Electronic Colloquium on Computational Complexity, Revision 1 of Report No. 181 (2023)

Hardness Condensation by Restriction

Mika Göös
EPFL

Ilan Newman
University of Haifa

Artur Riazanov
EPFL

Dmitry Sokolov
EPFL

April 29, 2024

Abstract. Can every n-bit boolean function with deterministic query complexity $k \ll n$ be restricted to $O(k)$ variables such that the query complexity remains $\Omega(k)$? That is, can query complexity be *condensed* via restriction? We study such hardness condensation questions in both query and communication complexity, proving two main results.

- **Negative:** Query complexity *cannot* be condensed in general: There is a function f with query complexity k such that any restriction of f to $O(k)$ variables has query complexity $\tilde{O}(k^{3/4})$.
- **Positive:** Randomised communication complexity can be condensed for the *sink-of-xor* function. This yields a quantitatively improved counterexample to the log-approximate-rank conjecture, achieving parameters conjectured by Chattopadhyay, Garg, and Sherif (2021).

Along the way we show the existence of *Shearer extractors*—a new type of seeded extractor whose output bits satisfy prescribed dependencies across distinct seeds.

Contents

ISSN 1433-8092

1 Introduction

Hardness condensation is a lower-bound technique in boolean function complexity, where one transforms an n-variate problem f of complexity $k \ll n$ into a related problem f' defined over $\Theta(k)$ variables such that the complexity is preserved at $\Theta(k)$. This approach was first introduced by Buresh-Oppenheim and Santhanam [BS06] in the context of circuit complexity. Later, it was put to concrete use in the context of proof complexity by Razborov [Raz16] and then further developed in [Raz17b, Raz17a, BN20, FPR22]. In these works, the function f' was obtained from f by expander-based function composition.

In this work, we prove two results on hardness condensation in query and communication complexity. We investigate whether hardness can be condensed by using the simplest possible operation that reduces the number of variables: ***restriction***.

1.1 Query complexity: Negative result

We first study the usual deterministic query complexity $\mathrm{D}(f)$ of a boolean function $f\colon \{0,1\}^n \to \{0,1\}$. Namely, $\mathrm{D}(f)$ equals the minimum number of queries an algorithm (decision tree) needs to make, on the worst-case input $x \in \{0,1\}^n$, to the variables $x_i \in \{0,1\}$ in order to compute $f(x)$; see the textbook [Juk12, §14] or the classic survey [BdW02] for background on decision trees. We say that f *condenses* to k variables if there is a partial assignment $\rho\colon [n] \to \{0,1,*\}$ that fixes all but $|\rho^{-1}(*)| = k$ variables, such that the resulting k-bit function $f' \coloneqq f|_\rho$ has maximum query complexity $\mathrm{D}(f|_\rho) = \Theta(k)$. Moreover, we say that f condenses *losslessly* if it condenses to $\Theta(\mathrm{D}(f))$ variables. We ask:

Can every f be condensed losslessly?

Example (Sink). To illustrate our question, we consider the *sink* function [BvEL74] as a recurring example. The function $\textsc{Sink}\colon \{0,1\}^m \to \{0,1\}$ where $m \coloneqq \binom{n}{2}$ is defined as follows. Let G denote the complete graph on nodes $v_1, \dots, v_n$. We interpret an input $x \in \{0,1\}^m$ as an assignment of orientations to the edges of G, which defines a directed graph G_x. We say a node v_i is a *sink* in G_x iff v_i has in-degree $n-1$. We define

$$\textsc{Sink}(x) \coloneqq 1 \quad \iff \quad G_x \text{ contains a sink.}$$

It is well known that $\mathrm{D}(\textsc{Sink}) = \Theta(n)$. For example, we can show a query lower bound using the basic fact that $\mathrm{D}(f) \geq \mathrm{s}(f)$ where $\mathrm{s}(f)$ is the *sensitivity* of f (maximum over all inputs x of the number of *sensitive coordinates* $i \in [n]$ satisfying $f(x) \neq f(x^i)$ where x^i is obtained from x by flipping the i-th bit). A highly sensitive input $x \in \{0,1\}^m$ is given by any orientation where v_1 has in-degree $n-1$ and the remaining nodes lie on a cycle, say, $(v_2, v_3, \dots, v_n, v_2)$. Then $\textsc{Sink}(x) = 1$, but flipping the orientation of any edge incident to v_1 flips the function value to 0. This shows $\mathrm{s}(\textsc{Sink}) \geq n-1$.

We claim that $\textsc{Sink}$ condenses losslessly to $\Theta(n)$ variables. In fact, *any* function f condenses to $\mathrm{s}(f)$ variables: Consider a partial assignment ρ that is obtained from a maximally sensitive input by assigning $*$ to all of its sensitive coordinates. We have $\mathrm{D}(f|_\rho) = \mathrm{s}(f)$ as desired. □

Our first main result (proved in Section 2) shows that deterministic—and even randomised—query complexity cannot be condensed losslessly in general. (Below, the notation $\tilde{O}(k)$ hides $\mathrm{poly}(\log k)$ factors.)

Theorem 1. *There exists an n-bit function f with deterministic (or randomised) query complexity k (in fact, $k \geq n^{\Omega(1)}$) such that for every partial assignment ρ with $|\rho^{-1}(*)| \leq O(k)$ we have $\mathrm{D}(f|_\rho) \leq \tilde{O}(k^{3/4})$.*

Our incondensable function f must necessarily exhibit a polynomial gap between $\mathrm{D}(f)$ and $\mathrm{s}(f)$. This is because, as discussed above, any function condenses to $\mathrm{s}(f)$ variables. Similarly, our f must also exhibit a gap between $\mathrm{D}(f)$ and $\deg(f)$, another standard complexity measure (defined as the degree of the unique multilinear polynomial $p \in \mathbb{R}[x_1, \dots, x_n]$ that agrees with f on boolean inputs), because any function condenses to $\deg(f)$ variables.[1] Prior work has constructed several examples that have $\mathrm{D}(f)$ simultaneously larger than both $\mathrm{s}(f)$ and $\deg(f)$ [GPW18, ABK16, ABB+17, BHT17, BBG+22]. We are able to prove Theorem 1 for one of these off-the-shelf functions, namely, a function constructed using *cheat sheets* [ABK16].

[1] Consider the polynomial p computing f and pick any monomial m in p of maximal degree $d \coloneqq \deg(f)$. Let ρ assign $*$ to variables in m and arbitrary boolean values to variables not in m. Then the restricted polynomial $p|_\rho$ still contains the monomial m (it cannot get cancelled). Hence $\mathrm{D}(f|_\rho) = \deg(p|_\rho) = d$.

1.2 Deterministic communication: Open problem

Next we study condensation in the standard two-party communication model [KN97, RY20]. Let $\mathrm{D}^{cc}(F)$ denote the deterministic communication complexity of a two-party function $F\colon [N]\times[N]\to\{0,1\}$. Here, Alice receives $x\in[N]$, Bob receives $y\in[N]$, and their goal is to compute $F(x,y)$. We often view F as an N-by-N boolean matrix. We say that F *condenses* to k variables if F can be restricted to a submatrix $R := X\times Y\subseteq[N]\times[N]$ of size $|X|=|Y|=2^k$ such that the resulting function $F|_R$ has maximum communication complexity $\mathrm{D}^{cc}(F|_R)=\Theta(k)$. We ask:

Open Problem 2. Does every F condense losslessly to $\Theta(\mathrm{D}^{cc}(F))$ variables?

Open Problem 2 was first posed by Hatami [Hat22]. The question is connected to the log-rank conjecture [LS88, LS23], which posits that for some universal constant $c\geq 1$, every F has $\mathrm{D}^{cc}(F)\leq \log^c \mathrm{rk}(F)$, where $\mathrm{rk}(F)$ is the rank (over reals) of the matrix F. We note that every F condenses to $\log \mathrm{rk}(F)$ variables because of two basic facts: (1) $\mathrm{D}^{cc}(F)\geq \log \mathrm{rk}(F)$, and (2) any matrix of rank r has an r-by-r submatrix of rank r. In particular, the log-rank conjecture implies that every F condenses to $\mathrm{D}^{cc}(F)^{1/c}$ variables. In concurrent work, Hrubes [Hru24] confirmed this implication: D^{cc} indeed condenses to $\Omega(\mathrm{D}^{cc}(f)^{1/2})$ variables.

We conjecture, in analogy to query complexity, that D^{cc} cannot be condensed losslessly in general, that is, that the answer to Open Problem 2 is negative. A popular approach to constructing counterexamples to questions like this is to start with an analogous counterexample f in query complexity (namely, Theorem 1) and then applying a *lifting theorem* [RM99, GPW18, CKLM19] to convert f into a two-party function $F := f\circ g$ obtained from f by composing it with a small two-party gadget g. However, by extending previous expander-based techniques, we show as a bonus result (Section 5) that this approach *fails*: Under mild technical assumptions, every function $f\circ g$ indeed condenses losslessly for D^{cc} (even if f did not for D).

Theorem 3 (Informal; see Section 5 for details)**.** *Let g be the inner-product gadget over $b := \Theta(\log^2 n)$ bits. For every n-bit boolean function f with $\mathrm{D}(f)\geq n^{\Omega(1)}$, the composed function $F := f\circ g$ condenses losslessly: there exists an 2^k-by-2^k submatrix R such that $k=\Theta(\mathrm{D}^{cc}(F))=\Theta(\mathrm{D}^{cc}(F|_R))=\Theta(\mathrm{D}(f)\cdot b)$.*

1.3 Randomised communication: Positive result

What about bounded-error randomised (public-coin) communication complexity $\mathrm{R}^{cc}(F)$? Can every F be condensed losslessly to a 2^k-by-2^k submatrix R such that $k=\Theta(\mathrm{R}^{cc}(F))=\Theta(\mathrm{R}^{cc}(F|_R))$? It is known that this is impossible in general.

Example (Equality). The *equality* function $\mathrm{EQ}_n\colon[2^n]\times[2^n]\to\{0,1\}$ is defined by $\mathrm{EQ}_n(x,y) := 1$ iff $x=y$. In the private-coin model, it is well known that $\mathrm{R}^{cc}_{\mathrm{priv}}(\mathrm{EQ}_n)=\Theta(\log n)$. Note that for $k=\Theta(\log n)$, every 2^k-by-2^k submatrix $R\subseteq[2^n]\times[2^n]$ of EQ_n is itself a submatrix of EQ_k (perhaps with all-0 rows/columns). Thus $\mathrm{R}^{cc}_{\mathrm{priv}}(\mathrm{EQ}_n|_R)\leq \mathrm{R}^{cc}_{\mathrm{priv}}(\mathrm{EQ}_k)\leq O(\log\log n)$. We conclude that EQ_n cannot be condensed losslessly in the private-coin model. In the public-coin model, a similar counterexample can be obtained by considering the *greater-than* function GT_n for which $\mathrm{R}^{cc}(\mathrm{GT}_n)=\Theta(\log n)$ [BW15, Vio15]. □

Example (Sparse random matrices). A more dramatic counterexample is provided by Hambardzumyan, Hatami, and Hatami [HHH22]. They show that a sparse random 2^n-by-2^n matrix F, of an appropriately chosen density, satisfies $\mathrm{R}^{cc}(F)=\Theta(n^{0.9})$ yet every $2^{n/2}$-by-$2^{n/2}$ submatrix R has $\mathrm{R}^{cc}(F|_R)=O(1)$. This shows that R^{cc} cannot be condensed even with a modest (polynomial) loss in parameters. □

Given the above counterexamples, can we at least show a lossless condensation result for a particular function of interest? We do so for the *sink-of-xor* function, which was recently used by Chattopadhyay, Mande, and Sherif [CMS20] to disprove the log-approximate-rank conjecture of Lee and Shraibman [LS07, LS23]. This conjecture stated that, for some universal constant $c\geq 1$, every F satisfies $\mathrm{R}^{cc}(F)\leq \log^c \mathrm{rk}_{1/3}(F)$ where $\mathrm{rk}_\varepsilon(F)$ is the *ε-approximate rank* defined by (here $\|M\|_\infty := \max_{x,y}|M_{x,y}|$ is the infinity norm)

$$\mathrm{rk}_\varepsilon(F) \;:=\; \min\big\{\mathrm{rk}(M) : M\in\mathbb{R}^{N\times N} \text{ and } \|F-M\|_\infty\leq\varepsilon\big\}.$$

Example (Sink-of-xor). The function $\textsc{Sink} \circ \oplus \colon \{0,1\}^m \times \{0,1\}^m \to \{0,1\}$ maps $(x, y) \mapsto \textsc{Sink}(x \oplus y)$ where $x \oplus y$ is the coordinate-wise xor and $m := \binom{n}{2}$. Chattopadhyay, Mande, and Sherif [CMS20] proved that this function exhibits an exponential gap between R^{cc} and log-approximate-rank:

$$\begin{aligned} \mathrm{R}^{cc}(\textsc{Sink} \circ \oplus) &= \Theta(n), \\ \mathrm{rk}_{1/3}(\textsc{Sink} \circ \oplus) &\leq O(n^4). \end{aligned}$$

□

A follow-up work by Chattopadhyay, Garg, and Sherif [CGS21] studied whether the above counterexample could be improved. They proposed candidates of n-bit functions F with $\mathrm{rk}_{1/3}(F) \leq O(n^3)$ that they conjectured should have $\mathrm{R}^{cc}(F) = \Theta(n)$, although proving this is still open. This would improve over $\textsc{Sink} \circ \oplus$ in two respects: (1) the relative gap between R^{cc} and $\mathrm{rk}_{1/3}$ would be improved from quartic to cubic, and (2) the functions have maximum possible randomised complexity in terms of number of variables.

In our second main result (proved in Section 4), we achieve the improved separation sought by [CGS21], in both respects above, albeit not for their function candidates, but for a lossless condensate of sink-of-xor.

Theorem 4. *There exists a $2^{O(n)}$-by-$2^{O(n)}$ submatrix R such that $F := (\textsc{Sink} \circ \oplus)|_R$ satisfies*

$$\begin{aligned} \mathrm{R}^{cc}(F) &= \Theta(n), \\ \mathrm{rk}_{1/3}(F) &\leq O(n^3). \end{aligned}$$

1.4 New tool: Shearer extractors

The central ingredient in our randomised lower bound for sink-of-xor (in Theorem 4) is a new type of seeded extractor whose output bits satisfy prescribed dependencies across distinct seeds. We call these objects ***Shearer extractors***, named after *Shearer's lemma* in information theory. To our knowledge, this is the first time that seeded extractors are used to prove a communication lower bound. In comparison, *two-source extractors* have been closely connected to two-party communication complexity since their conception [CG88].

A Shearer extractor with n output bits will be defined relative to a set family $\mathcal{S} \subseteq \binom{[m]}{n} := \{S \subseteq [m] : |S| = n\}$. Its seeds correspond to sets in $\mathcal{S}$ and the outputs for a pair of seeds $S, S' \in \mathcal{S}$ share $|S \cap S'|$ many bits. To define these extractors formally, we recall the following standard notions [Vad12, §6]: The *min-entropy* of a random variable $\boldsymbol{X}$ is defined as $\mathrm{H}_\infty(\boldsymbol{X}) := \min_x \log(1/\Pr[\boldsymbol{X} = x])$. We say $\boldsymbol{X}$ is a k-source if $\mathrm{H}_\infty(\boldsymbol{X}) \geq k$. We use $\Delta(\boldsymbol{X}, \boldsymbol{Y}) := \max_E \left|\Pr[\boldsymbol{X} \in E] - \Pr[\boldsymbol{Y} \in E]\right|$ to denote the statistical distance between random variables $\boldsymbol{X}$ and $\boldsymbol{Y}$. In this paper, we define all extractors in the "strong" sense (output is close to uniform even when conditioned on the seed).

Definition 5 (Seeded extractor [Vad12, §6]). We say that a function $\mathrm{Ext} \colon \{0,1\}^t \times [r] \to \{0,1\}^n$ is an (ε, k)-*extractor* if for every k-source $\boldsymbol{X} \in \{0,1\}^t$ and a uniform random $\boldsymbol{Y} \sim [r]$, $\boldsymbol{U} \sim \{0,1\}^n$,

$$\Delta((\boldsymbol{Y}, \mathrm{Ext}(\boldsymbol{X}, \boldsymbol{Y})), (\boldsymbol{Y}, \boldsymbol{U})) \leq \varepsilon.$$

Definition 6 (Shearer extractor). Let $\mathrm{Ext} \colon \{0,1\}^t \to \{0,1\}^m$ be a function and $\mathcal{S} = \{S_1, \ldots, S_r\} \subseteq \binom{[m]}{n}$ a set family. For $i \in [r]$ we write $\mathrm{Ext}(x, i) := \mathrm{Ext}(x)_{S_i}$ for the projection of the output onto coordinates S_i. We say that Ext is an (ε, k)-*Shearer extractor* for $\mathcal{S}$ if $\mathrm{Ext}(\cdot, \cdot)$ is an (ε, k)-extractor.

Every seeded extractor can be viewed as a Shearer extractor defined relative to a *pairwise disjoint* set family. Namely, if $\mathrm{Ext} \colon \{0,1\}^t \times [r] \to \{0,1\}^n$ is a seeded extractor, then $\mathrm{Ext}' \colon \{0,1\}^t \to \{0,1\}^{r \times n}$, defined as the concatenation of $\mathrm{Ext}(x, i)$ over all $i \in [r]$, is a Shearer extractor for the partition of the grid $[r] \times [n]$ into r rows, each of size n.

Example (Limited intersections). Shearer extractors for set families with *limited intersections* can be constructed from seeded extractors. For example, consider the set family $\mathcal{S} := \mathcal{R} \cup \mathcal{C}$ consisting of the rows $\mathcal{R} := \{\{i\} \times [n] : i \in [n]\}$ and columns $\mathcal{C} := \{[n] \times \{j\} : j \in [n]\}$ of an n-by-n grid. We claim that a randomly chosen function $\mathrm{Ext} \colon \{0,1\}^{2n} \to \{0,1\}^{n \times n}$ is a $(0.1, 1.9n)$-Shearer extractor for $\mathcal{S}$. Indeed, standard probabilistic arguments [Vad12, Thm 6.14] show that a random function is a $(0.1, 1.9n)$-Shearer extractor for the pairwise disjoint subfamily $\mathcal{R}$ with high probability. Similarly, a random function is a Shearer extractor for $\mathcal{C}$. By a union bound, we conclude that a random function is *simultaneously* a Shearer extractor for both $\mathcal{R}$ and $\mathcal{C}$ and hence for $\mathcal{S}$. This argument can be used more generally to show the existence of Shearer extractors for any $\mathcal{S}$ that can be partitioned into subfamilies consisting of pairwise disjoint sets. □

Given the above example, the most interesting setting for Shearer extractors is when the sets in $\mathcal{S}$ are *pairwise intersecting*, as in the following example.

Example (Sink family). Let us define a set family $\mathcal{S}_{\text{SINK}}$ underlying the sink function. Let $G = (V, E)$ denote the complete graph with $|V| = n$ nodes and $|E| = \binom{n}{2}$ edges. Define $\mathcal{S}_{\text{SINK}} \coloneqq \{S_1, \dots, S_n\}$ where $S_i \subseteq E$ is the set of edges incident to the i-th node. Note that $|S_i \cap S_j| = 1$ for $i \neq j$. □

Our main technical result (proved in Section 3) shows the existence of a Shearer extractor for $\mathcal{S}_{\text{SINK}}$.

Theorem 7. *For every $\varepsilon > 0$ and sufficiently large n, there exists an $(\varepsilon, (2 - c\varepsilon^6)n)$-Shearer extractor $\text{Ext}\colon \{0,1\}^{2n} \to \{0,1\}^E$ for $\mathcal{S}_{\text{SINK}}$ where $c > 0$ is an absolute constant. Moreover, a randomly chosen function satisfies this with high probability.*

We use this existence result to prove the lower bound for sink-of-xor (in Theorem 4) in Section 4. In particular, starting with a Shearer extractor Ext for $\mathcal{S}_{\text{SINK}}$, we prove Theorem 4 for the submatrix $R \coloneqq X \times Y$ where $X = Y = \text{Ext}(\{0,1\}^{2n})$ is the image of the extractor.

The proof of Theorem 7 is a surprisingly delicate probabilistic argument, analysing a martingale process featuring—unsurprisingly—an application of Shearer's lemma. While our proof is presented specifically for $\mathcal{S}_{\text{SINK}}$, the argument generalises to any set family that is "well-spread" enough in the sense of Shearer's lemma (formally stated as Lemma 20 in Section 3).

Example (Poorly spread family). We note that for nontrivial Shearer extractors to exist the family needs to be somewhat well spread. Consider $\mathcal{R}' \coloneqq \big\{\{(1,1)\} \cup \{i\} \times [n] : i \in [n]\big\}$ that is obtained from the row family $\mathcal{R}$ (which was discussed above) by including the point $(1,1)$ in every set. Then $\mathcal{R}'$ has pairwise intersections of size 1 but there is no nontrivial Shearer extractor $\text{Ext}\colon \{0,1\}^{2n} \to \{0,1\}^{n \times n}$ relative to $\mathcal{R}'$, as there will always be a $(2n-1)$-source $\boldsymbol{X}$ that fixes the $(1,1)$-th output bit, and hence every $\text{Ext}(\boldsymbol{X})_S$ for $S \in \mathcal{R}'$ is far from uniform. □

1.5 Open problems

We leave the following natural questions open for further work.

(1) Is the gap in Theorem 1 optimal for deterministic query complexity? That is, is there a function for which hardness is even less condensible than that given in Theorem 1? The maximum achievable bound for any function is $O(\text{D}(f)^{1/3})$, because we always have $\deg(f) \geq \Omega(\text{D}(f)^{1/3})$ [Mid04] and every function condenses to $\deg(f)$ variables. (We show in Section 2.3 that $O(\text{D}(f)^{3/4})$ is tight for our function.)

(2) Can unambiguous certificate complexity be condensed losslessly? (Hrubes [Hru24] proved the communication analogue: the log of partition number condenses losslessly.) How about block sensitivity?

(3) In the context of Open Problem 2, prove our conjecture that D^{cc} does not condense losslessly.

(4) Can we prove Theorem 4 for a submatrix $R = X \times Y$ where X and Y are subspaces of $\mathbb{Z}_2^n$? This would yield an even more structured counterexample to the log-approximate-rank conjecture, which would be closer to the function candidates proposed in [CGS21]. To achieve this, it would suffice to prove Theorem 7 for a *linear function*—however, it is already a long-standing open problem to prove the existence of linear seeded extractors with close to optimal seed length [CZ18].

(5) The quantum analogue of the log-approximate-rank conjecture was disproved by [ABT19, SdW19], also using sink-of-xor as a counterexample. Can condensation be used to improve these separations to get closer to the upper bound of Gál and Syed [GS21]?

(6) Are there *explicit constructions* of Shearer extractors for interesting pairwise intersecting set families?

2 Proof of Theorem 1: An incondensable function

In this section, we prove the following theorem.

Theorem 1. *There exists an n-bit function f with deterministic (or randomised) query complexity k (in fact, $k \geq n^{\Omega(1)}$) such that for every partial assignment ρ with $|\rho^{-1}(*)| \leq O(k)$ we have $\mathrm{D}(f|_\rho) \leq \tilde{O}(k^{3/4})$.*

We prove the theorem for deterministic query complexity (the randomised case is analogous). We start by describing the construction of our incondensable function f in Section 2.1, and then we prove Theorem 1 in Section 2.2. Finally, we show that the exponent 3/4 is tight for our function in Section 2.3.

2.1 Construction

Our incondensable function is the cheat sheet version of the usual *tribes* function. To define this precisely, let us first introduce the cheat sheet framework from [ABK16, AKK16]. The definition below uses the standard notion of a *certificate*: for $b \in \{0,1\}$, a b-certificate for a function $f\colon \{0,1\}^n \to \{0,1\}$ is a partial assignment $\rho\colon [n] \to \{0,1,*\}$ such that $f|_\rho$ is a constant function equal to b. A certificate for an input $x \in \{0,1\}^n$ is an $f(x)$-certificate ρ that is consistent with x, that is, x and ρ agree on the non-$*$ coordinates. The *certificate complexity* of x is the least size (number of non-$*$ coordinates, $|\rho^{-1}(\{0,1\})|$) of a certificate for x. Finally, the *certificate complexity* of f is the maximum certificate complexity over all inputs x.

Definition 8 (Cheat sheets). Let $g\colon \{0,1\}^N \to \{0,1\}$ be an N-bit function. Suppose its certificate complexity is k and let $c \coloneqq 10 \log N$ and $m \coloneqq k \log N$. We define the *cheat sheet* version of g as a function

$$g_{\mathrm{CS}}\colon (\{0,1\}^N)^c \times (\{0,1\}^{cm})^{2^c} \to \{0,1\}.$$

The input to g_{CS} is a string $(x, \mathsf{C}) = (x_1, \dots, x_c, \mathsf{C}_1, \dots, \mathsf{C}_{2^c})$ where $x_i \in \{0,1\}^N$ are inputs to g, and $\mathsf{C}_j \in \{0,1\}^{cm}$ are called *cells*. Let $\ell_i \coloneqq g(x_i)$ and $\ell \in [2^c]$ be the positive integer corresponding to the binary string $\ell_1, \dots, \ell_c$. We define $g_{\mathrm{CS}}(x, \mathsf{C}) \coloneqq 1$ iff the cell C_ℓ contains, for each $i \in [c]$, a binary encoding of a certificate ρ for x_i. Specifically, ρ is encoded as a binary string of length $m = k \log N$ that encodes the set $\rho^{-1}(\{0,1\})$ as a list of k numbers, each using $\log N$ bits.

Lemma 9 ([ABK16, Lemma 6])**.** *Suppose that $g\colon \{0,1\}^N \to \{0,1\}$ has deterministic (resp. randomised) query complexity k, then g_{CS} has deterministic (randomised) query complexity $\Omega(k)$.*

We now instantiate the cheat sheet framework for the function $\textsc{Tribes}_n\colon \{0,1\}^{n \times n} \to \{0,1\}$ given by

$$\textsc{Tribes}_n(x) \coloneqq \bigwedge_{i=1}^{n} \bigvee_{j=1}^{n} x[i,j],$$

where we treat the input x as an n-by-n boolean matrix and write $x[i,j]$ for the j-th bit in the i-th row. Thus $\textsc{Tribes}_n(x) = 1$ iff every row contains a 1-entry. It is well known and easy to see that this function has deterministic query complexity $\mathrm{D}(\textsc{Tribes}_n) = n^2$ and certificate complexity n.

Consider the cheat sheet function $f \coloneqq (\textsc{Tribes}_n)_{\mathrm{CS}}$ mapping $(\{0,1\}^{n^2})^c \times (\{0,1\}^{c \cdot m})^{2^c} \to \{0,1\}$ where $c \coloneqq 10 \log(n^2)$ and $m \coloneqq n \log(n^2)$. We have $\mathrm{D}(f) = \tilde{\Theta}(n^2)$ where the lower bound is from Lemma 9 and the upper bound is straightforward. We adopt the following conventions about how the input (x, C) encodes certificates (see Figure 1).

- The cn^2 bits of x describe c instances $x_1, \dots, x_c$ of $\textsc{Tribes}_n$, each an n-by-n boolean matrix.
- We have 2^c different cells C_ℓ. The ℓ-th cell C_ℓ contains encodings of c certificates: the j-th of them claims to be a ℓ_j-certificate for x_j. We interpret each certificate as an element of $[n]^n$, that is, a list of n *pointers*. We denote the k-th pointer of the j-th certificate in C_ℓ by $\mathsf{C}_{\ell,j,k} \in [n]$.
- We have $f(x, \mathsf{C}) = 1$ iff there exists cell $\ell \in [2^c]$ such that for each $j \in [c]$:
  - If $\ell_j = 1$, then C_ℓ contains a 1-certificate for x_j, that is, $x_j[t, \mathsf{C}_{\ell,j,t}] = 1$ for all $t \in [n]$. In words, we require the t-th pointer of the certificate to point to some 1-entry in the t-th row of x_j.
  - If $\ell_j = 0$, then C_ℓ contains a 0-certificate for x_j, that is, $x_j[\mathsf{C}_{\ell,j,1}, t] = 0$ for all $t \in [n]$. In words, the first pointer in the certificate indicates an all-0 row (we ignore the remaining $n-1$ pointers).

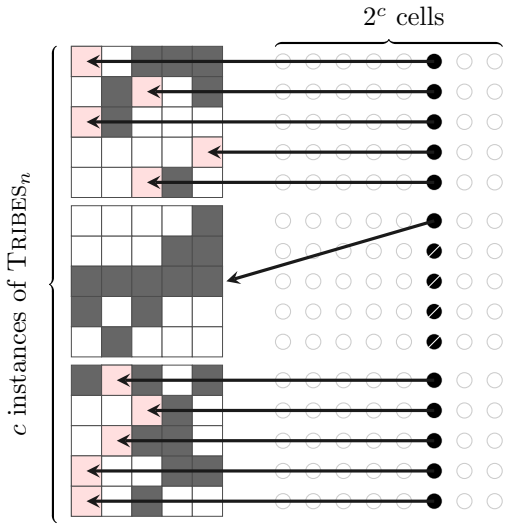

Figure 1: Function $f \coloneqq (\textsc{Tribes}_n)_{\text{CS}}$. The 1-entries of the n-by-n matrices x_j are drawn as white, and the 0-entries are drawn as black. The arrows illustrate the pointers encoded inside a single cell C_ℓ that certifies $\textsc{Tribes}_n(x_1) = 1$, $\textsc{Tribes}_n(x_2) = 0$, and $\textsc{Tribes}_n(x_3) = 1$.

2.2 Proof of Theorem 1

We now prove Theorem 1 for the function $f \coloneqq (\textsc{Tribes}_n)_{\text{CS}}$ that has $\mathrm{D}(f) = \tilde{\Theta}(n^2)$. Our goal is to show for every partial assignment $\rho\colon [n^2c + m2^c] \to \{0, 1, *\}$ with $|\rho^{-1}(*)| \le \tilde{O}(n^2)$ that $\mathrm{D}(f|_\rho) \le \tilde{O}(n^{3/2})$.

First, we make some simplifying assumptions about $S \coloneqq \rho^{-1}(*)$. We may assume that S contains all the $n^2c = \tilde{O}(n^2)$ bits of x; if not, we can modify S to include these bits since this can only increase $\mathrm{D}(f|_\rho)$. Similarly, for each pointer $\mathsf{C}_{\ell,j,k} \in [n]$ encoded using $\log n$ bits, we may assume a dichotomy: either S contains all the $\log n$ bits of that pointer or none at all. To ensure this dichotomy, for every pointer that S contains some bits, we may include all the $\log n$ bits of the pointer in S. This modification increases the size of S by at most a $\log n$ factor, so we still have $|S| = \tilde{O}(n^2)$.

To show $\mathrm{D}(f|_\rho) \le \tilde{O}(n^{3/2})$, we prove the following key lemma.

Lemma 10. *For every $j \in [c]$ there is an $\tilde{O}(n^{3/2})$-query algorithm $\mathcal{A}_j$ outputting* True/False *such that*

- *If $\mathcal{A}_j$ outputs* True*, then $\textsc{Tribes}_n(x_j) = 1$.*
- *If $\mathcal{A}_j$ outputs* False*, then no cell C_ℓ with $\ell_j = 1$ contains a 1-certificate for x_j.*

It is easy to show $\mathrm{D}(f|_\rho) \le \tilde{O}(n^{3/2})$ using Lemma 10. Indeed, we run $\mathcal{A}_j$ for each $j \in [c]$ and define $\ell_j \coloneqq 1$ if $\mathcal{A}_j$ outputs True, and otherwise we define $\ell_j \coloneqq 0$. Then the only cell which might contain certificates consistent with its index is $\ell \coloneqq \ell_1, \ldots, \ell_c$. We then check all the certificates in the ℓ-th cell and output 1 iff they are all correct. We now proceed to prove Lemma 10.

2.2.1 Simple special case

Before we tackle Lemma 10 in full generality, let us focus on a particular simple special case that illustrates the key ideas in our algorithm. For convenience, let us write $\mathsf{C}_{\ell,j,k} \in [n] \cup \{*\}$ for the *current state* of a pointer, where $\mathsf{C}_{\ell,j,k} = *$ if the pointer is not fixed by ρ and not yet queried by our algorithm. The following is our simplifying assumption.

(†) *Suppose that, at start, the first n cells contain only free pointers, and the remaining cells contain only fixed pointers. That is, $\mathsf{C}_{\ell,j,k} = *$ for all $\ell \le n$, and $\mathsf{C}_{\ell,j,k} \ne *$ for all $\ell > n$.*

Our algorithm $\mathcal{A}_j$ consists of two sub-procedures that will separately handle free cells and fixed cells.

Algorithm 1 Useful for cells with many **free** pointers. In case (†), the algorithm is invoked with $C = [n]$.

Input: A set of cells $C \subseteq [2^c]$, index $j \in [c]$, and $\tau \in ([n] \cup \{*\})^n$ (initialised to $\tau = *^n$ by default)
Output: Either $\tau \in [n]^n$, which is a 1-certificate for x_j, or $\perp$, if no cell in C contains such a certificate.

while there exist a cell $\ell \in C$ and a row $k \in [n]$ with $\tau_k = *$ **do**
  Query $t \leftarrow \mathsf{C}_{\ell,j,k} \in [n]$ and $x_j[k,t] \in \{0,1\}$
  if $x_j[k,t] = 1$ **then** $\tau_k \leftarrow t$ ▷ Successful
  else $C \leftarrow C \setminus \{\ell\}$ ▷ Unsuccessful: Eliminate the cell C_ℓ
return τ if it has no $*$'s and $\perp$ otherwise

Free cells (Algorithm 1). Say $j = 1$ for simplicity. The goal of Algorithm 1 is to check if one of the cells $\mathsf{C}_1, \dots \mathsf{C}_n$—which contain only free pointers under (†)—encodes a 1-certificate for x_1. For Lemma 10 it would be enough to check only the *relevant* cells C_ℓ with $\ell_1 = 1$, but for simplicity we design the algorithm to check every cell, even those C_ℓ with $\ell_1 = 0$. Thus, we consider each of the cells $\mathsf{C}_1, \dots \mathsf{C}_n$ in sequence, and for each C_ℓ find either a 1-certificate for x_1 or conclude that C_ℓ does not contain a 1-certificate for x_1. Recall that each (relevant) cell purports to contain n pointers picking out a 1-entry in each row of x_1. Whenever we query a pointer $\mathsf{C}_{\ell,1,k} \in [n]$ (by querying the $\log n$ underlying bits), we include a verification step that queries $x_1[k, \mathsf{C}_{\ell,1,k}] \in \{0,1\}$ to check that the pointer indeed pointed to a 1-entry in the k-th row. We call such a verification query *successful* if $x_1[k, \mathsf{C}_{\ell,1,k}] = 1$ and *unsuccessful* otherwise.

A naive algorithm would query n pointers in each $\mathsf{C}_1, \dots, \mathsf{C}_n$ making $\tilde{\Theta}(n^2)$ queries in total, but this is way more than our claimed budget of $\tilde{O}(n^{3/2})$. The idea in Algorithm 1 is that every time we make a successful query to row k in x_1, we do not need to revisit row k in subsequent iterations as we already know it contains a 1-entry. In each iteration, we thus accumulate information (stored in $\tau \in ([n] \cup \{*\})^n$) about rows in x_1 that contain a 1-entry, or we possibly encounter a cell C_ℓ with an unsuccessful pointer for row k, in which case we may eliminate the cell C_ℓ (it cannot contain a 1-certificate) and move to the next cell $\mathsf{C}_{\ell+1}$.

In the end, we may eliminate all $\mathsf{C}_1, \dots \mathsf{C}_n$ in which case we can safely output $\perp$, or we have identified a 1-entry in each row, in which case we can output the certificate $\tau \in [n]^n$. (Note that, in the latter case, we may not have identified a specific cell C_ℓ containing τ, but this was not required by Lemma 10.)

Finally, we claim the query cost of Algorithm 1 is $\tilde{O}(n + |C|)$, which equals $\tilde{O}(n)$ when run on the free cells $C := [n]$. This follows because every time we query a pointer, we either decrease the number of $*$'s in τ (successful case), which can happen n times, or we have eliminated a new cell in C (unsuccessful case), which can happen $|C|$ times.

Fixed cells (Algorithm 2). The goal of Algorithm 2 is to either find a 1-entry in each row of x_j, or eliminate as many cells in C as possible. In our special case (†), when we run the algorithm on the

Algorithm 2 Useful for cells with many **fixed** pointers. In case (†), the algorithm is invoked with $C = [2^c] \setminus [n]$; note that the property (1) is always true: *every* cell in C contains a fixed pointer into row k.

Input: A set of cells $C \subseteq [2^c]$, index $j \in [c]$.
Output: Either $\tau \in [n]^n$, which is a 1-certificate for x_j, or $\perp$

$\tau \leftarrow *^n$
while $C \neq \emptyset$ and there exists a row $k \in [n]$ with $\tau_k = *$ that satisfies the following property:

At least half of the cells in C contain a fixed pointer into the k-th row of x_j (1)

do
  $t \leftarrow$ the most popular value among $\mathsf{C}_{\ell,j,k}$ for $\ell \in C$ with $\mathsf{C}_{\ell,j,k} \neq *$
  $T \leftarrow \{\ell \in C \mid \mathsf{C}_{\ell,j,k} = t\}$
  Query $x_j[k,t] \in \{0,1\}$
  if $x_j[k,t] = 1$ **then** $\tau_k \leftarrow t$ ▷ Successful
  else $C \leftarrow C \setminus T$ ▷ Unsuccessful: Eliminate all the cells in T
return τ if it has no $*$'s and $\perp$ otherwise

cells $C \coloneqq [2^c] \setminus [n]$, we claim that

1. Property (1) of the *while*-loop is always satisfied.
2. If the algorithm outputs $\perp$, then it must have eliminated all cells, that is, no cell in C contains a 1-certificate for x_j.

The first claim is true simply because every (relevant) cell in $C = [2^c] \setminus [n]$ contains only fixed pointers, so *all* of them (not just half) will contain a fixed pointer into every row of x_j. The second claim is true since every iteration of the *while*-loop either decreases the number of $*$'s in τ (successful case) or shrinks the set C (unsuccessful case). Hence, if the output is $\perp$, we must have terminated with $C = \emptyset$.

How fast does the set C shrink? Note that for each row $k \in [n]$, there are n possible columns $v \in [n]$ for the pointer into row k to point. Hence there exists a *popular* column $v^* \in [n]$ such that at least $1/n$-fraction of cells in C point to the matrix entry (k, v^*). The algorithm queries such a popular entry. In case of an unsuccessful query, we may now eliminate $1/n$ fraction of cells in C. After at most $n + n \log |C|$ many iterations we must either have found a 1-entry in each row, or we have eliminated all cells in C. We have $n \log |C| = \tilde{O}(n)$ which shows that Algorithm 2 terminates in $\tilde{O}(n)$ queries.

Algorithm $\mathcal{A}_j$. Our final algorithm for the simplified case (†) is to first run Algorithm 1 on the free cells $C = [n]$ (and $\tau = *^n$) and then run Algorithm 2 on the remaining cells $C = [2^c] \setminus [n]$. We output 1 iff one of them outputs a certificate. The query cost of the algorithm is only $\tilde{O}(n)$.

2.2.2 General case

We now remove the simplifying assumption (†). In the general case, every cell may have a *mixture* of free and fixed pointers. The final algorithm is given as Algorithm 3. It first runs Algorithm 2 on all cells $C = [2^c]$. This run terminates after $\tilde{O}(n)$ queries for a similar reason as in the simple case (†): each unsuccessful query shrinks the set C by a factor of $1/(2n)$ rather than $1/n$ as in case (†) (the extra factor $1/2$ stems from property (1)). The following claim captures the key property of Algorithm 2 at termination.

Claim 11. *Let C and τ be as on line (§) of Algorithm 3. Denote by $L \coloneqq \{k \in [n] \mid \tau_k = *\}$ the set of rows for which we have not yet found 1-entries. Then $|C| \leq \tilde{O}(n^2/|L|)$.*

Proof. Assume $C \neq \emptyset \neq L$ as otherwise the claim is vacuously true. Thus the algorithm terminated because property (1) failed. This means that, for each $k \in L$, at least half of C contain a free pointer to row k. Hence there are at least $|L||C|/2$ free pointers remaining. But there can be at most $|S| \leq \tilde{O}(n^2)$ free pointers altogether and thus $|L||C|/2 \leq \tilde{O}(n^2)$, as desired. □

We finish the analysis depending on the size of L. If $|L| \leq \sqrt{n}$, then Algorithm 3 queries all the remaining rows L, making $n|L| \leq O(n^{3/2})$ queries, to determine $\textsc{Tribes}_n(x_j)$. Suppose then $|L| > \sqrt{n}$. In this case, Claim 11 implies that $|C| \leq \tilde{O}(n^{3/2})$. Hence when we finish by running Algorithm 1, it will terminate after $\tilde{O}(n + |C|) = \tilde{O}(n^{3/2})$ many queries. In all cases, the total query cost is $\tilde{O}(n^{3/2})$. This proves Lemma 10 and hence concludes the proof of Theorem 1.

Algorithm 3 The final algorithm for Lemma 10.

Input: $j \in [c]$.
Output: Either $\tau \in [n]^n$, which is a 1-certificate for x_j, or $\perp$, if no cell contains such a certificate.

```
C ← [2^c]
τ ← Algorithm 2 on (C, j)
if τ ≠ ⊥ then return τ
Update the values of C and τ with the values at the termination of Algorithm 2   (§)
L ← {k ∈ [n] | τ_k = *}
if |L| ≤ √n then
    Query x_j[k, t] for all k ∈ L and t ∈ [n], set τ accordingly
    return τ if it has no *'s and ⊥ otherwise
Run Algorithm 1 on (C, j, τ) and output its result
```

Remark 12. We note that the above proof yields a slightly stronger statement, showing that our function is robust: If we restrict all but $n^{2+2\varepsilon}$ variables, then the deterministic query complexity is at most $\tilde{O}(n^{3/2+\varepsilon})$.

2.3 Tightness of our analysis

Finally—as a bonus—we show here that Theorem 1 is optimal (up to log factors) for our function $f = (\textsc{Tribes}_n)_{\text{CS}}$. Namely, we show how to restrict f to a set S of $\tilde{O}(n^2)$ variables and assign values $\rho\colon \bar{S} \to \{0,1\}$ to all other variables, so that the restricted function $f|_\rho$ has query complexity $\Omega(n^{3/2})$.

Theorem 13. *There is a set of variables S of f with $|S| = \tilde{\Theta}(n^2)$ and an assignment $\rho\colon \bar{S} \to \{0,1\}$ to all other variables, and so that the restricted function $f|_\rho$ has query complexity $\Omega(n^{3/2})$.*

Our proof strategy is to find a ρ such that $f|_\rho$ becomes "isomorphic" to a cheat sheet version of a function g of query complexity $\mathrm{D}(g) = n^{3/2}$. A technical detail is that the standard cheat sheet construction g_{CS} (Definition 8) of a function of query complexity $\mathrm{D}(g) = n^{3/2}$ can have $n^{10\times 3/2}$ variables, while we are only allowed to have $\tilde{O}(n^2)$. Our proof starts by optimising this construction.

Proof. Let us define a *succinct* cheat sheet version of a function $g\colon \{0,1\}^N \to \{0,1\}$ of certificate complexity k. Let $c \coloneqq \log N$ and $m \coloneqq k \log N$. Then define $g^*_{\text{CS}}\colon (\{0,1\}^N)^c \times (\{0,1\}^{cm})^{2^c} \to \{0,1\}$ according to the Definition 8—the only change here is that we take $c = \log N$ instead of $c = 10 \log N$ as in the original construction. Then we still claim that $\mathrm{D}(g^*_{\text{CS}}) = \Omega(\mathrm{D}(g))$: the original proof [ABK16, Lemma 21] works even under this optimised choice of c.

Consider a "skewed" tribes $g\colon \{0,1\}^{\sqrt{n}\times n} \to \{0,1\}$ given by $g(y) \coloneqq \bigwedge_{i\in[\sqrt{n}]} \bigvee_{j\in[n]} y[i,j]$. We have that $\mathrm{D}(g) = n^{3/2}$ similarly as for the original tribes function. We then use the succinct cheat sheet from the above paragraph to construct g^*_{CS}. In particular, we encode 1-certificates of g as a list of $k \coloneqq \sqrt{n}$ pointers (picking a 1-entry in each row), and 0-certificates of g as a single pointer (picking out an all-0 row). Thus g^*_{CS} features $n^{3/2}$ cells, each encoding $\tilde{O}(\sqrt{n})$ pointers. In summary, g^*_{CS} has $\tilde{O}(n^2)$ variables and query complexity $\mathrm{D}(g^*_{\text{CS}}) = \Omega(n^{3/2})$.

Our goal will now be to construct a partial assignment ρ for f such that $f|_\rho$ becomes *isomorphic* to g^*_{CS}, that is, there is a 1-to-1 correspondence between the variables of $f|_\rho$ and g^*_{CS} such that the two functions are equivalent. This would show $\mathrm{D}(f|_\rho) = \mathrm{D}(g^*_{\text{CS}}) = \Omega(n^{3/2})$, as desired.

To define ρ, recall that the input of f consists of c instances $x_1, \dots, x_c$ of $n \times n$ tribes together with cells $\mathsf{C}_1, \dots, \mathsf{C}_{2^c}$. The assignment ρ will leave $\sqrt{n} \times n$ free variables in each of the first $c' = (3/2)\log n$ instances and fix all variables in the remaining $c - c'$ instances. We will also include $\tilde{O}(\sqrt{n})$ free pointers in each of the $2^{c'} = n^{3/2}$ many cells indexed by $F \coloneqq \{0,1\}^{c'} \times \{1\}^{c-c'}$. The remaining cells will be fully fixed.

Specifically, we make ρ fix all bits for instances x_j, $j \in [c] \setminus [c']$, to 1 so that $\textsc{Tribes}(x_j) = 1$. All bits in cells not in F we can fix arbitrarily (these cells have become irrelevant). Moreover, in the first c' instances x_j we fix all entries in the bottom-most rows $[n] \setminus [\sqrt{n}]$ to 1. Correspondingly, we arbitrarily fix for every cell in F the pointers corresponding to the last $c - c'$ instances and to the bottom-most rows of the first c' instances (note that all these fixed pointers will automatically point to 1-entries). It is now straightforward to check that f_ρ is isomorphic to g^*_{CS}. □

Remark 14. We end this with the note that the Theorem 13 is slightly stronger. We could use $n^{3/2+\varepsilon}$ free cells, each with $n^{1/2+\varepsilon}$ pointers (hence a total of $\tilde{O}(n^{2+2\varepsilon})$ variables), and show by the same argument as above that the query complexity is $\Omega(n^{3/2+\varepsilon})$ making Remark 12 tight.

3 Proof of Theorem 7: Existence of Shearer extractors

In this section, we prove Theorem 7, restated here for convenience.

Theorem 7. *For every $\varepsilon > 0$ and sufficiently large n, there exists an $(\varepsilon, (2 - c\varepsilon^6)n)$-Shearer extractor $\mathrm{Ext}\colon \{0,1\}^{2n} \to \{0,1\}^E$ for $\mathcal{S}_{\textsc{Sink}}$ where $c > 0$ is an absolute constant. Moreover, a randomly chosen function satisfies this with high probability.*

3.1 Overview of proof: Key concentration lemma

Our proof follows the textbook existence proof for extractors [Vad12, Thm 6.14], but featuring a novel concentration lemma for the sink family (Lemma 16), which we spend the rest of the section proving.

Let $\mathrm{Ext}\colon \{0,1\}^{2n} \to \{0,1\}^{\binom{n}{2}}$ be chosen uniformly at random. We will show that whp over this choice, for uniform $\boldsymbol{Y} \sim [n]$, $\boldsymbol{U} \sim \{0,1\}^{n-1}$, and every k-source $\boldsymbol{X}$ where $k := (2 - c\varepsilon^6)n$ (here $c > 0$ is fixed later),

$$\Delta\left((\boldsymbol{Y}, \mathrm{Ext}(\boldsymbol{X}, \boldsymbol{Y})), (\boldsymbol{Y}, \boldsymbol{U})\right) \le \varepsilon.$$

In proving this, we may assume that $\boldsymbol{X}$ is *flat*, that is, uniformly distributed over its support (indeed, every k-source is a mixture of flat k-sources [Vad12, Lemma 6.10], and the statistical distance is maximised at a flat source). Using the definition $\Delta(\boldsymbol{X}, \boldsymbol{Y}) := \max_D \left|\Pr[\boldsymbol{X} \in D] - \Pr[\boldsymbol{Y} \in D]\right|$, it is sufficient to show that for every event $D \subseteq [n] \times \{0,1\}^{n-1}$,

$$\left|\Pr[(\boldsymbol{Y}, \mathrm{Ext}(\boldsymbol{X}, \boldsymbol{Y})) \in D] - \Pr[(\boldsymbol{Y}, \boldsymbol{U}) \in D]\right| \le \varepsilon.$$

We call an event $D \subseteq [n] \times \{0,1\}^{n-1}$ *a balanced distinguisher* if $\Pr_{\boldsymbol{U} \sim \{0,1\}^{n-1}}[(i, \boldsymbol{U}) \in D] = 1/2$ for all $i \in [n]$. Let $D_i := \{x \in \{0,1\}^{n-1} \mid (i,x) \in D\}$ for $i \in [n]$. The following helper lemma (proved in Section 3.3) allows us to only consider such balanced distinguishers wlog.

Lemma 15. *Let* $\mathrm{Ext}\colon \{0,1\}^t \to \{0,1\}^{\binom{n}{2}}$ *be an arbitrary function. Suppose that for a source* $\boldsymbol{X} \in \{0,1\}^t$*, and uniform* $\boldsymbol{Y} \sim [n]$*,* $\boldsymbol{U} \sim \{0,1\}^{n-1}$ *we have* $\Delta\left((\boldsymbol{Y}, \mathrm{Ext}(\boldsymbol{X}, \boldsymbol{Y})), (\boldsymbol{Y}, \boldsymbol{U})\right) \ge \varepsilon$*. Then there exists a balanced distinguisher* $D \subseteq [n] \times \{0,1\}^{n-1}$ *such that* $\left|\Pr[(\boldsymbol{Y}, \mathrm{Ext}(\boldsymbol{X}, \boldsymbol{Y})) \in D] - 1/2\right| \ge \varepsilon/2$.

There are $\binom{2^{2n}}{2^k}$ flat sources of min-entropy k, and at most $2^{n \cdot 2^{n-1}}$ balanced distinguishers. Our proof proceeds by a union bound over all such source–distinguisher pairs: We only need to show that a random Ext fools a given pair with sufficiently high probability.

More formally, let $\boldsymbol{X}$ and D be some fixed pair of a source and a balanced distinguisher. Let $\mathcal{X}$ be the support of $\boldsymbol{X}$. Then by Lemma 15 it suffices to show that for $\boldsymbol{Y} \sim [n]$ we have $\left|\Pr[(\boldsymbol{Y}, \mathrm{Ext}(\boldsymbol{X}, \boldsymbol{Y})) \in D] - 1/2\right| \le \varepsilon/2$, with high probability over the choice of Ext. That is equivalent to bounding from above

$$\Pr_{\mathbf{Ext}}\left[\frac{1}{n \cdot |\mathcal{X}|} \sum_{x \in \mathcal{X}} \sum_{i \in [n]} [\![\mathbf{Ext}(x, i) \in D_i]\!] \notin (-\varepsilon, \varepsilon)\right].$$

Here we use $[\![P]\!] \in \{\pm 1\}$ for a proposition P to denote $+1$ if P is true and -1 if P is false. In Section 3.2 we prove the following concentration lemma, which is the crux of our proof. This is the only point in the present proof that relies on the structure of $\mathcal{S}_{\textsc{Sink}}$.

Lemma 16. *Assume (i)–(ii). Let* $\boldsymbol{T}^1, \dots, \boldsymbol{T}^N \sim \{0,1\}^{\binom{n}{2}}$ *be uniform and independent. Then for* $\varepsilon > 0$*,*

$$\Pr\left[\left|\sum_{j \in [N]} \sum_{i \in [n]} [\![\boldsymbol{T}^j_{S_i} \in D_i]\!]\right| \ge \varepsilon N n\right] \le \exp(-\Omega(\varepsilon^6 n N)).$$

Using Lemma 16 we can conclude the proof of Theorem 7. Indeed, applying the above with $N := |\mathcal{X}| = 2^k$ and $\boldsymbol{T}^x := \mathbf{Ext}(x)$, a union bound says that we fail to fool some source–distinguisher pair with probability at most

$$\begin{aligned}\binom{2^{2n}}{2^k} 2^{n \cdot 2^{n-1}} \cdot \exp(-\Omega(\varepsilon^6 2^k n)) &\le \left(\frac{2^{2n} e}{2^k}\right)^{2^k} 2^{n \cdot 2^{n-1}} \exp(-\Omega(\varepsilon^6 2^k n)) \\ &\le \exp(O(2^k \cdot (2n - k) + n \cdot 2^{n-1}) - \Omega(\varepsilon^6 2^k n)).\end{aligned}$$

Since with $\varepsilon \ge 1$ the theorem is trivial, we may assume that $\varepsilon < 1$ and $c < 1/2$. Then $k = n(2 - c\varepsilon^6) > 1.5n$, so $2^k = \omega(2^{n-1} \cdot n)$. Then letting c_1 be the constant in the big-O and c_2 be the constant in the big-Ω, the probability of failure is

$$\exp(2^k \cdot (c_1(2n - k) - c_2(\varepsilon^6 n))) = \exp(2^k \cdot n\varepsilon^6(c_1 \cdot c - c_2)).$$

By choosing $c < c_2/c_1$ we establish that the probability is $o(1)$. This proves Theorem 7.

3.2 Proof of Lemma 16

We start by informally explaining two reductions that help us identify the technical core of the proof. We use the following notation throughout.

(*i*) $\mathcal{S}_{\text{SINK}} = \{S_1, \dots, S_n\}$.
(*ii*) $D \coloneqq \bigcup_{i\in[n]}\{i\} \times D_i$ where $D_i \subseteq \{0,1\}^{n-1}$, $|D_i| = 2^{n-1}/2$, is a balanced distinguisher.

First reduction. Lemma 16 is a concentration inequality for the sum of random variables $[\![\boldsymbol{T}^j_{S_i} \in D_i]\!]$. Notice that the variables with distinct values of $j \in [N]$ are independent, so the main challenge is to show, for a fixed j, that $\sum_{i\in[n]}[\![\boldsymbol{T}^j_{S_i} \in D_i]\!]$ is concentrated around 0. Lemma 16 then follows via a Chernoff bound. Hence our first reduction is to the following.

Lemma 17. *Assume (i)–(ii). For every $\delta > 0$,*

$$\Pr_{\boldsymbol{X}\sim\{0,1\}^{\binom{n}{2}}}\left[\left|\sum_{i\in[n]}[\![\boldsymbol{X}_{S_i} \in D_i]\!]\right| \geq \delta n\right] \leq \exp(-\Omega(\delta^5 n)).$$

Second reduction. Establishing the concentration in Lemma 17 is tricky as the sets $S_i \in \mathcal{S}_{\text{SINK}}$ intersect one another, and consequently the events $\boldsymbol{X}_{S_i} \in D_i$ are not even pairwise independent. For a concrete example, suppose $S_1 \cap S_2 = \{a\}$ where $a \in [\binom{n}{2}]$, and consider balanced events D_1, D_2 defined by $x_{S_1} \in D_1 \Leftrightarrow x_{S_2} \in D_2 \Leftrightarrow x_a = 1$. Then $[\![\boldsymbol{X}_{S_1} \in D_1]\!] = [\![\boldsymbol{X}_{S_2} \in D_2]\!]$ with probability 1.

It is intuitively plausible, however, that for a well-spread set family such effects are local. To formalise this intuition, we pick $\boldsymbol{K} \sim \binom{[n]}{\alpha n}$ for a small constant α and show that $\sum_{i\in\boldsymbol{K}}[\![\boldsymbol{X}_{S_i} \in D_i]\!]$ is concentrated around 0. Considering only a handful of the sets S_i helps to treat the associated variables as more independent. Having shown such concentration for small random sets, it is easy to prove Lemma 17: we pick a uniformly random partition of $[n]$ into $1/\alpha$ sets of size αn and apply the concentration to each of them. Thus, the key concentration result we need here is the following.

Lemma 18. *Assume (i)–(ii). For every $\varepsilon, \delta > 0$ such that $\varepsilon < \delta^3/100$, and an integer $\ell \leq \varepsilon n$,*

$$\Pr_{\boldsymbol{K}\sim\binom{[n]}{\ell};\boldsymbol{X}\sim\{0,1\}^{\binom{n}{2}}}\left[\left|\sum_{i\in\boldsymbol{K}}[\![\boldsymbol{X}_{S_i} \in D_i]\!]\right| \geq \delta\ell\right] \leq \exp(-\Omega(\delta^2\ell)).$$

The plan for the rest of the section is as follows. In Section 3.2.1 we introduce the tools we need to prove Lemma 18, namely Shannon entropy and martingales; in Section 3.2.2 we prove Lemma 18; in Section 3.2.3 we prove the second reduction by deriving Lemma 17 from Lemma 18; finally, in Section 3.2.4 we prove the first reduction thereby finishing the proof of Lemma 16.

3.2.1 Tools

The usual Shannon entropy of a random variable $\boldsymbol{X}$ with support $\mathcal{X}$ is defined by

$$\mathrm{H}(\boldsymbol{X}) \coloneqq \sum_{x\in\mathcal{X}} \Pr[\boldsymbol{X} = x] \log \frac{1}{\Pr[\boldsymbol{X} = x]}.$$

We will use the following result showing that if a random variable has nearly maximum possible entropy, then it is close to uniform. This result appears in [Gav16] and was used (in a slightly different form) in [CMS20]. We include their proof for completeness.

Lemma 19. *Suppose a variable $\boldsymbol{X} \in \mathcal{X}$ has $\mathrm{H}(\boldsymbol{X}) \geq \log|\mathcal{X}| - \varepsilon$. Then $\Delta(\boldsymbol{X}, \boldsymbol{U}) \leq \sqrt{\varepsilon}$ where $\boldsymbol{U} \sim \mathcal{X}$.*

Proof. The proof is a simple corollary of Pinsker's inequality. For two random variables $\boldsymbol{X}$ and $\boldsymbol{Y}$ with the support $\mathcal{X}$, the Kullback–Leibner divergence is defined as

$$\mathrm{D}_{\mathrm{KL}}(\boldsymbol{X}\|\boldsymbol{Y}) \coloneqq \sum_{x\in\mathcal{X}} \Pr[\boldsymbol{X} = x] \log \frac{\Pr[\boldsymbol{Y} = x]}{\Pr[\boldsymbol{X} = x]}.$$

Pinsker's inequality (see e.g. [CT05, Lemma 11.6.1]) then states that

$$\mathrm{D}_{\mathrm{KL}}(\boldsymbol{X}\|\boldsymbol{Y}) \geq \frac{1}{2\ln 2}\|\boldsymbol{X}-\boldsymbol{Y}\|_1^2 \geq \Delta(\boldsymbol{X},\boldsymbol{Y})^2.$$

Then we have

$$\Delta(\boldsymbol{X},\boldsymbol{U})^2 \leq \mathrm{D}_{\mathrm{KL}}(\boldsymbol{X}\|\boldsymbol{U}) = \sum_{x\in\mathcal{X}} \Pr[\boldsymbol{X}=x]\left(\log\Pr[\boldsymbol{X}=x] - \log\frac{1}{|\mathcal{X}|}\right) = \log|\mathcal{X}| - \mathrm{H}(\boldsymbol{X}) \leq \varepsilon.$$

□

Naturally, we will also employ Shearer's lemma [CGFS86].

Lemma 20. *Let $\boldsymbol{X}$ be a random variable over Σ^m and let $\boldsymbol{S}$ be a random variable over the subsets of $[m]$ such that $\Pr[i\in\boldsymbol{S}] \geq \mu$ for every $i\in[m]$. Then*

$$\mathrm{H}(\boldsymbol{X}) \leq \frac{1}{\mu}\,\mathbb{E}_{\boldsymbol{S}}\left[\mathrm{H}(\boldsymbol{X}_{\boldsymbol{S}})\right].$$

Our probabilistic proof is based on the concentration inequality for martingales due to Azuma [Azu67]. We assume here that the reader is familiar with the standard conditional expected value $\mathbb{E}[\boldsymbol{X}\mid\boldsymbol{Y}]$.

Definition 21. A sequence $\boldsymbol{X}_1,\ldots,\boldsymbol{X}_n$ is called *a martingale* relative to another sequence $\boldsymbol{Y}_1,\ldots,\boldsymbol{Y}_n$ if, for every $i\in[n]$ we have that $\boldsymbol{X}_i$ is a function of $\boldsymbol{Y}_{\leq i} \coloneqq (\boldsymbol{Y}_1,\ldots,\boldsymbol{Y}_i)$ and

$$\mathbb{E}[\boldsymbol{X}_{i+1}\mid\boldsymbol{Y}_{\leq i}] = \boldsymbol{X}_i.$$

Moreover, we say that a sequence $\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n$ is a *martingale difference sequence* if $\boldsymbol{X}_i \coloneqq \sum_{j\in[i]}\boldsymbol{x}_j$ is a martingale (relative to the same sequence).

Theorem 22 (Azuma's inequality, [Azu67])**.** *Let $\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n$ be a martingale difference sequence (relative to some other sequence) such that $|\boldsymbol{x}_i|\leq c$. Then for any $t>0$*

$$\Pr\left[\left|\sum_{i\in[n]}\boldsymbol{x}_i\right| \geq t\right] \leq 2\exp\left(\frac{-t^2}{2nc^2}\right).$$

3.2.2 Proof of Lemma 18

Lemma 18. *Assume (i)–(ii). For every $\varepsilon,\delta>0$ such that $\varepsilon<\delta^3/100$, and an integer $\ell\leq\varepsilon n$,*

$$\Pr_{\boldsymbol{K}\sim\binom{[n]}{\ell};\boldsymbol{X}\sim\{0,1\}^{\binom{n}{2}}}\left[\left|\sum_{i\in\boldsymbol{K}}[\![\boldsymbol{X}_{S_i}\in D_i]\!]\right| \geq \delta\ell\right] \leq \exp(-\Omega(\delta^2\ell)).$$

Overview. Let $\boldsymbol{X}\sim\{0,1\}^{\binom{n}{2}}$ and $\boldsymbol{K}\sim\binom{[n]}{\ell}$. For convenience, we treat $\boldsymbol{K}$ as an ordered sequence $\boldsymbol{K}=(\boldsymbol{k}_1,\ldots,\boldsymbol{k}_\ell)\in[n]^\ell$ generated by picking $\boldsymbol{k}_{i+1}$ uniformly at random from $[n]\setminus\{\boldsymbol{k}_1,\ldots,\boldsymbol{k}_i\}$. We define

$$\boldsymbol{y}_i \coloneqq [\![\boldsymbol{X}_{S_{\boldsymbol{k}_i}}\in D_{\boldsymbol{k}_i}]\!].$$

Let us write $\boldsymbol{y}_{\leq i}\coloneqq(\boldsymbol{y}_1,\ldots,\boldsymbol{y}_i)\in\{\pm1\}^i$ and $\boldsymbol{k}_{\leq i}\coloneqq(\boldsymbol{k}_1,\ldots,\boldsymbol{k}_i)\in[n]^i$ for short. Our goal is to show that $\sum_{i\in[\ell]}\boldsymbol{y}_i$ concentrates around 0. In the hopes of applying Azuma's inequality, our dream would be that $\boldsymbol{y}_1,\ldots,\boldsymbol{y}_n$ was a martingale difference sequence relative to $(\boldsymbol{y}_i,\boldsymbol{k}_i)$, that is,

$$\mathbb{E}[\boldsymbol{y}_{i+1}\mid\boldsymbol{y}_{\leq i},\boldsymbol{k}_{\leq i}] = 0. \tag{2}$$

Unfortunately, our dream does not come true, because the events $\boldsymbol{X}_{S_{\boldsymbol{k}_i}}\in D_{\boldsymbol{k}_i}$ are dependent, as discussed previously. Fortunately, we are able to use Shearer's inequality to show that the dependencies are limited enough that we can recover a concentration result.

Let us discuss this strategy more formally. First, for $k \in [n]^i$ and $y \in \{\pm 1\}^i$ where $i \in [\ell]$, we define

$$\text{Event } E_y^k \quad \iff \quad \forall j \in [i]\colon [\![\boldsymbol{X}_{S_{k_j}} \in D_{k_j}]\!] = y_j.$$

Our dream condition (2) now becomes equivalent to requiring that, for every $(k, y) \in [n]^i \times \{\pm 1\}^i$,

$$\mathbb{E}_{\boldsymbol{j} \sim [n] \setminus \{k_1, \dots, k_i\}} \left[[\![\boldsymbol{X}_{S_{\boldsymbol{j}}} \in D_{\boldsymbol{j}}]\!] \mid E_y^k \right] = 0. \tag{3}$$

We show below in Lemma 23 that (3) is approximately true: for a *typical* (k, y) the expectation is close to 0. More formally, for $(k, y) \in [n]^i \times \{\pm 1\}^i$, we define

$$(k, y) \text{ is } \textbf{\textit{typical}} \quad \iff \quad \Pr[E_y^k] \geq 2^{-2i}.$$

Let $\mathcal{T}_i$ denote the set of typical pairs for length i. Deriving concentration from Lemma 23 is quite straightforward, so this lemma is the heart of the proof. To prove it, we show, using Shearer's lemma, that under typical E_y^k, the entropy of $\boldsymbol{X}_{S_j}$ is very large for most $j \in [n]$. Then by Lemma 19, we have that $(\boldsymbol{X}_{S_j} | E_y^k)$ is typically close to uniform, so the probability $(\boldsymbol{X}_{S_j} | E_y^k) \in D_i$ is close to $\boldsymbol{U} \in D_i$ for $\boldsymbol{U} \sim \{0,1\}^{n-1}$. Since the latter is $1/2$ we recover an approximate version of (3) in the typical case.

Lemma 23. *Assume (i)–(ii). Suppose $(k, y) \in \mathcal{T}_i$ is typical and $i \leq \varepsilon n$. Then with probability at least $1 - \delta$ over the choice of $\boldsymbol{j} \sim [n] \setminus \{k_1, \dots, k_i\}$,*

$$|\Pr[\boldsymbol{X}_{S_{\boldsymbol{j}}} \in D_{\boldsymbol{j}} \mid E_y^k] - 1/2| \leq \sqrt{4\varepsilon/\delta}.$$

Proof. Let $\boldsymbol{Y} \coloneqq (\boldsymbol{X} \mid E_y^k)$. Note that $\boldsymbol{Y}$ is flat and hence $\mathrm{H}(\boldsymbol{Y}) = \mathrm{H}_\infty(\boldsymbol{Y}) \geq \binom{n}{2} - 2i$. Now we use the fact that the sets in $\mathcal{S}_{\text{SINK}}$ satisfy the preconditions of Shearer's lemma (Lemma 20) with $\Pr_{\boldsymbol{j} \sim [n]}[t \in S_{\boldsymbol{j}}] = 2/n$ for every $t \in [\binom{n}{2}]$. We get $\binom{n}{2} - 2i \leq \mathrm{H}(\boldsymbol{Y}) \leq n/2 \cdot \mathbb{E}_{\boldsymbol{j} \sim [n]}[\mathrm{H}(\boldsymbol{Y}_{S_{\boldsymbol{j}}})]$. Then $\mathbb{E}_{\boldsymbol{j} \sim [n]}[\mathrm{H}(\boldsymbol{Y}_{S_{\boldsymbol{j}}})] \geq n - 1 - 4\varepsilon$. By Markov's inequality applied to the non-negative random variable $n - 1 - \mathrm{H}(\boldsymbol{Y}_{S_{\boldsymbol{j}}})$, we get that

$$\Pr_{\boldsymbol{j} \sim [n]} \Big[\underbrace{\mathrm{H}(\boldsymbol{Y}_{S_{\boldsymbol{j}}}) \geq (n-1) - 4\varepsilon/\delta}_{=:\text{ event } L} \Big] \geq 1 - \delta.$$

We may assume $4\varepsilon/\delta < 1$ as otherwise the lemma is trivial. Note that for every outcome $j \in \{k_1, \dots, k_i\}$, the variable $\boldsymbol{Y}_{S_j}$ is supported either on D_j or its complement $\{0,1\}^{n-1} \setminus D_j$, and hence $\mathrm{H}(\boldsymbol{Y}_{S_j}) \leq n - 2$. We conclude that no outcome $j \in \{k_1, \dots, k_i\}$ satisfies L. Therefore

$$\Pr_{\boldsymbol{j} \sim [n] \setminus \{k_1, \dots, k_i\}}[L] \geq \Pr_{\boldsymbol{j} \sim [n]}[L] \geq 1 - \delta.$$

Consider any j that satisfies event L and apply Lemma 19 to the random variable $\boldsymbol{Y}_{S_j}$. This results in $\Delta\left(\boldsymbol{Y}_{S_j}, \boldsymbol{U}\right) \leq \sqrt{4\varepsilon/\delta}$ where $\boldsymbol{U} \sim \{0,1\}^{S_i}$. Hence

$$\Pr[\boldsymbol{X}_{S_j} \in D_j \mid E_y^k] = \Pr[\boldsymbol{Y}_{S_j} \in D_j] \in 1/2 \pm \sqrt{4\varepsilon/\delta}.$$

□

Corollary 24. $|\mathbb{E}[\boldsymbol{y}_{i+1} \mid (\boldsymbol{k}_{\leq i}, \boldsymbol{y}_{\leq i}) \in \mathcal{T}_i]| \leq 2\delta + 4\sqrt{\varepsilon/\delta}$ *for every δ, ε, i.*

Proof. We calculate

$$\begin{aligned}
|\mathbb{E}[\boldsymbol{y}_{i+1} \mid (\boldsymbol{k}_{\leq i}, \boldsymbol{y}_{\leq i}) \in \mathcal{T}_i]| &\leq \sum_{(k,y) \in \mathcal{T}_i} \Pr[(\boldsymbol{k}_{\leq i}, \boldsymbol{y}_{\leq i}) = (k, y)] \cdot |\mathbb{E}[\boldsymbol{y}_{i+1} \mid (\boldsymbol{k}_{\leq i}, \boldsymbol{y}_{\leq i}) = (k, y)]| \\
&= \sum_{(k,y) \in \mathcal{T}_i} \Pr[(\boldsymbol{k}_{\leq i}, \boldsymbol{y}_{\leq i}) = (k, y)] \cdot |\mathbb{E}[\boldsymbol{y}_{i+1} \mid E_y^k \wedge \boldsymbol{k}_{\leq i} = k]| \\
&\leq \max_{(k,y) \in \mathcal{T}_i} |\mathbb{E}[\boldsymbol{y}_{i+1} \mid E_y^k \wedge \boldsymbol{k}_{\leq i} = k]| \\
&= \max_{(k,y) \in \mathcal{T}_i} |\mathbb{E}_{\boldsymbol{j} \sim [n] \setminus \{k_1, \dots, k_i\}} \left[\mathbb{E}_{\boldsymbol{X}}[[\![\boldsymbol{X}_{S_{\boldsymbol{j}}} \in D_{\boldsymbol{j}}]\!] \mid E_y^k] \right]| \\
&\leq 2 \max_{(k,y) \in \mathcal{T}_i} \left(\delta + \Pr_{\boldsymbol{j} \sim [n] \setminus \{k_1, \dots, k_i\}} \left[\Pr[\boldsymbol{X}_{S_{\boldsymbol{j}}} \in D_{\boldsymbol{j}} \mid E_y^k] \notin 1/2 \pm \delta \right] \right) \\
&\text{(Lemma 23)} \leq 2(\delta + 2\sqrt{\varepsilon/\delta}).
\end{aligned}$$

□

Proof of Lemma 18. Let $\boldsymbol{w}_i \coloneqq \mathbb{E}[\boldsymbol{y}_i \mid \boldsymbol{y}_{<i}, \boldsymbol{k}_{<i}]$. Let us decompose this random variable as

$$\boldsymbol{w}_i = \boldsymbol{w}_i^{\mathrm{typ}} + \boldsymbol{w}_i^{\mathrm{rare}},$$

where $\boldsymbol{w}_i^{\mathrm{typ}} = \boldsymbol{w}_i$ if $(\boldsymbol{k}_{<i}, \boldsymbol{y}_{<i})$ is typical and $\boldsymbol{w}_i^{\mathrm{typ}} = 0$ otherwise. To summarise,

$$\boldsymbol{y}_i = (\boldsymbol{y}_i - \boldsymbol{w}_i) + \boldsymbol{w}_i^{\mathrm{typ}} + \boldsymbol{w}_i^{\mathrm{rare}}.$$

Let us analyze the sum over $i \in [\ell]$ of each of these summands.

Term $\boldsymbol{y}_i - \boldsymbol{w}_i$. The sequence $\boldsymbol{y}_i - \boldsymbol{w}_i$ is a martingale difference sequence with $|\boldsymbol{y}_i - \boldsymbol{w}_i| \le 2$, so by Azuma,

$$\Pr\left[\left|\sum_{i\in[\ell]} (\boldsymbol{y}_i - \boldsymbol{w}_i)\right| \ge \delta\ell\right] \le 2\exp(-\Omega(\delta^2\ell)). \tag{4}$$

Term $\boldsymbol{w}_i^{\mathrm{rare}}$. Observe that $\Pr[(\boldsymbol{k}_{\le i}, \boldsymbol{y}_{\le i}) \notin \mathcal{T}_i] \le \max_k \Pr[(k, \boldsymbol{y}_{\le i}) \notin \mathcal{T}_i]$. Moreover, for every k,

$$\Pr[(k, \boldsymbol{y}_{\le i}) \notin \mathcal{T}_i] \le \sum_{y\colon (k,y)\notin\mathcal{T}_i} \Pr[\boldsymbol{y}_{\le i} = y] \le 2^i \cdot 2^{-2i} = 2^{-i}.$$

Using this, we estimate

$$\begin{aligned}\Pr\left[\left|\sum_{i=1}^{\ell} \boldsymbol{w}_i^{\mathrm{rare}}\right| \ge \delta\ell\right] &\le \Pr\left[\exists i \in [\delta\ell, \ell]\colon \boldsymbol{w}_i^{\mathrm{rare}} \ne 0\right] \\ &\le \Pr\left[\exists i \in [\delta\ell, \ell]\colon (\boldsymbol{k}_{\le i}, \boldsymbol{y}_{\le i}) \notin \mathcal{T}_i\right] \le \sum_{i=\delta\ell}^{\infty} 2^{-i} = 2^{-\delta\ell+1}.\end{aligned} \tag{5}$$

Term $\boldsymbol{w}_i^{\mathrm{typ}}$. One the one hand, $\boldsymbol{w}_i^{\mathrm{typ}} = \boldsymbol{w}_i$ whenever $(\boldsymbol{k}_{<i}, \boldsymbol{y}_{<i}) \in \mathcal{T}_{i-1}$. From Corollary 24,

$$|(\boldsymbol{w}_i \mid (\boldsymbol{k}_{<i}, \boldsymbol{y}_{<i}) \in \mathcal{T}_{i-1})| = |\,\mathbb{E}[\boldsymbol{y}_i \mid (\boldsymbol{k}_{<i}, \boldsymbol{y}_{<i}) \in \mathcal{T}_{i-1}]| \le 2\delta + 4\sqrt{\varepsilon/\delta} \le 4\delta.$$

On the other hand, $(\boldsymbol{w}_i^{\mathrm{typ}} \mid (\boldsymbol{k}_{<i}, \boldsymbol{y}_{<i}) \notin \mathcal{T}_{i-1}) = 0$. Thus we conclude $|\boldsymbol{w}_i^{\mathrm{typ}}| \le 4\delta$. Hence

$$\Pr\left[\left|\sum_{i\in[\ell]} \boldsymbol{w}_i^{\mathrm{typ}}\right| > 4\delta\ell\right] = 0. \tag{6}$$

Putting Equations (4) to (6) together, we get for large enough n (with ε and δ fixed),

$$\Pr\left[\left|\sum_{i\in[\ell]} \boldsymbol{y}_i\right| > 6\delta\ell\right] \le 3\exp(-\Omega(\delta^2\ell)) = \exp(-\Omega(\delta^2\ell)).$$

Reparameterizing by $\delta' \coloneqq \delta/7$ we get the desired bound. □

3.2.3 Proof of the Second Reduction

Proof of Lemma 17. Let $\varepsilon \coloneqq \delta^3/100$ and $\ell \coloneqq \lfloor \varepsilon n \rfloor$. Consider a random partition $\boldsymbol{K}_1, \dots, \boldsymbol{K}_{n/\ell}$ of $[n]$ so that $\boldsymbol{K}_j \sim \binom{[n]}{\ell}$; here we assume for simplicity that ℓ divides n (if not, consider parts of size $\ell \pm 1$). Then

$$\begin{aligned}\Pr_{\boldsymbol{X}}\left[\left|\sum_{i\in[n]} [\![\boldsymbol{X}_{S_i} \in D_i]\!]\right| \ge \delta n\right] &\le \Pr_{\boldsymbol{X};\boldsymbol{K}_1,\dots,\boldsymbol{K}_{n/\ell}}\left[\exists j \in [n/\ell]\colon \left|\sum_{i\in\boldsymbol{K}_j} [\![\boldsymbol{X}_{S_i} \in D_i]\!]\right| \ge \delta\ell\right] \\ &\le \sum_{j\in[n/\ell]} \Pr_{\boldsymbol{X},\boldsymbol{K}_j}\left[\left|\sum_{i\in\boldsymbol{K}_j} [\![\boldsymbol{X}_{S_i} \in D_i]\!]\right| \ge \delta\ell\right] \\ \text{(Lemma 18)}\quad &\le \lceil 1/\varepsilon \rceil \exp(-\Omega(\delta^2\ell)) \\ \text{(for large } n\text{)}\quad &\le \exp(-\Omega(\delta^5 n)).\end{aligned}$$

□

3.2.4 Proof of the First Reduction

Proof of Lemma 16. Recall the Chernoff bound, $\Pr[\boldsymbol{X} \geq a] \leq e^{-ta}\,\mathbb{E}[e^{t\boldsymbol{X}}]$ for $t > 0$. Let $\boldsymbol{R}_j \coloneqq \sum_{i\in[n]} [\![\boldsymbol{T}^j_{S_i} \in D_i]\!]$. By Lemma 17 there exists a constant $C > 0$ such that $\Pr[\boldsymbol{R}_j \geq \varepsilon n/2] \leq e^{-C\varepsilon^5 n}$. Using this, we get

$$\mathbb{E}\left[e^{t\cdot \boldsymbol{R}_j}\right] \leq e^{tn} \cdot \Pr[\boldsymbol{R}_j \geq \varepsilon n/2] + e^{t\varepsilon n/2} \leq e^{tn - C\varepsilon^5 n} + e^{t\varepsilon n/2}. \tag{7}$$

Now we can compute our tail bound by

$$\begin{aligned}
\Pr\left[\sum_{j\in[N]} \boldsymbol{R}_j \geq \varepsilon N n\right] &\leq e^{-t\varepsilon N n}\,\mathbb{E}\left[e^{t\cdot \sum_{j\in[N]} \boldsymbol{R}_j}\right] \\
\text{(independence of } \boldsymbol{T}^j\text{)} \quad &= e^{-t\varepsilon N n} \prod_{j\in[N]} \mathbb{E}\left[e^{t\cdot \boldsymbol{R}_j}\right] \\
\text{(using (7))} \quad &\leq e^{-t\varepsilon N n} \prod_{j\in[N]} \left(e^{tn - C\varepsilon^5 n} + e^{t\varepsilon n/2}\right) \\
\text{(choosing } t \coloneqq C\varepsilon^5\text{)} \quad &\leq e^{-C\varepsilon^6 N n}(1 + e^{C\varepsilon^6 n/2})^N \\
&\leq e^{-C\varepsilon^6 N n} \cdot e^{(1 + C\varepsilon^6 n/2)\cdot N} = e^{-C\varepsilon^6 N n/2 + N} \\
\text{(for large } n\text{)} \quad &\leq \exp(-\Omega(\varepsilon^6 N n)).
\end{aligned}$$

The inequality bounding the sum from below is handled similarly. □

3.3 Proof of Lemma 15

In this section, we prove that it is sufficient to show that the extractor fools *balanced* distinguishers.

Lemma 15. *Let* $\mathrm{Ext}\colon \{0,1\}^t \to \{0,1\}^{\binom{n}{2}}$ *be an arbitrary function. Suppose that for a source* $\boldsymbol{X} \in \{0,1\}^t$*, and uniform* $\boldsymbol{Y} \sim [n]$*,* $\boldsymbol{U} \sim \{0,1\}^{n-1}$ *we have* $\Delta\left((\boldsymbol{Y}, \mathrm{Ext}(\boldsymbol{X}, \boldsymbol{Y})), (\boldsymbol{Y}, \boldsymbol{U})\right) \geq \varepsilon$*. Then there exists a balanced distinguisher* $D \subseteq [n] \times \{0,1\}^{n-1}$ *such that* $|\Pr[(\boldsymbol{Y}, \mathrm{Ext}(\boldsymbol{X}, \boldsymbol{Y})) \in D] - 1/2| \geq \varepsilon/2$*.*

Proof. Recall that for two distributions S and T over the same domain $\mathcal{X}$,

$$\Delta(S, T) = \frac{1}{2} \sum_{a\in\mathcal{X}} \left| \Pr_{\boldsymbol{x}\sim S}[\boldsymbol{x} = a] - \Pr_{\boldsymbol{x}\sim T}[\boldsymbol{x} = a] \right|.$$

Then let us expand the statistical distance as follows:

$$\varepsilon \leq \Delta\left((\boldsymbol{Y}, \mathrm{Ext}(\boldsymbol{X}, \boldsymbol{Y})), (\boldsymbol{Y}, \boldsymbol{U})\right) = \frac{1}{2n} \sum_{i\in[n];\alpha\in\{0,1\}^{n-1}} \left|\Pr[\mathrm{Ext}(\boldsymbol{X}, i) = \alpha] - 2^{-(n-1)}\right|.$$

Let $R^+_i \coloneqq \{\alpha \in \{0,1\}^{n-1} \mid \Pr[\mathrm{Ext}(\boldsymbol{X}, i) = \alpha] \geq 2^{-(n-1)}\}$ and $R^-_i = \{0,1\}^{n-1} \setminus R^+_i$. Then

$$\varepsilon \leq \frac{1}{2n} \sum_{i\in[n];\alpha\in R^+_i} \left(\Pr[\mathrm{Ext}(\boldsymbol{X}, i) = \alpha] - 2^{-(n-1)}\right) + \frac{1}{2n} \sum_{i\in[n];\alpha\in R^-_i} \left(2^{-(n-1)} - \Pr[\mathrm{Ext}(\boldsymbol{X}, i) = \alpha]\right).$$

Let $\varepsilon^+_i \coloneqq \sum_{\alpha\in R^+_i} \left(\Pr[\mathrm{Ext}(\boldsymbol{X}, i) = \alpha] - 2^{-(n-1)}\right)$, $\varepsilon^-_i \coloneqq -\sum_{\alpha\in R^-_i} \left(\Pr[\mathrm{Ext}(\boldsymbol{X}, i) = \alpha] - 2^{-(n-1)}\right)$, and $\varepsilon_i = (\varepsilon^+_i + \varepsilon^-_i)/2$. It is then sufficient to show that there exist $D_1, \ldots, D_n$ such that for every $i \in [n]$

$$|\Pr[\mathrm{Ext}(\boldsymbol{X}, i) \in D_i] - |D_i|/2^{n-1}| \geq \frac{\varepsilon_i}{2}.$$

Fix i. Let us assume that $\varepsilon^+_i \geq \varepsilon_i$ (the proof in the case of $\varepsilon^-_i \geq \varepsilon_i$ is the same). If $|R^+_i| \geq 2^{n-1}/2$ let D_i be the subset of R^+_i of size $2^{n-1}/2$ where the largest values of the difference $\Pr[\mathrm{Ext}(\boldsymbol{X}, i) = \alpha] - 2^{-(n-1)}$ are achieved. Then $\Pr[\mathrm{Ext}(\boldsymbol{X}, i) \in D_i] \geq \varepsilon^+_i/2 \geq \varepsilon_i/2$. If $|R^+_i| \leq 2^{n-2}$ let D_i be the union of R^+_i and the subset of R^-_i of size $2^{n-2} - |R^+_i| \leq |R^-_i|/2$ where the smallest values of the difference $2^{-(n-1)} - \Pr[\mathrm{Ext}(\boldsymbol{X}, i) = \alpha]$ are achieved. Then $|D_i \setminus R^+_i|/2^{n-1} - \Pr[\mathrm{Ext}(\boldsymbol{X}, i) \in (D_i \setminus R^+_i)] \leq \varepsilon^-_i/2 \leq \varepsilon_i/2$. Then $\Pr[\mathrm{Ext}(\boldsymbol{X}, i) \in D_i] - |D_i|/2^{n-1} \geq \varepsilon^+_i - \varepsilon_i/2 \geq \varepsilon_i/2$. □

4 Proof of Theorem 4: Condensing sink-of-xor

In this section, we prove Theorem 4, restated here for convenience.

Theorem 4. *There exists a $2^{O(n)}$-by-$2^{O(n)}$ submatrix R such that $F := (\textsc{Sink} \circ \oplus)|_R$ satisfies*

$$\begin{aligned} \mathrm{R}^{cc}(F) &= \Theta(n), \\ \mathrm{rk}_{1/3}(F) &\leq O(n^3). \end{aligned}$$

4.1 Preliminaries

We use the notation $\mathbb{1}[P]$ to indicate 1 if P is true and 0 if P is false. (Compare this with the ± 1-indicator notation $[\![P]\!]$ used in the previous section.) We introduce the following additional technical property of extractors to use them for our communication lower bounds.

Definition 25. We say that a function $\mathrm{Ext}\colon \{0,1\}^t \to \{0,1\}^m$ is *α-smooth* wrt $\{S_1, \dots, S_r\} \subseteq \binom{[m]}{n}$ if for every $i \in [r]$ and every $x \in \{0,1\}^n$ we have $\Pr_{\boldsymbol{y} \sim \{0,1\}^t}[\mathrm{Ext}(\boldsymbol{y}, i) = x] \leq \alpha 2^{-n}$.

Proposition 26. *A random function $\mathrm{Ext}\colon \{0,1\}^{2n} \to \{0,1\}^m$ is $\sqrt{n + \ln r}$-smooth wrt any fixed family $\{S_1, \dots, S_r\} \subseteq \binom{[m]}{n}$ with probability $1 - o(1)$.*

Proof. Fix an arbitrary $i \in [r]$ and $x \in \{0,1\}^n$. Now for a uniformly random $\mathbf{Ext}$

$$\Pr\left[\sum_{\alpha \in \{0,1\}^{2n}} \mathbb{1}[\mathbf{Ext}(\alpha, i) = x] \geq \sqrt{n + \ln r} \cdot 2^{2n-n}\right] \leq \exp(-2^{2n} 2^{-2n} \cdot (n + \ln r)) \leq e^{-n} r = o(2^{-n} r)$$

by Hoeffding inequality, since $\mathbb{E}\left[\sum_{\alpha \in \{0,1\}^t} \mathbb{1}[\mathbf{Ext}(\alpha, i) = x]\right] = 2^{2n-n} = 2^n$. Then by the union bound the total probability is $o(1)$. □

We also recall Yao's corruption bound for proving a randomised communication lower bound.

Lemma 27 ([Yao83])**.** *Let $F\colon X \times Y \to \{0,1\}^n$ be a function and let ν be a distribution over $X \times Y$ such that $\nu(F^{-1}(0)) = 1/2$. Let $\varepsilon < 1/8$. Then whenever $\mathrm{R}^{cc}_\varepsilon(F) \leq c$ there exists a rectangle $R \subseteq X \times Y$ such that $\nu(R \cap F^{-1}(1)) < 4\varepsilon\nu(R)$ and $\nu(R) \geq 2^{-c-3}$.*

4.2 Communication lower bound

We start by proving the communication lower bound in Theorem 4, that is, $\mathrm{R}^{cc}((\textsc{Sink} \circ \oplus)|_{A \times A}) \geq \Omega(n)$ where A is the image of a Shearer extractor. We show a general statement that applies not only to the xorification of Sink but to the xorification of any function that is a union of subcubes.

Theorem 28. *Let $S_1, \dots, S_r \subseteq [m]$, $a_i \in \{0,1\}^{S_i}$ for each $i \in [r]$. Suppose that $\mathrm{Ext}\colon \{0,1\}^t \to \{0,1\}^m$ is an nr-smooth (γ, k)-Shearer extractor for $\{S_1, \dots, S_r\}$. Suppose that $\gamma < 0.01$ and $r < 2^{n/2-1}/n$. Then for $F(x,y) := \bigvee_{i \in [r]} \mathbb{1}[\mathrm{Ext}(x)_{S_i} \oplus \mathrm{Ext}(y)_{S_i} = a_i]$ we have $\mathrm{R}^{cc}(F) = \Omega(t - k)$.*

Proof. Let $A_j := \{(x,y) \in (\{0,1\}^t)^2 \mid \mathrm{Ext}(x)_{S_j} \oplus \mathrm{Ext}(y)_{S_j} = a_j\}$ for $j \in [r]$. Then $F^{-1}(1) = \bigcup_{j \in [r]} A_j$. By nr-smoothness of Ext we have $|A_j| \leq 2^t \cdot 2^{t-n} \cdot nr$, so $|F^{-1}(0)| \leq r 2^{2t-n} \cdot nr = 2^{2t-1} \cdot (2nr^2/2^n) < 2^{2t}/2$. We are going to apply the corruption bound for F wrt a measure ν where $(\boldsymbol{x}, \boldsymbol{y}) \sim \nu$ is sampled as follows:

- With probability $1/2$, output a uniform $(\boldsymbol{x}, \boldsymbol{y}) \sim F^{-1}(0)$.
- With probability $1/2$, sample a uniform $\boldsymbol{j} \sim [r]$ and output a uniform $(\boldsymbol{x}, \boldsymbol{y}) \sim A_{\boldsymbol{j}}$.

Let $\varepsilon = 1/100$, let $c = \mathrm{R}^{cc}_\varepsilon(F)$, the ε-error communication complexity of F. It suffices to prove a lower bound on c, since the error parameter of any constant-error protocol can be reduced to ε by repeating the protocol constantly many times. Assume for contradiction that $c \leq t - k - 3$. Then by Lemma 27 there exists a rectangle $R = X \times Y \subseteq (\{0,1\}^m)^2$ such that $\nu(R \cap F^{-1}(1)) \leq 4\varepsilon\nu(R)$ and $\nu(R) \geq 2^{-c-3}$. Let us estimate

$$|R| \geq |R \cap F^{-1}(0)| \geq 2 \cdot 2^{2t} \cdot \nu(R \cap F^{-1}(0)) \geq 2^{2t+1}(1 - 4\varepsilon)\nu(R) \geq 2^{2t-c-3}. \tag{8}$$

The second inequality follows from $|F^{-1}(0)| \le 1/2 \cdot 2^{2t}$, the fourth requires $\varepsilon < 1/8$. Then $\mathrm{H}_\infty(X), \mathrm{H}_\infty(Y) \ge t - c - 3 \ge k$. The rest of the proof goes as follows: using extractor properties we establish that $\nu(R \cap F^{-1}(1))$ is large, which contradicts the fact that it comprises only a small fraction of R as measured by ν.

Claim 29. $\nu(R \cap F^{-1}(1)) \ge |R|(1 - 12\gamma)/(3 \cdot 2^{2t+3})$.

Proof. Let $A_j^\alpha := \{x \in \{0,1\}^t \mid \mathrm{Ext}(x, j) = \alpha\}$. Observe that $A_j = \bigcup_{\alpha \in \{0,1\}^n} A_j^\alpha \times A_j^{\alpha \oplus a_j}$. By the extractor property we have $(2r)^{-1} \sum_{j \in [r]} \sum_{\alpha \in \{0,1\}^n} \left| |A_j^\alpha \cap X|/|X| - 2^{-n} \right| \le \gamma$ and the same property for Y. Then by the Markov's inequality for the fraction $2/3$ of the seeds j we have $\sum_{\alpha \in \{0,1\}^n} \left| |A_j^\alpha \cap X|/|X| - 2^{-n} \right| \le 6\gamma$. Applying Markov's inequality for the same sum for Y we get that for the fraction $1/3$ of the seeds j we have

$$\sum_{\alpha \in \{0,1\}^n} \left| \frac{|A_j^\alpha \cap Y|}{|Y|} - 2^{-n} \right| \le 6\gamma \quad \text{and} \quad \sum_{\alpha \in \{0,1\}^n} \left| \frac{|A_j^\alpha \cap X|}{|X|} - 2^{-n} \right| \le 6\gamma.$$

Denote the set of such seeds as J. Applying Markov's inequality again we get that for each $j \in J$ for at least $2^n(1 - 12\gamma)$ values α we have $|A_j^\alpha \cap X|/|X| \ge 2^{-n-1}$ and for at least $2^n(1 - 12\gamma)$ values α we have $|A_j^{\alpha \oplus a_j} \cap Y|/|Y| \ge 2^{-n-1}$. So in total for $2^n(1 - 24\gamma)$ values α both inequalities hold, so multiplying them together we get

$$\frac{|A_j^\alpha \times A_j^{\alpha \oplus a_j} \cap R|}{|R|} \ge 2^{-2n-2}.$$

Summing these up yields $|A_j \cap R|/|R| \ge (1 - 24\gamma)2^{-n-2}$ for each $j \in J$. Then

$$\begin{aligned}
\nu(R \cap F^{-1}(1)) &= \frac{1}{r} \sum_{j \in [r]} \frac{|A_j \cap R|}{|A_j|} \ge \frac{1}{r} \sum_{j \in J} \frac{|A_j \cap R|}{|A_j|} \\
&\ge \frac{1}{r} \sum_{j \in J} \frac{|A_j \cap R|}{2^{2t-n/2}} \\
&= \frac{1}{r} \frac{|R|}{2^{2t-n+1}} \sum_{j \in J} \frac{|A_j \cap R|}{|R|} \ge \frac{|R|(1 - 24\gamma)}{3 \cdot 2^{2t+3}}
\end{aligned}$$

□

From (8) we get that $\nu(R) \le |R|2^{-2t}/(2(1 - 4\varepsilon))$, so $\nu(R \cap F^{-1}(1)) \le \varepsilon |R| 2^{-2t}/(2(1 - 4\varepsilon))$. Combining this with Claim 29 we get $(1 - 24\gamma)/24 \le \varepsilon/(2(1 - 4\varepsilon))$, which is a contradiction for $\varepsilon, \gamma \le 0.01$. □

Now let us recover the communication lower bound for the submatrix of $\textsc{Sink} \circ \oplus$. As it is observed in [CMS20], $\textsc{Sink}^{-1}(1)$ is a union of disjoint subcubes: let $\{S_1, \dots, S_n\} = \mathcal{S}_{\textsc{Sink}}$ and let $a_i \in \{0,1\}^{S_i}$ be chosen such that each two cubes in $\{x \mid x_{S_i} = a_i\}$ conflict. Then

$$\textsc{Sink}_n(x) = \bigvee_{i \in [n]} \mathbb{1}[x_{S_i} = a_i] = \sum_{i \in [n]} \mathbb{1}[x_{S_i} = a_i]. \tag{9}$$

Corollary 30. *Let R be a random subset of $\{0,1\}^{\binom{n}{2}}$ of size 2^{2n}. Then for the function $F_R \colon R \times R \to \{0,1\}$ defined as $F_R(x, y) := \textsc{Sink}_n(x \oplus y)$ we have $\mathrm{R}^{cc}(F_R) = \Omega(n)$ with probability $1 - o(1)$ over R.*

Proof. By Theorem 7 and Proposition 26 a uniformly random function $\mathrm{Ext} \colon \{0,1\}^{2n} \to \{0,1\}^{\binom{n}{2}}$ is a $\sqrt{n - 1 + \ln n}$-smooth $(0.01, \Omega(n))$-Shearer extractor. By Theorem 28 we get that $G(x, y) = \textsc{Sink}(\mathrm{Ext}(x) \oplus \mathrm{Ext}(y))$ has $\mathrm{R}^{cc}(G) = \Omega(n)$. To finish the proof observe that the communication cost of G coincides with the communication cost of $F_{\mathrm{Ext}(\{0,1\}^{2n})}$. □

4.3 Approximate rank upper bound

Finally, to prove the approximate rank upper bound in Theorem 4, we show that, in fact, *all* small submatrices of sink-of-xor have low approximate rank.

Theorem 31. *Let $R \subseteq \{0,1\}^{\binom{n}{2}}$ have size 2^{Cn} for any constant C. Then the matrix $F_R \colon R \times R \to \{0,1\}$ defined as $F_R(x, y) := \textsc{Sink}_n(x \oplus y)$ has $1/3$-approximate rank $O(n^3)$.*

We consider the factorization norm γ_2 [Tom89, LMSS04] that is often used as a smooth proxy for approximate rank. To define it, we denote by $r(M)$ the maximum ℓ_2-norm of a row of a matrix M. Then

$$\gamma_2(A) \coloneqq \min\{r(B)r(C) : BC^\top = A\}.$$

This matrix norm is connected to approximate rank through the following lemma. Note below how the bound depends on the matrix size N; our improvement over [CKLM19] is due to us focusing on only a small submatrix of sink-of-xor. (The dependency on N is quite analogous to how Newman's theorem [New91] converts public-coin protocols to private-coin ones at a cost depending on N.)

Lemma 32 ([LS09, Thm 10])**.** *For every $A \in \{0,1\}^{N\times N}$ we have $\mathrm{rk}_{1/3}(A) \le O(\gamma_2^2(A)\cdot \log N)$.*

Proof of Theorem 31. By Lemma 32 it suffices to show that $\gamma_2(F_R) = O(n)$. Let $M_{\textsc{Sink}}$ be a $2^{\binom{n}{2}}\times 2^{\binom{n}{2}}$ binary matrix defined as $(M_{\textsc{Sink}})_{x,y} \coloneqq \textsc{Sink}_n(x\oplus y)$. Then F_R is a submatrix of $M_{\textsc{Sink}}$, hence $\gamma_2(F_R) \le \gamma_2(M_{\textsc{Sink}})$. We show that $\gamma_2(M_{\textsc{Sink}}) \le n$. By (9) we have

$$(M_{\textsc{Sink}})_{x,y} = \bigvee_{i\in[n]} (x_{S_i} \oplus y_{S_i} = a_i) = \sum_{i\in[n]} \mathbb{1}[x_{S_i}\oplus y_{S_i} = a_i].$$

The last equality holds since the cubes $\{x_{S_i} = a_i\}$ are disjoint. Let M_i be a $2^{\binom{n}{2}}\times 2^{\binom{n}{2}}$ matrix defined by $M_i(x,y) \coloneqq \mathbb{1}[x_{S_i}\oplus y_{S_i} = a_i]$. Then $M_{\textsc{Sink}} = M_1 + \cdots + M_n$, so $\gamma_2(M_{\textsc{Sink}}) \le \sum_{i\in[n]}\gamma_2(M_i)$ where we used the sub-additivity of the γ_2-norm. It remains to show that $\gamma_2(M_i) = 1$ for every i. The factorization is as follows: A, B are $2^{\binom{n}{2}}\times 2^{n-1}$ boolean matrices defined by $A_{x,y} \coloneqq \mathbb{1}[x_{S_i} = y]$ and $B_{x,y} \coloneqq \mathbb{1}[x_{S_i} = y\oplus a_i]$. It is easy to check that $AB^\top = M_i$ and that every row of A and B contains exactly one 1-entry. □

5 Open Problem 2: Case of lifted functions

In this section, we prove Theorem 3, which we state more formally as follows. Given a boolean function $f\colon \{0,1\}^n \to \{0,1\}$ and a two-party function $g\colon \Sigma\times\Sigma \to \{0,1\}$ (often called a *gadget*), we define the composed (or *lifted*) function $F \coloneqq f\circ g$ of type $\Sigma^n\times\Sigma^n \to \{0,1\}$ that maps $(x,y)\in\Sigma^n\times\Sigma^n$ to

$$F(x,y) = (f\circ g)(x,y) \coloneqq f(g(x_1,y_1),\dots,g(x_n,y_n)) \qquad \text{where } x_i, y_i \in \Sigma.$$

We consider the usual *inner-product* gadget $\mathrm{IP}_b\colon \{0,1\}^b\times\{0,1\}^b \to \{0,1\}$ given by $\mathrm{IP}_b(x,y) \coloneqq \langle x,y\rangle \bmod 2$.

Theorem 33 (Formal version of Theorem 3)**.** *For every $\varepsilon > 0$ there exists $C > 0$ such that for every function $f\colon \{0,1\}^n \to \{0,1\}$ with $\mathrm{D}(f) \ge n^\varepsilon$ the composed function $F \coloneqq f\circ \mathrm{IP}_{C\log^2 n}$ condenses losslessly: there exists an 2^k-by-2^k submatrix R such that $k = \Theta(\mathrm{D}^{cc}(F)) = \Theta(\mathrm{D}^{cc}(F|_R)) = \Theta(\mathrm{D}(f)\cdot\log^2 n)$.*

One can consider this theorem as a *positive* result: functions that are lifted by inner-product can be losslessly condensed; as well as *negative* result: lifting with inner-product cannot be used to prove our conjecture that deterministic communication cannot be condensed for every function.

Proof overview. To prove Theorem 33 we go back to query complexity and show (in Theorem 36) that if instead of restricting by a partial assignment we allow substitutions of certain linear functions then we can condense *any* function losslessly. More formally, let $f\colon \{0,1\}^n \to \{0,1\}$ be a boolean function. For some $m \le n$ and a bipartite graph $G \coloneqq ([n],[m],E)$ define $f\circ\oplus_G\colon \{0,1\}^m \to \{0,1\}$ by

$$(f\circ\oplus_G)(x) \coloneqq f\Big(\bigoplus_{i\in N_G(1)} x_i, \dots, \bigoplus_{i\in N_G(n)} x_i\Big)$$

where $N_G(i) \coloneqq \{j\in[m] \mid (i,j)\in E\}$ is the set of neighbours of i in G. We also think of $\oplus_G$ as a function $\oplus_G\colon \{0,1\}^m \to \{0,1\}^n$ where i-th bit of output is computed as the parity $\oplus_{i\in N_G(n)} x_i$. We show that for a sufficiently well expanding graph G, the query complexity of $f\circ\oplus_G$ can be losslessly condensed to $\mathrm{D}(f\circ\oplus_G)$ variables. To conclude the proof, we lift this query result to communication complexity using a lifting theorem.

Comparison to prior work. Our condensation result for query complexity (Theorem 36) may be viewed as a version of Razborov's hardness condensation theorem [Raz16, FPR22]. However, these results are incomparable with ours. In both papers [Raz16, FPR22], the authors study the decision model with bounded memory that corresponds to Resolution proofs with bounded width. To get rid of dependency on memory our proof uses the "sequential closure" strategy that appears in [Sok20, Section A].

5.1 Condensing query complexity via parities

A bipartite graph $G \coloneqq ([n], [m], E)$ is an (r, Δ, α)-*expander* if each left node in $[n]$ has degree at most Δ, and for every $I \subseteq [n]$, $|I| \leq r$ implies that $|N_G(I)| \geq \alpha|I|$. Moreover if $|\partial I| \geq \alpha|I|$ then we say that G is an (r, Δ, α)-*boundary expander* where $\partial S \subseteq N_G(S)$ is the set of all nodes connected to S via *exactly* one edge.

Proposition 34. *If G is an $(r, \Delta, (1-\varepsilon)\Delta)$-expander then G is an $(r, \Delta, (1-2\varepsilon)\Delta)$-boundary expander.*

Proof. There are at least $|\partial S| + 2((1-\varepsilon)\Delta|S| - |\partial S|)$ edges between S and $N_G(S)$. On the other hand, there are at most $\Delta|S|$ of these edges. Thus $|\partial S| + 2\Delta|S| - 2\varepsilon\Delta|S| - 2|\partial S| \leq \Delta|S|$, so $|\partial S| \geq (1-2\varepsilon)\Delta|S|$. □

Lemma 35 (Folklore)**.** *For every constant $c \geq 1$ and for large enough m there exists $(r, \Delta, 0.9\Delta)$-expander $G = ([n], [m], E)$ with $r = \Theta(m/\Delta)$, $n = m^c$, $\Delta = \Theta(\log m)$.*

Proof. This is a variant of the classical proof of expander existence with somewhat unusual parameters. In a similar setting, it appears for instance in [ABRW00, Raz16, SS22]. The latter paper [SS22, Section C] shows that there exists a random bipartite graph $\boldsymbol{G}$ over nodes $([n], [m])$ and with left degree Δ such that

$$p \coloneqq \Pr[\boldsymbol{G} \text{ is not a } (r, \Delta, 0.9 \cdot \Delta)\text{-expander}] \leq \sum_{s \in [r]} \left(e^{1+0.9\Delta} \frac{n}{s} \cdot \left(\frac{0.9 \cdot \Delta s}{m} \right)^{0.1\Delta} \right)^s.$$

Let us pick $\Delta = 3c \log m$ and r such that $0.9 \cdot \Delta \cdot r/m < e^{-100}$. Then each summand is at most $e^{1+2.7c \log m} \cdot n^c \cdot e^{-30c \log m} < e^{-20c \log m} \ll 1/m < 1/r$, so $p < 1$. Therefore there exists an expander with the required parameters. □

Theorem 36. *Suppose $f \colon \{0,1\}^n \to \{0,1\}$ is such that $\mathrm{D}(f) = k$ and $G \coloneqq ([n], [m], E)$ is an $(r, \Delta, 0.9 \cdot \Delta)$-expander, where $r \geq k$ and $\Delta \geq 7$. Then $\mathrm{D}(f \circ \oplus_G) = \Theta(k\Delta)$.*

Remark 37. We note that the condition $r \geq k$ above is crucial. For example, consider $f \coloneqq \textsc{And}_n$ as the outer function, and $m \coloneqq n - 1$. We may assume wlog, that 1^n is not in the image of $\oplus_G$ for some reasonable choice of G. Thus, $\textsc{And}_n \circ \oplus_G$ is a constant-0 function and $\mathrm{D}(\textsc{And}_n \circ \oplus_G) = 0$ even though G can be a good enough expander, but only for $r < k$.

5.1.1 Proof of Theorem 36

The upper bound $\mathrm{D}(f \circ \oplus_G) \leq O(k\Delta)$ is straightforward, so we concentrate on proving the corresponding lower bound. To this end, we are going to convert an adversary strategy for f that survives for $k-1$ rounds of queries (without fixing the value of f) to an adversary strategy for $f \circ \oplus_G$ that survives for $\Omega(k\Delta)$ rounds. At the ℓ-th round of the game, we maintain two partial assignments: $\rho_\ell \colon [m] \to \{0, 1, *\}$ which corresponds to the answers our strategy for $f \circ \oplus_G$ has given, and $\tau_\ell \colon [n] \to \{0, 1, *\}$ corresponding to the answers given by the adversary strategy for f that we invoke. We will also maintain a partial assignment $\widehat{\rho}_\ell$, an extension of ρ_ℓ such that the following invariants hold (here we write $\mathsf{fix}(\tau) \coloneqq \tau^{-1}(\{0,1\}) = \{i : \tau(i) \neq *\}$).

(i) For every $i \in \mathsf{fix}(\tau_\ell)$ we have $N_G(i) \subseteq \mathsf{fix}(\widehat{\rho}_\ell)$ and $\tau_\ell(i) = \bigoplus_{j \in N(i)} \widehat{\rho}_\ell(j)$.
(ii) The graph $G_\ell \coloneqq G - \mathsf{fix}(\tau_\ell) - \mathsf{fix}(\widehat{\rho}_\ell)$ is an $(r, \Delta, 0.8 \cdot \Delta)$-expander.

Our proof is divided into two parts. In the first part, assuming that we can maintain invariants *(i)–(ii)* and the property $|\mathsf{fix}(\tau_\ell)| < k$, we show that $(f \circ \oplus_G)|_{\widehat{\rho}_\ell}$ is not constant. Hence our adversary strategy can survive for more rounds, which shows $\mathrm{D}(f \circ \oplus_G) > \ell$. In the second part, we show how to maintain invariants *(i)–(ii)* as well as analyse how fast $|\mathsf{fix}(\tau_\ell)|$ grows as a function of ℓ.

Before we carry out this plan, we first show a useful property of $\oplus_G$ when G expands.

Lemma 38. *Let $G := (U, V, E)$ be an (r, Δ, ε)-boundary expander for some $\varepsilon > 0$. Then for every set $I \subseteq U$ of size at most r and any $a \in \{0,1\}^I$ there is an assignment $\beta \in \{0,1\}^V$ such that $\oplus_G(\beta)_I = a$.*

Proof. Since $\oplus_G$ is a linear function, the statement of the lemma is violated iff there is a nontrivial set of outputs $I \subseteq U$ of size at most r such that linear equations that correspond to I are linearly dependent. Pick the smallest such set I. Since I is the smallest and we are doing our computations over $\mathbb{F}_2$, this implies that for every $j \in V$, its neighborhood size in I is even. However, since $|I| \leq r$ then there is at least one $j \in \partial I$ whose neighborhood size in I is one, contradicting the above. □

First part. Here we prove that $(f \circ \oplus_G)|_{\widehat{\rho}_\ell}$ is not constant. We focus on showing that $(f \circ \oplus_G)|_{\widehat{\rho}_\ell}$ is not the constant-0 function (showing it is not constant-1 is similar). First note that $f|_{\tau_\ell}$ is not constant since τ_ℓ comes from the adversary strategy for f after $|\mathsf{fix}(\tau_\ell)| < k$ queries. Thus, if we run the optimal k-query decision tree for f starting from τ_ℓ, we can find an extension $\tau_\ell \cup \sigma$ of τ_ℓ, corresponding to a leaf of the optimal tree that outputs 1, such that

- $|\mathsf{fix}(\sigma)| \leq k$
- $f|_{\tau_\ell \cup \sigma}$ is constant 1.

Let $\widehat{\rho}^0_\ell \in \{0,1\}^m$ be the full assignment obtained from $\widehat{\rho}_\ell$ by replacing every $*$ with a 0. Note that the output $y := \oplus_G(\widehat{\rho}^0_\ell)$ is consistent with τ_ℓ. By Proposition 34 and *(ii)*, the graph G_ℓ is a (r, Δ, ε)-boundary expander for some $\varepsilon > 0$. Thus Lemma 38 implies that there is some $x \in \{0,1\}^{[m] \setminus \mathsf{fix}(\widehat{\rho}_\ell)}$ such that the output $z := \oplus_{G_\ell}(x) \in \{0,1\}^{[n] \setminus \mathsf{fix}(\tau_\ell)}$ is consistent with $\sigma \oplus y_{\mathsf{fix}(\sigma)}$. Let $x^0 \in \{0,1\}^m$ and $z^0 \in \{0,1\}^n$ denote the extensions of x and z by 0s. Note that $z^0 = \oplus_G(x^0)$. Then $x^* := \widehat{\rho}^0_\ell + x^0$ is consistent with $\widehat{\rho}_\ell$ and its output $\oplus_G(x^*) = y \oplus z^0$ is consistent with $\tau_\ell \cup \sigma$. This implies $(f \circ \oplus_G)(x^*) = 1$, which means $(f \circ \oplus_G)|_{\widehat{\rho}_\ell}$ is not constant-0, as desired.

Second part. At the $(\ell+1)$-th round we receive a query $i \in [m]$. If $i \in \mathsf{fix}(\widehat{\rho}_\ell)$, then we respond with $\widehat{\rho}_\ell(i)$, set $\rho_{\ell+1} := \rho_\ell \cup \{(i, \widehat{\rho}_\ell(i))\}$, and continue. Neither invariant depends on $\rho_{\ell+1}$, so they continue to hold.

Suppose that we receive a query $i \in [m] \setminus \mathsf{fix}(\widehat{\rho}_\ell)$. We set $\rho_{\ell+1} := \rho_\ell \cup \{(i,b)\}$ and $\widehat{\rho}_{\ell+1} := \widehat{\rho}_\ell \cup \{(i,b)\}$ where $b \in \{0,1\}$ can be chosen arbitrarily. Let $G'_\ell := G_\ell - i$, and let $B_{\ell+1} \subseteq [n] \setminus \mathsf{fix}(\tau_\ell)$ be the largest subset such that $|B_{\ell+1}| \leq r$ and $|N_{G'_\ell}(B_{\ell+1})| \leq 0.8 \cdot \Delta |B_{\ell+1}|$ ($B_{\ell+1}$ may be empty). We then query the adversary for f with all elements of $B_{\ell+1}$ extending $\tau_{\ell+1}$ from τ_ℓ with the received answers. Now we need to extend $\widehat{\rho}_{\ell+1}$ by fixing all elements of $N_{G'_\ell}(B_{\ell+1})$ such that Item *(i)* is satisfied. This is possible by Lemma 38, since by Proposition 34 and Item *(ii)*, we have that G'_ℓ is an $(r, \Delta, 0.8 \cdot \Delta - 1)$-expander and $\Delta \geq 7$ we can conclude that G'_ℓ is a boundary expander.

We set $G_{\ell+1} := G'_\ell - B_{\ell+1} - N_{G'_\ell}(B_{\ell+1})$, since $\mathsf{fix}(\widehat{\rho}_{\ell+1}) = \mathsf{fix}(\widehat{\rho}_\ell) \cup \{i\} \cup N_{G'_\ell}(B_{\ell+1})$. It remains to show that $|\mathsf{fix}(\tau_\ell)|$ grows slowly as a function of ℓ, and to verify Item *(ii)*.

Claim 39. *If $\ell \leq \Delta r/220$, then $|\mathsf{fix}(\tau_\ell)| \leq 10 \cdot \ell/\Delta$. Moreover $|B_i| \leq r/2$ for each $i \leq \ell$.*

Proof. We show that $|\mathsf{fix}(\tau_t)| \leq 10 \cdot t/\Delta$ by induction on t where $t \in [\ell]$. The base case of the induction is satisfied as τ_0 does not assign any variable. Now suppose that $|\mathsf{fix}(\tau_t)| \leq 10 \cdot t/\Delta$ for some $t < \ell$; we aim to prove $|\mathsf{fix}(\tau_{t+1})| \leq 10 \cdot (t+1)/\Delta$. Observe that $N_{G'_t}(B_{t+1}) = N_G(B_{t+1}) \setminus (N_G(\mathsf{fix}(\tau_t)) \cup \mathsf{fix}(\rho_{t+1}))$. By the choice of B_{t+1} we have $|N_{G'_t}(B_{t+1})| \leq 0.8 \cdot \Delta |B_{t+1}|$, and by the expansion of G we have $|N_G(B_{t+1})| \geq 0.9 \cdot \Delta |B_{t+1}|$. Hence $|N_G(\mathsf{fix}(\tau_t)) \cup \mathsf{fix}(\rho_{t+1})| \geq \Delta |B_{t+1}|/10$. By the construction $|\mathsf{fix}(\rho_{t+1})| = t+1$ and by induction hypothesis $|\mathsf{fix}(\tau_t)| \leq 10t/\Delta$ implying $|N_G(\mathsf{fix}(\tau_t))| \leq 10t$. Then we get

$$|B_{t+1}| \leq \frac{10}{\Delta} \cdot (10(t+1)) \leq r/2.$$

Then $|\mathsf{fix}(\tau_{t+1})| \leq |\mathsf{fix}(\tau_t)| + |B_{t+1}| \leq r$, so by expansion of G we get $|N_G(\mathsf{fix}(\tau_{t+1}))| \geq 0.9 \cdot \Delta |\mathsf{fix}(\tau_{t+1})|$. By the constructions of G_i we have

$$N_G(\mathsf{fix}(\tau_{t+1})) \subseteq \bigcup_{i=0}^{t} N_{G'_i}(B_{i+1}) \cup \mathsf{fix}(\rho_{t+1}).$$

By the choice of B_i it implies $|N_G(\mathsf{fix}(\tau_{t+1}))| \leq 0.8 \cdot \Delta |\mathsf{fix}(\tau_{t+1})| + (t+1)$. Putting the two bounds on $|N_G(\mathsf{fix}(\tau_{t+1}))|$ together we get $|\mathsf{fix}(\tau_{t+1})| \leq 10(t+1)/\Delta$. □

Claim 40. *If $\ell \le \Delta r/220$, then G_ℓ is an $(r, \Delta, 0.8 \cdot \Delta)$-expander.*

Proof. Pick the minimal $t \in \{0, 1, \dots, \ell - 1\}$ such that G_{t+1} is not an $(r, \Delta, 0.8 \cdot \Delta)$-expander. Then there exists a set S of size at most r such $|N_{G_{t+1}}(S)| < 0.8 \cdot \Delta|S|$. By expansion of G we have $|N_G(S)| \ge 0.9 \cdot \Delta|S|$, so since $N_{G_{t+1}}(S) = N_G(S) \setminus (N_G(\mathsf{fix}(\tau_{t+1})) \cup \mathsf{fix}(\rho_{t+1}))$ we get

$$\Delta r/20 \ge \Delta r/22 + (t+1) \ge |N_G(\mathsf{fix}(\tau_{t+1})) \cup \mathsf{fix}(\rho_{t+1}))| \ge \Delta|S|/10,$$

where the right-hand side follows from the previous bound on $N_{G_{t+1}}(S)$ and the left-hand side from Claim 39 and the fact that at round $t+1 < \ell$ at most $t+1$ places are fixed. We conclude that $|S| \le r/2$. Since by the "moreover" part of Claim 39 we have $|B_{t+1}| \le r/2$, $|S \cup B_{t+1}| \le r$. Then $N_{G'_t}(S \cup B_{t+1}) = N_{G'_t}(B_{t+1}) \cup N_{G_{t+1}}(S)$, hence $|N_{G'_t}(S \cup B_{t+1})| < 0.8 \cdot \Delta|B_{t+1}| + 0.8 \cdot \Delta|S| \le 0.8 \cdot \Delta|B_{t+1} \cup S|$. This contradicts the choice of B_{t+1}. □

Altogether. After $t := \Delta r/220$ answers we still are able to maintain the invariant. Thus the function $(f \circ \oplus_G)|_{\widehat{\rho}_\ell}$ is not a constant. Hence $\mathrm{D}(f \circ \oplus_G) > \Delta r/220$.

5.2 Proof of Theorem 33

We now prove Theorem 33. Let f be an n-bit function with $\mathrm{D}(f) := k \ge n^{\Omega(1)}$. We first transform f into the function $f \circ \oplus_G$ and then lift it by inner-product to yield $H := (f \circ \oplus_G) \circ \mathrm{IP}_t$ for an appropriate expander G and a parameter t. By using the following query-to-communication lifting theorem we get a lower bound on the communication complexity of H, and finally we complete the proof by showing that H is a submatrix of our target function $F := f \circ \mathrm{IP}_{\Theta(\log^2 n)}$.

Theorem 41 ([CFK+19])**.** *For every $f\colon \{0,1\}^n \to \{0,1\}$ we have $\mathrm{D}^{cc}(f \circ \mathrm{IP}_{b \log n}) = \Theta(\mathrm{D}(f) \log n)$ where $b \ge 1$ is a universal constant.*

Let $G := ([n], [m], E)$ be an $(k, \Delta, 0.9 \cdot \delta \log n)$-expander where $\Delta := \delta \log n$, $m = O(k \log n)$ and $\delta > 0$ is a universal constant, which exists by Lemma 35 and observation that $k = n^{\Omega(1)}$. We define an auxiliary two-party function $H := (f \circ \oplus_G) \circ \mathrm{IP}^m_{b \log n}$ where b is an absolute constant from Theorem 41. Note that $\mathrm{D}(f \circ \oplus_G) = \Theta(\mathrm{D}(f) \log n)$ by Theorem 36 and by Theorem 41 we get $\mathrm{D}^{cc}(H) = \Theta(\mathrm{D}(f) \log^2 n)$.

Define $F := f \circ \mathrm{IP}_{\Delta t}$ where $t := b \log n$. To complete the proof we show that F contains H as a submatrix. Consider an input $(x, y) = (x^1, \dots, x^m, y^1, \dots, y^m) \in (((\{0,1\}^t)^m)^2$ to H. Then we can write

$$H(x,y) = f\left(\bigoplus_{j \in N_G(1)} \mathrm{IP}_t(x^j, y^j), \dots, \bigoplus_{j \in N_G(n)} \mathrm{IP}_t(x^j, y^j) \right).$$

Now observe that each argument of f in the expression above is an instance of $\mathrm{IP}_{\Delta t}$, for example,

$$\bigoplus_{j \in N_G(1)} \mathrm{IP}_t(x^j, y^j) = \bigoplus_{\substack{j \in N_G(1) \\ i \in [t]}} x^j_i y^j_i = \mathrm{IP}_{\Delta t}((x^j_i)_{j \in N_G(1); i \in [t]}, (y^j_i)_{j \in N_G(1); i \in [t]}).$$

Hence we can embed H as a submatrix of F via the mapping $(x, y) \mapsto ((x^j_i)_{j \in N_G(k); i \in [t]}, (y^j_i)_{j \in N_G(k); i \in [t]})_{k \in [n]}$.

Acknowledgements

We thank Robin Kothari for asking about hardness condensation for query complexity and also informing us of Open Problem 2 of [Hat22]. We thank Suhail Sherif for helpful e-mail correspondence and Arkadev Chattopadhyay, Gil Cohen, Nathan Harms, and Rahul Santhanam for discussions. We thank Pavel Hrubes for discussions about his concurrent work [Hru24]. Finally, special thanks to Victoria Göös for serving as an uncritical sounding board during the writing of this paper. M.G., A.R., and D.S. were supported by the Swiss State Secretariat for Education, Research and Innovation (SERI) under contract number MB22.00026.

References

[ABB+17] Andris Ambainis, Kaspars Balodis, Aleksandrs Belovs, Troy Lee, Miklos Santha, and Juris Smotrovs. Separations in query complexity based on pointer functions. *Journal of the ACM*, 64(5):1–24, 2017. doi:10.1145/3106234.

[ABK16] Scott Aaronson, Shalev Ben-David, and Robin Kothari. Separations in query complexity using cheat sheets. In *Proceedings of the 48th Symposium on Teory of Computing (STOC)*, pages 863–876, 2016. doi:10.1145/2897518.2897644.

[ABRW00] Michael Alekhnovich, Eli Ben-Sasson, Alexander Razborov, and Avi Wigderson. Pseudorandom generators in propositional proof complexity. In *Proceedings of the 41st Symposium on Foundations of Computer Science (FOCS)*, pages 43–53, 2000. doi:10.1109/SFCS.2000.892064.

[ABT19] Anurag Anshu, Naresh Goud Boddu, and Dave Touchette. Quantum log-approximate-rank conjecture is also false. In *Proceedings of the 60th Symposium on Foundations of Computer Science (FOCS)*. IEEE, 2019. doi:10.1109/focs.2019.00063.

[AKK16] Andris Ambainis, Martins Kokainis, and Robin Kothari. Nearly optimal separations between communication (or query) complexity and partitions. In *Proceedings of the 31st Conference on Computational Complexity (CCC)*, volume 50, pages 4:1–4:14. Schloss Dagstuhl, 2016. doi:10.4230/LIPIcs.CCC.2016.4.

[Azu67] Kazuoki Azuma. Weighted sums of certain dependent random variables. *Tohoku Mathematical Journal, Second Series*, 19(3):357–367, 1967. doi:10.2748/tmj/1178243286.

[BBG+22] Kaspars Balodis, Shalev Ben-David, Mika Göös, Siddhartha Jain, and Robin Kothari. Unambiguous DNFs and Alon–Saks–Seymour. In *Proceedings of the 62nd Symposium on Foundations of Computer Science (FOCS)*. IEEE, feb 2022. doi:10.1109/focs52979.2021.00020.

[BdW02] Harry Buhrman and Ronald de Wolf. Complexity measures and decision tree complexity: A survey. *Theoretical Computer Science*, 288(1):21–43, 2002. doi:10.1016/S0304-3975(01)00144-X.

[BHT17] Shalev Ben-David, Pooya Hatami, and Avishay Tal. Low-sensitivity functions from unambiguous certificates. In *Proceedings of the 8th Innovations in Theoretical Computer Science Conference (ITCS)*, volume 67, pages 28:1–28:23. Schloss Dagstuhl, 2017. doi:10.4230/LIPICS.ITCS.2017.28.

[BN20] Christoph Berkholz and Jakob Nordström. Supercritical space-width trade-offs for resolution. *SIAM Journal on Computing*, 49(1):98–118, jan 2020. doi:10.1137/16m1109072.

[BS06] Joshua Buresh-Oppenheim and Rahul Santhanam. Making hard problems harder. In *Proceedings of the 21st Conference on Computational Complexity (CCC)*, pages 73–87. IEEE, 2006. doi:10.1109/ccc.2006.26.

[BvEL74] Marc Best, Peter van Emde Boas, and Hendrik Lenstra. A sharpened version of the Aanderaa-Rosenberg conjecture. Technical Report ZW 30/74, Mathematisch Centrum Amsterdam, 1974. URL: https://hdl.handle.net/1887/3792.

[BW15] Mark Braverman and Omri Weinstein. A discrepancy lower bound for information complexity. *Algorithmica*, 76(3):846–864, nov 2015. doi:10.1007/s00453-015-0093-8.

[CFK+19] Arkadev Chattopadhyay, Yuval Filmus, Sajin Koroth, Or Meir, and Toniann Pitassi. Query-to-communication lifting for BPP using inner product. In *Proceedings of the 46th International Colloquium on Automata, Languages, and Programming (ICALP)*, volume 132, pages 35:1–35:15. Schloss Dagstuhl, 2019. doi:10.4230/LIPIcs.ICALP.2019.35.

[CG88] Benny Chor and Oded Goldreich. Unbiased bits from sources of weak randomness and probabilistic communication complexity. *SIAM Journal on Computing*, 17(2):230–261, 1988. doi:10.1137/0217015.

[CGFS86] Fan Chung, Ronald Graham, Péter Frankl, and James Shearer. Some intersection theorems for ordered sets and graphs. *Journal of Combinatorial Theory, Series A*, 43(1):23–37, 1986. doi:10.1016/0097-3165(86)90019-1.

[CGS21] Arkadev Chattopadhyay, Ankit Garg, and Suhail Sherif. Towards stronger counterexamples to the log-approximate-rank conjecture. In *Proceedings of the 41st Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS)*, volume 213, pages 13:1–13:16, Dagstuhl, 2021. Schloss Dagstuhl. doi:10.4230/LIPIcs.FSTTCS.2021.13.

[CKLM19] Arkadev Chattopadhyay, Michal Koucký, Bruno Loff, and Sagnik Mukhopadhyay. Simulation theorems via pseudo-random properties. *Computational Complexity*, 28(4):617–659, 2019. doi:10.1007/s00037-019-00190-7.

[CMS20] Arkadev Chattopadhyay, Nikhil Mande, and Suhail Sherif. The log-approximate-rank conjecture is false. *Journal of the ACM*, 67(4):1–28, 2020. doi:10.1145/3396695.

[CT05] Thomas Cover and Joy Thomas. *Elements of Information Theory*. John Wiley & Sons, 2005. doi:10.1002/047174882X.ch11.

[CZ18] Xue Chen and David Zuckerman. Existence of simple extractors. Technical Report TR18-116, Electronic Colloquium on Computational Complexity (ECCC), 2018. URL: https://eccc.weizmann.ac.il/report/2018/116/.

[FPR22] Noah Fleming, Toniann Pitassi, and Robert Robere. Extremely deep proofs. In *Proceedings of the 13th Innovations in Theoretical Computer Science Conference (ITCS)*, volume 215, pages 70:1–70:23. Schloss Dagstuhl, 2022. doi:10.4230/LIPICS.ITCS.2022.70.

[Gav16] Dmitry Gavinsky. Entangled simultaneity versus classical interactivity in communication complexity. In *Proceedings of the 48th Symposium on Theory of Computing (STOC)*, pages 877–84. ACM, 2016. doi:10.1145/2897518.2897545.

[GPW18] Mika Göös, Toniann Pitassi, and Thomas Watson. Deterministic communication vs. partition number. *SIAM Journal on Computing*, 47(6):2435–2450, 2018. doi:10.1137/16m1059369.

[GS21] Anna Gál and Ridwan Syed. Upper bounds on communication in terms of approximate rank. In *Proceedings of the 16th International Computer Science Symposium in Russia (CSR)*, pages 116–130. Springer, 2021. doi:10.1007/978-3-030-79416-3_7.

[Hat22] Hamed Hatami. Problem presented at open problem session. Banff workshop on Communication Complexity and Applications, III, July 2022.

[HHH22] Lianna Hambardzumyan, Hamed Hatami, and Pooya Hatami. A counter-example to the probabilistic universal graph conjecture via randomized communication complexity. *Discrete Applied Mathematics*, 322:117–122, dec 2022. doi:10.1016/j.dam.2022.07.023.

[Hru24] Pavel Hrubes. Hard submatrices for non-negative rank and communication complexity. Technical report, Electronic Colloquium on Computational Complexity (ECCC), 2024. URL: https://eccc.weizmann.ac.il/report/2024/008/.

[Juk12] Stasys Jukna. *Boolean Function Complexity: Advances and Frontiers*, volume 27 of *Algorithms and Combinatorics*. Springer, 2012. doi:10.1007/978-3-642-24508-4.

[KN97] Eyal Kushilevitz and Noam Nisan. *Communication Complexity*. Cambridge University Press, 1997. doi:10.1017/CBO9780511574948.

[LMSS04] Nati Linial, Shahar Mendelson, Gideon Schechtman, and Adi Shraibman. Complexity measures of sign matrices. *Combinatorica*, 27, 09 2004. doi:10.1007/s00493-007-2160-5.

[LS88] László Lovász and Michael Saks. Lattices, Möbius functions and communication complexity. In *Proceedings of the 29th Symposium on Foundations of Computer Science (FOCS)*, pages 81–90. IEEE, 1988. doi:10.1109/SFCS.1988.21924.

[LS07] Troy Lee and Adi Shraibman. Lower bounds in communication complexity. *Foundations and Trends in Theoretical Computer Science*, 3(4):263–399, 2007. doi:10.1561/0400000040.

[LS09] Troy Lee and Adi Shraibman. An approximation algorithm for approximation rank. In *Proceedings of the 24th Conference on Computational Complexity (CCC)*, pages 351–357, 2009. doi:10.1109/CCC.2009.25.

[LS23] Troy Lee and Adi Shraibman. Around the log-rank conjecture. *Israel Journal of Mathematics*, 256(2):441–477, 2023. doi:10.1007/s11856-023-2517-5.

[Mid04] Gatis Midrijānis. Exact quantum query complexity for total boolean functions. Technical report, arXiv, 2004. arXiv:quant-ph/0403168.

[New91] Ilan Newman. Private vs. common random bits in communication complexity. *Information Processing Letters*, 39(2):67–71, 1991. doi:10.1016/0020-0190(91)90157-d.

[Raz16] Alexander Razborov. A new kind of tradeoffs in propositional proof complexity. *Journal of the ACM*, 63(2):1–14, 2016. doi:10.1145/2858790.

[Raz17a] Alexander Razborov. On space and depth in resolution. *Computational Complexity*, 27(3):511–559, 2017. doi:10.1007/s00037-017-0163-1.

[Raz17b] Alexander Razborov. On the width of semialgebraic proofs and algorithms. *Mathematics of Operations Research*, 42(4):1106–1134, 2017. doi:10.1287/moor.2016.0840.

[RM99] Ran Raz and Pierre McKenzie. Separation of the monotone NC hierarchy. *Combinatorica*, 19(3):403–435, 1999. doi:10.1007/s004930050062.

[RY20] Anup Rao and Amir Yehudayoff. *Communication Complexity: And Applications*. Cambridge University Press, 2020. doi:10.1017/9781108671644.

[SdW19] Makrand Sinha and Ronald de Wolf. Exponential separation between quantum communication and logarithm of approximate rank. In *Proceedings of the 60th Symposium on Foundations of Computer Science (FOCS)*. IEEE, 2019. doi:10.1109/focs.2019.00062.

[Sok20] Dmitry Sokolov. (Semi)algebraic proofs over $\{\pm 1\}$ variables. In *Proceedings of the 52nd Symposium on Theory of Computing (STOC)*, pages 78–90. ACM, 2020. doi:10.1145/3357713.3384288.

[SS22] Anastasia Sofronova and Dmitry Sokolov. A lower bound for k-DNF resolution on random CNF formulas via expansion. Technical report, Electronic Colloquium on Computational Complexity (ECCC), 2022. URL: https://eccc.weizmann.ac.il/report/2022/054.

[Tom89] Nicole Tomczak-Jaegermann. *Banach-Mazur Distances and Finite-dimensional Operator Ideals*. Pitman monographs and surveys in pure and applied mathematics. Longman Scientific & Technical, 1989. URL: https://books.google.ch/books?id=3TrvAAAAMAAJ.

[Vad12] Salil Vadhan. Pseudorandomness. *Foundations and Trends in Theoretical Computer Science*, 7(1–3):1–336, 2012. doi:10.1561/0400000010.

[Vio15] Emanuele Viola. The communication complexity of addition. *Combinatorica*, 35(6):703–747, 2015. doi:10.1007/s00493-014-3078-3.

[Yao83] Andrew Yao. Lower bounds by probabilistic arguments. In *Proceedings of the 24th Symposium on Foundations of Computer Science (SFCS)*. IEEE, 1983. doi:10.1109/sfcs.1983.30.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il