Deterministic Communication vs. Partition Number

Mika Göös
Harvard University

Toniann Pitassi
University of Toronto

Thomas Watson
University of Memphis

September 5, 2017

Abstract

We show that deterministic communication complexity can be superlogarithmic in the partition number of the associated communication matrix. We also obtain near-optimal deterministic lower bounds for the Clique vs. Independent Set problem, which in particular yields new lower bounds for the log-rank conjecture. All these results follow from a simple adaptation of a communication-to-query simulation theorem of Raz and McKenzie (Combinatorica 1999) together with lower bounds for the analogous query complexity questions.

1 Introduction

The *partition number* of a two-party function $F\colon \mathcal{X}\times\mathcal{Y}\to\{0,1\}$ is defined by

$$\chi(F) \;\coloneqq\; \chi_1(F)+\chi_0(F)$$

where $\chi_i(F)$ is the least number of rectangles (sets of the form $A\times B$ where $A\subseteq\mathcal{X}$, $B\subseteq\mathcal{Y}$) needed to partition the set $F^{-1}(i)$. Yao [Yao79] observed that $\log\chi(F)$ is a lower bound on the deterministic communication complexity of F and inquired about the exact relationship. For upper bounds, it is known that $O(\log^2\chi(F))$ bits [AUY83], or even $O(\log^2\chi_1(F))$ bits [Yan91], suffice.

Our results are as follows—here the notation $\tilde{\Omega}(m)$ hides factors polylogarithmic in m.

Theorem 1. *There is an F with deterministic communication complexity $\tilde{\Omega}(\log^{1.5}\chi(F))$.*

Theorem 2. *There is an F with deterministic communication complexity $\tilde{\Omega}(\log^{2}\chi_1(F))$.*

Theorem 1 implies that the logarithm of the partition number does not characterize (up to constant factors) deterministic communication complexity, which solves an old problem [KN97, Open Problem 2.10]. The previous best lower bound in this direction was about $2\cdot\log\chi(F)$ due to Kushilevitz, Linial, and Ostrovsky [KLO99]. In this work, we show—maybe surprisingly—that superlogarithmic lower bounds can be obtained using known techniques!

Theorem 2 is essentially tight in view of the upper bound $O(\log^2\chi_1(F))$ mentioned above. A recent work [Göö15] exhibited a different F with $\Omega(\log^{1.128}\chi_1(F))$ *conondeterministic* communication complexity (i.e., the logarithm of the least number of rectangles needed to *cover* the set $F^{-1}(0)$);

Supported by funding from NSERC and from NSF grant CCF-1657377. A preliminary version of this work was published as [GPW15].

this is quantitatively weaker than Theorem 2 and hence the two results are incomparable. The question about the relationship between $\log \chi_1(F)$ and deterministic communication complexity is sometimes referred to as the Clique vs. Independent Set problem; see [Juk12, §4.4] for an excellent overview. In particular, Theorem 2 implies that there exists a graph on n nodes for which the Clique vs. Independent Set problem (Alice is given a clique and Bob is given an independent set: do they intersect?) requires $\tilde{\Omega}(\log^2 n)$ communication. (The upper bound $O(\log^2 n)$ holds for all graphs.) Theorem 2 also gives improved lower bounds for the log-rank conjecture [LS88] (see [Lov14] for a survey): Viewing rectangles as all-1 submatrices, we have $\chi_1(F) \geq \operatorname{rank}(F)$ where the rank is over the reals. Hence Theorem 2 implies a communication lower bound of $\tilde{\Omega}(\log^2 \operatorname{rank}(F))$. The previous record was $\Omega(\log^{1.63} \operatorname{rank}(F))$ due to Kushilevitz [Kus94].

1.1 Our approach

We follow a recurring theme (e.g., [NW95, RM99, SZ09, She11, HN12, CLRS16, GLM+16, LRS15]):

> Instead of proving an ad hoc communication lower bound directly, we prove a lower bound in the simpler-to-understand world of *query complexity* [BdW02], and then "lift" the result over to the world of communication complexity.

The general idea is to start with a boolean function $f\colon \{0,1\}^n \to \{0,1\}$ (called the *outer function*) and then study a composed function $F \coloneqq f \circ g^n$ where $g\colon \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ is a small two-party function (called the *gadget*). More precisely, the communication problem is to compute, on input $x \in \mathcal{X}^n$ to Alice, and $y \in \mathcal{Y}^n$ to Bob, the output

$$F(x,y) \;\coloneqq\; f(g(x_1,y_1),\ldots,g(x_n,y_n)).$$

Deterministic simulation. We use tools that were introduced already in 1997 by Raz and McKenzie [RM99] (building on [EIRS01]). They proved a *simulation theorem* that converts a deterministic protocol for $F \coloneqq f \circ g^n$ (where f is arbitrary but the gadget g is chosen carefully) into a deterministic decision tree for f. Unfortunately, their result was originally formulated only in case f was a certain "structured" search problem (canonical search problem associated with a DNF tautology), and this is how their result has been applied subsequently [BEGJ00, Joh01]. However, we observe that, with minor modifications, their proof actually works without any assumptions on f. Such a simulation theorem (for functions f) was conjectured in [Dru09]. We provide (in Section 3) a self-contained and streamlined exposition (including some simplifications) of the following version of the Raz–McKenzie result—here $\mathsf{P}^{\mathsf{cc}}(F)$ denotes the deterministic communication complexity of F and $\mathsf{P}^{\mathsf{dt}}(f)$ denotes the deterministic decision tree complexity of f.

Theorem 3 (Simulation Theorem). *There is a gadget $g\colon \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ where the size of Alice's input is $\log|\mathcal{X}| = \Theta(\log n)$ bits such that for all $f\colon \{0,1\}^n \to \{0,1\}$ we have*

$$\mathsf{P}^{\mathsf{cc}}(f \circ g^n) \;=\; \mathsf{P}^{\mathsf{dt}}(f) \cdot \Theta(\log n).$$

The gadget in the above can be taken to be the usual indexing function $g\colon [m] \times \{0,1\}^m \to \{0,1\}$ where $m \coloneqq \operatorname{poly}(n)$ and $g(x,y) \coloneqq y_x$. The upper bound in Theorem 3,

$$\mathsf{P}^{\mathsf{cc}}(f \circ g^n) \;\leq\; \mathsf{P}^{\mathsf{dt}}(f) \cdot O(\log n), \tag{1}$$

follows simply because a communication protocol can always simulate a decision tree for f with an overhead of factor $\mathsf{P}^{\mathsf{cc}}(g) \leq \lceil \log m \rceil + 1 = \Theta(\log n)$. Indeed, whenever the decision tree queries the

i-th input bit of f, Alice and Bob exchange $\Theta(\log n)$ bits to compute the output $g(x_i, y_i)$ of the i-th gadget. The nontrivial part of Theorem 3 is to show that this type of protocol is optimal: there are no shortcuts to computing $f \circ g^n$ other than to "query" individual input bits of f in some order.

Nondeterministic models. Recall that a *nondeterministic protocol* (e.g., [KN97, Juk12]) is a protocol that is allowed to make guesses—an input is accepted iff there is at least one accepting computation. Combinatorially, a nondeterministic protocol for F of communication cost k can be visualized as a covering of the set $F^{-1}(1)$ using at most 2^k (possibly overlapping) rectangles. Thus, the nondeterministic communication complexity of F, denoted $\mathsf{NP}^{\mathsf{cc}}(F)$ in analogy to the classical (Turing machine) complexity class NP, is just the logarithm of the least number of rectangles needed to cover $F^{-1}(1)$. A nondeterministic protocol is *unambiguous* if for each input, there is at most one accepting computation. Combinatorially, this means that the associated rectangles covering $F^{-1}(1)$ do not overlap. Hence we use the notation $\mathsf{UP}^{\mathsf{cc}}(F) := \lceil \log \chi_1(F) \rceil$ in analogy to the classical class UP. We also define $\mathsf{coUP}^{\mathsf{cc}}(F) := \lceil \log \chi_0(F) \rceil$, and, using the shorthand $\mathsf{2UP} := \mathsf{UP} \cap \mathsf{coUP}$, we define the two-sided measure $\mathsf{2UP}^{\mathsf{cc}}(F) := \lceil \log \chi(F) \rceil \in \max\{\mathsf{UP}^{\mathsf{cc}}(F), \mathsf{coUP}^{\mathsf{cc}}(F)\} \pm O(1)$.

Analogously, a *nondeterministic decision tree* (e.g., [Juk12, §14.2]) is a decision tree that is allowed to make guesses. Formally, we treat a nondeterministic decision tree for f as a collection of 1-certificates (accepting computations), that is, partial assignments to variables of f that force the output of the function to be 1; the cost is the maximum number of variables fixed by a partial assignment. In other words, a nondeterministic decision tree is just a DNF formula; the cost is the maximum width of its terms. We denote by $\mathsf{NP}^{\mathsf{dt}}(f)$ the minimum cost of a nondeterministic decision tree for f, that is, its DNF width. A nondeterministic decision tree is *unambiguous* if for each input, there is at most one accepting certificate. We denote by $\mathsf{UP}^{\mathsf{dt}}(f)$ the minimum cost of an unambiguous decision tree for f. We also let $\mathsf{coUP}^{\mathsf{dt}}(f) := \mathsf{UP}^{\mathsf{dt}}(\neg f)$ and $\mathsf{2UP}^{\mathsf{dt}}(f) := \max\{\mathsf{UP}^{\mathsf{dt}}(f), \mathsf{coUP}^{\mathsf{dt}}(f)\}$.

Communication $\leftrightarrow$ query. Generalizing (1), it is straightforward to check that a communication protocol can simulate a corresponding type of decision tree also in the case of our nondeterministic models. That is, for any $f\colon \{0,1\}^n \to \{0,1\}$ and for the gadget g from Theorem 3 we have

$$\mathcal{C}^{\mathsf{cc}}(f \circ g^n) \;\leq\; \mathcal{C}^{\mathsf{dt}}(f) \cdot O(\log n) \qquad \forall \mathcal{C} \in \{\mathsf{2UP}, \mathsf{UP}\}. \tag{2}$$

(It is not known whether the corresponding lower bounds hold above—we conjecture they do—but luckily we need only the upper bounds in this work.)

We can rephrase our communication results with the new notation defined above.

Theorem 1 (Rephrased). *There is an F such that* $\mathsf{P}^{\mathsf{cc}}(F) \geq \tilde{\Omega}(\mathsf{2UP}^{\mathsf{cc}}(F)^{1.5})$.

Theorem 2 (Rephrased). *There is an F such that* $\mathsf{P}^{\mathsf{cc}}(F) \geq \tilde{\Omega}(\mathsf{UP}^{\mathsf{cc}}(F)^{2})$.

Our goal is to prove analogous query complexity separations (in Section 2):

Theorem 4. *There is an f such that* $\mathsf{P}^{\mathsf{dt}}(f) \geq \tilde{\Omega}(\mathsf{2UP}^{\mathsf{dt}}(f)^{1.5})$.

Theorem 5. *There is an f such that* $\mathsf{P}^{\mathsf{dt}}(f) \geq \tilde{\Omega}(\mathsf{UP}^{\mathsf{dt}}(f)^{2})$.

Theorems 1–2 can now be derived by simply applying Theorem 3 and the upper bounds (2) to Theorems 4–5. We only add that the functions in Theorems 4–5 will actually satisfy $\mathsf{P}^{\mathsf{dt}}(f) = n^{\Theta(1)}$ and hence the factor $\Theta(\log n)$ overhead that is introduced by the gadget gets hidden in our $\tilde{\Omega}$-notation.

A few comments about Theorems 4–5 are in order. Firstly, Savický [Sav02] and Belovs [Bel06] have previously exhibited a function with $\mathsf{P}^{\mathsf{dt}}(f) \geq \Omega(2\mathsf{UP}^{\mathsf{dt}}(f)^{1.261})$. This means that a quantitatively weaker (but still superlogarithmic) version of Theorem 1 follows already by combining the Savický–Belovs result with Theorem 3. Secondly, it is not hard to see that $\mathsf{UP}^{\mathsf{dt}}(f) \geq \deg(f)$ where $\deg(f)$ is the minimum degree of a multilinear real polynomial that agrees with f on boolean inputs. (The communication analogue of this inequality, namely $\mathsf{UP}^{\mathsf{cc}}(F) \geq \log \operatorname{rank}(F)$, was discussed above.) Consequently, Theorem 5 gives the largest known gap between $\mathsf{P}^{\mathsf{dt}}(f)$ and $\deg(f)$. The previous record was $\mathsf{P}^{\mathsf{dt}}(f) \geq \Omega(\deg(f)^{1.63})$ by Kushilevitz [Kus94], and the current best upper bound in this context is $\mathsf{P}^{\mathsf{dt}}(f) \leq O(\deg(f)^3)$ for all f by Midrijānis [Mid04].

The work [Göö15] also uses the "query separation plus simulation theorem" approach to exhibit an F with $\mathsf{coNP}^{\mathsf{cc}}(F) \geq \Omega(\mathsf{UP}^{\mathsf{cc}}(F)^{1.128})$. The query separation in that paper involves a recursive composition that is intricate and delicate, due to the one-sided nature of coNP and UP. In contrast, our query separation is direct (without recursive composition) and much simpler to prove. However, the deterministic simulation theorem we employ is a fair bit more complicated to prove than the conondeterministic simulation theorem used in [Göö15] (which is a relatively simple special case of a general simulation theorem from [GLM$^+$16]).

2 Query Separations

In proving the query complexity separations it is convenient to work with functions $f\colon \Sigma^n \to \{0,1\}$ that have a larger-than-boolean input alphabet Σ. For such functions the understanding is that it still costs one query for a decision tree to learn a particular input variable. At the end, we may always convert such an f back to a boolean function $f \circ h^n$ where $h\colon \{0,1\}^{\lceil \log |\Sigma| \rceil} \to \Sigma$ is some surjection. The following trivial bounds suffice for us:

$$\mathcal{C}^{\mathsf{dt}}(f) \;\leq\; \mathcal{C}^{\mathsf{dt}}(f \circ h^n) \;\leq\; \mathcal{C}^{\mathsf{dt}}(f) \cdot \lceil \log |\Sigma| \rceil, \qquad \forall \mathcal{C} \in \{\mathsf{P}, 2\mathsf{UP}, \mathsf{UP}\}. \tag{3}$$

We start with the proof of Theorem 5 since the proof of Theorem 4 uses Theorem 5 (as a black box).

2.1 Proof of Theorem 5

Motivating example. Let $n \coloneqq k^2$, and consider the function $f\colon \{0,1\}^{k \times k} \to \{0,1\}$ defined on boolean matrices $M \in \{0,1\}^{k \times k}$ such that $f(M) = 1$ iff M contains a *unique* all-1 column. We claim that

$$\begin{aligned} \mathsf{NP}^{\mathsf{dt}}(f) \;&\leq\; 2k - 1, \\ \mathsf{P}^{\mathsf{dt}}(f) \;&\geq\; k^2. \end{aligned}$$

For the upper bound, consider 1-certificates that read the unique all-1 column and a single 0-entry from each of the other columns. (Note that this collection of certificates is not unambiguous!) For the lower bound, it suffices to give an *adversary argument* (see, e.g., [Juk12, §14]), that is, a strategy to answer queries made by a decision tree such that even after $k^2 - 1$ queries, the output of the function is not yet determined. Here is the strategy: Suppose the decision tree queries M_{ij}. If M_{ij} is the last unqueried entry in the j-th column, answer $M_{ij} = 0$. Otherwise answer $M_{ij} = 1$. It is straightforward to check that this strategy forces the decision tree to query all of the entries.

Actual gap example. We modify the function described above with the goal of establishing

$$\begin{aligned} \mathsf{UP}^{\mathsf{dt}}(f) &\leq 2k-1, \\ \mathsf{P}^{\mathsf{dt}}(f) &\geq k^2. \end{aligned}$$

The modified function, which we still call f, has input variables that take on values from the alphabet $\Sigma := \{0,1\} \times ([k] \times [k] \cup \{\bot\})$. Here $[k] \times [k] \cup \{\bot\}$ is a set of *pointer values*, where we interpret an entry $M_{ij} = (m_{ij}, p_{ij}) \in \Sigma$ as *pointing* to another entry $M_{p_{ij}}$ given that $p_{ij} \neq \bot$. If $p_{ij} = \bot$ then we have a null pointer. We define the function $f\colon \Sigma^{k\times k} \to \{0,1\}$ by describing an unambiguous decision tree computing it. (We give an "algorithmic" definition rather than writing a list of certificates.)

> *Unambiguous decision tree:* Nondeterministically guess a column index $j \in [k]$. Read the entries $M_{ij} = (m_{ij}, p_{ij})$ for $i \in [k]$ while checking that $m_{ij} = 1$ for all i and that $p_{ij} \neq \bot$ for at least one i. Let i be the first index for which $p_{ij} \neq \bot$. Next, iteratively follow pointers for $k-1$ steps starting at $(i_1, j_1) := p_{ij}$. Namely, at the s-th step, read M_{i_s,j_s} and if $s \leq k-2$ then check that $p_{i_s,j_s} \neq \bot$ and define $(i_{s+1}, j_{s+1}) := p_{i_s,j_s}$. Finally, check that the resulting sequence $(i_1, j_1), \ldots, (i_{k-1}, j_{k-1})$ visits all but the j-th column (i.e., $\{j_1, \ldots, j_{k-1}\} = [k] \smallsetminus \{j\}$) and that $m_{i_s,j_s} = 0$ for all $s \in [k-1]$.

Thus the upper bound holds by construction. For the lower bound, we use the below strategy; here a query to an entry M_{ij} is called *critical* if M_{ij} is the last unqueried entry in its column.

> *Adversary strategy:* Always answer queries with $(1, \bot)$ unless the query is critical. On the first critical query, answer $(0, \bot)$. On subsequent critical queries, answer $(0, p)$ where $p \in [k] \times [k]$ points to where the previous critical query took place.

The function value remains undetermined after $k^2 - 1$ queries, because we can answer the last (k^2-th) query with $(0, \bot)$ to make the function evaluate to 0, or with $(1, p)$, where p is as above, to make the function evaluate to 1. This proves $\mathsf{P}^{\mathsf{dt}}(f) \geq \Omega(\mathsf{UP}^{\mathsf{dt}}(f)^2)$ for a function with a non-boolean alphabet. If we convert f into a boolean function $f' := f \circ h^n$ (where $n := k^2$) as in (3) we end up with the claimed gap $\mathsf{P}^{\mathsf{dt}}(f') \geq \tilde{\Omega}(\mathsf{UP}^{\mathsf{dt}}(f')^2)$ since the conversion introduces only some $\lceil \log |\Sigma| \rceil = \Theta(\log n)$ factors.

2.2 Proof of Theorem 4

Let g be given by Theorem 5 such that $\mathsf{P}^{\mathsf{dt}}(g) = \tilde{\Theta}(q^2)$ where $q := \mathsf{UP}^{\mathsf{dt}}(g)$. We define $f := \mathrm{AND} \circ g^q$, that is, $f(z_1, \ldots, z_q) = 1$ iff $g(z_i) = 1$ for all $i \in [q]$. We claim that

$$\begin{aligned} 2\mathsf{UP}^{\mathsf{dt}}(f) &\leq \tilde{O}(q^2), \\ \mathsf{P}^{\mathsf{dt}}(f) &\geq \tilde{\Omega}(q^3). \end{aligned}$$

For the upper bound, an unambiguous certificate for an input z will contain unambiguous 1-certificates for $g(z_i) = 1$ for all $i \in [\ell - 1]$ where ℓ is the least index such that $g(z_\ell) = 0$, or $\ell := q+1$ if no such index exists. If $\ell \leq q$ we also include an unambiguous 0-certificate for $g(z_\ell) = 0$ that just mimics the execution of an optimal decision tree for g on input z_ℓ. In other words, we use the fact that $\mathsf{coUP}^{\mathsf{dt}}(g) \leq \mathsf{P}^{\mathsf{dt}}(g)$. The cost is at most $(\ell - 1) \cdot \mathsf{UP}^{\mathsf{dt}}(g) + \mathsf{P}^{\mathsf{dt}}(g) \leq \tilde{O}(q^2)$. For the lower bound, we have $\mathsf{P}^{\mathsf{dt}}(\mathrm{AND} \circ g^q) = \mathsf{P}^{\mathsf{dt}}(\mathrm{AND}) \cdot \mathsf{P}^{\mathsf{dt}}(g) = q \cdot \tilde{\Theta}(q^2) = \tilde{\Theta}(q^3)$ by the basic fact (e.g., [Sav02, Lemma 3.2]) that P^{dt} behaves multiplicatively with respect to composition.

3 Raz–McKenzie Simulation

The goal of this section is to give a self-contained, streamlined, and somewhat simplified proof of the Simulation Theorem that works without any assumptions on the outer function

$$f\colon \{0,1\}^N \to \{0,1\}.$$

(Here we use N for the input length instead of n, which we reserve for later use.) In fact, f can be taken to be anything: a partial function, a search problem (a general relation), or have a non-boolean codomain. However, we stick with the boolean function case for concreteness.

The gadget $g\colon [m] \times \{0,1\}^m \to \{0,1\}$, where $m \coloneqq N^{20}$, is chosen to be the indexing function defined by $g(x,y) \coloneqq y_x$. Recall that for the composed function $F \coloneqq f \circ g^N$, Alice's input is $x \coloneqq (x_1,\ldots,x_N) \in [m]^N$ and Bob's input is $y \coloneqq (y_1,\ldots,y_N) \in (\{0,1\}^m)^N$. We denote by $z_i \coloneqq g(x_i,y_i)$ the i-th input bit of f so that $F(x,y) \coloneqq f(z_1,\ldots,z_N)$.

We prove the nontrivial part of the Simulation Theorem, namely the lower bound

$$\mathsf{P}^{\mathsf{cc}}(f \circ g^N) \;\geq\; \mathsf{P}^{\mathsf{dt}}(f)\cdot\Omega(\log m).$$

3.1 High-level overview

Once and for all, we fix a deterministic protocol for $F \coloneqq f\circ g^N$ of communication cost $k \leq o(N\cdot\log m)$. The basic strategy is to use the protocol to build a decision tree of cost $O(k/\log m)$ for evaluating the outer function f on an unknown input $z \in \{0,1\}^N$. The simulation algorithm proceeds in iterations, where in each iteration we either descend one level in the communication protocol tree (by making the protocol send a bit), or descend one level in the decision tree (by querying a bit of z). To show the simulation is correct, we maintain invariants ensuring that when we reach a leaf in the protocol tree, the value it outputs must be the correct value of $f(z)$ (hence we can make the current node in the decision tree a leaf). To show the simulation is efficient, we use a potential function argument showing that in each "communication iteration" the potential increases by at most $O(1)$, and in each "query iteration" the potential decreases by at least $\Omega(\log m)$, and hence the number of query iterations is at most $O(k/\log m)$ since there are at most k communication iterations.

In a little more detail, let R_v denote the rectangle associated with the current node v in the communication protocol tree. The simulation maintains a "cleaned up" subrectangle $A \times B \subseteq R_v$ with the property that the set of all outputs of g^N over points in $A \times B$ is exactly the set of all possible z's that are consistent with the results of the queries made so far. This ensures the correctness when we reach a leaf. The analysis has two key lemmas: the Thickness Lemma helps us update $A \times B$ in a communication iteration, and the Projection Lemma helps us update $A \times B$ in a query iteration.

To determine which type of iteration should be next, we examine, for each unqueried coordinate, how predictable it is (in some sense) from the values of the other unqueried coordinates of g^N within $A \times B$. If no coordinate is too predictable, then it is "safe" to have a communication iteration; the protocol partitions the rectangle R_v into two parts, and we restrict to the part that is "bigger" (from the perspective of the unqueried coordinates), then use the Thickness Lemma to do some further clean-up that restores our invariants. On the other hand, if say the i-th coordinate is too predictable from the others, then its value (within $A \times B$) is in danger of becoming a function of the values of the other coordinates (which would violate our invariants). In this case, we query z_i while we are still able to accommodate either possible value for it (which might become impossible if we delayed querying z_i), and the Projection Lemma allows us to clean up $A \times B$ and restore our invariants.

We describe our notation and state the two key lemmas in Section 3.2. Then we describe the simulation algorithm itself in Section 3.3 and analyze it in Section 3.4. Finally, we provide the proofs of the two key lemmas in Section 3.5 and Section 3.6.

3.2 Notation and lemmas

For a node v in the communication protocol tree, let $R_v \coloneqq X^v \times Y^v$ denote its associated rectangle, let $X^{v,b} \subseteq X^v$ be the set of Alice's inputs on which the bit b would be sent (if Alice sends), and let $Y^{v,b} \subseteq Y^v$ be the set of Bob's inputs on which the bit b would be sent (if Bob sends).

Supposing $A \subseteq [m]^n$ and $B \subseteq (\{0,1\}^m)^n$ for some $n \le N$, we make the following definitions.

- **Size of sets:** Let $\alpha(A)$ be such that $|A| = 2^{-\alpha(A)} \cdot |[m]^n|$, and let $\beta(B)$ be such that $|B| = 2^{-\beta(B)} \cdot \big|(\{0,1\}^m)^n\big|$ (assuming $|A|, |B| > 0$).
- **Projections:** If $I \subseteq [n]$ then let $A_I \coloneqq \big\{(x_i)_{i \in I} : (x_1, \ldots, x_n) \in A \text{ for some } (x_j)_{j \in [n] \smallsetminus I}\big\} \subseteq [m]^{|I|}$ be the projection of A onto the coordinates in I, and similarly $B_I \coloneqq \big\{(y_i)_{i \in I} : (y_1, \ldots, y_n) \in B \text{ for some } (y_j)_{j \in [n] \smallsetminus I}\big\} \subseteq (\{0,1\}^m)^{|I|}$.
- **Pruning:** If $U \subseteq [m]$, $V \subseteq \{0,1\}^m$, and $i \in [n]$, then let $A^{i,U} \coloneqq \big\{x \in A : x_i \in U\big\}$ and $B^{i,V} \coloneqq \big\{y \in B : y_i \in V\big\}$.
- **Auxiliary graph:** If $i \in [n]$ then let $\mathrm{Graph}_i(A)$ be the bipartite graph defined as follows. The left nodes are $[m]$, the right nodes are $[m]^{n-1}$, and each tuple $x \coloneqq (x_1, \ldots, x_n) \in A$ is viewed as an edge between the left node x_i and the right node $(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n)$. Note that $A_{[n] \smallsetminus \{i\}}$ is the set of nonzero-degree right nodes.
- **Average/minimum degree:** Let $\mathrm{AvgDeg}_i(A) \coloneqq |A| / |A_{[n] \smallsetminus \{i\}}|$ and $\mathrm{MinDeg}_i(A)$ be, respectively, the average and minimum degrees of a nonzero-degree right node in $\mathrm{Graph}_i(A)$.
- **Thickness:** We say A is *thick* iff $\mathrm{MinDeg}_i(A) \ge m^{17/20}$ for all $i \in [n]$.

The following lemma is helpful for when we need to let the communication protocol send a bit.

Lemma 6 (Thickness Lemma). *If $n \ge 2$ and $A \subseteq [m]^n$ is such that $\mathrm{AvgDeg}_i(A) \ge d$ for all $i \in [n]$, then there exists an $A' \subseteq A$ such that*

(1) $\mathrm{MinDeg}_i(A') \ge d/2n$ *for all* $i \in [n]$,
(2) $\alpha(A') \le \alpha(A) + 1$.

The following lemma is helpful for when we need to have the decision tree query a bit.

Lemma 7 (Projection Lemma). *Suppose $n \ge 2$, $A \subseteq [m]^n$ is thick, and $B \subseteq (\{0,1\}^m)^n$ is such that $\beta(B) \le m^{2/20}$. Then for every $i \in [n]$ and every $b \in \{0,1\}$ there exists a b-monochromatic rectangle $U \times V \subseteq [m] \times \{0,1\}^m$ in g such that*

(1) $A^{i,U}_{[n] \smallsetminus \{i\}}$ *is thick,*
(2) $\alpha\big(A^{i,U}_{[n] \smallsetminus \{i\}}\big) \le \alpha(A) - \log m + \log \mathrm{AvgDeg}_i(A)$,
(3) $\beta\big(B^{i,V}_{[n] \smallsetminus \{i\}}\big) \le \beta(B) + 1$.

Algorithm 1: Simulation algorithm for Theorem 3

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

initialize $v = \text{root}$, $I = [N]$, $A = [m]^N$, $B = (\{0,1\}^m)^N$
while v is not a leaf **do**
  if $\text{AvgDeg}_i(A_I) \geq m^{19/20}$ for all $i \in I$ **then**
    let v_0, v_1 be the children of v
    if Alice sends a bit at v **then**
      let $b \in \{0,1\}$ be such that $\alpha\big((A \cap X^{v,b})_I\big) \leq \alpha(A_I) + 1$
      let $A' \subseteq (A \cap X^{v,b})_I$ be such that
        (1) A' is thick
        (2) $\alpha(A') \leq \alpha\big((A \cap X^{v,b})_I\big) + 1$
      update $A = \big\{x \in A \cap X^{v,b} : (x_i)_{i \in I} \in A'\big\}$ and $v = v_b$ (so now $A_I = A'$)
    else if Bob sends a bit at v **then**
      let $b \in \{0,1\}$ be such that $\beta\big((B \cap Y^{v,b})_I\big) \leq \beta(B_I) + 1$
      update $B = B \cap Y^{v,b}$ and $v = v_b$
    end
  else if $\text{AvgDeg}_i(A_I) < m^{19/20}$ for some $i \in I$ **then**
    query z_i
    let $U \times V \subseteq [m] \times \{0,1\}^m$ be a z_i-monochromatic rectangle of g such that
      (1) $A^{i,U}_{I \smallsetminus \{i\}}$ is thick
      (2) $\alpha\big(A^{i,U}_{I \smallsetminus \{i\}}\big) \leq \alpha(A_I) - (\log m)/20$
      (3) $\beta\big(B^{i,V}_{I \smallsetminus \{i\}}\big) \leq \beta(B_I) + 1$
    update $A = A^{i,U}$, $B = B^{i,V}$, and $I = I \smallsetminus \{i\}$
  end
end
output the same value that v does

3.3 Description of the simulation algorithm

The Simulation Theorem is witnessed by Algorithm 1, which is a decision tree for f that employs the hypothesized communication protocol for F. Algorithm 1 uses the following variables: v is a node in the communication protocol tree, $I \subseteq [N]$ is the set of unqueried coordinates, $A \subseteq [m]^N$ is a set of inputs to Alice, and $B \subseteq (\{0,1\}^m)^N$ is a set of inputs to Bob. We now exposit what Algorithm 1 is doing, with reference to the high-level overview in Section 3.1.

On input $z \in \{0,1\}^N$, the node variable v traces a root-to-leaf path (of length at most k) in the protocol tree, which is used to determine which z_i bits to query, and when. The set $A \times B$ is the "cleaned up" subrectangle of R_v (so we maintain $A \subseteq X^v$ and $B \subseteq Y^v$). We maintain the invariant that every $(x, y) \in A \times B$ is consistent with the results of the queries made so far (i.e., $g^N(x, y)$ agrees with z on queried coordinates), or in other words, $A_{\{i\}} \times B_{\{i\}}$ is z_i-monochromatic in g for $i \in [N] \smallsetminus I$. Thus we never need to worry about any coordinate that has previously been queried. The interesting structure in the sets A and B is what they look like on the unqueried coordinates, i.e., the projections A_I and B_I. Since all $2^{|I|}$ settings of the unqueried bits of z remain possible, we must maintain that all these settings are indeed possible outcomes of $g^{|I|}$ on points in $A_I \times B_I$. In fact we maintain a stronger property that turns out to entail this, namely that A_I is thick ($\text{MinDeg}_i(A_I) \geq m^{17/20}$ for every $i \in I$) and B_I is "large" (as measured by $\beta(B_I)$). The potential function is $\alpha(A_I)$; i.e., we look at the set of all projections of elements of A to the unqueried coordinates, and we consider how large this set is compared to its domain $[m]^{|I|}$. Smaller potential corresponds to a larger set.

We caution that the sets A and B in the statements of the Thickness Lemma and Projection Lemma will not be the $A \subseteq [m]^N$ and $B \subseteq (\{0,1\}^m)^N$ maintained by the algorithm, but rather will be subsets of the projected spaces $([m]^N)_I = [m]^n$ and $((\{0,1\}^m)^N)_I = (\{0,1\}^m)^n$, where n is the size of I.

Lines 2–23 are the main loop, with each iteration being either a "communication iteration" (if line 3 holds) in which we update v, A, B, or a "query iteration" (if line 15 holds) in which we update I, A, B. The type of iteration is determined by $\min_{i \in I} \text{AvgDeg}_i(A_I)$, which is our measure of how much the values of the unqueried coordinates are unpredictable from each other within $A \times B$.

In a communication iteration, there are two subcases depending on whether it is Alice's turn (line 5) or Bob's turn (line 11) to communicate. In either subcase, the bit of communication partitions R_v (and hence $A \times B$) into two parts, and we restrict our attention to the "bigger" part (lines 6 and 12) by having the communication protocol "send" the corresponding bit. Here, "bigger" is actually in terms of the projections A_I and B_I. This ensures the potential does not increase too much if Alice sends, and B_I stays large enough if Bob sends. However, if Alice sends, then the restriction to the bigger part may destroy the thickness invariant, and the Thickness Lemma is used (lines 7–9) to repair this.

In a query iteration, we have the decision tree query a bit z_i for which $\text{AvgDeg}_i(A_I)$ is too small (line 16). Then we can use the Projection Lemma (lines 17–20) to restrict $A \times B$ to a subrectangle on which the i-th output bit of g^N is fixed to z_i (for either possible value of $z_i \in \{0,1\}$); this exploits the fact that $\text{MinDeg}_i(A_I)$ is large by the thickness invariant. Furthermore, the fact that $\text{AvgDeg}_i(A_I)$ is small allows us to ensure a $\Omega(\log m)$ decrease in potential (i.e., the density of A_I increases). (Although the absolute size of A_I decreases, recall that the measure $\alpha(A_I)$ is relative to the current set I; by fixing the i-th coordinate, I becomes $I \smallsetminus \{i\}$, and since we fixed a coordinate of small average degree, the density projected to $I \smallsetminus \{i\}$ will increase a lot.)

3.4 Analysis of the simulation algorithm

We now formally argue that Algorithm 1 witnesses the Simulation Theorem (assuming the Thickness Lemma and the Projection Lemma). Assuming lines 7–9 and 17–20 always succeed (which we argue below), in each iteration one of the following three cases occurs.

- If lines 3 and 5 hold, then $\alpha(A_I)$ increases by ≤ 2 and $\beta(B_I)$ stays the same.
- If lines 3 and 11 hold, then $\alpha(A_I)$ stays the same and $\beta(B_I)$ increases by ≤ 1.
- If line 15 holds, then $\alpha(A_I)$ decreases by $\geq (\log m)/20$ and $\beta(B_I)$ increases by ≤ 1.

Since there are at most k iterations in which line 3 holds, and since $\alpha(A_I)$ is initially 0 and always nonnegative, it follows that there are at most $40k/\log m$ iterations in which line 15 holds, and hence the decision tree makes at most $40k/\log m$ queries. Moreover, since there are at most $k + 40k/\log m \leq m^{2/20}$ iterations, and $\beta(B_I)$ is initially 0, at all times we have $\beta(B_I) \leq m^{2/20}$.

Claim 8. *Lines 7–9 and 17–20 always succeed, and the following loop invariants are maintained.*

(i) *A_I is thick.*
(ii) *$A \times B \subseteq R_v$.*

(iii) $g(x_i, y_i) = z_i$ *for all* $(x, y) \in A \times B$ *and all* $i \in [N] \smallsetminus I$.

Proof. The invariants trivially hold initially. Now assume they hold at the beginning of an iteration.

Suppose lines 3 and 5 hold. For all $i \in I$, we have $\mathrm{AvgDeg}_i\big((A \cap X^{v,b})_I\big) = \big|(A \cap X^{v,b})_I\big| \,/\, \big|(A \cap X^{v,b})_{I \smallsetminus \{i\}}\big| \geq (|A_I|/2) \,/\, |A_{I \smallsetminus \{i\}}| = \mathrm{AvgDeg}_i(A_I)/2 \geq m^{19/20}/2$. Thus we may apply the Thickness Lemma with $(A \cap X^{v,b})_I$ (in place of A in the lemma), I identified with $[n]$, and $d \coloneqq m^{19/20}/2$ (noting that $d/2n \geq m^{19/20}/4m^{1/20} \geq m^{17/20}$) to conclude that lines 7–9 succeed, and hence (i) is maintained. Also, (ii) is maintained by line 10. If lines 3 and 11 hold, then (i) is trivially maintained and (ii) is maintained by line 13. Supposing line 3 holds, in either case (iii) is maintained since the new $A \times B$ is a subset of the old $A \times B$ and I is unchanged.

Now suppose line 15 holds. Since (i) holds and $\beta(B_I) \leq m^{2/20}$,[1] we may apply the Projection Lemma with A_I and B_I (in place of A and B in the lemma), I identified with $[n]$, and $b \coloneqq z_i$ (noting that $-\log m + \log \mathrm{AvgDeg}_i(A_I) \leq -(\log m)/20$) to conclude that lines 17–20 succeed, and hence (i) is maintained. The new $A \times B$ is a subset of the old $A \times B$; therefore, (ii) is maintained since v is unchanged, and (iii) is maintained since $U \times V$ is z_i-monochromatic in g. $\square$

Let v be the leaf reached at termination. We claim that there exists an $(x, y) \in R_v$ such that $g^N(x, y) = z$, and hence the algorithm indeed outputs $f(z) = F(x, y)$. Imagine that instead of terminating, the algorithm continues by executing lines 16–21 repeatedly, once for each remaining coordinate $i \in I$ in arbitrary order until only one coordinate remains unqueried—except that we ignore condition (2) (line 19). In this "extended" execution there are a total of $k + N - 1 \leq m^{2/20}$ iterations, so we have $\beta(B_I) \leq m^{2/20}$ at all times, and thus as in the proof of Claim 8, the application of the Projection Lemma always succeeds and invariants (i), (ii), (iii) are maintained.

Consider the state (i.e., v, I, A, B) at the end of this extended execution. Then I is a singleton, say $\{1\}$, and $|A_{\{1\}}| = \mathrm{MinDeg}_1(A_{\{1\}}) \geq m^{17/20}$ by (i), and $|B_{\{1\}}| \geq 2^{-m^{2/20}} \cdot 2^m = 2^{m - m^{2/20}}$. Hence $A_{\{1\}} \times B_{\{1\}}$ is not monochromatic in g, since the largest monochromatic rectangle with rows $A_{\{1\}}$ has at most $2^{m - |A_{\{1\}}|} < |B_{\{1\}}|$ columns. Pick an $(x_1, y_1) \in A_{\{1\}} \times B_{\{1\}}$ such that $g(x_1, y_1) = z_1$, and pick an $(x, y) \in A \times B$ with this value of (x_1, y_1). By (ii) we have $(x, y) \in R_v$, and by (iii) we also have $g(x_i, y_i) = z_i$ for all $i \in [N] \smallsetminus \{1\}$, and thus $g^N(x, y) = z$. The correctness is established.

3.5 Proof of the Thickness Lemma

The Thickness Lemma is witnessed by Algorithm 2, which constructs a sequence $A = A^0 \supseteq A^1 \supseteq A^2 \supseteq \cdots$ that converges to the desired set A'.

Algorithm 2: Algorithm for Lemma 6

```
let A^0 := A
for j = 0, 1, 2, ... do
    if MinDeg_i(A^j) ≥ d/2n for all i ∈ [n] then stop and output A' := A^j
    let i be such that MinDeg_i(A^j) < d/2n, and assume i = 1 for convenience of notation
    let (x*_2, ..., x*_n) be a nonzero-degree right node in Graph_1(A^j) with degree < d/2n
    let A^{j+1} := A^j ∖ {(x_1, x*_2, ..., x*_n) : x_1 ∈ [m]}
end
```

[1] There is no circular reasoning here; in showing that $\beta(B_I) \leq m^{2/20}$ we just needed that lines 7–9 and 17–20 succeeded in all iterations before this one.

If the algorithm terminates, then A' satisfies (1). We just need to argue that it does terminate, moreover with $|A'| \geq |A|/2$ (which is equivalent to (2)). In an iteration, it obtains $\mathrm{Graph}_i(A^{j+1})$ from $\mathrm{Graph}_i(A^j)$ by removing all edges incident to some right node in $A^j_{[n]\setminus\{i\}}$. Hence $\big|A^{j+1}_{[n]\setminus\{i\}}\big| = \big|A^j_{[n]\setminus\{i\}}\big| - 1$, and for every $i' \neq i$, $\big|A^{j+1}_{[n]\setminus\{i'\}}\big| \leq \big|A^j_{[n]\setminus\{i'\}}\big|$. Therefore, the total number of iterations is at most $\sum_{i=1}^n |A_{[n]\setminus\{i\}}| = \sum_{i=1}^n |A|/\operatorname{AvgDeg}_i(A) \leq n \cdot |A|/d$. Since $|A^{j+1}| > |A^j| - d/2n$ in each iteration, in total at most $(n \cdot |A|/d) \cdot (d/2n) = |A|/2$ elements of A can be removed throughout the execution. Thus the algorithm must terminate with $|A'| \geq |A|/2$.

3.6 Proof of the Projection Lemma

Assume $i = n$ for convenience of notation, so $A^{i,U}_{[n]\setminus\{i\}} = A^{n,U}_{[n-1]} = \big\{(x_1, \ldots, x_{n-1}) : (x_1, \ldots, x_n) \in A \text{ for some } x_n \in U\big\}$ (which is the set of right nodes in $\mathrm{Graph}_n(A)$ that have a neighbor in U) and $B^{i,V}_{[n]\setminus\{i\}} = B^{n,V}_{[n-1]} = \big\{(y_1, \ldots, y_{n-1}) : (y_1, \ldots, y_n) \in B \text{ for some } y_n \in V\big\}$.

We claim that if we take a uniformly random $U \subseteq [m]$ of size $m^{7/20}$ and let $V \coloneqq \big\{w \in \{0,1\}^m : w_j = b \text{ for all } j \in U\big\}$, then

(0) $A^{n,U}_{[n-1]} = A_{[n-1]}$ with probability greater than $1 - 2^{-m^{3/20}}$,
(1) $A_{[n-1]}$ is thick,
(2) $\alpha(A_{[n-1]}) \leq \alpha(A) - \log m + \log \operatorname{AvgDeg}_n(A)$,
(3) $\beta\big(B^{n,V}_{[n-1]}\big) \leq \beta(B) + 1$ with probability greater than $2^{-m^{3/20}}$.

The Projection Lemma then follows by a union bound. (We mention that our argument for property (3) is substantially different from and simpler than the corresponding part of the proof in [RM99].)

Property (0). For every nonzero-degree right node $(x_1, \ldots, x_{n-1}) \in A_{[n-1]}$ of $\mathrm{Graph}_n(A)$, let $L_{x_1,\ldots,x_{n-1}} \coloneqq \big\{x_n \in [m] : (x_1, \ldots, x_{n-1}, x_n) \in A\big\}$ denote the set of all left nodes adjacent to it. We have $|L_{x_1,\ldots,x_{n-1}}| \geq \operatorname{MinDeg}_n(A) \geq m^{17/20}$, and $(x_1, \ldots, x_{n-1}) \in A^{n,U}_{[n-1]}$ iff U intersects $L_{x_1,\ldots,x_{n-1}}$. Since U has size $m^{7/20}$, the probability U does not intersect $L_{x_1,\ldots,x_{n-1}}$ is at most $(1 - m^{17/20}/m)^{m^{7/20}} \leq e^{-m^{4/20}}$. Since the number of elements $(x_1, \ldots, x_{n-1}) \in A_{[n-1]}$ is at most $m^{n-1} \leq 2^{m^{1/20} \cdot \log m}$, by a union bound the probability that one of them is not in $A^{n,U}_{[n-1]}$ is at most $2^{m^{1/20} \cdot \log m} \cdot e^{-m^{4/20}} < 2^{-m^{3/20}}$.

Property (1). For this it suffices to show that $\operatorname{MinDeg}_j(A_{[n-1]}) \geq \operatorname{MinDeg}_j(A)$ for all $j \in [n-1]$. Assume $j = n-1$ for convenience of notation. For every nonzero-degree right node $(x_1, \ldots, x_{n-2})$ in $\mathrm{Graph}_{n-1}(A_{[n-1]})$, there exists x_{n-1} such that $(x_1, \ldots, x_{n-2}, x_{n-1}) \in A_{[n-1]}$. Thus by the definition of $A_{[n-1]}$ there exists x_n such that $(x_1, \ldots, x_{n-2}, x_{n-1}, x_n) \in A$. Therefore, by the definition of $\operatorname{MinDeg}_{n-1}(A)$ applied to the nonzero-degree right node $(x_1, \ldots, x_{n-2}, x_n)$ of $\mathrm{Graph}_{n-1}(A)$, we have that $(x_1, \ldots, x_{n-2}, x'_{n-1}, x_n) \in A$ holds for at least $\operatorname{MinDeg}_{n-1}(A)$ different elements x'_{n-1}. All these elements satisfy $(x_1, \ldots, x_{n-2}, x'_{n-1}) \in A_{[n-1]}$. Hence, the degree of the right node $(x_1, \ldots, x_{n-2})$ in $\mathrm{Graph}_{n-1}(A_{[n-1]})$ is at least $\operatorname{MinDeg}_{n-1}(A)$.

Property (2). We have $|A_{[n-1]}| = |A|/\operatorname{AvgDeg}_n(A)$ and $|[m]^{n-1}| = |[m]^n|/m$, and hence $\alpha(A_{[n-1]}) = \log\big(|[m]^{n-1}|/|A_{[n-1]}|\big) = \log\big(|[m]^n|/|A|\big) - \log\big(m/\operatorname{AvgDeg}_n(A)\big) = \alpha(A) - \log m + \log \operatorname{AvgDeg}_n(A)$. (Thus (2) holds with equality, but we only needed the inequality.)

Property (3). We first state a claim, whose proof we give later.

Claim 9. *For every $W \subseteq \{0,1\}^m$ with $\beta(W) \le m^{11/20}$, we have $\mathbf{Pr}_U[V \cap W \neq \emptyset] \ge 3/4$.*

In particular, for every $W \subseteq \{0,1\}^m$ we have $\mathbf{Pr}_U[V \cap W \neq \emptyset] \ge \frac{3}{4} \cdot |W|/2^m - 2^{-m^{11/20}}$. For every $(y_1, \dots, y_{n-1}) \in (\{0,1\}^m)^{n-1}$, let $W_{y_1,\dots,y_{n-1}} \coloneqq \big\{y_n \in \{0,1\}^m : (y_1, \dots, y_{n-1}, y_n) \in B\big\}$. Letting $(y_1, \dots, y_{n-1})$ be uniformly random in $(\{0,1\}^m)^{n-1}$, we have

$$\begin{aligned}
\mathbf{E}_U\Big[\big|B^{n,V}_{[n-1]}\big|/2^{m(n-1)}\Big] &= \mathbf{E}_{y_1,\dots,y_{n-1}}\ \mathbf{Pr}_U\Big[(y_1, \dots, y_{n-1}) \in B^{n,V}_{[n-1]}\Big] \\
&= \mathbf{E}_{y_1,\dots,y_{n-1}}\ \mathbf{Pr}_U\big[V \cap W_{y_1,\dots,y_{n-1}} \neq \emptyset\big] \\
&\ge \mathbf{E}_{y_1,\dots,y_{n-1}}\Big(\tfrac{3}{4} \cdot \big|W_{y_1,\dots,y_{n-1}}\big|/2^m - 2^{-m^{11/20}}\Big) \\
&= \tfrac{3}{4} \cdot |B|/2^{mn} - 2^{-m^{11/20}} \\
&\ge \tfrac{5}{8} \cdot |B|/2^{mn}
\end{aligned}$$

where the last line follows since $|B|/2^{mn} = 2^{-\beta(B)} \ge 2^{-m^{2/20}}$. It follows that with probability at least $\frac{1}{8} \cdot |B|/2^{mn} > 2^{-m^{3/20}}$ over U, we have $\big|B^{n,V}_{[n-1]}\big|/2^{m(n-1)} \ge \frac{1}{2} \cdot |B|/2^{mn}$, which is equivalent to (3). This finishes the proof of the Projection Lemma, except for the proof of Claim 9.

Recall that $b \in \{0,1\}$ is fixed. For $W \subseteq \{0,1\}^m$ and $j \in [m]$, define $W^j \coloneqq \{w \in W : w_j = b\}$ and $\mathrm{Bad}(W) \coloneqq \{j \in [m] : |W^j| < |W|/4\}$.

Claim 10. *For every $W \subseteq \{0,1\}^m$, $|\mathrm{Bad}(W)| \le 6\beta(W)$.*

Proof of Claim 10. Let w be a random variable uniformly distributed over W, and let $H(\cdot)$ denote Shannon entropy. There are at most $6\beta(W)$ coordinates j such that $\mathbf{Pr}[w_j = b] < 1/4$, since otherwise $H(w) \le \sum_{j=1}^m H(w_j) < 6\beta(W) \cdot H(1/4) + (m - 6\beta(W)) \cdot 1 \le m - 6\beta(W) \cdot (1 - 0.82) \le m - \beta(W)$, contradicting the fact that $H(w) = \log|W| = m - \beta(W)$. ☐

Proof of Claim 9. Suppose we sample $U \coloneqq \{j_1, \dots, j_{m^{7/20}}\}$ by iteratively picking each $j_{i+1} \in [m] \smallsetminus \{j_1, \dots, j_i\}$ uniformly at random. We write V as V_U, as a reminder that it depends on U. For $i \in \{0, 1, \dots, m^{7/20}\}$, define $W_i \coloneqq \big\{w \in W : w_{j_1} = w_{j_2} = \cdots = w_{j_i} = b\big\}$, and note that $W_0 = W$, $W_{i+1} = W_i^{j_{i+1}}$, and $W_{m^{7/20}} = V_U \cap W$. Let E_{i+1} denote the event that $j_{i+1} \notin \mathrm{Bad}(W_i)$, and note that if E_{i+1} occurs then $\beta(W_{i+1}) \le \beta(W_i) + 2$. Thus if $E_1 \cap \dots \cap E_{m^{7/20}}$ occurs then $\beta(V_U \cap W) \le \beta(W) + 2m^{7/20} < \infty$ and hence $V_U \cap W \neq \emptyset$. Conditioned on any particular outcome of $j_1, \dots, j_i$ for which $E_1 \cap \dots \cap E_i$ occurs, by Claim 10 we have $|\mathrm{Bad}(W_i)| \le 6\beta(W_i) \le 6(\beta(W) + 2i)$ and thus

$$\mathbf{Pr}\big[E_{i+1} \mid j_1, \dots, j_i\big] \ \ge\ 1 - \frac{|\mathrm{Bad}(W_i)|}{m - i} \ \ge\ 1 - \frac{6(\beta(W) + 2i)}{(6/7)m} \ \ge\ e^{-14(\beta(W)+2i)/m}$$

where the last inequality uses the fact that $1 - x \ge e^{-2x}$ if $x \in [0, 1/2]$, applied to $x \coloneqq 6(\beta(W) + 2i)/(6/7)m \le 7(m^{11/20} + 2m^{7/20})/m \le 1/2$. We conclude that

$$\begin{aligned}
\mathbf{Pr}[V_U \cap W \neq \emptyset] &\ge \mathbf{Pr}\big[E_1 \cap \dots \cap E_{m^{7/20}}\big] \\
&= \textstyle\prod_{i=0}^{m^{7/20}-1} \mathbf{Pr}\big[E_{i+1} \mid E_1 \cap \dots \cap E_i\big] \\
&\ge \textstyle\prod_{i=0}^{m^{7/20}-1} e^{-14(\beta(W)+2i)/m} \\
&= \exp\big(-\textstyle\sum_{i=0}^{m^{7/20}-1} 14(\beta(W) + 2i)/m\big)
\end{aligned}$$

$$
\begin{aligned}
&= \exp\bigl(-\tfrac{14}{m}\bigl(\beta(W)m^{7/20} + (m^{7/20}-1)m^{7/20}\bigr)\bigr) \\
&\geq \exp\bigl(-14\bigl(m^{-2/20} + m^{-6/20}\bigr)\bigr) \\
&\geq 3/4. \qquad \square
\end{aligned}
$$

4 Conclusion

Further observations. There is not much room for improvement in Theorem 2, since an $O(\mathsf{UP}^{\mathsf{cc}}(F)^2)$ upper bound is known. A closer inspection of our proof shows that $\mathsf{P}^{\mathsf{cc}}(F) \geq \Omega\bigl(\mathsf{UP}^{\mathsf{cc}}(F)^2/\log^3 \mathsf{UP}^{\mathsf{cc}}(F)\bigr)$. The lower bound can be improved to $\Omega\bigl(\mathsf{UP}^{\mathsf{cc}}(F)^2/\log^2 \mathsf{UP}^{\mathsf{cc}}(F)\bigr)$ by letting $h\colon \{0,1\}^{O(\log n)} \to \Sigma$ (as in (3) at the top of Section 2, where $|\Sigma| = \mathrm{poly}(n)$) be the decoder of any asymptotically good error-correcting code (such as the Justesen code). For such an h, any adversary strategy has the property that unless at least some small constant fraction of the input bits to h have been queried, every element of Σ remains a possible output of h. Thus an adversary strategy for h composes with the adversary strategy for f (from Section 2.1) to give $\mathsf{P}^{\mathsf{dt}}(f \circ h^n) \geq \Omega(n \log n)$. The upper bound $\mathsf{UP}^{\mathsf{dt}}(f \circ h^n) \leq O(\sqrt{n}\log n)$ is unchanged. The Simulation Theorem introduces another $\log n$ factor.

Grolmusz and Tardos [GT03] (building on [KNSW94]) generalized the fact that for all F, $\mathsf{P}^{\mathsf{cc}}(F) \leq O(\mathsf{UP}^{\mathsf{cc}}(F)^2)$ by showing that for all F and $\ell \geq 1$, $\mathsf{P}^{\mathsf{cc}}(F)$ is at most $O(\ell)$ times the square of the "ℓ-ambiguous nondeterministic communication complexity of F", which is the logarithm of the least number of rectangles so that each 1-input is covered between 1 and ℓ times, and no 0-input is covered. This can be shown to be tight by applying the Simulation Theorem to the corresponding query complexity separation. The latter is witnessed by taking the OR of ℓ independent copies of the function witnessing Theorem 5 (rather than taking the AND as in Theorem 4): the deterministic query complexity goes up by a factor of ℓ, but the ℓ-ambiguous query complexity of the new function is at most the unambiguous query complexity of the original function.

Subsequent developments. Our techniques have been adapted in several subsequent works. In [SW86], Saks and Wigderson conjectured that the largest separation between deterministic and zero-sided randomized query complexities should be a power roughly 1.326 (witnessed by the recursive AND–OR tree). The paper [ABB+16a] used variants of our function from Theorem 5 to disprove the Saks–Wigderson conjecture and, in fact, to give optimal (quadratic) separations between deterministic and zero-sided randomized query complexities and between zero-sided and one-sided randomized query complexities, as well as a fourth power separation between deterministic and quantum query complexities, among other separations. The paper [MS15] independently showed that our function from Theorem 5 already disproves the Saks–Wigderson conjecture. Inspired by our techniques, [ABK16] exhibited a function witnessing a power 2.5 separation between two-sided randomized and quantum query complexities, as well as a function witnessing a power 4 separation between quantum query complexity and approximate polynomial degree.

The paper [GJPW17] strengthened Theorem 1 and Theorem 2 to have randomized ($\mathsf{BPP}^{\mathsf{cc}}$) rather than deterministic lower bounds (using different functions). In particular, the query complexity analogues of these results (the $\mathsf{BPP}^{\mathsf{dt}}$ analogues of Theorem 4 and Theorem 5) were also shown, thereby quantitatively improving the main result from [KRS15] (which was the first superlinear $\mathsf{BPP}^{\mathsf{dt}}$ vs. $2\mathsf{UP}^{\mathsf{dt}}$ separation).

The exponent in our Theorem 1 and Theorem 4 has been improved from 1.5 to $2 - o(1)$ (which is essentially tight) in [AKK16]. Furthermore, several other query complexity separations were proven in [AKK16]: a power 2 separation between $\mathsf{BPP}^{\mathsf{dt}}$ and $\mathsf{2UP}^{\mathsf{dt}}$ (which has since been lifted to communication complexity [ABB+16b, GPW17]), a power 1.5 separation between $\mathsf{BQP}^{\mathsf{dt}}$ and $\mathsf{2UP}^{\mathsf{dt}}$, and a power 2 separation between $\mathsf{BQP}^{\mathsf{dt}}$ and $\mathsf{UP}^{\mathsf{dt}}$.

The papers [RW16, BR17] adapted and applied the Simulation Theorem to prove lower bounds on the communication complexity of finding Nash equilibria.

The Simulation Theorem itself has been improved in various ways: [dNV16] proved a version where rounds of communication in the protocol correspond directly to rounds of adaptivity in the decision tree (and the paper gave applications to proof complexity and circuit complexity), and [CKLM17, WYY17] improved the gadget to be the inner-product function on $O(\log n)$ bits (as was used in [GLM+16]).

Finally, our techniques have inspired the development of simulation theorems for other models: bounded-error randomized (BPP-type) protocols/decision trees [GPW17], as well as P^{NP}-type protocols/decision trees [GKPW17].

Acknowledgements

We thank Raghu Meka and Ran Raz for very helpful discussions.

References

[ABB+16a] Andris Ambainis, Kaspars Balodis, Aleksandrs Belovs, Troy Lee, Miklos Santha, and Juris Smotrovs. Separations in query complexity based on pointer functions. In *Proceedings of the 48th Symposium on Theory of Computing (STOC)*, pages 800–813. ACM, 2016. doi:10.1145/2897518.2897524.

[ABB+16b] Anurag Anshu, Aleksandrs Belovs, Shalev Ben-David, Mika Göös, Rahul Jain, Robin Kothari, Troy Lee, and Miklos Santha. Separations in communication complexity using cheat sheets and information complexity. In *Proceedings of the 57th Symposium on Foundations of Computer Science (FOCS)*, pages 555–564, 2016. doi:10.1109/FOCS.2016.66.

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

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

[AUY83] Alfred Aho, Jeffrey Ullman, and Mihalis Yannakakis. On notions of information transfer in VLSI circuits. In *Proceedings of the 15th Symposium on Theory of Computing (STOC)*, pages 133–139. ACM, 1983. doi:10.1145/800061.808742.

[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.

[BEGJ00] Maria Luisa Bonet, Juan Luis Esteban, Nicola Galesi, and Jan Johannsen. On the relative complexity of resolution refinements and cutting planes proof systems. *SIAM Journal on Computing*, 30(5):1462–1484, 2000. doi:10.1137/S0097539799352474.

[Bel06] Aleksandrs Belovs. Non-intersecting complexity. In *Proceedings of the 32nd Conference on Current Trends in Theory and Practice of Computer Science (SOFSEM)*, pages 158–165. Springer, 2006. doi:10.1007/11611257_13.

[BR17] Yakov Babichenko and Aviad Rubinstein. Communication complexity of approximate Nash equilibria. In *Proceedings of the 49th Symposium on Theory of Computing (STOC)*, pages 878–889, 2017. doi:10.1145/3055399.3055407.

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

[CLRS16] Siu On Chan, James Lee, Prasad Raghavendra, and David Steurer. Approximate constraint satisfaction requires large LP relaxations. *Journal of the ACM*, 63(4):34:1–34:22, 2016. doi:10.1145/2811255.

[dNV16] Susanna de Rezende, Jakob Nordström, and Marc Vinyals. How limited interaction hinders real communication (and what it means for proof and circuit complexity). In *Proceedings of the 57th Symposium on Foundations of Computer Science (FOCS)*, pages 295–304. IEEE, 2016. doi:10.1109/FOCS.2016.40.

[Dru09] Andrew Drucker. Multitask efficiencies in the decision tree model. In *Proceedings of the 24th Conference on Computational Complexity (CCC)*, pages 286–297. IEEE, 2009. doi:10.1109/CCC.2009.33.

[EIRS01] Jeff Edmonds, Russell Impagliazzo, Steven Rudich, and Jiří Sgall. Communication complexity towards lower bounds on circuit depth. *Computational Complexity*, 10(3):210–246, 2001. doi:10.1007/s00037-001-8195-x.

[GJPW17] Mika Göös, T.S. Jayram, Toniann Pitassi, and Thomas Watson. Randomized communication vs. partition number. In *Proceedings of the 44th International Colloquium on Automata, Languages, and Programming (ICALP)*, pages 52:1–52:15. Schloss Dagstuhl, 2017. doi:10.4230/LIPIcs.ICALP.2017.52.

[GKPW17] Mika Göös, Pritish Kamath, Toniann Pitassi, and Thomas Watson. Query-to-communication lifting for P^{NP}. In *Proceedings of the 32nd Computational Complexity Conference (CCC)*, pages 12:1–12:16. Schloss Dagstuhl, 2017. doi:10.4230/LIPIcs.CCC.2017.12.

[GLM+16] Mika Göös, Shachar Lovett, Raghu Meka, Thomas Watson, and David Zuckerman. Rectangles are nonnegative juntas. *SIAM Journal on Computing*, 45(5):1835–1869, 2016. doi:10.1137/15M103145X.

[Göö15] Mika Göös. Lower bounds for clique vs. independent set. In *Proceedings of the 56th Symposium on Foundations of Computer Science (FOCS)*, pages 1066–1076. IEEE, 2015. doi:10.1109/FOCS.2015.69.

[GPW15] Mika Göös, Toniann Pitassi, and Thomas Watson. Deterministic communication vs. partition number. In *Proceedings of the 56th Symposium on Foundations of Computer Science (FOCS)*, pages 1077–1088. IEEE, 2015. doi:10.1109/FOCS.2015.70.

[GPW17] Mika Göös, Toniann Pitassi, and Thomas Watson. Query-to-communication lifting for BPP. In *Proceedings of the 58th Symposium on Foundations of Computer Science (FOCS)*. IEEE, 2017. To appear.

[GT03] Vince Grolmusz and Gábor Tardos. A note on non-deterministic communication complexity with few witnesses. *Theory of Computing Systems*, 36(4):387–391, 2003. doi:10.1007/s00224-003-1158-7.

[HN12] Trinh Huynh and Jakob Nordström. On the virtue of succinct proofs: Amplifying communication complexity hardness to time–space trade-offs in proof complexity. In *Proceedings of the 44th Symposium on Theory of Computing (STOC)*, pages 233–248. ACM, 2012. doi:10.1145/2213977.2214000.

[Joh01] Jan Johannsen. Depth lower bounds for monotone semi-unbounded fan-in circuits. *RAIRO - Theoretical Informatics and Applications*, 35:277–286, 2001. doi:10.1051/ita:2001120.

[Juk12] Stasys Jukna. *Boolean Function Complexity: Advances and Frontiers*, volume 27 of *Algorithms and Combinatorics*. Springer, 2012.

[KLO99] Eyal Kushilevitz, Nathan Linial, and Rafail Ostrovsky. The linear-array conjecture in communication complexity is false. *Combinatorica*, 19(2):241–254, 1999. doi:10.1007/s004930050054.

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

[KNSW94] Mauricio Karchmer, Ilan Newman, Michael Saks, and Avi Wigderson. Non-deterministic communication complexity with few witnesses. *Journal of Computer and System Sciences*, 49(2):247–257, 1994. doi:10.1016/S0022-0000(05)80049-2.

[KRS15] Robin Kothari, David Racicot-Desloges, and Miklos Santha. Separating decision tree complexity from subcube partition complexity. In *Proceedings of the 19th International Workshop on Randomization and Computation (RANDOM)*, pages 915–930. Schloss Dagstuhl, 2015. doi:10.4230/LIPIcs.APPROX-RANDOM.2015.915.

[Kus94] Eyal Kushilevitz. Unpublished. Cited in [NW95], 1994.

[Lov14] Shachar Lovett. Recent advances on the log-rank conjecture in communication complexity. *Bulletin of EATCS*, 1(112), 2014. URL: http://bulletin.eatcs.org/index.php/beatcs/article/view/260/245.

[LRS15] James Lee, Prasad Raghavendra, and David Steurer. Lower bounds on the size of semidefinite programming relaxations. In *Proceedings of the 47th Symposium on Theory of Computing (STOC)*, pages 567–576. ACM, 2015. doi:10.1145/2746539.2746599.

[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.

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

[MS15] Sagnik Mukhopadhyay and Swagato Sanyal. Towards better separation between deterministic and randomized query complexity. In *Proceedings of the 35th Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS)*, pages 206–220. Schloss Dagstuhl, 2015. doi:10.4230/LIPIcs.FSTTCS.2015.206.

[NW95] Noam Nisan and Avi Wigderson. On rank vs. communication complexity. *Combinatorica*, 15(4):557–565, 1995. doi:10.1007/BF01192527.

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

[RW16] Tim Roughgarden and Omri Weinstein. On the communication complexity of approximate fixed points. In *Proceedings of the 57th Symposium on Foundations of Computer Science (FOCS)*, pages 229–238. IEEE, 2016. doi:10.1109/FOCS.2016.32.

[Sav02] Petr Savický. On determinism versus unambiguous nondeterminism for decision trees. Technical Report TR02-009, Electronic Colloquium on Computational Complexity (ECCC), 2002. URL: http://eccc.hpi-web.de/report/2002/009.

[She11] Alexander Sherstov. The pattern matrix method. *SIAM Journal on Computing*, 40(6):1969–2000, 2011. doi:10.1137/080733644.

[SW86] Michael Saks and Avi Wigderson. Probabilistic boolean decision trees and the complexity of evaluating game trees. In *Proceedings of the 27th Symposium on Foundations of Computer Science (FOCS)*, pages 29–38. IEEE, 1986. doi:10.1109/SFCS.1986.44.

[SZ09] Yaoyun Shi and Yufan Zhu. Quantum communication complexity of block-composed functions. *Quantum Information and Computation*, 9(5–6):444–460, 2009.

[WYY17] Xiaodi Wu, Penghui Yao, and Henry Yuen. Raz-McKenzie simulation with the inner product gadget. Technical Report TR17-010, Electronic Colloquium on Computational Complexity (ECCC), 2017. URL: https://eccc.weizmann.ac.il/report/2017/010/.

[Yan91] Mihalis Yannakakis. Expressing combinatorial optimization problems by linear programs. *Journal of Computer and System Sciences*, 43(3):441–466, 1991. doi:10.1016/0022-0000(91)90024-Y.

[Yao79] Andrew Chi-Chih Yao. Some complexity questions related to distributive computing. In *Proceedings of the 11th Symposium on Theory of Computing (STOC)*, pages 209–213. ACM, 1979. doi:10.1145/800135.804414.