Electronic Colloquium on Computational Complexity, Report No. 175 (2018)

Lifting Theorems for Equality

Bruno Loff* Sagnik Mukhopadhyay †

Abstract

We show a *deterministic simulation (or lifting) theorem* for composed problems $f \circ \mathsf{Eq}_n$ where the inner function (the gadget) is Equality on n bits. When f is a total function on p bits, it is easy to show via a rank argument that the communication complexity of $f \circ \mathsf{Eq}_n$ is $\Omega(\deg(f) \cdot n)$. However, there is a surprising counter-example of a partial function f on p bits, such that any completion f' of f has $\deg(f') = \Omega(p)$, and yet $f \circ \mathsf{Eq}_n$ has communication complexity $O(n)$. Nonetheless, we are able to show that the communication complexity of $f \circ \mathsf{Eq}_n$ is at least $D(f) \cdot n$ for a complexity measure $D(f)$ which is closely related to the AND-query complexity of f and is lower-bounded by the logarithm of the leaf complexity of f. As a corollary, we also obtain lifting theorems for the set-disjointness gadget, and a lifting theorem in the context of parity decision-trees, for the NOR gadget.

As an application, we prove the first tight lower-bound for the deterministic communication complexity of the natural Gap-Equality problem, where Alice and Bob are each given p-many n-bit strings, with the promise that either $\frac{3}{4}$ of them are equal or $\frac{3}{4}$ of them are different, and they wish to know which is the case. We show that the complexity of this problem is $\Theta(p \cdot n)$.

Contents

Acknowledgements. We are thankful to Suhail Sherif, Mark Vinyals, and Susanna de Rezende for many helpful discussions, and Or Meir for pointing out an important bug in an earlier draft of the paper. We owe an extraordinary debt to Arkadev Chattopadhyay, an outstanding companion of many tea-break conversations on the subject of this paper.

*INESC-TEC and University of Porto, Porto, `bruno.loff@gmail.com`

†Computer Science Institute of Charles University, Prague, `sagnik@kam.mff.cuni.cz`. Most of the work is done while the author was a post-doctoral researcher at KTH Royal Institute of Technology, Stockholm.

ISSN 1433-8092

1 Introduction

In the same paper of Karchmer and Wigderson [KW90], where the notion of formula depth was shown to be equivalent to the communication complexity of their since-homonymous games, was also the first proof separating monotone NC_2 from monotone NC_1. Although not formulated explicitly in this way, their separation of these two circuit classes can be nowadays be presented as a two-part argument: (I) one first shows that the monotone Karchmer–Wigderson game for connectivity on $n^{\Theta(1)}$-node graphs is equivalent to a composition problem in communication complexity, namely $\mathsf{Fork}_n \circ \mathsf{Ind}_n$, the composition of the Fork relation on n bits with the Indexing gadget on $\log n$ bits (given to Alice) and n bits (given to Bob); and (II) one then shows lower-bounds for $\mathsf{Fork}_n \circ \mathsf{Ind}_n$ by lifting an $\Omega(\log n)$ adversarial lower-bound against decision trees trying to solve the Fork_n relation, into an $\Omega((\log n)^2)$ adversarial lower-bound against communication protocols for $\mathsf{Fork}_n \circ \mathsf{Ind}_n$.

Their seminal paper led to the following general approach for proving lower-bounds against a given complexity measure. One first (I) finds a composed problem $f \circ g$ whose communication complexity is upper-bounded by our complexity measure, and (II) one then proves a lower-bound for the communication complexity of $f \circ g$ by arguing that a lower-bound for f in a simple model (such as decision trees) will *lift* to a lower-bound against protocols for $f \circ g$.

Complexity theory has profited greatly from this approach. It appears in the celebrated Raz-McKenzie separation of the monotone NC hierarchy, [RM99] but also in the best known lower-bounds on monotone formula depth and monotone span programs [RPRC16, PR18]. Several lower-bounds on the length of proofs in various proof systems were first established using this approach [dRNV16, PR18, GGKS18], and it is the only known way of proving various separations between complexity classes in communication complexity [GPW15b, GPW15a, GLM+15, GKPW17, GJPW17, Wat17]. It may even be used for proving lower-bounds against data-structure schemes [CKLM18], and lower-bounds on the extension complexity of linear programs [KMR17, LMV, GJW18].

Owing partly to this long list of discoveries, and partly to the Karchmer-Raz-Wigderson approach [KRW95] for proving lower-bounds against (non-monotone) NC_1 [HW90, EIRS01, GMWW14, DM18], the lower-bounds community developed a specific interest in understanding the computational complexity of composition, and devoted a large effort to understanding composition problems. Under this heading we should include Sherstov's pattern matrix method [She11], and the closely related block-composition method of Shi and Zhu [SZ09], which were developed further in [Cha07, LSS08, CA08, She12b, She13, RY15], and resulted in many different applications. The problem of understanding the communication complexity of XOR functions [Raz95, HHL18, TWXZ13] is another example of a composition problem, and particularly pertinent to our case since Equality is itself a XOR function, and so composition with Equality in the communication complexity world can be seen as composition with the NOR function in the parity-decision-tree world. Even though the approach used is different to Karchmer–Wigderson's approach, work on the direct-sum and direct-product problems [JRS03, BPSW05, HJMR07, JKN08, Dru12, Pan12, JPY12, JY12, BBCR13, BRWY13a, BRWY13b, BBK+13, BR14, KLL+15, Jai15] is also a study of composition, where the outer function f in $f \circ g$ is the hardest possible: the identity function; even this case remains unsolved in various settings.

The complexity of composition is a difficult problem—not just because, generally speaking, lower-bounds are hard to establish, but also because the composition of two hard problems is sometimes not as hard as one may expect: sometimes there is a "collapse" of hardness. A classic example is the case of direct sum in communication complexity: a perfect direct sum result holds in the non-deterministic case [Lov75, KKN95], but fails to hold in the deterministic model [Orl90, FKNN95], and is still an open problem in the randomized model. The following recent example is also of great interest. In the case of deterministic decision-trees, the depth-complexity of $f \circ g$ is the product of the complexities of f and g; this both intuitive and easy to establish, and holds whether f is a total function, a partial function, or relation of any kind. But already if we look into *randomized* decision-trees, Gavinsky *et al.* [GLS18] and Sanyal [San18] show that the depth-complexity of the composition $f \circ g$ will be as high as the product of the *square-root* of the complexity of f with the complexity of g; and, surprisingly, [GLS18] exhibit a *relation* f and a function g for which this bound is tight. This "collapse" of hardness when composing relations or partial functions seems to make such problems difficult to understand. As we will see, composition with Equality provides another instance of this phenomenon.

1.1 A tea-break puzzle

Alice and Bob, two renowned complexity theorists, get together during the conference's tea break: *Communication complexity is the most successful area in complexity theory* — Alice says — *at least the natural examples of functions are really well understood.* Bob raises his eyebrows — *do you mean total functions, like Equality, or partial functions, like Gap-Hamming-distance?* — *Both* — replies Alice — *Equality has been well understood since the invention of the field [Yao79], and even Gap-Hamming-Distance is at this point understood for every gap — the constant gap case is a simple result [Woo07], and even $\frac{1}{\sqrt{n}}$ fraction gap was eventually understood [CR12, Vid13, She12a].*

Ok — Bob replied, wryly — *how about Gap-Equality? Suppose you are given p-many n-bit strings $x_1, \ldots, x_p$, and I am given $y_1, \ldots, y_p$, and we are promised that at least $\frac{3}{4}p$ of these strings are equal, or at least $\frac{3}{4}p$ of these strings are different... show me that we need to communicate $\Omega(n \cdot p)$ bits in order to know which is the case...*

Alice thinks for a while — *I know, we can do it via a rank argument. Gap-Equality is the composition $f \circ \mathsf{Eq}_n$, where f is the Gap-Hamming-distance function and Eq_n is Equality on n bits; now, by Minsky and Papert's symmetrization method [MP88, BdW02], we may show that the degree of any completion f' of Gap-Hamming-distance is $\Omega(p)$, and Eq_n has rank 2^n, so the rank of the communication matrix of $f' \circ \mathsf{Eq}_n$ is $2^{\Omega(pn)}$* (see Lemma 6).

Bob nods — *That is true, but that only means that any protocol for $f' \circ \mathsf{Eq}_n$ needs $\Omega(n \cdot p)$ bits. However, even though a protocol for Gap-Equality does give you a completion of the partial communication matrix for $f \circ \mathsf{Eq}_n$, this completion does not need to be in the composed form $f' \circ g$ where f' is a completion of f...*

At this point Alice does not know what to answer, and rightly so. We will see below an example of a p-bit partial function f, such that any completion of f must have degree $\Omega(p^{1/4})$, and yet the communication complexity of $f \circ g$ is $O(n)$, instead of $\Omega(n \cdot p^{1/4})$, which is what one might expect from a rank argument. The protocol that shows this will precisely take advantage of the fact that a completion of $f \circ g$ does not have to be of the form $f' \circ g$ for some completion of f' of f.

A solution to Bob's tea-break puzzle appears as Corollary 18, in page 7. Using our lifting theorem (Theorem 13, page 6) the desired lower-bound of $\Omega(n \cdot p)$ is a 2-line argument.

1.2 Composition with Equality

In this work, we answer a question pertaining to the communication complexity of composition of Boolean relations with the Equality gadget. Before stating the question and our main results, we explain the context surrounding this question. We begin with some definitions.

- Define the Fork relation: $\mathsf{Fork}_p = \{(z, i) \in \{0,1\}^p \times \{0, 1, \ldots, p\} \mid z_i = 1, z_{i+1} = 0\}$, where we use $z_0 = 1$ and $z_{p+1} = 0$, *i.e.*, we are given p bits and wish to find a "forking point", a position i where a 1-bit flips into a 0-bit. If $z = 0^p$ we must output $i = 0$ and if $z = 1^p$ we must output $i = p$.

- Let $\mathsf{Ind}_n : [n] \times \{0,1\}^n \to \{0,1\}$ denote the two-player Indexing function on n-bits, so that $\mathsf{Ind}_n(x, y) = y_x$.

- Then $\mathsf{Fork}_p \circ \mathsf{Ind}_n$ denotes the composed Boolean relation:

$$\mathsf{Fork}_p \circ \mathsf{Ind}_n = \{(\bar{x}; \bar{y}; i) \in [n]^p \times (\{0,1\}^n)^p \times \{0, 1, \ldots, p\} \mid (y_i)_{x_i} = 1, (y_{i+1})_{x_{i+1}} = 0\}.$$

- Let $\mathsf{Eq}_n : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ denote two-player Equality on n-bits, so that $\mathsf{Eq}_n(x, y) = 1$ iff $x = y$.

- Then $\mathsf{Fork}_p \circ \mathsf{Eq}_n$ denotes the composed Boolean relation:

$$\mathsf{Fork}_p \circ \mathsf{Eq}_n = \{(\bar{x}; \bar{y}; i) \in (\{0,1\}^n)^p \times (\{0,1\}^n)^p \times \{0, 1, \ldots, p\} \mid x_i = y_i, x_{i+1} \neq y_{i+1}\}.$$

- Let $F \subseteq \mathcal{A} \times \mathcal{B} \times \mathcal{C}$ be a relation. The *deterministic communication complexity of F*, $\mathsf{D}^{cc}(F)$, is the minimum communication cost of a protocol for solving the communication problem where Alice is given $a \in \mathcal{A}$, Bob is given $b \in \mathcal{B}$, and they wish to find c such that $(a, b, c) \in F$, whenever one such c exists (see [KN97], Chapter 5).

- Let $f \subseteq \{0,1\}^p \times \mathcal{C}$ be a relation. The *deterministic query complexity of* f, $\mathsf{D}^{dt}(f)$, is the minimum number of queries made by a deterministic decision-tree which, given query access to $z \in \{0,1\}^p$, finds a $c \in \mathcal{C}$ such that $(z,c) \in F$, whenever one such c exists.

In STOC'88, Karchmer and Wigderson [KW90] presented a proof that connectivity is not in monotone NC_1. At the heart of their result was an argument which may be reinterpreted as a proof of the following theorem:

Theorem 1 (Karchmer and Wigderson, [KW90])**.** $\mathsf{D}^{cc}(\mathsf{Fork}_p \circ \mathsf{Ind}_n) = \Omega((\log n) \cdot \log p)$.

In Structures'91, the conference now known as CCC, Grigni and Sipser [GS91] provided an alternative proof that connectivity is not in monotone NC_1. Their proof uses Eq in place of Ind, and this allows for a simpler argument:

Theorem 2 (Grigni and Sipser, [GS91])**.** $\mathsf{D}^{cc}(\mathsf{Fork}_p \circ \mathsf{Eq}_n) = \Omega(n \cdot \log p)$.

It is not hard to see that Theorem 2 implies Theorem 1, by reducing $\mathsf{Eq}_{\log n}$ to Ind_n. Later, in FOCS'97, Raz and McKenzie [RM99] separated the entire monotone NC hierarchy. At the heart of their proof was an argument for a vast generalization of Theorem 1:

Theorem 3 (Raz and McKenzie, [RM99])**.** *For any Boolean relation $f \subseteq \{0,1\}^p \times \mathcal{C}$, whenever $n \geq p^{20}$,* $\mathsf{D}^{cc}(f \circ \mathsf{Ind}_n) = \Omega\left((\log n) \cdot \mathsf{D}^{dt}(f)\right)$.

Theorem 3 was not stated with such generality in [RM99], but appeared in this form in a recent work of Göös, Pitasi and Watson [GPW15a]. Theorem 3 has been the basis of several papers [GPW15a, CKLM17, Koz18].

Knowing the above history, one naturally comes to the question of whether one can prove a similar generalization for Grigni and Sipser's Theorem 2, i.e., whether we can prove the conjecture:

Conjecture 4. *For any Boolean relation $f \subseteq \{0,1\}^p \times \mathcal{C}$,* $\mathsf{D}^{cc}(f \circ \mathsf{Eq}_n) = \Omega\left(n \cdot \mathsf{D}^{dt}(f)\right)$.

Very general lifting theorems may be proven using rank arguments, and the current state of the art [RPRC16, PR18] is a lifting of the *Nullstellensatz degree* of f to the rank of $f \circ g$, which works for a large class of gadgets g having a certain algebraic property[1]. This property, unfortunately, does not hold for equality. However, there is an *ad-hoc* degree-to-rank lifting theorem which works for equality, when f is a total function, and which is in the same spirit as [RPRC16, PR18]. It uses the following characterization:

Proposition 5 ([BC99])**.** *If h is a Boolean function and F is the communication matrix of $h \circ \mathsf{XOR}_2$, then* $\mathsf{rank}(F) = \|h\|_0$.

Above, $\mathsf{rank}(F)$ is the real rank of the communication matrix of F, and $\|h\|_0$ is the Fourier sparsity (the number of non-zero Fourier coefficients) of h. We can view $f \circ \mathsf{Eq}_n$ as an XOR_2 function, $f \circ \mathsf{NOR}_n \circ \mathsf{XOR}_2$. The following observation is easy to prove (see Section 7), and is in the same spirit as .

Lemma 6. *For every $f : \{+1,-1\}^p \to \{+1,-1\}$ with $\deg(f) \geq 1$, and every $g : \{+1,-1\}^n \to \{+1,-1\}$, we have $\|f \circ g\|_0 \geq (\|g\|_0 - 1)^{\deg(f)}$.*

Lemma 6 implies that $\|f \circ \mathsf{NOR}_n\|_0 = \Omega(2^{\deg(f) \cdot n})$, since $\|\mathsf{NOR}_n\|_0 = 2^n$. By the rank-lower bound for communication complexity, we thus have $\mathsf{D}^{cc}(f \circ \mathsf{Eq}) \geq \Omega(\deg(f) \cdot n)$. Now we can use the following connection between $\deg(f)$ and $\mathsf{D}^{dt}(f)$.

Proposition 7 (Nisan and Smolensky, see [BdW02])**.** $\deg(f) = \Omega(\mathsf{D}^{dt}(f)^{1/4})$.

Combining the three above facts, we get that when f is a total Boolean function, then $\mathsf{D}^{cc}(f \circ \mathsf{Eq}_n) = \Omega(\mathsf{D}^{dt}(f)^{1/4} \cdot n)$. This easy-to-prove result is similar to Conjecture 4, except for the 1/4 loss in the exponent, and works for all total functions. But surprisingly, when allow f to be a partial function, Conjecture 4 is false! The following counter-example was given to us by Arkadev Chattopadhyay, Suhail Sherif, and Mark Vinyals. Let $f \subseteq \{0,1\}^p \times \{0,1\}$ be the relation

$$f = \{(z,1) \mid |z| = p \text{ or } |z| < p-1\} \cup \{(z,0) \mid |z| = p-1 \text{ or } |z| < p-1\},$$

[1] These results are explained in Robert Robere's excellent PhD thesis [Rob18]. The algebraic property appears in Section 5.1.

i.e., we are given a Boolean string $z \in \{0,1\}^p$, and wish to distinguish the case when z has Hamming weight p from the case when z has Hamming weight $p-1$. It is easy to show that $\mathsf{D}^{dt}(f) \geq p$: an adversary keeps answering 1 to all queries, and $f(z)$ will remain unknown until the very last query. This adversary also shows that $\mathsf{D}^{dt}(f') \geq p$ for any "completion" of f, i.e. any total function $f' : \{0,1\}^p \to \{0,1\}$ which agrees with f on the inputs with Hamming weight p or $p-1$; and hence $\deg(f') = \Omega(p^{1/4})$ for any such f', by Proposition 7. So one might mistakenly hope, like Alice did in Section 1.1, that a rank/degree argument would serve to prove a lower-bound of $\Omega(p^{1/4} \cdot n)$ for $f \circ \mathsf{Eq}_n$.

However, a protocol for $f \circ \mathsf{Eq}_n(x_1, \ldots, x_p; y_1, \ldots, y_p)$ may proceed as follows. Think of each of Alice and Bob's inputs for $f \circ \mathsf{Eq}_n$ as a matrix with p rows and n columns. Then let $a \in \{0,1\}^n$ be the XOR of each column of Alice's input, and $b \in \{0,1\}^n$ be the XOR of each column of Bob's input. Then Alice sends a to Bob, and Bob replies whether $a = b$. Now, if every x_i equals the corresponding y_i, then clearly $a = b$; and if every x_i equals y_i, except for a single value of $i \in [p]$, then there must exist a $j \in [n]$ such that $a_j \neq b_j$. It then holds that $\mathsf{D}^{cc}(f \circ \mathsf{Eq}_n) \leq n + 1$, and so Conjecture 4 is false. Remarkably, this seems to suggest that rank/degree arguments will fail to hold.

This counter-example also shows that $\mathsf{Eq}_{\log n}$ behaves differently from Ind_n, when used as the inner function in a composition — indeed Theorem 3 implies that $\mathsf{D}^{cc}(f \circ \mathsf{Ind}_n) \geq p \log n$, which is strictly higher when $p = \omega(1)$. The difference between Equality and Indexing may be further explained with the help of a recent paper of Chattopadhyay, Koucký, and the authors [CKLM17]. There it is shown that a theorem like Conjecture 4 will hold for any inner function g, in place of Eq_n, which admits certain *hitting distributions*[2]. As it turns out, all gadgets for which we could prove a deterministic simulation theorem, namely, Indexing [RM99], Inner-product and gap-Hamming [CKLM17], and several others [Koz18], all admit such hitting distributions. But it may be seen that although Equality has a 0-hitting distribution, it fails to have any 1-hitting distribution.

The existence of such a counter-example was surprising to us, because in the case of the Fork relation, the Karchmer–Wigderson theorem and Grigni–Sipser theorem behave the same way (by lifting a decision-tree adversary for the Fork relation). The main purpose of this work was to understand what is happening.

1.3 Almost Conjecture 4

We will be able to prove a simulation theorem for composition with Equality, but for a notion different than decision-tree depth. In order to avoid long preliminaries for now, we postpone the full list of our results until the end of Section 2. However, one of our results is sufficiently close to what was already discussed, that it may be easily stated in the present section, and may thus serve as motivation for the remainder.

For a given relation $f \subseteq \{0,1\}^p \times \mathcal{C}$, let $\mathsf{L}^{dt}(f)$ denote the smallest number of leaves of any deterministic decision-tree which, given query access to $z \in \{0,1\}^p$, finds a $c \in \mathcal{C}$ such that $(z, c) \in F$, whenever such a c exists. Notice that $\mathsf{D}^{dt}(f) \geq \log \mathsf{L}^{dt}(f)$, and so if Conjecture 4 were true, a consequence would be that $\mathsf{D}^{cc}(f \circ \mathsf{Eq}_n) = \Omega\left(n \cdot \log \mathsf{L}^{dt}(f)\right)$. The following theorem, thus, may be considered a weak variant of Conjecture 4:

Theorem 8 (Lifting for $\log \mathsf{L}^{dt}$)**.** *For any Boolean relation $f \subseteq \{0,1\}^p \times \mathcal{C}$, whenever $n \geq 100 \cdot \log p$,*

$$\mathsf{D}^{cc}(f \circ \mathsf{Eq}_n) = \Omega\left(n \cdot \frac{\log \mathsf{L}^{dt}(f)}{\log p}\right).$$

1.4 Organization

In Section 2 we state the definitions required to understand the statements of our results, and then state all our results in full; in this section we give the first new concept required by our results, namely the notion of 0-query complexity. In Section 3, we introduce the combinatorial invariants required to prove our main result, including the notion of *thickness*, which comes from Raz and McKenzie [RM99, GPW15a, CKLM17], but also the notion of *square*, which is the second new concept required by our proofs. In Section 4 we prove a *projection lemma*—the crucial lemma required to prove the simulation theorem—which is then proven in

[2]A (δ, h)-hitting rectangle-distribution (for $\delta \in (0,1)$ and $h \in \mathbb{N}$) is a distribution over rectangles such that a random rectangle from this distribution will intersect any 2^{-h}-large rectangle with probability $\geq 1 - \delta$. By a Boolean function g having (δ, h)-hitting monochromatic rectangle-distributions, we mean that there are two (δ, h)-hitting rectangle-distributions σ_0 and σ_1, such that σ_c only samples rectangles which are c-monochromatic with respect to g.

Section 5. In Section 6, we prove all the remaining results mentioned in Section 2. Lemma 6 is proven in Section 7.

2 Preliminaries, and precise statements of our results

In this section we provide basic notations and precise statements of all our results.

We will assume the reader is familiar with various basic concepts pertaining to complexity of Boolean functions, namely: decision trees, query complexity, leaf complexity, protocol trees, communication complexity, combinatorial rectangles, and Fourier analysis of Boolean functions. See [KN97, Juk12] for reference.

We will be studying the decision-tree complexity of relations. A *Boolean relation* f is a subset of $\{0,1\}^p \times \mathcal{C}$ where $\mathcal{C}$ is a finite set; associated with f is the search problem where we are given a string $z \in \{0,1\}^p$, and wish to find an element $c \in \mathcal{C}$ such that $(z, c) \in \mathcal{C}$, if such an element exists.[3] If to each z corresponds exactly one c, we call f a *total Boolean function*.

For a given Boolean relation, we let $\mathsf{D}^{dt}(f)$, called the *query complexity of* f, be the minimum height of T, taken over deterministic decision-trees T which solve the search problem associated with f. We let $\mathsf{L}^{dt}(f)$, called the *leaf complexity of* f, be the minimum number of leaves of T, again taken over deterministic decision-trees T which solve the search problem associated with f.

We will also be interested in the communication complexity of relations. A *two-player relation* F is a subset $F \subseteq \mathcal{A} \times \mathcal{B} \times \mathcal{C}$ where $\mathcal{A}, \mathcal{B}, \mathcal{C}$ are finite sets; associated with F is the communication problem where Alice is given $a \in \mathcal{A}$, Bob is given $b \in \mathcal{B}$, and they wish to find $c \in \mathcal{C}$ such that $(a, b, c) \in F$, if one such c exists. If $g \subseteq \mathcal{A} \times \mathcal{B} \times \{0, 1\}$ is a two-player relation such that to each pair $(a, b) \in \mathcal{A} \times \mathcal{B}$ corresponds exactly one $c \in \{0,1\}$ with $(a, b, c) \in g$, we call g a *gadget*. The Equality and Indexing function defined in page 2 are examples of gadgets. A third example is the *Set-disjointness function* $\mathsf{Disj}_n : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, where $\mathsf{Disj}_n(x, y) = 0$ iff $x_i = y_i = 1$ for some $i \in [n]$.

For a given two-player relation $F \subseteq \mathcal{A} \times \mathcal{B} \times \mathcal{C}$, we let $\mathsf{D}^{cc}(F)$, called the *communication complexity of* F, be the height of the shortest deterministic protocol-tree for solving the communication problem associated with F.

The *composition* of a Boolean relation $f \subseteq \{0,1\}^p \times \mathcal{C}$ with a gadget $g : \mathcal{A} \times \mathcal{B} \to \{0, 1\}$ is the two-player relation $f \circ g \subseteq \mathcal{A}^p \times \mathcal{B}^p \times \mathcal{C}$, given by

$$f \circ g = \{(a_1, \ldots, a_p; b_1, \ldots, b_p; c) \mid (g(a_1, b_1) \ldots g(a_p, b_p), c) \in f\}.$$

The following definition is crucial to our result and, to our knowledge, has not been used prior to this work:

Definition 9. Given a deterministic decision-tree T over $\{0,1\}^p$, the 0-*depth of* T is the maximum number of queries which are answered 0, in any root-to-leaf path of T. The 0-*query complexity* of f, denoted $\mathsf{D}_0^{dt}(f)$, to be the smallest 0-depth of T, taken over deterministic decision-trees T which solve the search problem associated with f.

It is unusual to make a query complexity notion depend on the specific outcome of the queries, instead of just the number of queries. However, the above notion is closely related to a notion analogous to parity decision-trees. Indeed, we may define AND decision-trees to be like parity decision-trees, but where the algorithm is allowed the query an AND of the input bits, instead of a parity of the input bits:

Definition 10. An AND decision-tree over $\{0,1\}^p$ is a rooted tree where each internal node v is labeled by a *set of variables* $Q_v \subseteq [p]$ and each edge is labeled 0 or 1. As in the case of deterministic decision-tree, the execution of T on an input $z \in \{0,1\}^p$ traces a path in this tree: at each internal node v the execution is given the value of the conjunction $q = \bigwedge_{i \in Q_v} z_i$, and follows the edge labeled q into one of v's children. With each node v of the tree we may associate the set $S_v \subseteq \{0,1\}^p$ of those inputs whose execution follows the path down to the node v; the set S_v is given by a system of conjunctive equations.

[3] Although when considering functions the difference between a total function and a partial function (a promise problem) is very important, this distinction is irrelevant when thinking more generally about relations, at least in computational models which are guaranteed to produce an output. Indeed, a partial Boolean relation $f \subseteq \{0,1\}^n \times \mathcal{C}$ may be replaced by the total Boolean relation $f' = f \cup \{(x, c) \in \{0,1\}^n \times \mathcal{C} \mid (x, c') \notin f \text{ for any } c' \in \mathcal{C}\}$, meaning if the input is outside the promise we allow the algorithm to output anything.

An AND decision-tree over $\{0,1\}^p$ is said to solve the search problem associated with a Boolean relation $f \subseteq \{0,1\}^p \times \mathcal{C}$ if, for every leaf v, there exists a choice of $c \in \mathcal{C}$ such that $(z,c) \in f$ for every $z \in S_v$.

Then, the AND-*query complexity* of f, denoted $\mathsf{D}^{dt}_{\mathsf{AND}}(f)$, is defined as the minimum depth of T, taken over AND decision-trees T which solve the search problem associated with f.

We are then able to establish the following relationship:

Lemma 11. *Let $f \subseteq \{0,1\}^p \times \mathcal{C}$ be any Boolean relation. Then*

$$\mathsf{D}^{dt}_{\mathsf{AND}}(f) \geq \mathsf{D}^{dt}_{0}(f) \geq \frac{\mathsf{D}^{dt}_{\mathsf{AND}}(f)}{\lceil \log(p+1) \rceil}$$

Since these measures are within a $\log p$ factor of each other, it is possible to think of the more natural $\mathsf{D}^{dt}_{\mathsf{AND}}(f)$ as a proxy for $\mathsf{D}^{dt}_{0}(f)$. The proof is simple and appears in Section 6.

The following lemma connects 0-query complexity with leaf complexity. The proof applies Dinur and Meir's notion of *fortification* [DM18] to decision trees, and appears in Section 6.

Lemma 12. *Let $f \subseteq \{0,1\}^p \times \mathcal{C}$ be any Boolean relation. Then*

$$\mathsf{D}^{dt}_{0}(f) \geq \log \frac{\mathsf{L}^{dt}(f)}{\log p}.$$

Lifting theorems for Equality. Our main result is a simulation theorem which lifts 0-query complexity of a Boolean relation $f \subseteq \{0,1\}^p \times \mathcal{C}$ to the communication complexity $f \circ \mathsf{Eq}_n$:

Theorem 13 (Lifting for D^{dt}_{0})**.** *Let $f \subseteq \{0,1\}^p \times \mathcal{C}$ be any Boolean relation. Then, whenever $n \geq 100 \cdot \log p$,*

$$\mathsf{D}^{cc}(f \circ \mathsf{Eq}_n) = \Omega\left(n \cdot \mathsf{D}^{dt}_{0}(f)\right).$$

The proof of Theorem 13 uses the notion of thickness and average-thickness from Raz-McKenzie [RM99], and a new invariant, called a *square*, which is inspired by Grigni-Sipser [GS91]. These notions are presented in Section 3.

Remark 14. It is not hard to verify that $\mathsf{D}^{dt}_{0}(f) = 1$ when f is the counter-example to Conjecture 4, which we described in Section 1.2: a decision tree for f queries coordinates one at a time until it finds the first 0. Then it follows from Theorem 13 that the protocol for $f \circ \mathsf{Eq}_n$ appearing in page 3 is optimal, up to constant factors.

Theorem 8 follows from Theorem 13 and Lemma 12. Theorem 13 and Lemma 11 give us the following:

Corollary 15 (Lifting for $\mathsf{D}^{dt}_{\mathsf{AND}}$)**.** *Let $f \subseteq \{0,1\}^p \times \mathcal{C}$ be any Boolean relation. Then, whenever $n \geq 100 \cdot \log p$,*

$$\mathsf{D}^{cc}(f \circ \mathsf{Eq}_n) = \Omega\left(n \cdot \frac{\mathsf{D}^{dt}_{\mathsf{AND}}(f)}{\log p}\right).$$

Lifting theorems for Set-disjointness. By a simple reduction, we are also able to show the first lifting theorem known for set-disjointness. Indeed, we may reduce an instance of Eq_n to an instance of Disj_{2n}. Alice maps each of her bits x_i into the pair of bits $a_i = (1-x_i)x_i$, and Bob maps each of his bits y_i into $b_i = y_i(1-y_i)$; it now holds that $x_i = y_i$ iff a_i and b_i are disjoint, and hence $\mathsf{Eq}_n(x,y) = \mathsf{Disj}_{2n}(a,b)$. As a corollary, we find:

Corollary 16 (Lifting for disjointness)**.** *Let $f \subseteq \{0,1\}^p \times \mathcal{C}$ be a Boolean relation and $n \geq 100 \cdot \log p$. Then*

$$\mathsf{D}^{cc}(f \circ \mathsf{Disj}_n) = \Omega\left(n \cdot \mathsf{D}^{dt}_{0}(f)\right).$$

Naturally, Theorem 8 and Corollary 15 will hold for Set-disjointness.

Lifting theorems for parity decision-trees. A composition with Equality, $f \circ \mathsf{Eq}_n$, is a XOR function $f \circ \mathsf{NOR}_n \circ \mathsf{XOR}_2$. It is well known and easy to see that $\mathsf{D}^{cc}(F \circ \mathsf{XOR}_2) \leq \mathsf{D}^{dt}_{\oplus}(F)$ [HHL18], where $\mathsf{D}^{dt}_{\oplus}(F)$ is the parity-query complexity of F. Hence a consequence of our lifting theorem for Equality in communication complexity is also a lifting theorem for the NOR function, with respect to parity decision-trees:

Corollary 17. *For any Boolean relation $f \subseteq \{0,1\}^p \times \mathcal{C}$, whenever $n \geq 100 \cdot \log p$,*

$$\mathsf{D}^{dt}_{\oplus}(f \circ \mathsf{NOR}_n) = \Omega\left(n \cdot \mathsf{D}^{dt}_0(f)\right).$$

It may be seen that $\mathsf{D}^{dt}_0(f)$ cannot be replaced by $\mathsf{D}^{dt}(f)$, by the same counter-example f of page 3.

A solution to the tea-break puzzle. A lifting theorem such as Theorem 13 is a powerful tool for proving lower-bounds in communication complexity. The theorem is very general and many such results may be proven, but let us here give an example of lower-bound for a natural problem in communication complexity.

Consider the promise problem $\mathsf{GEq}_{p,n}$ where Alice and Bob are each given p-many n-bit strings, with the promise that either at least $\frac{3}{4}p$ of the strings are equal, or at least $\frac{3}{4}p$ of the strings are different, and they wish to know which is the case.

Take the partial Constant-gap-Hamming function $\mathsf{GH}_p : \{0,1\}^p \to \{0,1\}$ where $\mathsf{GH}_p(z) = 1$ if the Hamming weight $|z| \geq \frac{3}{4}p$ and $\mathsf{GH}_p(z) = 0$ if $|z| \leq \frac{1}{4}p$. This is a partial function, so we may not use Lemma 6 to prove a lower-bound on it. However an adversary may answer 0 $\frac{p}{4}$ times before fixing $\mathsf{GH}_p(z)$; hence $\mathsf{D}^{dt}_0(\mathsf{GH}_p) \geq \frac{p}{4}$, and it follows immediately from Theorem 13:

Corollary 18. *Whenever $n \geq 100 \cdot \log p$, $\mathsf{D}^{cc}(\mathsf{GEq}_{p,n}) = \Omega(n \cdot p)$.*

To the best of our knowledge, there is currently no other way to establish this lower-bound.

3 Thickness and squares

Notation. If p is a natural number, we write $[p]$ for the set $\{1, \ldots, p\}$. For sets A and B, we use $A \to B$ to denote the set of total functions from A to B. We write $f : A \to B$ to mean $f \in (A \to B)$. We also use B^A to denote the set of total functions from A to B, but in this case we think of them as A-indexed sequences of elements from B, and if we first write $f \in B^A$, instead of $f : A \to B$, we will later write f_a instead of $f(a)$. If $f : A \to B$ (or $f \in B^A$) and $A' \subseteq A$, then $f\big|_{A'}$ is the restriction of f to A'. A disjoint union is denoted by $\uplus$, i.e. $A \uplus B$ denotes the union of two disjoint sets A and B.

We will look at sets $A \subseteq (\{0,1\}^n)^{[p]}$, and we will often want to think of some set of coordinates $I \subseteq [p]$ as being *alive*, and the corresponding complement $D = [p] \setminus I$ will be the set of *dead* coordinates. We be working with partial assignments of elements from $(\{0,1\}^n)^{[p]}$, which can be encoded as total functions from I to $\{0,1\}^n$. Hence the following two definitions will be helpful.

Definition 19 (Join)**.** Let $n \geq 1$ and $p \geq 2$ be integers, $\varnothing \neq I \subsetneq [p]$ and $D = [p] \setminus I$.

If $s' \in (\{0,1\}^n)^I$ and $s'' : (\{0,1\}^n)^D$, then their *join* $s' \times s'' \in (\{0,1\}^n)^{[p]}$ is given by:

$$(s' \times s'')_i = \begin{cases} s'_i & \text{if } i \in I \\ s''_i & \text{if } i \in D. \end{cases}$$

This notation is extended to subsets of $(\{0,1\}^n)^I$ and $(\{0,1\}^n)^D$ in the natural way.

If $i \in I \subseteq [p]$, $s' \in \{0,1\}^n$ and $s'' \in (\{0,1\}^n)^{I \setminus \{i\}}$, then their *join at* i is the sequence $s' \times_i s'' \in (\{0,1\}^n)^I$ with $(s' \times_i s'')_i = s'$, and $\forall j \in I \setminus \{i\}$ $(s' \times_i s'')_j = s''_j$.

Definition 20. Let $n \geq 1$ and $p \geq 2$ be integers, $I \subseteq [p]$, $i \in I$ and $S \subseteq (\{0,1\}^n)^I$. We define the *projections*: $S_i = \{s_i \mid s \in S\} \subseteq \{0,1\}^n$ and $S_{\neq i} = \{s\big|_{I \setminus \{i\}} \mid s \in S\} \subseteq (\{0,1\}^n)^{I \setminus \{i\}}$.

Likewise if $\varnothing \neq E \subset I$, we define $S_E = \{s\big|_E \mid s \in S\} \subseteq (\{0,1\}^n)^E$ and, for each $s'' \in (\{0,1\}^n)^{I \setminus E}$, the extensions of s'' in S is the set $\mathsf{Ext}_S(s'') = \{s' \in (\{0,1\}^n)^E \mid s' \times s'' \in S\}$.

For a subset $U \subseteq \{0,1\}^n$, the *restriction of S to U at coordinate i* is the set $S^{i,U} = \{s \in S \mid s(i) \in U\}$. We will also write $S^{i,U}_{\neq i}$ for the set $(S^{i,U})_{\neq i}$ (i.e. we first restrict the i-th coordinate then project onto the remaining coordinates in I): $S^{i,U}_{\neq i} = \{s\big|_{I \setminus \{i\}} \mid s \in S, s_i \in U\}$.

3.1 Thickness and its properties

The notion of *thickness* was first used by Raz and McKenzie in [RM99], and is by now a well-known notion. But whereas previously the notion of thickness was only looked at with respect to all coordinates simultaneously, we will be interested in the notion of thickness with respect to a subset of coordinates. This difference is non-essential, and all the relevant properties are proven mutatis mutandis.

Definition 21 (Aux graph, average and min-degrees)**.** Let $n \geq 1, p \geq 2$ be integers, $I \subseteq [p]$, and $S \subseteq (\{0,1\}^n)^I$. For each $i \in I$, the aux graph $G(S,i)$ is the bipartite graph with left-side vertices S_i, right-side vertices $S_{\neq i}$ and edges corresponding to the set S, i.e., (s', s'') is an edge iff $s' \times_i s'' \in S$.

We define the average degree of $G(S,i)$ to be the average right-degree: $d_{\text{avg}}(S,i) = \frac{|S|}{|S_{\neq i}|}$, and the min-degree of $G(S,i)$, to be the minimum right-degree: $d_{\min}(S,i) = \min_{s'' \in S_{\neq i}} |\mathsf{Ext}_S(s'')|$.

Definition 22 (Thickness and average thickness)**.** Let $n \geq 1, p \geq 2$ be integers, $\varnothing \neq F \subseteq I \subseteq [p]$, and $S \subseteq (\{0,1\}^n)^I$. Then S is called *τ-thick on F* if $d_{\min}(S,i) \geq \tau \cdot 2^n$ for all $i \in F$. (By convention an empty set S is τ-thick.) Similarly, S is called *φ-average-thick on F* if $d_{\text{avg}}(S,i) \geq \varphi \cdot 2^n$ for all $i \in F$. For $p = 1$, set S is τ-thick if $|S| \geq \tau \cdot 2^n$.

We will need the following two lemmas.

Lemma 23 (Average thickness implies thickness)**.** *Let $n \geq 1, p \geq 2$ be integers, $\varnothing \neq F \subseteq I \subseteq [p]$, and $S \subseteq (\{0,1\}^n)^I$. If S is φ-average-thick on F, then for every $\delta \in (0,1)$ there is a subset $S' \subseteq S$ which is $\frac{\delta}{p}\varphi$-thick on F and has $|S'| \geq (1-\delta) \cdot |S|$.*

Proof. The proof of this lemma is similar to the proof of Theorem 3.9 of [CKLM17]. We include it here for completeness. The set S' is obtained by running Algorithm 1, which greedily prunes S until there are no small-degree nodes in any $G(S,i)$.

Algorithm 1

Set $S^0 = S$, $j = 0$.
while $d_{\min}(S^j, i) < \frac{\delta}{p}\varphi \cdot 2^n$ for some $i \in F$ **do**
    Let s' be a right node of $G(S^j, i)$ with (non-zero) degree less than $\frac{\delta}{p}\varphi \cdot 2^n$.
    Set $S^{j+1} = S^j \setminus \{s'\} \times_i \mathsf{Ext}(s')$, i.e., remove every extension of s'. Increment j.
end while
Set $S' = S^j$.

The total number of iteration of the algorithm is at most $\sum_{i \in [p]} |S_{\neq i}|$. (We remove at least one node in some $G(S^j, i)$ in each iteration which was a node also in the original $G(S,i)$.) So the number of iterations is at most

$$\sum_{i \in F} |S_{\neq i}| = \sum_{i \in F} \frac{|S|}{d_{\text{avg}}(S,i)} \leq \frac{p|S|}{\varphi \cdot 2^n}.$$

As the algorithm removes at most $\frac{\delta}{p}\varphi \cdot 2^n$ elements of S in each iteration, the total number of elements removed from S is at most $\delta|S|$, so $|S'| \geq (1-\delta)|S|$. Hence, the algorithm always terminates with a non-empty set S' that must be $\frac{\delta}{p}\varphi$-thick. $\square$

A recent example by Kozachinskiy [Koz18] shows that the $\frac{1}{p}$ loss in Lemma 23 is needed. This loss is the core reason why we need the gadget to have size $n = \Omega(\log p)$ in Theorem 13.

Lemma 24. *Let $n \geq 1, p \geq 2$ be integers, $i \in F \subseteq I \subseteq [p]$, and $S \subseteq (\{0,1\}^n)^I$ be τ-thick on F. Then for any set $U \subseteq \{0,1\}^n$, $S^{i,U}_{\neq i}$ will also be τ-thick on $F \setminus \{i\}$, and $S^{i,U}_{\neq i}$ will be empty iff $U \cap S_i$ is empty.*

Proof. This proof of similar to the proof of Theorem 3.10 of [CKLM17]. Notice that $S^{i,U}_{\neq i}$ is non-empty iff $U \cap S_i$ is non-empty. Consider the case of $|F| \geq 3$. Let $s \in S$, where $s_i \in U$. and set $s' = s_{\neq i}$, so that s' is an arbitrary element of $S^{i,U}_{\neq i}$. Take any $j \in F \setminus \{i\}$. Every extension of $s_{\neq j}$ in S is also an extension of $s'_{\neq j}$ in

$S^{i,U}_{\neq i}$, i.e., $\mathsf{Ext}_S(s_{\neq j}) \subseteq \mathsf{Ext}_{S^{i,U}_{\neq i}}(s'_{\neq j'})$, hence the degree of s' in $G(S^{i,U}_{\neq i}, j')$ is at least the degree of s in $G(S, j)$ which is at least $\tau \cdot 2^n$. Hence, $S^{i,U}_{\neq i}$ is τ-thick on $F \setminus \{i\}$.

To see the case $|F| = 2$, assume there is some string $s' \in S_{\neq i}$ which has some extension $s'' \in U$; but S itself is τ-thick on F, so there have to be at least $\tau \cdot 2^n$ many such s', which will then all be in $S^{i,U}_{\neq i}$. $\square$

3.2 Squares

We will be interested in rectangles $R = A \times B$, where A, B both are subsets of $(\{0,1\}^n)^{[p]}$. But some pairs (x, y) of R will be more interesting than others. Throughout our constructions we will maintain such a rectangle $R = A \times B$, and also a set $I \subseteq [p]$ of *live* coordinates, with a corresponding set $D = [p] \setminus I$ of *dead* coordinates, and also a family $S \subseteq (\{0,1\}^n)^I$, for which one can do the following:

For any $s \in S$, there exist $\alpha(s), \beta(s) \in (\{0,1\}^n)^D$, such that $s \times \alpha(s) \in A$ and $s \times \beta(s) \in B$, and, furthermore, $\alpha(s)_i \neq \beta(t)_i$ for every $s \in S, t \in S, i \in I$.

I.e., given any s in S, which is a way of filling the live coordinates, there are two ways of filling the dead coordinates, $\alpha(s)$ and $\beta(s)$, such that the join $s \times \alpha(s)$ will be on Alice's side of the rectangle, and the join $s \times \beta(s)$ will be on Bob's side of the rectangle; furthermore, $\alpha(s)_i \neq \beta(t)_i$ always holds. To such a configuration we will call a *square*:

Definition 25 (Square)**.** A *square* is a tuple $\mathcal{S} = \langle n, p, R = A \times B, I, S, \alpha, \beta\rangle$ where:

- $n \geq 1$, $p \geq 2$ are integers;
- $R = A \times B$ where $A, B \subseteq (\{0,1\}^n)^{[p]}$;
- $\varnothing \neq I \subseteq [p]$ is a non-empty set of so-called *live* coordinates, and
- $D = [p] \setminus I$ is the corresponding set of *dead* coordinates;
- $S \subseteq (\{0,1\}^n)^I$;
- $\alpha : S \to (\{0,1\}^n)^D$ and $\beta : S \to (\{0,1\}^n)^D$ are such that
$$A = \{s \times \alpha(s) \mid s \in S\} \text{ and } B = \{s \times \beta(s) \mid s \in S\};$$
- for every $s \in S$, $t \in S$, $i \in I$, we have $\alpha(s)_i \neq \beta(t)_i$.

Definition 26. The *density* of square $\mathcal{S} = \langle n, p, R = A \times B, I, S, \alpha, \beta\rangle$ is given by
$$\mathsf{Density}(\mathcal{S}) = \frac{|S|}{2^{n|I|}}.$$

Definition 27. We say a square $\mathcal{S} = \langle n, p, R = A \times B, I, S, \alpha, \beta\rangle$ is τ-thick on $F \subseteq I$ if S is τ-thick on F, and is φ-average-thick on F if S is φ-average-thick on F.

One may justify the name *square* by the observation that a square $\mathcal{S} = \langle n, p, R = A \times B, I, S, \alpha, \beta\rangle$ induces a bijection between A and B, where $s \times \alpha(s) \in A$ corresponds to $s \times \beta(s) \in B$.

4 The projection lemma

The main technical lemma of our simulation theorem is a *projection lemma*, which allow us to constrain coordinates of a square while preserving thickness, in such a way that $\alpha(s)_i \neq \beta(t)_i$ always holds.

Lemma 28. *Let $\mathcal{S} = \langle n, p, R = A \times B, I, S, \alpha, \beta\rangle$ be a square and $\tau, \varphi \in [0,1]$ be real numbers. Suppose that $p \leq \frac{1}{12} \cdot 2^{\tau \cdot 2^n}$. Suppose also that $\mathcal{S}$ is τ-thick, but not φ-average-thick, on $F \subseteq I$.*

Then, for any $z \in \{0,1\}^F$, there exists a non-empty set $E = E(z) \subseteq F$ such that, letting $E_0 = \{i \in E \mid z_i = 0\}$, we may construct a square $\mathcal{S}' = \mathcal{S}'(z) = \langle n, p, R' = A' \times B', I', S', \alpha', \beta'\rangle$, where:

(i) $A' \subseteq A$ and $B' \subseteq B$;

(ii) $I' = I \setminus E_0$;

(iii) $\mathsf{Density}(\mathcal{S}') \geq (\frac{1}{2\varphi})^{|E_0|} \cdot \mathsf{Density}(\mathcal{S})$; *and*

(iv) $\mathcal{S}'$ *is* $\frac{1}{2}\varphi$*-average-thick on* $F \setminus E$.

Furthermore, the set $E = E(z) \subseteq F$ *is obtained by a query procedure on the string* z*, and is exactly the set of positions queried by this procedure.*

Proof. We will explain the projection procedure in three steps. The entire procedure is achieved by running Procedure 1, 2 and 3 one after another.

Procedure 1. *Choosing* E*.*

- We start by letting $E = \varnothing$.
- As long as $S_{I\setminus E_0}$ is not φ-average-thick on $F \setminus E$, there exists some $i \in F \setminus E$ such that

$$\frac{|S_{I\setminus E_0}|}{|S_{I\setminus(E_0\cup\{i\})}|} \leq \varphi \cdot 2^n.$$

- We will then add i to E and query z_i (to know if $i \in E_0$ or not).

To begin with, S is not φ-average-thick on F, and so we are assured we will add at least one coordinate to E. Every time we add an index i to E we have, immediately prior to this, that $\frac{|S_{I\setminus E_0}|}{|S_{I\setminus(E_0\cup\{i\})}|} \leq \varphi \cdot 2^n$, and hence $|S_{I\setminus(E_0\cup\{i\})}| \geq \frac{|S_{I\setminus E_0}|}{\varphi\cdot 2^n}$. This means that if $z_i = 0$ and we then add i to E_0, we will have $|S_{I\setminus E_0}|$ grow by a factor of $\varphi \cdot 2^n$. By the end of this process, $S_{I\setminus E_0}$ must be φ-average-thick on $F \setminus E$ (otherwise we would add another coordinate to E), and furthermore $|S_{I\setminus E_0}| \geq \frac{|S|}{(\varphi\cdot 2^n)^{|E_0|}}$, which is to say

$$\frac{|S_{I\setminus E_0}|}{2^{|I\setminus E_0|\cdot n}} \geq \frac{1}{\varphi^{|E_0|}} \cdot \frac{|S|}{2^{|I|\cdot n}}. \qquad (*)$$

This will later ensure our density increase. Now consider the following procedure:

Procedure 2. *Choosing* $W = (U_i, V_i)_{i\in E_0}$*,* X *and* Y*.*

- Independently for each $i \in E_0$, choose a partition $\{0,1\}^n = U_i \cup V_i$, so that each string $x \in \{0,1\}^n$ is placed in U_i with probability $\frac{1}{2}$, and is placed in V_i otherwise. Let us use $W = (U_i, V_i)_{i\in E_0}$ to denote all the partitions chosen in this step.
- Now let us start by letting $X = Y = S$.
- Then for each index $i \in E_0$ in turn, we change X to $X^{i,U_i}_{\neq i}$ and change Y to $Y^{i,V_i}_{\neq i}$.

At the end of this process, we have both $X, Y \subseteq S_{I\setminus E_0}$. Now we may ask how much of $S_{I\setminus E_0}$ survived inside both X and Y. Let us first consider the difficult case when $|E_0| \geq 1$. We make the following claim:

Claim 29. *If* $|E_0| \geq 1$*, then for some choice of the partitions* $(U_i, V_i)_{i\in E_0}$ *we will have* $|X \cap Y| \geq \frac{1}{2} \cdot |S_{I\setminus E_0}|$.

Before proving this claim, let us see why it is enough to give us our new square $\mathcal{S}'$. Let $U \subseteq (\{0,1\}^n)^{E_0}$ be the product of the various U_i sets, for $i \in E_0$, and likewise let $V \subseteq (\{0,1\}^n)^{E_0}$ be the product of the various V_i sets, for $i \in E_0$. The square $\mathcal{S}'$ is chosen thus:

Procedure 3. *Choosing the square* $\mathcal{S}'$*.*

- We set $S' = X \cap Y$.
- For each $s' \in S'$, we choose a string $u(s') \in U \cap \mathsf{Ext}_S(s') \subseteq (\{0,1\}^n)^{E_0}$; such a $u(s')$ exists because

of how X was constructed; letting $s = s' \times u(s') \in S$, for each $i \in [p] \setminus I' = ([p] \setminus I) \cup E_0$, set

$$\alpha'(s')_i = \begin{cases} u(s')_i & \text{if } i \in E_0, \\ \alpha(s)_i & \text{if } i \in [p] \setminus I. \end{cases}$$

- We proceed symmetrically to choose $\beta'(s')$.
- A' and B' are simply the images of S' under α' and β'.

For any $s \in S$, $t \in S$ and $i \in ([p] \setminus I) \cup E_0$, we have $\alpha(s)_i \neq \beta(t)_i$. This follows, on coordinates $i \in E_0$ because U_i and V_i are disjoint, and on coordinates $i \in [p] \setminus I$ because square $\mathcal{S}$ has the same property for α and β.

Properties (i) and (ii) are by construction. Property (iii) is a calculation using Claim 29 and ($*$):

$$\mathsf{Density}(\mathcal{S}') = \frac{|S'|}{2^{|I'|\cdot n}} \underset{\text{Claim 29}}{\geq} \frac{\frac{1}{2} \cdot |S_{I\setminus E_0}|}{2^{|I\setminus E_0|\cdot n}} \underset{\text{Using } (*)}{\geq} \frac{1}{2} \cdot \frac{1}{\varphi^{|E_0|}} \cdot \frac{|S|}{2^{|I|\cdot n}} = \frac{1}{2} \cdot \frac{1}{\varphi^{|E_0|}} \cdot \mathsf{Density}(\mathcal{S}).$$

Now Property (iii) follows using the fact that $|E_0| \geq 1$. Property (iv) follows by Claim 29, because $S_{I\setminus E_0}$ is φ-average-thick on $F \setminus E$, and S' is a subset of $S_{I\setminus E_0}$ with $|S'| \geq \frac{1}{2} \cdot |S_{I\setminus E_0}|$.

In the simple case when $|E_0| = 0$, we have $X = Y = S$, and so we set $\mathcal{S}'$ to be exactly $\mathcal{S}$. Properties (i) and (ii) are easy to check, and Property (iii) is trivial, and property (iii) holds even without the $1/2$ factor loss, by our choice of E.

Now to prove Claim 29. Let $\delta = 2^{-\tau\cdot 2^n}$. Let us think of a matrix M where the rows are indexed by the various possible $s' \in S_{I\setminus E_0}$ and the columns are indexed by the different possible choices $W = (U_i, V_i)_{i\in E_0}$. The entry $M(s', W)$ equals 1 if $s' \in X$, where X is obtained from S and $(U_i)_{i\in E_0}$ by Procedure 2. In other words, again denoting by U the product of the various sets U_i, we have $M(s', W) = 1$ iff $U \cap \mathsf{Ext}_S(s') \neq \varnothing$.

Now fix some $s' \in S_{I\setminus E_0}$, and let us estimate the probability that $M(s', W) = 1$, i.e. that $s' \in X$, over the randomized choice of W. At the beginning of Procedure 2, we have $X = S$, and X is τ-thick on F. Then for each index $i \in E_0 \subseteq F$ in turn, we will change X to $X^{i,U_i}_{\neq i}$. Before we do this for the first time, s' will have at least one extension $s \in \mathsf{Ext}_X(s') \subseteq (\{0,1\}^n)^{E_0}$; at this point X is τ-thick on F, and so, taking any extension $s'' \in \mathsf{Ext}_{X_{\neq i}}(s') \subseteq (\{0,1\}^n)^{E\setminus\{i\}}$, there will be at least $\tau \cdot 2^n$ strings $s''' \in \{0,1\}^n$ such that $(s' \times s'') \times_i s''' \in S$. Each of these strings s''' is placed in U_i with probability $1/2$; hence the probability that $(s' \times s'') \in X^{i,U_i}_{\neq i}$ is at least $1 - 2^{-\tau\cdot 2^n} = 1 - \delta$, i.e., some extension s'' of s' survived with at least $1 - \delta$ probability over the choice of this first U_i. By Lemma 24, changing X to $X^{i,U_i}_{\neq i}$ gives us a set which is again thick on $F \setminus \{i\}$. Hence we may apply the same reasoning to the next index in E_0.

Changing X in this way $|E_0|$ times, we conclude that, in the end,

$$\Pr[M(s', W) = 1] = \Pr[s' \in X] \geq (1 - \delta)^{|E_0|} \geq 1 - |E_0|\delta,$$

where the probability is with respect to the distribution of W given by the above process. Now call a certain choice of W X-good if the W-column of M has at least a $1 - 3|E_0|\delta$ fraction of the rows $s' \in S_{I\setminus E_0}$ with $M(s', W) = 1$. Then, by a standard averaging argument, we must have $\Pr[W \text{ is } X\text{-good}] > 1/2$ (where again the probability is with respect to the distribution of W).

Arguing in the same way with respect to Y, we conclude that the probability that W is Y-good will also be more than $\frac{1}{2}$. Hence there must exist a choice of W which is both X-good and Y-good. For this choice of W we will have both $|X|, |Y| \geq (1 - 3|E_0|\delta)|S_{I\setminus E_0}|$, and given that $X, Y \subseteq S_{I\setminus E_0}$, this implies that $|X \cap Y| \geq (1 - 6|E_0|\delta) \cdot |S_{I\setminus E_0}| \geq (1 - 6p\delta) \cdot |S_{I\setminus E_0}|$. This is at least $\frac{1}{2}|S_{I\setminus E_0}|$ by our assumed bound on p. The claim is thus proven. ☐

Lemma 30. *Let $\mathcal{S} = \langle n, p, R = A \times B, I, S, \alpha, \beta\rangle$ be a square which is τ-thick on $F \subseteq I$, and let $z \in \{0,1\}^p$ be such that $z_i = 1$ for every $i \in I \setminus F$, and $z_i = 0$ for every $i \in [p] \setminus I$. Then there exists some $(x, y) \in A \times B$ with $\mathsf{Eq}^p(x, y) = z$.*

Proof. This is proven very similarly to Lemma 28. Instead of using Procedure 1 to choose E and E_0, we choose them directly based on z.

If there are no $i_0 \in I$ with $z_{i_0} = 0$, then any $s \in S$ will give $\mathsf{Eq}^p(s \times \alpha(s), s \times \beta(s)) = z$. Otherwise, let $E = F \setminus \{i_0\}$, so that $E_0 = \{i \in I \mid i \neq i_0, z_i = 0\}$. We may then use Procedure 2 to construct sets X and Y

such that $X, Y \subseteq S_{I \setminus E_0}$. Note now that Claim 29 will still hold, because it only requires that S be thick on F. We may then use Procedure 3 to construct $\mathcal{S}'$, and Properties (i) and (ii) will hold as before. $\mathcal{S}'$ is a square on coordinates $I \setminus E_0 = \{i \mid z_i = 1\} \cup \{i_0\}$. By Lemma 24, we know that $\mathcal{S}'$ is τ-thick on $\{i_0\}$ and thus there are two strings $s \in S'$ and $t \in S'$ with $s_{i_0} \neq t_{i_0}$ but $s_i = t_i$ for all $i \in I' \setminus \{i_0\}$. Then $x = s \times \alpha(s)$ and $y = t \times \beta(t)$ give us $\mathsf{Eq}^p(x, y) = z$. □

5 Lifting 0-query complexity

We now prove our main simulation theorem (Theorem 13). Suppose $p \leq 2^{n/100}$, and let us fix $\tau = 2^{-n/10}$ and $\varphi = 2^{-n/20}$. Suppose we are given a C-bit communication protocol π for $f \circ \mathsf{Eq}_n$

We will then construct a decision-tree τ for f. On input $z \in \{0,1\}^p$, τ will find a leaf v of the protocol-tree of π, such that the associated rectangle R_v has some $(x, y) \in R_v$ with $\mathsf{Eq}^p(x, y) = z$. The label of such a leaf then equals $f(\mathsf{Eq}^p(x, y)) = f(z,)$. We now present an informal description of τ, and in Algorithm 2 below we provide pseudocode for τ. We will then show that the algorithm for τ is correct, i.e. that it is always able to find such a leaf v, and then show that the number of 0-queries that τ makes is $O(\frac{C}{n})$, which completes the proof of Theorem 13.

Given an input $z \in \{0,1\}^p$, τ starts traversing a path from the root of the protocol tree of π. A variable v is maintained, indicating the node of the protocol tree of π which is the current-node during the ongoing simulation; associated with v is the rectangle R_v of inputs which cause the protocol to reach node v. The decision-tree τ, when traversing node v, maintains a rectangle $R = A \times B$ and a square $\mathcal{S} = \langle n, p, R = A \times B, I = F \cup O, S, \alpha, \beta \rangle$, such that R is a sub-rectangle of R_v. The set F corresponds to coordinates of the input z that were not queried yet, and O is set of coordinates i which have been queried and found to have $z_i = 1$. Throughout the execution of the algorithm, it is maintained as an invariant that the square $\mathcal{S}$ is τ-thick in the coordinates F. At the beginning, $I = F = [p]$, $O = \varnothing$, and $A = B = (\{0,1\}^n)^{[p]}$, so the invariant is trivially true.

In each iteration of the simulation, the algorithm checks whether $\mathcal{S}$ is φ-average-thick on F. If this fails to hold, the algorithm will use the projection lemma (Lemma 28) and change $\mathcal{S}$ to ensure this requirement, as follows. Using Procedure 1 of Lemma 28, it chooses the set $E \subseteq F$; this requires querying z_i for $i \in E$, and gives us the set $E_0 \subseteq E$ of coordinates where $z_i = 0$, and the set $E_1 = E \setminus E_0$ of coordinates where $z_i = 1$. The algorithm then uses Procedure 3 of Lemma 28 to construct a square $\mathcal{S}'$. Lemma 28 guarantees that $\mathcal{S}'$ is $\frac{\varphi}{2}$-average-thick on $F \setminus E$, and that $\mathsf{Density}(\mathcal{S}')$ grows by a factor of $(2\varphi)^{-|E_0|}$. If E_0 is non-empty, i.e. if we have made some 0 queries, the density will grow significantly; otherwise the density will not change. The algorithm proceeds with $\mathcal{S} = \mathcal{S}'$, $I = I \setminus E_0$, $O = O \cup E_1$, and $F = F \setminus E$.

Now the algorithm is promised to have a square $\mathcal{S}$ which is at least $\frac{1}{2}\varphi$-average-thick. The algorithm then proceeds to a child v_c of v which has at least $1/2$ fraction of the density of $\mathcal{S}$, as follows. Suppose Alice communicated in v, and for each $c \in \{0,1\}$, let $R_{v_c} = A_{v_c} \times B_{v_c}$ be the rectangle which π associates with v_c. We then fix a choice $c \in \{0,1\}$ such that $|R \cap R_{v_c}| \geq |R|/2$. Now consider the set $S' = \{s \in S \mid s \times \alpha(s) \in A_{v_c}\}$. This set is still $\frac{1}{4}\varphi$-average-thick. We may then apply Lemma 23, with $\delta = \frac{1}{2}$, to S', which gives us a subset $S'' \subseteq S'$ which is τ-thick on F. The new square $\mathcal{S}$ is then given by restricting α and β to the set S''. By changing $\mathcal{S}$ in this way, we have preserved a $\frac{1}{4}$ fraction of the density.

Eventually, when we reach a leaf node v of the protocol tree, we are left with a square $\mathcal{S}$ which is τ-thick on F. The algorithm outputs the labeling of R_v in π, and we will now argue that this must equal $f(z)$.

Correctness. Because π correctly solves $f \circ \mathsf{Eq}^n$, then for each leaf v of π we have $(x, y, \pi(v)) \in f$ for all $(x, y) \in R_v$; the rectangle R obtained at the termination of Algorithm 2 is a sub-rectangle of R_v for a leaf of π, hence $(x, y, \pi(v)) \in f$ for all $(x, y) \in R$. On the other hand, we have preserved a square $\mathcal{S} = \langle n, p, R = A \times B, I, S, \alpha, \beta \rangle$ which is τ-thick on $F \subseteq I$, and such that $z_i = 1$ for every $i \in O = I \setminus F$, and $z_i = 0$ for every $i \in [p] \setminus I$. Then Corollary 30 tells us that some pair $(x, y) \in R$ is such that $\mathsf{Eq}^p(x, y) = z$; hence $(z, \pi(v)) \in f$.

Algorithm 2 Decision-tree procedure τ

Input: $z \in \{0,1\}^p$
Output: $f(z)$

Initialization:

- Set v to be the root of the protocol tree for π,
- $I = F = [p]$, $O = \varnothing$,
- $\mathcal{S} = \langle n, p, R, I, S, \alpha, \beta \rangle$, where $R = A \times B$, $A = B = S = (\{0,1\}^n)^{[p]}$, and α, β are the empty functions.

while v is not a leaf **do**
  if $\mathcal{S}$ is not φ-average-thick on F **then**
    Use Lemmas 28 and 23, to obtain $E \subseteq F$, $E_0 \subseteq E$, and
      a square $\mathcal{S}' = \langle n, p, R', I \setminus E_0, S', \alpha', \beta' \rangle$ such that
     (1) $\mathcal{S}'$ is τ-thick on $F \setminus E$,
     (2) $\mathsf{Density}(\mathcal{S}') \geq \frac{1}{4}\frac{1}{\varphi^{|E_0|}}\mathsf{Density}(\mathcal{S})$,
    Update $\mathcal{S} = \mathcal{S}'$, $O = O \cup (E \setminus E_0)$, $F = F \setminus E$, $I = I \setminus E_0$.
  end if
          $\triangleright$ At this point we are assured that $\mathcal{S}$ is at least $\frac{1}{2}\varphi$-average-thick on F.
  Choose $c \in \{0,1\}$ such that $|R \cap R_{v_c}| \geq \frac{1}{2}|R|$.
  Using Lemma 23, choose $\mathcal{S}' = \langle n, p, R', I, S', \alpha', \beta' \rangle$ such that
   (1) $R' \subseteq R \cap R_{v_c}$
   (2) $\mathsf{Density}(\mathcal{S}') \geq \frac{1}{4}\mathsf{Density}(\mathcal{S})$.
   (3) $\mathcal{S}'$ is τ-thick on F.
  Update $\mathcal{S} = \mathcal{S}'$, and $v = v_c$.
end while
Output $\pi(v)$.

Number of queries. In each time when the simulation goes down the protocol tree of π, $\mathsf{Density}(\mathcal{S})$ drops by a factor of at most $\frac{1}{4}$ and hence, in total, by a factor of 4^{-C}. For each set E of queries that the algorithm makes in a round, the density of the current square increases by a factor of $(2\varphi)^{-|E_0|}$ — this is Property (iii) of Lemma 28. So, if Q_0 is the total number of queries which the algorithm makes, and which are answered 0, then the total gain in $\mathsf{Density}(\mathcal{S})$ is at least $(2\varphi)^{-Q_0}$. Since the density can be at most 1, we have,

$$4^{-C} \cdot (2\varphi)^{-Q_0} \leq 1$$

and so

$$Q_0 \leq \frac{-2C}{\log(2\varphi)} = \frac{2C}{\frac{n}{20} - 1} = O\left(\frac{C}{n}\right).$$

This concludes the proof. $\square$

6 Consequences of the main result

In this section, we prove Lemmas 11 and 12.

Lemma 31 (Lemma 11 restated)**.** *Let $f \subseteq \{0,1\}^p \times \mathcal{C}$ be any Boolean relation. Then*

$$\mathsf{D}_0^{dt}(f) \underset{(1)}{\leq} \mathsf{D}_{\mathsf{AND}}^{dt}(f) \underset{(2)}{\leq} \mathsf{D}_0^{dt}(f) \cdot \lceil \log(p+1) \rceil.$$

Proof. Inequality (1) is simple. Usual queries can simulate an AND query $Q(z) = \bigwedge_{i \in S} z_i$, by querying the bit positions $i \in S$ one after another in any order; the first time we find $z_i = 0$, we may stop, knowing that $Q(z) = 0$ holds true; if we found no $i \in S$ with $z_i = 0$, then $Q(z) = 1$. Although we do $|S|$ queries, at most one such query will give us the answer 0. Hence, $\mathsf{D}_0^{dt}(f) \leq \mathsf{D}_{\mathsf{AND}}^{dt}(f)$.

Inequality (2) is proven using binary search. Let us take a decision tree τ' whose 0-depth is $\mathsf{D}_0^{dt}(f)$ and simulate it by an AND decision tree τ which makes $\leq \mathsf{D}_0^{dt}(f) \cdot \lceil \log p \rceil$ AND queries. Consider the root-to-leaf path $p = v_0 v_1 \dots v_d$ of τ' where every query is answered 1; we may assume no repeated queries are asked along the same path, so $d \le p$; let $i_0, \dots, i_{d-1}$ be the bit positions which are queried along path p; then τ may use binary search to find the first i_j such that $z_{i_j} = 0$, or find that no such i_j exists; this requires τ to make $\lceil \log(d+1) \rceil$-many AND queries. If no such i_j exists, then the computation $\tau'(z)$ ends in node v_d. Otherwise, the computation $\tau'(z)$ goes through the nodes $v_0, \dots, v_j v'$, where v' is the child of v_j other than v_{j+1}.

Since τ' made a 0-query to reach v', the subtree of τ' under v' has 0-depth at most $\mathsf{D}_0^{dt}(f) - 1$. Then τ repeats the above procedure inside this subtree, recursively until it reaches a leaf of τ'. ☐

Lemma 32 (Lemma 12 restated)**.** *Let $f \subseteq \{0,1\}^p \times \mathcal{C}$ be any Boolean relation. Then*

$$\mathsf{D}_0^{dt}(f) \ge \frac{\log \mathsf{L}^{dt}(f)}{\log p}.$$

Proof. The proof employs the *fortification* technique of Irit Dinur and Or Meir [DM18], which they applied in the context of Karchmer–Wigderson games, and we here apply in the much simpler model of decision trees.

We may think of a partial assignment $Z \in \{0,1,*\}^p$, as the subset $Z \subseteq \{0,1\}^p$ of those strings consistent with the assignment, in which case we call Z a *subcube*. Let $\mathsf{free}(Z) = \{i \in [p] \mid Z_i = *\}$ denote the set of unassigned, or *free*, variables of Z.

For a subcube $Z \subseteq \{0,1\}^p$, let $\mathsf{L}_f(Z) = \mathsf{L}^{dt}(f\big|_Z)$ be the leaf-complexity of solving the restricted relation:

$$f\big|_Z = \{(z,c) \in f \mid z \in Z\} \cup \{(z,c) \in \{0,1\}^p \times \mathcal{C} \mid z \notin Z\}$$

(i.e., when solving $f\big|_Z$ we do not require the decision tree to be correct outside of Z). The sub-additivity property of L^{dt} states that, for any $i \in \mathsf{free}(Z)$,

$$\mathsf{L}_f(Z) \le \mathsf{L}_f(Z_{z_i=0}) + \mathsf{L}_f(Z_{z_i=1}),$$

where $Z_{z_i=b}$ for $b \in \{0,1\}$ is the restriction of Z when fixing the i-th bit to b. It might be the case that the two terms on the right-hand side are very unbalanced; however, given a subcube Z, we may always find a large subcube Z' of Z, where the leaf complexity of f is mostly preserved, but where $\mathsf{L}_f(Z'_{z_i=0})$ and $\mathsf{L}_f(Z'_{z_i=1})$ are reasonably close, for every free variable $i \in \mathsf{free}(Z')$:

Claim 33. *For any given subcube $Z \in \{0,1,*\}^p$, there exists a* balanced subcube *$Z' \subseteq Z$, such that $\mathsf{L}_f(Z') \ge \frac{1}{2} \cdot \mathsf{L}_f(Z)$, and*

$$\mathsf{L}_f(Z'_{z_i=b}) > \frac{1}{2p} \cdot \mathsf{L}_f(Z'),$$

for every free variable $i \in \mathsf{free}(Z')$, and every $b \in \{0,1\}$.

The proof of this claim follows a greedy strategy. Start by letting $Z' = Z$, and suppose that there exists some $i \in \mathsf{free}(Z')$ and $b \in \{0,1\}$ such that $\mathsf{L}_f(Z'_{z_i=b}) \le \frac{1}{2p} \cdot \mathsf{L}_f(Z')$. It follows from sub-additivity that

$$\mathsf{L}_f(Z'_{z_i=1-b}) \ge \mathsf{L}_f(Z') - \mathsf{L}_f(Z'_{z_i=b}) \ge \left(1 - \frac{1}{2p}\right) \cdot \mathsf{L}_f(Z').$$

Then fix the i-th variable to $1-b$, i.e., change partial assignment Z'_i from $*$ to $1-b$. This can only be done p times, and hence eventually we will have $\mathsf{L}_f(Z'_{z_i=b}) > \frac{1}{2p} \cdot \mathsf{L}_f(Z')$ for every $i \in \mathsf{free}(Z')$ and $b \in \{0,1\}$, as intended. Furthermore, at this point we find that

$$\mathsf{L}_f(Z') \ge \left(1 - \frac{1}{2p}\right)^p \cdot \mathsf{L}_f(Z) \ge \frac{1}{2} \cdot \mathsf{L}_f(Z).$$

This proves the claim. Now we show a lower-bound on $\mathsf{D}_0^{dt}(f)$ by an adversarial argument: we are given an arbitrary decision tree for f, and will force it to query at least $\frac{\log \mathsf{L}^{dt}(f)}{\log(2p)}$-many zeros. Before making any query, the decision tree is trying to solve $f\big|_Z$ where $Z = \{0,1\}^p$ is the entire Boolean hypercube. We immediately use Claim 33 to obtain a balanced subcube Z' of Z, potentially loosing $\frac{1}{2}$ of the leaf complexity of f. Then, if the decision tree queries a non-free variable i of Z', we answer Z'_i; but if the decision tree queries a free variable $i \in \mathsf{free}(Z')$, we answer 0 and change Z'_i from $*$ to 0 — we now know that at least a $\frac{1}{2p}$ fraction of the leaf complexity of f is preserved, because Z' was a balanced subcube before setting variable i. We then apply Claim 33 again to obtain a balanced subcube of Z'. We are able to do this as long as $\mathsf{L}_f(Z') > 1$, which ensures $f\big|_{Z'}$ is not constant. For each 0-query the decision tree is forced to make, a $\frac{1}{4p}$ fraction of the leaf complexity of f remains, hence we are able to do this $\log_{4p} \mathsf{L}^{dt}(f) = \Omega\left(\frac{\log \mathsf{L}^{dt}(f)}{\log p}\right)$ times before f becomes constant on Z'. $\square$

7 Lifting degree to communication via a rank argument

We restate Lemma 6 for convenience:

Lemma 34 (Lemma 6 restated)**.** *For every $f : \{+1,-1\}^p \to \{+1,-1\}$ with $\deg(f) \geq 1$, and every $g : \{+1,-1\}^n \to \{+1,-1\}$, we have $\|f \circ g\|_0 \geq (\|g\|_0 - 1)^{\deg(f)}$.*

Proof. For $z \in \{+1,-1\}^p$ and $x \in \{+1,-1\}^n$, let us write the Fourier expansion

$$f(z) = \sum_{S \subseteq [p]} \hat{f}(S) \prod_{i \in S} z_i, \qquad \text{and} \qquad g(x) = \sum_{T \subseteq [n]} \hat{g}(T) \prod_{j \in T} x_j.$$

Let $S' \subseteq [p]$ with $|S'| = \deg(f)$ be any maximal-degree monomial in the Fourier expansion of f. Assume without loss of generality that $S' = \{1, \ldots, k\}$, where $k = \deg(f)$. Now think of the expression $E = (f \circ g)(x^{(1)}, \ldots, x^{(p)})$, which we obtain if we replace each variable z_i in the Fourier expansion of $f(z)$ with the Fourier expansion of $g(x^{(i)})$. The term in E which corresponds to the monomial S' will be:

$$\hat{f}(S') \times \left[\sum_{T \subseteq [n]} g(T) \cdot \prod_{j \in T} x_j^{(1)}\right] \times \ldots \times \left[\sum_{T \subseteq [n]} g(T) \cdot \prod_{j \in T} x_j^{(k)}\right].$$

Distributing the outer products over the sums, we get

$$\hat{f}(S') \sum_{T_1, \ldots, T_k \subseteq [n]} g(T_1) \ldots g(T_k) \cdot \prod_{i=1}^{k} \prod_{j \in T_i} x_j^{(i)}. \tag{$*$}$$

Now let us *single out* the monomials which are obtained from the sum ($*$), in the case when every T_i is non-empty. There are $(\|g\|_0 - 1)^k$ unique singled-out monomials. The lemma follows from the observation that these monomials cannot appear elsewhere in E, and thus cannot be cancelled out. Indeed, the other monomials in E are either:

(1) monomials appearing for the choice S' above, but corresponding to one or more empty T_i, or

(2) monomials appearning for a different choice $S \neq S'$.

Any such monomials have non-empty symmetric difference with any of the singled out monomials, and hence none of the singled-out monomials can be cancelled out.[4] $\square$

[4] Note, however, that monomials of type (1) *may* cancel out with monomials of type (2); for example if we take $f(z_1, z_2) = z_1 z_2 + z_1 + z_2 + 1$ and $g(x) = x - 1$, we then have

$$(f \circ g)(x^{(1)}, x^{(2)}) = (x^{(1)} - 1)(x^{(2)} - 1) + x^{(1)} - 1 + x^{(2)} - 1 + 1 = x^{(1)} x^{(2)},$$

so that $\deg(f) = 2$ and $\|g\|_0 = 2$, but giving us $\|f \circ g\|_0 = 1$, which indeed is equal to $(\|g\|_0 - 1)^{\deg(f)}$, and no better.

References

[BBCR13] Boaz Barak, Mark Braverman, Xi Chen, and Anup Rao. How to compress interactive communication. *SIAM Journal on Computing*, 42(3):1327–1363, 2013.

[BBK+13] Joshua Brody, Harry Buhrman, Michal Kouckỳ, Bruno Loff, Florian Speelman, and Nikolay Vereshchagin. Towards a reverse Newman's theorem in interactive information complexity. In *Proceedings of the 28th CCC*, pages 24–33, 2013.

[BC99] Anna Bernasconi and Bruno Codenotti. Spectral analysis of boolean functions as a graph eigenvalue problem. *IEEE Transactions on Computers*, 48(3):345–351, 1999.

[BdW02] Harry Buhrman and Ronald de Wolf. Complexity measures and decision tree complexity: a survey. *Theoretical Computer Science*, 288(1):21–43, 2002.

[BPSW05] Paul Beame, Toniann Pitassi, Nathan Segerlind, and Avi Wigderson. A direct sum theorem for corruption and the multiparty nof communication complexity of set disjointness. In *Proceedings of the 20th CCC*, pages 52–66, 2005.

[BR14] Mark Braverman and Anup Rao. Information equals amortized communication. *IEEE Transactions on Information Theory*, 60(10):6058–6069, 2014.

[BRWY13a] Mark Braverman, Anup Rao, Omri Weinstein, and Amir Yehudayoff. Direct product via round-preserving compression. In *Proceedings of the 40th ICALP*, pages 232–243, 2013.

[BRWY13b] Mark Braverman, Anup Rao, Omri Weinstein, and Amir Yehudayoff. Direct products in communication complexity. In *Proceedings of the 54th FOCS*, pages 746–755, 2013.

[CA08] Arkadev Chattopadhyay and Anil Ada. Multiparty communication complexity of disjointness. Technical Report TR08-002, ECCC, 2008.

[Cha07] Arkadev Chattopadhyay. Discrepancy and the power of bottom fan-in in depth-three circuits. In *Proceedings of the 48th FOCS*, pages 449–458, 2007.

[CKLM17] Arkadev Chattopadhyay, Michal Koucký, Bruno Loff, and Sagnik Mukhopadhyay. Simulation theorems via pseudorandom properties. *CoRR*, abs/1704.06807, 2017.

[CKLM18] Arkadev Chattopadhyay, Michal Kouckỳ, Bruno Loff, and Sagnik Mukhopadhyay. Simulation beats richness: new data-structure lower bounds. In *Proceedings of the 50th STOC*, pages 1013–1020. ACM, 2018.

[CR12] Amit Chakrabarti and Oded Regev. An optimal lower bound on the communication complexity of gap-hamming-distance. *SIAM Journal on Computing*, 41(5):1299–1317, 2012.

[DM18] Irit Dinur and Or Meir. Toward the KRW composition conjecture: Cubic formula lower bounds via communication complexity. *Computational Complexity*, 27(3):375–462, 2018.

[dRNV16] Susanna F. de Rezende, Jakob Nordström, and Marc Vinyals. How limited interaction hinders real communication. In *Proceedings of the 56th FOCS*, 2016.

[Dru12] Andrew Drucker. Improved direct product theorems for randomized query complexity. *Computational Complexity*, 21(2):197–244, 2012.

[EIRS01] Jeff Edmonds, Russell Impagliazzo, Steven Rudich, and Jirí Sgall. Communication complexity towards lower bounds on circuit depth. *Computational Complexity*, 10(3):210–246, 2001.

[FKNN95] Tomás Feder, Eyal Kushilevitz, Moni Naor, and Noam Nisan. Amortized communication complexity. *SIAM Journal on Computing*, 24(4):736–750, 1995.

[GGKS18] Ankit Garg, Mika Göös, Pritish Kamath, and Dmitry Sokolov. Monotone circuit lower bounds from resolution. In *Proceedings of the 50th STOC*, pages 902–911, 2018.

[GJPW17] Mika Göös, TS Jayram, Toniann Pitassi, and Thomas Watson. Randomized communication versus partition number. In *Proceedings of the 44th ICALP*, 2017.

[GJW18] Mika Göös, Rahul Jain, and Thomas Watson. Extension complexity of independent set polytopes. *SIAM Journal on Computing*, 47(1):241–269, 2018.

[GKPW17] Mika Göös, Pritish Kamath, Toniann Pitassi, and Thomas Watson. Query-to-communication lifting for P^{np}. In *Proceedings of the 32nd CCC*, 2017.

[GLM+15] Mika Göös, Shachar Lovett, Raghu Meka, Thomas Watson, and David Zuckerman. Rectangles are nonnegative juntas. In *Proceedings of the 47th STOC*, pages 257–266. ACM, 2015.

[GLS18] Dmitry Gavinsky, Troy Lee, and Miklos Santha. On the randomised query complexity of composition. *CoRR*, abs/1801.02226, 2018.

[GMWW14] Dmitry Gavinsky, Or Meir, Omri Weinstein, and Avi Wigderson. Toward better formula lower bounds: an information complexity approach to the krw composition conjecture. In *Proceedings of the 46th STOC*, pages 213–222, 2014.

[GPW15a] Mika Göös, Toniann Pitassi, and Thomas Watson. Deterministic communication vs. partition number. In *Proceedings of the 56th FOCS*, 2015.

[GPW15b] Mika Göös, Toniann Pitassi, and Thomas Watson. The landscape of communication complexity classes. *Computational Complexity*, pages 1–60, 2015.

[GS91] Michelangelo Grigni and Michael Sipser. Monotone separation of logspace from NC. In *Proceedings of the 6th Annual Structure in Complexity Theory Conference*, pages 294–298, 1991.

[HHL18] Hamed Hatami, Kaave Hosseini, and Shachar Lovett. Structure of protocols for XOR functions. *SIAM Journal on Computing*, 47(1):208–217, 2018.

[HJMR07] Prahladh Harsha, Rahul Jain, David McAllester, and Jaikumar Radhakrishnan. The communication complexity of correlation. In *Proceedings of the 22nd CCC*, pages 10–23, 2007.

[HW90] Johan Håstad and Avi Wigderson. Composition of the universal relation. In *Proceedings of the DIMACS Workshop*, pages 119–134, 1990.

[Jai15] Rahul Jain. New strong direct product results in communication complexity. *Journal of the ACM*, 62(3):20, 2015.

[JKN08] Rahul Jain, Hartmut Klauck, and Ashwin Nayak. Direct product theorems for classical communication complexity via subdistribution bounds. In *Proceedings of the 40th STOC*, pages 599–608, 2008.

[JPY12] Rahul Jain, Attila Pereszlényi, and Penghui Yao. A direct product theorem for the two-party bounded-round public-coin communication complexity. In *Proceedings of the 53rd FOCS*, pages 167–176, 2012.

[JRS03] Rahul Jain, Jaikumar Radhakrishnan, and Pranab Sen. A direct sum theorem in communication complexity via message compression. In *Proceedings of the 20th ICALP*, pages 300–315, 2003.

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

[JY12] Rahul Jain and Penghui Yao. A strong direct product theorem in terms of the smooth rectangle bound. *CoRR*, abs/1209.0263, 2012.

[KKN95] Mauricio Karchmer, Eyal Kushilevitz, and Noam Nisan. Fractional covers and communication complexity. *SIAM Journal on Discrete Mathematics*, 8(1):76–92, 1995.

[KLL+15] Iordanis Kerenidis, Sophie Laplante, Virginie Lerays, Jérémie Roland, and David Xiao. Lower bounds on information complexity via zero-communication protocols and applications. *SIAM Journal on Computing*, 44(5):1550–1572, 2015.

[KMR17] Pravesh K Kothari, Raghu Meka, and Prasad Raghavendra. Approximating rectangles by juntas and weakly-exponential lower bounds for lp relaxations of csps. In *Proceedings of the 49th STOC*, pages 590–603. ACM, 2017.

[KN97] Eyal Kushilevitz and Noam Nisan. *Communication complexity.* Cambridge University Press, 1997.

[Koz18] Alexander Kozachinskiy. From expanders to hitting distributions and simulation theorems. In *Proceedings of the 43rd MFCS*, pages 4:1–4:15, 2018.

[KRW95] Mauricio Karchmer, Ran Raz, and Avi Wigderson. Super-logarithmic depth lower bounds via the direct sum in communication complexity. *Computational Complexity*, 5(3/4):191–204, 1995.

[KW90] Mauricio Karchmer and Avi Wigderson. Monotone circuits for connectivity require super-logarithmic depth. *SIAM Journal on Discrete Mathematics*, 3(2):255–265, 1990.

[LMV] James Lee, Raghu Meka, and Thomas Vidick. (less) heavy lifting: from conic junta degree to non-negative rank. Presented in the workshop *Hardness Escalation in Communication Complexity and Query Complexity*, FOCS 2017.

[Lov75] László Lovász. On the ratio of optimal integral and fractional covers. *Discrete mathematics*, 13(4):383–390, 1975.

[LSS08] Troy Lee, Adi Shraibman, and Robert Spalek. A direct product theorem for discrepancy. In *Proceedings of the 23rd CCC*, pages 71–80, 2008.

[MP88] Marvin Minsky and Seymour A Papert. *Perceptrons: Expanded edition.* MIT Press, 1988.

[Orl90] Alon Orlitsky. Worst-case interactive communication. i. two messages are almost optimal. *IEEE Transactions on Information Theory*, 36(5):1111–1126, 1990.

[Pan12] Denis Pankratov. Direct sum questions in classical communication complexity. Master's thesis, University of Chicago, 2012.

[PR18] Toniann Pitassi and Robert Robere. Lifting nullstellensatz to monotone span programs over any field. In *Proceedings of the 50th STOC*, pages 1207–1219, 2018.

[Raz95] Ran Raz. Fourier analysis for probabilistic communication complexity. *Computational Complexity*, 5(3-4):205–221, 1995.

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

[Rob18] Robert Robere. *Unified Lower Bounds for Monotone Computation.* PhD thesis, University of Toronto, 2018.

[RPRC16] Robert Robere, Toniann Pitassi, Benjamin Rossman, and Stephen A Cook. Exponential lower bounds for monotone span programs. In *Proceedings of the 57th FOCS*, pages 406–415, 2016.

[RY15] Anup Rao and Amir Yehudayoff. Simplified lower bounds on the multiparty communication complexity of disjointness. In *Proceedings of the 30th CCC*, pages 88–101, 2015.

[San18] Swagato Sanyal. A composition theorem via conflict complexity. *CoRR*, abs/1801.03285, 2018.

[She11] Alexander A Sherstov. The pattern matrix method. *SIAM Journal on Computing*, 40(6):1969–2000, 2011.

[She12a] Alexander A. Sherstov. The communication complexity of gap hamming distance. *Theory of Computing*, 8(1):197–208, 2012.

[She12b] Alexander A. Sherstov. The multiparty communication complexity of set disjointness. In *Proceedings of the 44th STOC*, pages 525–548, 2012.

[She13] Alexander A. Sherstov. Communication lower bounds using directional derivatives. In *Proceedings of the 45th STOC*, pages 921–930, 2013.

[SZ09] Yaoyun Shi and Yufan Zhu. Quantum communication complexity of block-composed functions. *Quantum Information & Computation*, 9(5):444–460, 2009.

[TWXZ13] Hing Yin Tsang, Chung Hoi Wong, Ning Xie, and Shengyu Zhang. Fourier sparsity, spectral norm, and the log-rank conjecture. In *Proceedings of the 54th FOCS*, pages 658–667, 2013.

[Vid13] Thomas Vidick. A concentration inequality for the overlap of a vector on a large set, with application to the communication complexity of the Gap-Hamming-Distance problem. *Chicago Journal of Theoretical Computer Science*, 2013, 2013.

[Wat17] Thomas Watson. A $\mathsf{ZPP}^{\mathsf{NP}}$ lifting theorem. *Unpublished preprint*, 2017.

[Woo07] David P. Woodruff. *Efficient and private distance approximation in the communication and streaming models*. PhD thesis, Massachusetts Institute of Technology, 2007.

[Yao79] Andrew Chi-Chih Yao. Some complexity questions related to distributive computing (preliminary report). In *Proceedings of the 11h STOC*, pages 209–213, 1979.

ECCC ISSN 1433-8092
https://eccc.weizmann.ac.il