Noam Nisan. Fractional covers and communication complexity. *SIAM J. Discrete Math.*, 8(1):76–92, 1995.

20 Mauricio Karchmer, Ran Raz, and Avi Wigderson. Super-logarithmic depth lower bounds via the direct sum in communication complexity. *Computational Complexity*, 5(3/4):191–204, 1995.

21 Mauricio Karchmer and Avi Wigderson. Monotone circuits for connectivity require super-logarithmic depth. *SIAM J. Discrete Math.*, 3(2):255–265, 1990.

22 V. M. Khrapchenko. A method of obtaining lower bounds for the complexity of π-schemes. *Mathematical Notes Academy of Sciences USSR*, 10:474–479, 1972.

23 Ilan Komargodski and Ran Raz. Average-case lower bounds for formula size. In *Symposium on Theory of Computing Conference, STOC'13, Palo Alto, CA, USA, June 1-4, 2013*, pages 171–180, 2013.

24 Ilan Komargodski, Ran Raz, and Avishay Tal. Improved average-case lower bounds for de Morgan formula size. In *54th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2013, 26-29 October, 2013, Berkeley, CA, USA*, pages 588–597, 2013.

25 Eyal Kushilevitz and Noam Nisan. *Communication complexity*. Cambridge University Press, 1997.

26 Dana Moshkovitz. Parallel repetition from fortification. In *55th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2014, Philadelphia, PA, USA, October 18-21, 2014*, pages 414–423, 2014.

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

28 Ran Raz and Avi Wigderson. Monotone circuits for matching require linear depth. *J. ACM*, 39(3):736–744, 1992.

29 Alexander A. Razborov. Applications of matrix methods to the theory of lower bounds in computational complexity. *Combinatorica*, 10(1):81–93, 1990.

30 Rahul Santhanam. Fighting perebor: New and improved algorithms for formula and QBF satisfiability. In *51th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2010, October 23-26, 2010, Las Vegas, Nevada, USA*, pages 183–192, 2010.

31 Philip M. Spira. On time-hardware complexity tradeoffs for boolean functions. In *Proceedings of the Fourth Hawaii International Symposium on System Sciences*, pages 525–527, 1971.

32 Bella Abramovna Subbotovskaya. Realizations of linear functions by formulas using +,.,-. *Soviet Mathematics Doklady*, 2:110–112, 1961.

33 Madhu Sudan. Algorithmic introduction to coding theory (lecture notes), 2001. Available from `http://theory.csail.mit.edu/~madhu/FT01/`.

34 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, 2014.

35 Andrew Chi-Chih Yao. Some complexity questions related to distributive computing (preliminary report). In *STOC 1979*, pages 209–213, 1979.