Communication Complexity

Yassine Hamoudi
under the supervision of Anil Ada

February 1 - June 17, 2016

Abstract

In this internship report we describe several results about communication complexity, both in the two player and number on the forehead (NOF) models. Our first contribution is a construction for Ramsey numbers over $\mathbb{F}_p^n$ using communication complexity ideas. We then describe a new efficient protocol for composed functions of constant block-width, and the implications on the $\log n$ barrier problem. Finally, we recall the links of decision tree complexities to the log-rank conjecture, and fully characterize the decision tree complexities of symmetric functions.

Contents

1 Introduction

Communication complexity measures the amount of information to be exchanged in order to compute a function whose input is distributed between a certain number of players. For instance, if Alice and Bob each have an integer x and y, what is the minimum number of bits they need to communicate before deciding whether $x = y$? And what if they are allowed a small probability of error?

The two player communication complexity model was first formalized in the seminal paper [Yao79] from Yao. Later, Chandra, Furst and Lipton [CFL83] proposed the *number on the forehead* (NOF) model that generalizes to $k \geq 2$ players. Roughly speaking, each player now sees all the input, except the part which is written on her forehead. The computational power of everyone is unlimited, but the number of exchanged bits has to be minimized.

Communication complexity has proved to be of value in the study of many areas of computer science. It has applications in circuit complexity [HG91, BT94], streaming algorithms [AMS96], Ramsey theory [CFL83], branching programs [CFL83], proof complexity [BPS07], quasirandom graphs [CT93], etc. On the other hand, many basic questions in communication complexity remain open and the NOF model is still poorly understood.

In this report, we first study the links of communication complexity to Ramsey theory. We especially devise the first efficient construction for Ramsey numbers over $\mathbb{F}_p^n$. Then, two of the main open problems in communication complexity are addressed. The first one consists of finding a function which is hard to compute for $\geq \log n$ players (where n is the size of the input on each player's forehead). We prove that some candidates to break this barrier turn out to have efficient communication protocols. The second open problem is the famous log-rank conjecture, which states that the two party communication complexity is upper bounded by the log-rank of the *communication matrix*. One of the approaches to tackle this question uses a link between communication and decision tree complexities. We study the latter in the context of symmetric functions.

1.1 Definitions

Let $\mathcal{X}$, $\mathcal{Y}$ and $\mathcal{Z}$ be three arbitrary sets, and $F : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$. Consider two players, Alice and Bob, who respectively know $x \in \mathcal{X}$ and $y \in \mathcal{Y}$. They want to collaboratively evaluate $F(x, y)$. To this end, they *communicate* bits to each other according to a predetermined *protocol*. This protocol specifies whose turn it is to speak, and which bit is to be sent given the information exchanged so far and the input of the speaking player. It also determines when communication stops. At the end, both Alice and Bob must be able to recover $F(x, y)$ from their input and the transcript of the exchange. The cost of the protocol on input (x, y) is the number of exchanged bits. The total cost of the protocol is the worst case cost on all inputs (x, y).

Definition 1. The *deterministic (two player) communication complexity* of a function $F : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ is the smallest cost of a protocol computing $F(x, y)$ for all $(x, y) \in \mathcal{X} \times \mathcal{Y}$. This quantity is denoted by $\mathrm{D}_2(F)$.

The usual setting is $\mathcal{X} = \mathcal{Y} = \{0, 1\}^n$ and $\mathcal{Z} = \{0, 1\}$. It is always possible for one player to send her entire input to the other party (n bits), who then computes $F(x, y)$ and sends back the result (1 bit). Thus, we always have $\mathrm{D}_2(F) \leq n + 1$. On the other hand, a protocol is considered to be *efficient* if it has cost $\operatorname{polylog} n$. For instance, the $\textsc{Equality} : \{0, 1\}^n \times \{0, 1\}^n \to \{0, 1\}$ function, that outputs 1 if and only if $x = y$, is *hard* since it does not have any efficient protocol (we will see later that $\mathrm{D}_2(\textsc{Equality}) = \Omega(n)$).

The previous model can be extended by allowing the players to make decisions based on a shared random string. A protocol is then said to compute F with error ϵ if it correctly outputs $F(x,y)$ with probability $\geq 1-\epsilon$, for any $(x,y) \in \mathcal{X} \times \mathcal{Y}$.

Definition 2. The *randomized (public coin) communication complexity* of a function $F : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ is the smallest cost of a protocol computing F with error ϵ when the two players have access to a shared random string. This quantity is denoted by $\mathrm{R}_2^\epsilon(F)$. We also drop the ϵ when it is equal to $1/3$.

The Equality function is easy to compute in the randomized model. Indeed, Alice and Bob can exchange $x \cdot r$ and $y \cdot r$ for sufficiently many random $r \in \{0,1\}^n$ (where $x \cdot r = x_1 r_1 \oplus \cdots \oplus x_n r_n \in \{0,1\}$), and then decide $x = y$ if and only if they always observed $x \cdot r = y \cdot r$. It it easy to prove that this protocol succeeds with probability $3/4$ when two random r are used. Thus, the randomized model can be much more efficient than the deterministic one: $\mathrm{R}_2(\textsc{Equality}) = \mathcal{O}(1)$ whereas $\mathrm{D}_2(\textsc{Equality}) = \Omega(n)$.

Other kinds of models can be similarly defined (non-deterministic, quantum, randomized private coin, etc.). For a complete introduction to communication complexity, see the book [KN97]. We will also study the *simultaneous* model in which Alice and Bob do not interact with each other, but instead send information to a referee. The latter does not know the players' inputs, and cannot give any information back. At the end, the referee must be able to recover $F(x,y)$ from what she obtained. The simultaneous deterministic communication complexity is denoted by $\mathrm{D}_2^{||}(F)$, and the randomized one is $\mathrm{R}_2^{||}(F)$.

The *communication matrix* of a function $F : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ is the $|\mathcal{X}| \times |\mathcal{Y}|$ matrix whose entry (x,y) contains $F(x,y)$. The following result is a well-known lower bound in communication complexity:

Proposition 3 ([MS82]). *For any $F : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$, we have:*

$$\log \operatorname{rank} M_F \leq \mathrm{D}_2(F)$$

where the rank can be taken over any field.

Similar results hold in other frameworks (randomized, quantum, etc.), see [LS09] for a recent survey. In particular, it proves $\mathrm{D}_2(\textsc{Equality}) = \Omega(n)$ since $\operatorname{rank} M_{\textsc{Equality}} = 2^n$.

The two player model was later generalized by Chandra et al. [CFL83] to an arbitrary number of parties k. Given a function $F : \mathcal{X}_1 \times \cdots \times \mathcal{X}_k \to \mathcal{Z}$, player i now sees all of the input $(x_1, \ldots, x_k) \in \mathcal{X}_1 \times \cdots \times \mathcal{X}_k$, except x_i. The situation is as if input x_i was written on the forehead of player i, hence the name *number on the forehead* (NOF) model. The players still follow a protocol, and each bit someone sends is seen by all the other players. Note that when $k = 2$ this is equivalent to the two player model previously defined. The deterministic and randomized communication complexities are denoted by $\mathrm{D}_k(F)$ and $\mathrm{R}_k(F)$. The simultaneous model is also generalized to $\mathrm{D}_k^{||}(F)$ and $\mathrm{R}_k^{||}(F)$.

One of the interesting aspects of the NOF model is the increasing overlap of information as k grows up. For instance, the generalized $\textsc{Equality}_k$ function, which outputs 1 if and only if $x_1 = \cdots = x_k$, has complexity $\mathrm{D}_k(\textsc{Equality}_k) = \mathcal{O}(1)$ when $k \geq 3$. It suffices for player 1 to check if $x_2 = \cdots = x_k$ and for player 2 to check if $x_1 = x_3 = \cdots = x_k$, in order to know whether the output is 1.

1.2 Motivations and open problems

The log-rank result from Proposition 3 is a convenient way to obtain lower bounds in the two player model. Indeed, it converts a communication problem into the study of a well-known object in linear algebra. The *log-rank conjecture* proposed in [LS88] asserts that this result is tight:

Problem 1 (The log-rank conjecture)**.** Prove that $\mathrm{D}_2(F) \leq \log^c \operatorname{rank} M_f$ for some absolute constant c, and all $F : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ (where the rank is taken over $\mathbb{R}$).

This conjecture is a long-standing open problem in communication complexity. It was proved to be true for many classes of functions [BdW01, ZS09, TWXZ13], but the general case does not seem within easy reach (see [Lov14] for the latest advances). Recently, the attention has focused on two classes called the XOR and AND functions, for which the log-rank conjecture can be linked to Fourier analysis and decision tree complexity. We will further investigate this relation in Sections 1.3 and 4.

The log-rank method does not generalize to the NOF model. More generally, very few techniques are known to produce lower bounds for $k \geq 3$ players. However, the NOF model has much richer applications than the two player model. For instance, the communication complexity of the $\textsc{Eval}_G$ function (defined and studied in Section 2) is directly linked to Ramsey theory. Another major challenge is to find a function which is hard to compute for more than $\log n$ players:

Problem 2 (The $\log n$ barrier)**.** Find a function F such that $\mathrm{D}_k^{||}(F) = \omega(\operatorname{polylog} n)$ when $k \geq \operatorname{polylog} n$. The non-simultaneous case $\mathrm{D}_k(F) = \omega(\operatorname{polylog} n)$ is also of interest.

The main motivation for solving this problem comes from circuit complexity. Recall that any function in P can be computed by polynomial size circuits made of AND, OR and NOT gates. On the other hand, it is believed that not all functions in NP can be computed by such circuits. In particular, this conjecture implies $\mathsf{P} \neq \mathsf{NP}$. A first step toward this end is to prove an easier separation, namely $\mathsf{ACC}^0 \neq \mathsf{NP}$, where ACC^0 stands for the functions computable by polynomial size constant-depth circuits made of AND, OR, NOT and MOD_m gates. Finding a function f which is is NP but not in ACC^0 is directly linked to Problem 2:

Proposition 4 ([HG91])**.** *For any function f in ACC^0 and any partition of the input between $k = \Omega(\operatorname{polylog} n)$ players, there exists an efficient k-party simultaneous protocol of cost* $\operatorname{polylog} n$ *computing f.*

Proof. Let's define $\mathsf{SYM}^+(s,k)$ to be the class of functions computable by depth-2 circuits whose top gate is a symmetric gate (i.e. its output only depends on the number of inputs set to 1) of fan-in s, and each bottom gate is an AND gate of fan-in k. Yao, Beigel and Tarui [Yao90, BT94] proved that $\mathsf{ACC}^0 \subset \mathsf{SYM}^+(2^{\operatorname{polylog} n}, \operatorname{polylog} n)$.

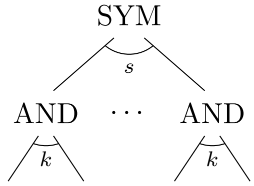

Figure 1: Structure of a $\mathsf{SYM}^+(s,k)$ circuit.

Consider now a function f computed by a $\mathsf{SYM}^+(s,k-1)$ circuit, and a partition of the input between k players. Each bottom gate has fan-in $k-1$, so it can be computed by at least one of the players. We fix a partition of the AND gates between the k players, such that player i only receives gate which she can evaluate. Then, each player sends to the referee the number of her gates that evaluate to 1. These information are enough to recover the output of the function. The total cost of the protocol is $\mathcal{O}(k \log s)$. $\square$

Consequently, any function solving Problem 2 cannot be in ACC^0. The majority function MAJ is conjectured to be outside of ACC^0 [Smo87], but the multiparty communication complexity of the functions involving MAJ is widely unknown. More generally, the strongest known lower bounds in the NOF model are of the form $\Omega(n/2^k)$, which does not give any information when $k \geq \log n$. It might seem like these bounds are not optimal, but there exist surprising protocols that start to be efficient when $k = \operatorname{polylog} n$. We will build such a protocol in Section 3, which prevents a new class of functions from solving Problem 2.

1.3 Fourier analysis of boolean functions

We introduce some notions of Fourier analysis for boolean functions. This tool is of great interest in the study of theoretical computer science, and will be used throughout this report. We refer the reader to the book [O'D14] for more details on the topic.

Any function $f : \{0,1\}^n \to \mathbb{R}$ can be uniquely written as:

$$f(x) = \sum_{s \in \{0,1\}^n} \hat{f}(s) \cdot (-1)^{x \cdot s}$$

where $x \cdot s = x_1 s_1 \oplus \cdots \oplus x_n s_n$ and $\hat{f}(s) \in \mathbb{R}$ are the *Fourier coefficients*. This expression is called the *Fourier transform* of f.

For $x \in \{0,1\}^n$, we define $|x| = \sum_{i=1}^n x_i$ the Hamming weight of x. The number of nonzero Fourier coefficients of f is the *monomial complexity* $\operatorname{mon}(f)$. The degree, $\deg(f)$, is the largest $|s|$ such that $\hat{f}(s) \neq 0$. If we replace each $(-1)^{x \cdot s}$ by $\prod\limits_{i : s_i = 1} (1 - 2x_i)$, we obtain a polynomial in $x_1, \ldots, x_n$ whose number of monomials is denoted by $\operatorname{mon}^\star(f)$.

We also define the *polynomial representation over* $\mathbb{F}_p$ of $f : \mathbb{F}_p^k \to \mathbb{F}_p$ as the unique polynomial:

$$f(x) = \sum_{0 \leq i_1, \ldots, i_k \leq p-1} f_p(i_1, \ldots, i_k) \cdot x_1^{i_1} \cdots x_k^{i_k}$$

with $f_p(i_1, \ldots, i_k) \in \mathbb{F}_p$. If we embedded $\{0,1\}$ into $\mathbb{F}_p$, this is a different way to represent boolean functions. This expression is sometimes more convenient to use, since the coefficients now lie in $\mathbb{F}_p$ instead of $\mathbb{R}$. We will denote by $\deg_p(f)$ and $\operatorname{mon}_p(f)$ the degree and number of monomials of the polynomial representation of f over $\mathbb{F}_p$.

Example 5. Let's consider the majority function $\mathrm{MAJ}_3 : \{0,1\}^3 \to \{0,1\}$ which outputs the input most frequent bit. The Fourier transform of MAJ_3 is:

$$\mathrm{MAJ}_3(x_1, x_2, x_3) = \frac{1}{2} - \frac{1}{4}(-1)^{x_1} - \frac{1}{4}(-1)^{x_2} - \frac{1}{4}(-1)^{x_3} + \frac{1}{4}(-1)^{x_1 + x_2 + x_3}$$

The polynomial representation over $\mathbb{F}_2$ is:

$$\mathrm{MAJ}_3(x_1, x_2, x_3) = x_1 x_2 + x_1 x_3 + x_2 x_3$$

To illustrate the use of Fourier analysis in communication complexity, let's focus on the following functions:

Definition 6. A function $F : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ is an *XOR function* if there exists $f : \{0,1\}^n \to \{0,1\}$ such that $F(x,y) = f(x \oplus y)$, where $\oplus$ is the bit-wise XOR. Similarly, F is said to be an *AND function* if $F(x,y) = f(x \wedge y)$, for some function $f : \{0,1\}^n \to \{0,1\}$.

Example 7. Here are some famous XOR and AND functions:

1. $\textsc{Equality}(x,y) = \mathrm{NOR}(x \oplus y)$, which outputs 1 if $x = y$.

2. $\textsc{Hamming}_d(x, y) = \mathrm{GAP}_d(x \oplus y)$ (where $\mathrm{GAP}_d(z) = 1$ if $\sum z_i \leq d$), which outputs 1 if the Hamming distance between x and y is less than d.

3. $\textsc{Disjointness}(x, y) = \mathrm{NOR}(x \wedge y)$, which outputs 1 if the sets X and Y, associated to the characteristic vectors x and y, are disjoint.

4. $\textsc{InnerProduct}(x, y) = \mathrm{MOD}_2(x \wedge y)$, which outputs 1 if $\sum x_i y_i = 1 \mod 2$.

The Fourier transforms of XOR and AND functions have interesting links with communication complexity (see [BdW01, TWXZ13] for instance). One of the main motivations for studying them is the two following results:

Proposition 8 ([BC99])**.** *For any $F(x, y) = f(x \oplus y)$, we have* $\mathrm{rank}(M_F) = \mathrm{mon}(f)$.

Proof. Define $H = [(-1)^{x \cdot y}]_{x,y \in \{0,1\}^n}$ to be the Hadamard matrix, and let D be the $2^n \times 2^n$ diagonal matrix with entries $[\hat{f}(s)]_{s \in \{0,1\}^n}$ on the diagonal. Then, it is easy to see that $M_F = HDH$, and since H is orthogonal:

$$\mathrm{rank}(M_F) = \mathrm{rank}(HDH) = \mathrm{rank}(D) = \mathrm{mon}(f)$$

□

Proposition 9 ([BdW01])**.** *For any $F(x, y) = f(x \wedge y)$, we have* $\mathrm{rank}(M_F) = \mathrm{mon}^\star(f)$.

Thus, using Proposition 3, we obtain $\log \mathrm{mon}(f) \leq \mathrm{D}_2(F)$ for XOR functions, and $\log \mathrm{mon}^\star(f) \leq \mathrm{D}_2(F)$ for AND functions. Moreover, the log-rank conjecture for XOR and AND functions is equivalent to proving $\mathrm{D}_2(F) \leq \log^c \mathrm{mon}(f)$ and $\mathrm{D}_2(F) \leq \log^c \mathrm{mon}^\star(f)$ respectively.

Finally, we will sometimes restrict to the *symmetric* functions, which are invariant under any permutation of the input variables. Note that being symmetric for a boolean function $f : \{0,1\}^n \to \{0,1\}$ means that $f(x)$ only depends on $|x|$. Thus, we will sometimes use f as a function $f : \{0, \ldots, n\} \to \{0,1\}$ with the understanding that $f(|x|) = f(x)$. Some natural quantities can also be associated with the symmetric boolean functions over $\{0,1\}^n$. For instance, we let $\ell_0(f)$ and $\ell_1(f)$ be the minimum integers less than $n/2$ such that $f(i) = f(i+1)$ for $i \in [\ell_0(f), n - \ell_1(f) - 1]$, and $\ell(f) = \max\{\ell_0(f), \ell_1(f)\}$. Similarly, we define $r_0(f)$ and $r_1(f)$ as the minimum integers less than $n/2$ such that $f(i) = f(i+2)$ for $i \in [r_0(f), n - r_1(f) - 2]$, and $r(f) = \max\{r_0(f), r_1(f)\}$. We also call $t(f)$ the smallest integer such that $f(t(f) - 1) \neq f(t(f))$ (if f is constant then $t(f) = n$). We will link these quantities to some complexity measures in Section 4.

2 Ramsey numbers and $\textsc{Eval}_G$

For any Abelian group G, the $\textsc{Eval}_G : G^k \to \{0,1\}$ function outputs 1 on input $x_1, \ldots, x_k \in G$ if and only if $x_1 + \cdots + x_k = 0$. It is one of the very first functions studied in the NOF model. In particular, since $x_1 + \cdots + x_k = 0$ is equivalent to $x_1 = -(x_2 \cdots + x_k)$, applying the randomized protocol for $\textsc{Equality}$ leads to:

$$\mathrm{R}_k^{||}(\textsc{Eval}_G) = \mathcal{O}(1)$$

On the other hand, the deterministic communication complexity of $\textsc{Eval}_G$ is way harder to determine. Indeed, as observed in [CFL83], it is intricately linked to certain Ramsey numbers, which are poorly understood. We first recall what this connection is in Section 2.1. We then propose the first non-trivial construction for Ramsey numbers over $\mathbb{F}_p^n$. Our work is based on ideas from a recent multiparty communication protocol for composed functions (see Section 3.1), and a previous result over $\mathbb{F}_2^n$ obtained in [ACFN15].

We will also briefly talk of $\textsc{Exact}_N : \{1, \dots, N\}^k \to \{0,1\}$ that outputs 1 if and only if $x_1 + \cdots + x_k = N$. Most of the results of Section 2.1 were in fact established for $\textsc{Exact}_N$ in the seminal paper [CFL83] that introduced the NOF model.

2.1 The multidimensional corner problem

A $k-$dimensional *corner* is a set of $k+1$ points in G^k of the form:

$$(x_1, x_2, \dots, x_k), (x_1 + \lambda, x_2, \dots, x_k), (x_1, x_2 + \lambda, \dots, x_k), \dots, (x_1, x_2, \dots, x_k + \lambda)$$

where $\lambda \neq 0$. We denote by $c_k^{\angle}(G)$ the minimum number of colors needed to color G^k, so that no $k-$dimensional corner is monochromatic. Also, $r_k^{\angle}(G)$ is defined to be the size of the largest corner-free subset of G^k. The communication complexity of $\textsc{Eval}_G$ is essentially equal to $\log c_k^{\angle}(G)$:

Proposition 10 ([CFL83]). *We have:*

$$\log(c_k^{\angle}(G)) \leq \mathrm{D}_{k+1}(\textsc{Eval}_G) \leq \mathrm{D}_{k+1}^{||}(\textsc{Eval}_G) \leq k \cdot \log(c_k^{\angle}(G))$$

and:

$$\mathrm{D}_{k+1}(\textsc{Eval}_G) \leq k + \log(c_k^{\angle}(G))$$

Proof. The proof is presented in Appendix A. □

Remark 11. Proposition 10 implies that any protocol for $\textsc{Eval}_G$ can be made simultaneous with an extra cost factor k (since $\mathrm{D}_{k+1}^{||}(\textsc{Eval}_G) \leq k \cdot \log(c_k^{\angle}(G)) \leq k \cdot \mathrm{D}_{k+1}(\textsc{Eval}_G)$).

Thus, finding the complexity of $\textsc{Eval}_G$ reduces to estimating the value of $c_k^{\angle}(G)$. To this end, it is also relevant to define the minimum number $c_k(G)$ of colors needed to color G so that no $k-$term arithmetic progression is monochromatic. Similarly, $r_k(G)$ is the size of the largest subset of G that does not contain any $k-$term arithmetic progression. The following lemmas link all these Ramsey numbers:

Lemma 12. *For any Abelian group G, we have:*

$$\frac{|G|^k}{r_k^{\angle}(G)} \leq c_k^{\angle}(G) \leq \frac{2|G|^k \log |G|^k}{r_k^{\angle}(G)}$$

and:

$$\frac{|G|}{r_k(G)} \leq c_k(G) \leq \frac{2|G| \log |G|}{r_k(G)}$$

Proof. These are straightforward generalizations of Theorem 4.3 from [CFL83]. □

Lemma 13. *For any Abelian group G, we have:*

$$r_{k+1}(G) \leq \frac{r_k^{\angle}(G)}{|G|^{k-1}}$$

and for $G = \mathbb{F}_p^n$:

$$c_k^{\angle}(\mathbb{F}_p^n) \leq c_k(\mathbb{F}_{kp}^n)$$

Proof. The first inequality is proved via a standard reduction. See [Ada14] for a sketch of it when $k = 2$. The second one is straightforward generalization of Theorem 4.2 from [CFL83]. $\square$

The counterpart numbers $c_k^{\angle}(N)$, $r_k^{\angle}(N)$, $c_k(N)$ and $r_k(N)$ for $\textsc{Exact}_N$ are similarly defined in [CFL83]. They satisfy $\frac{N^k}{r_k^{\angle}(N)} \leq c_k^{\angle}(N) \leq \frac{2N^k \log N^k}{r_k^{\angle}(N)}$, $\frac{N}{r_k(N)} \leq c_k(N) \leq \frac{2N \log N}{r_k(N)}$ and $c_k^{\angle}(N) \leq c_k(kN)$. We will also let $N = |G|$ when working over an Abelian group G, in order to make the comparison easier.

The famous Van der Waerden's and Szemerédi's theorems prove that $c_k(N)$ and $N/r_k(N)$ are superconstant. The best (very weak) lower bounds [FK78, Gow07] known on $\frac{N^k}{r_k^{\angle}(N)}$ also implies that $\mathrm{D}_k(\textsc{Exact}_N)$ is superconstant (whereas $\mathrm{R}_k(\textsc{Exact}_N) = \mathcal{O}(1)$). On the other hand, using an upper bound on $c_2^{\angle}(N)$ due to Behrend [Beh46], Chandra, Furst and Lipton [CFL83] proved that $\mathrm{D}_3(\textsc{Exact}_N) = \mathcal{O}\left(\sqrt{\log N}\right)$. There have been few improvements since (see [ACFN15] for the recent results relevant to $\textsc{Exact}_N$).

On the other hand, it was observed that lower bounds on $N/r_k(N)$ were in fact simpler to handle in the finite field setting. Moreover, an argument from Bourgain [Bou99] makes possible to convert results over $\mathbb{F}_p^n$ into results over any Abelian group G (see [Gre05] for instance). Some results about Ramsey numbers over $\mathbb{F}_p^n$ are gathered in [ACFN15]. In particular, the best known lower bound on $N^2/r_2^{\angle}(\mathbb{F}_p^n)$ is due to [LM07]:

$$\frac{N^2}{r_2^{\angle}(\mathbb{F}_p^n)} \geq \frac{\log \log N}{\log \log \log N}$$

Using an efficient protocol for $\textsc{Eval}_{\mathbb{F}_2^n}$, the authors of [ACFN15] established the first non-trivial upper bound on $c_k^{\angle}(\mathbb{F}_2^n)$, namely $c_k^{\angle}(\mathbb{F}_2^n) \leq \mathcal{O}\left(N^{1/2^{k-2}} \log^{k+1} N\right)$. Moreover, they described an explicit large corner-free set that matches this bound.

The only known upper bound for general $\mathbb{F}_p^n$ stems from a recent communication protocol that applies to $\textsc{Eval}_{\mathbb{F}_p^n}$ (see [CS14] and Proposition 23):

Proposition 14. *If* $k > 1 + p\log(3n)$ *then:*

$$\frac{N^k}{r_k^{\angle}(\mathbb{F}_p^n)} \leq c_k^{\angle}(\mathbb{F}_p^n) \leq 2^{\mathcal{O}\left(p \log^2 n\right)} p^{\mathcal{O}(p \log n)}$$

This result is obtained via the reduction of Proposition 10. Thus, it does not give an explicit description of a large corner-free set over $\mathbb{F}_p^n$. A construction of such a set is provided for the first time in next section.

2.2 A large corner-free set over $\mathbb{F}_p^n$

We describe the first non-trivial corner-free set over $\mathbb{F}_p^n$. Our construction is inspired by the communication protocol from [CS14] (see the proof of Proposition 23) and the previous corner-free set built over $\mathbb{F}_2^n$ in [ACFN15].

We interpret each $M \in (\mathbb{F}_p^n)^k$ as a $k \times n$ matrix over $\mathbb{F}_p$, whose columns are $c_1, \ldots, c_n \in \mathbb{F}_p^k$. For all $c \in \mathbb{F}_p^k$, the Hamming distance $d(c, c_j)$ between c and c_j is the number of coordinates at which c and c_j differ. We also define the following quantity:

$$n_{i,c}(M) = |\{j \in \{1, \ldots, n\} : d(c, c_j) = i\}|$$

The next proposition provides a general way to build corner-free sets over $\mathbb{F}_p^n$.

Proposition 15. *Let $N_k = 0$ and $N_0, \ldots, N_{k-1} \geq 0$ such that $\sum_{i=0}^{k} N_i = n$. Then*

$$S_c^k = \{M \in (\mathbb{F}_p^n)^k : \forall i \in \{0, \ldots, k\}, n_{i,c}(M) = N_i\}$$

is a corner-free set.

Proof. The proof goes as in [ACFN15], Theorem 4.4.

Let's assume that S_c^k contains a corner. Then there exist $M \in S_c^k$ and $\lambda \in \mathbb{F}_p^n \setminus \{0\}$ such that $M + \lambda^\ell \in S_c^k$ for all $\ell \in \{1, \ldots, k\}$ (where $\lambda^\ell \in (\mathbb{F}_p^n)^k$ is zero everywhere, except for the ℓ-th row where it is equal to λ).

Consider the columns of M corresponding to indices j such that $\lambda_j \neq 0$. Let t denotes the maximum Hamming distance to c among these columns. Note that $t < k$ since the number $n_{k,c}$ of columns at distance k to c is zero ($N_k = 0$).

The columns of M at distance $t + 1$ to c remain intact in $M + \lambda^\ell$ for all ℓ. However, by definition of t, there exists j and ℓ' such that column j is at distance t from c in M, and at distance $t + 1$ from c in $M + \lambda^{\ell'}$ (because $\lambda_j \neq 0$). Thus, $n_{t+1,c}(M + \lambda^{\ell'}) > n_{t+1,c}(M)$. This is a contradiction since $n_{i,c}(X)$ is constant for all $X \in S_c^k$. $\square$

Remark 16. If we restrict our attention to $\mathbb{F}_2^n$ and take $c = (1, \ldots, 1)$, we obtain the reasoning carried out in [ACFN15].

It remains to choose the N_i's so as to maximize the size of S_c^k. We use the following parameters:

$$\begin{cases} N_i = \left\lfloor \binom{k}{i} \frac{(p-1)^i}{p^k} n \right\rfloor, \ 1 \leq i \leq k-1 \\ N_0 = n - \sum\limits_{i=1}^{k-1} N_i \end{cases}$$

We now estimate the size of the associated set:

Theorem 17. *Let $n, p \geq 2$ and $k \geq \left\lceil \frac{\log n}{\log\left(1+\frac{1}{p-1}\right)} \right\rceil$. The set S_c^k defined above does not contain a corner, and*

$$|S_c^k| \geq \frac{N^k}{C^{k^2} p^{k+k^2}}$$

for some absolute constant C, and $N = \left|\mathbb{F}_p^n\right|$.

The proof is rather computational, and is left to Appendix B. Note that the size we obtain is close to the lower bound $r_k^{\angle}(\mathbb{F}_p^n) \geq \frac{N^k}{2^{\mathcal{O}(p \log^2 n)} p^{\mathcal{O}(p \log n)}}$ from Proposition 14 when $k \approx p \log n$.

2.3 Future work

The Eval_G and Exact_N functions are undeniably among the most important ones in communication complexity. Their randomized communication complexity is $\mathcal{O}(1)$. However, the deterministic communication cost is poorly known, but is conjectured to be high in many interesting cases. For instance, $\mathrm{D}_3(\text{Exact}_N)$ is believed to be close to the known upper bound $\mathcal{O}\left(\sqrt{\log N}\right)$. If proved, it would be the first efficient separation between randomized and deterministic communication complexity for $k \geq 3$ players.

On the other hand, communication complexity has proved to be of interest for studying Ramsey theory. It provided the first upper bounds over $\mathbb{F}_p^n$ and inspired the construction of large corner-free sets. It is always relevant to try to convert communication protocols into Ramsey constructions, and vice versa.

Finally, the $\textsc{Eval}_G$ function (for well-chosen G) is conjectured to break the $\log n$ barrier. It turns out that $\textsc{Eval}_{\mathbb{F}_p^n}$ has a particular structure that makes it easier to study. Indeed, it belongs to the family of *composed functions* that will be studied in next section.

3 The $\log n$ barrier and composed functions

As explained in introduction, finding a function that breaks the $\log n$ barrier is one of the main open questions in communication complexity. Here we study this problem for a large class of functions called the *composed functions.* We especially give the first efficient simultaneous protocol for a certain class of composed functions of constant block-width.

3.1 Previous candidates to break the barrier

One of the first strong lower bounds in the NOF model was obtained for the GIP function, defined as follow:

Definition 18. Given $x_1, \dots, x_k \in \{0,1\}^n$, the *Generalized Inner Product (GIP)* function for k players outputs $\sum_{i=1}^{n} x_{1,i} \cdots x_{k,i} \mod 2$.

Babai, Nisan and Szegedy proved in [BNS92] that $\mathrm{R}_k(\mathrm{GIP}) \geq \Omega(n/4^k)$. Thus, GIP is hard up to $(1-\epsilon)\log n$ players. It might seem like GIP remains hard for $k \geq \log n$ players. However, Grolmusz [Gro94] found later an efficient (non simultaneous) protocol of cost $\log^2 n$ when $k \geq \log n$.

The GIP function can be seen as an element of a broader family, called the *composed functions*:

Definition 19. Let $f : \{0,1\}^n \to \{0,1\}$ and $\overrightarrow{g} = (g_1, \dots, g_n)$ where $g_i : \{0,1\}^k \to \{0,1\}$. Given $x_1, \dots, x_k \in \{0,1\}^n$, the *composed function* $f \circ \overrightarrow{g}$ for k players outputs $f \circ \overrightarrow{g}(x_1, \dots, x_k) = f(\dots, g_i(x_{1,i}, \dots, x_{k,i}), \dots)$. When $g = g_1 = \dots = g_n$, we will denote it by $f \circ g$.

It is convenient to visualize the input of a composed function as a $k \times n$ matrix M over $\{0,1\}$, where row i is the number x_i on the forehead of player i, and column j is the input of function g_j (see Figure 2). By definition of the NOF model, player i sees all of M except row j.

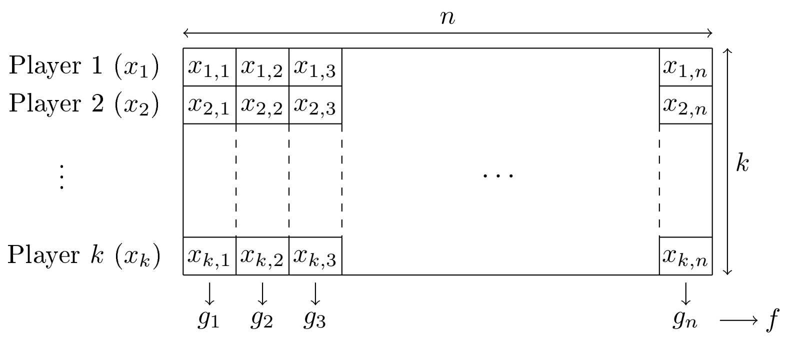

Figure 2: Matrix structure of a composed function $f \circ \overrightarrow{g}$ on input $(x_1, \dots, x_k)$.

We will call $\textsc{Any} \circ \overrightarrow{\textsc{Any}}$ (resp. $\textsc{Any} \circ \textsc{Any}$) the set of all composed functions $f \circ \overrightarrow{g}$ (resp. $f \circ g$). We define similarly $\textsc{Sym} \circ \textsc{Sym}$ for symmetric f and symmetric g, $\textsc{Sym} \circ \overrightarrow{\textsc{Any}}$ for symmetric f and any $\overrightarrow{g}$, etc.

Example 20. Here are some example of composed functions:

1. $\text{GIP} = \text{MOD}_2 \circ \text{AND} \in \textsc{Sym} \circ \textsc{Sym}$, the Generalized Inner Product.
2. $\text{MAJ} \circ \text{MAJ} \in \textsc{Sym} \circ \textsc{Sym}$, where MAJ is the Majority function.
3. $\text{DISJ} = \text{NOR} \circ \text{AND} \in \textsc{Sym} \circ \textsc{Sym}$, the generalized Disjointness function (for which is known [RY15] that $\text{R}_k(\text{DISJ}) \geq \Omega(n/4^k)$).

The (*non simultaneous*) protocol from [Gro94] applies to all composed functions in $\textsc{Sym} \circ \text{AND}$ (the inner function g must be the AND function).

Next, Babai, Kimmel and Lokam [BKL95] proposed $\text{MAJ} \circ \text{MAJ}$ as a candidate to break the barrier (since MAJ is conjectured to be outside ACC^0). However, they found later an efficient *simultaneous* protocol for $\textsc{Sym} \circ \textsc{Sym}$:

Proposition 21 ([BGKL04])**.** *Let M be a $k \times n$ matrix over $\{0,1\}$ with $k > 1 + \lceil \log n \rceil$. For $0 \leq i \leq k$, let y_i be the number of columns with i ones. For $j = 1, \ldots, k$, let player j see all of M except row j. Then there exists a simultaneous multiparty protocol in which each player sends $\mathcal{O}(k \log n)$ bits to the referee, after which the referee can calculate $y_0, \ldots, y_k$.*

According to the interpretation of Figure 2, this provides a protocol of total cost $\mathcal{O}\left(k^2 \log n\right)$ for any $f \circ g \in \textsc{Sym} \circ \textsc{Sym}$. Indeed, since f and g are symmetric, recovering the y_i's is enough to compute $f \circ g(x_1, \ldots, x_k)$. We briefly state the proof of Proposition 21:

Proof of Proposition 21. For all $1 \leq j \leq k$ and $0 \leq i \leq k-1$, player j sends to the referee the number $a_j(i)$ of columns she sees with exactly i ones. Note that player j does not see row j, so she cannot see k ones in a same column. If we denote $b_i = \sum_{j=1}^k a_j(i)$, we observe that the y_i's must satisfy the following equations:

$$\begin{cases} (k-i)y_i + (i+1)y_{i+1} = b_i, \ i = 0, 1, \ldots, k-1 \\ y_i \geq 0, \ 0 \leq i \leq k \quad \textit{and} \quad \sum_{i=0}^k y_i \leq n \end{cases}$$

If it admits only one *integral* solution, the referee can recover it and compute $f \circ g(x_1, \ldots, x_k)$. Let us assume that it is not the case, and denote by $y = (y_i)_{0 \leq i \leq k}$ and $y' = (y'_i)_{0 \leq i \leq k}$ two different solutions. For all i, define $d_i = y_i - y'_i$. Since $y_i + y'_i \geq |y_i - y'_i| = |d_i|$, we obtain:

$$\begin{cases} (k-i)d_i + (i+1)d_{i+1} = 0, \ i = 0, 1, \ldots, k-1 \\ \sum_{i=0}^k |d_i| \leq 2n \end{cases}$$

Thus, $d_1 = -kd_0 = -\binom{k}{1}d_0$, $d_2 = -\frac{k-1}{2}d_1 = \binom{k}{2}d_0$, and more generally:

$$d_i = (-1)^i \binom{k}{i} d_0$$

However, since $y \neq y'$, one of the d_i's is different from 0. It implies that $d_0 \neq 0$ and $|d_i| = \binom{k}{i}|d_0| \geq \binom{k}{i}$ for all i. We obtain a contradiction:

$$2n \geq \sum_{i=0}^k |d_i| \geq \sum_{i=0}^k \binom{k}{i} = 2^k > 2^{1+\log n} = 2n$$

□

When the number k of players is polylog n, this is an efficient protocol over $\textsc{Sym} \circ \textsc{Sym}$. For larger k, Babai et al. only showed how to handle efficiently $\textsc{Sym} \circ \textsc{Comp}$, where $\textsc{Comp}$ (*compressible symmetric functions*) is a subclass of $\textsc{Sym}$ that includes the functions of Example 20 (*compressibility* will be defined in Section 3.2). Later, combining ideas from [Gro94] and [BGKL04], Ada et al. [ACFN15] removed the compressibility condition and provided an efficient *simultaneous* protocol of cost $\mathcal{O}\left(\log^3 n\right)$ for any $f \circ \overrightarrow{g} \in \textsc{Sym} \circ \overrightarrow{\textsc{Any}}$ and $k > 1 + 2\log n$. In other words, none of the functions in $\textsc{Sym} \circ \overrightarrow{\textsc{Any}}$ can break the $\log n$ barrier.

The next step was to study composed functions of larger block-width t. Instead of g_i acting on a single $k \times 1$ column of M, we now have $g_i : \{0,1\}^{k \cdot t} \to \{0,1\}$ acting on t columns of M. See Figure 3 for the matrix representation.

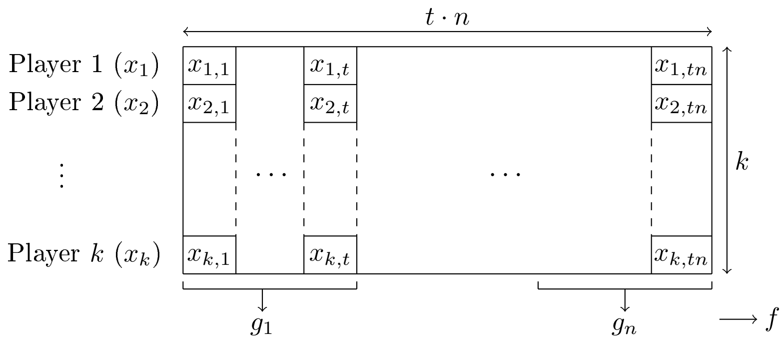

Figure 3: Matrix structure of a composed function $f \circ \overrightarrow{g}$ of block-width t.

The MAJ $\circ$ MAJ function is generalized to $\text{MAJ} \circ \text{MAJ}_t$ where $\text{MAJ}_t : \{0,1\}^{k \cdot t} \to \{0,1\}$ outputs 1 if at least $kt/2$ bits of the input are set to 1. It is conjectured that $\text{MAJ} \circ \text{MAJ}_{\sqrt{n}}$ breaks the $\log n$ barrier. However, even the case $t = 2$ is unsolved.

A more convenient way to look at composed functions of block-width t is to interpret each sub-row $r \in \{0,1\}^t$ of each block as a number in $\mathbb{F}_{2^t}$. Thus, a composed function over $\mathbb{F}_p$ is defined as $f \circ \overrightarrow{g}$ where $\overrightarrow{g} = (g_1, \ldots, g_n)$ and $g_i : \mathbb{F}_p^k \to \{0,1\}$. The corresponding $k \times n$ matrix M has now entries in $\mathbb{F}_p$ instead of $\{0,1\}$, and each g_i acts on a single column of M. We call $\textsc{Any} \circ \overrightarrow{\textsc{Any}_p}$ the family of composed functions over $\mathbb{F}_p$ (we define similarly $\textsc{Any} \circ \textsc{Any}_p$, $\textsc{Sym} \circ \textsc{Any}_p$, etc.). Note for instance that the class $\textsc{Any} \circ \textsc{Any}$ of Definition 19 is in fact $\textsc{Any} \circ \textsc{Any}_2$.

Example 22. The $\textsc{Eval}_{\mathbb{F}_p}$ function studied in Section 2 belongs to $\textsc{Sym} \circ \textsc{Sym}_p$ (since $\textsc{Eval}_{\mathbb{F}_p} = \text{NOR} \circ \text{MOD}_p$). We can also interpret $\text{MAJ} \circ \text{MAJ}_t$ as an element of $\textsc{Sym} \circ \textsc{Sym}_{2^t}$ with $\text{MAJ}_t : \mathbb{F}_{2^t}^n \to \{0,1\}$.

The first efficient protocol for $\textsc{Sym} \circ \overrightarrow{\textsc{Any}_p}$ was proposed by Chattopadhyay and Saks:

Proposition 23 ([CS14])**.** *Let $f : \{0,1\}^n \to \{0,1\}$ be a symmetric function and $\overrightarrow{g} = (g_1, \ldots, g_n)$ where $g_i : \mathbb{F}_p^k \to \{0,1\}$ are any functions. If $k > 1 + p\log(3n)$ then:*

$$\mathrm{D}_k(f \circ \overrightarrow{g}) \leq \mathcal{O}\left(p \log n \log(pn)\right)$$

and:

$$\mathrm{R}_k^{||}(f \circ \overrightarrow{g}) \leq \mathcal{O}\left(p \log^2 n\right)$$

Proof. When $k > 1 + p\log(3n)$, it is easy to see by a probabilistic argument that there exists $c = (s_1, \ldots, s_k) \in \mathbb{F}_p^k$ such that each column of M has at least one coordinate in common with c. We then take a prime $q \in [n, 2n]$ and consider the polynomial representation *shifted by* c over $\mathbb{F}_q$ of each of the g_i's:

$$g_i(x) = \sum_{0 \leq i_1, \dots, i_k \leq q-1} g_{iq}(i_1, \dots, i_k) \cdot (x_1 - s_1)^{i_1} \cdots (x_k - s_k)^{i_k}$$

For all $i_1, \dots, i_k \neq 0$, we know that $g_{iq}(i_1, \dots, i_k) \cdot (x_1 - s_1)^{i_1} \cdots (x_k - s_k)^{i_k}$ will evaluate to 0 on column i of M (by definition of c). Thus, we only care of the terms $g_{iq}(i_1, \dots, i_k) \cdot (x_1 - s_1)^{i_1} \cdots (x_k - s_k)^{i_k}$ where at least one of the i_j's is equal to 0. These terms are partitioned between the players such that player j only has terms for which $i_j = 0$.

Each player is able to evaluate her terms (since they do not contain values from her forehead), and send their sum (modulo q) to the referee. Finally, by summing up in $\mathbb{F}_q$ the values she received, the referee obtains the numbers of columns that evaluate to 1, and compute $f \circ \overrightarrow{g}(x_1, \dots, x_k)$.

The only non-simultaneous part of the protocol is the share of the vector c at the beginning. However, the players can agree simultaneously on c if they have access to a random public string. $\square$

Remark 24. The role of column c in the proof above inspired the construction of the corner-free set over $\mathbb{F}_p^n$ in Section 2.2.

This protocol is efficient for $\textsc{Sym} \circ \overrightarrow{\textsc{Any}}_p$ with p up to $\operatorname{polylog} n$ (i.e. blocks of width $\log\log n$). However, since it is not simultaneous it does not prevent any function from breaking the $\log n$ barrier. Next section, we will build the first *simultaneous* protocol for composed functions of block-width greater than one.

3.2 Composed functions of constant block-width

Using ideas from [BGKL04], we describe the first efficient simultaneous protocol for composed functions of constant block-width in $\textsc{Sym} \circ \textsc{Comp}_p$ (we will define later what $\textsc{Comp}$ is). We generalize the protocol of Proposition 21, by showing that the next system of equations admits at most one integral solution:

Theorem 25. *Let p, k and n be positive integers such that $k > 1 + 5^p \log n - p$. Let $(b_{i_1,\dots,i_p})_{0 \leq i_1 + \dots + i_p \leq k-1}$ be integers. Consider the following system of equations:*

$$\begin{cases} (k - (i_1 + \dots + i_p)) y_{i_1,\dots,i_p} + \sum_{j=1}^{p} (i_j + 1) y_{i_1,\dots,i_{j-1},i_j+1,i_{j+1},\dots,i_p} = b_{i_1,\dots,i_p} \\ 0 \leq i_1 + \dots + i_p \leq k-1 \end{cases} \tag{1}$$

Assume further that

$$y_{i_1,\dots,i_p} \geq 0,\ 0 \leq i_1 + \dots + i_p \leq k \quad \text{and} \quad \sum_{i_1 + \dots + i_p \leq k} y_{i_1,\dots,i_p} \leq n \tag{2}$$

Then, under constraints (1), the system of equations (2) has at most one integral solution.

Theorem 25 is implied by the following one:

Theorem 26. *Let p, k and n be positive integers such that $k > 1 + 5^p \log n - p$. Consider the following system of equations:*

$$\begin{cases} (k - (i_1 + \dots + i_p)) d_{i_1,\dots,i_p} + \sum_{j=1}^{p} (i_j + 1) d_{i_1,\dots,i_{j-1},i_j+1,i_{j+1},\dots,i_p} = 0 \\ 0 \leq i_1 + \dots + i_p \leq k-1 \end{cases} \tag{3}$$

Assume further that

$$\sum_{i_1+\cdots+i_p\le k} |d_{i_1,\ldots,i_p}| \le 2n \tag{4}$$

Then, under constraints (4), the system of equations (3) cannot have a non-zero integral solution.

Proof that Theorem 26 implies Theorem 25. We assume by contradiction that Equation (1) under constraints (2) has two different integer solutions $y = (y_{i_1,\ldots,i_p})_{0\le i_1+\cdots+i_p\le k}$ and $y' = (y'_{i_1,\ldots,i_p})_{0\le i_1+\cdots+i_p\le k}$. For $0 \le i_1 + \cdots + i_p \le k$, let $d_{i_1,\ldots,i_p} = y_{i_1,\ldots,i_p} - y'_{i_1,\ldots,i_p}$. Since $y \neq y'$, we know there exists at least one $d_{i_1,\ldots,i_p} \neq 0$.

From (1), we obtain the following relations:

$$\begin{cases} (k-(i_1+\cdots+i_p))d_{i_1,\ldots,i_p} + \sum_{j=1}^{p}(i_j+1)d_{i_1,\ldots,i_{j-1},i_j+1,i_{j+1},\ldots,i_p} = 0 \\ 0 \le i_1+\cdots+i_p \le k-1 \end{cases} \tag{5}$$

Moreover, since $y_{i_1,\ldots,i_p} + y'_{i_1,\ldots,i_p} \ge |y_{i_1,\ldots,i_p} - y'_{i_1,\ldots,i_p}| = d_{i_1,\ldots,i_p}$, we have:

$$2n \ge \sum_{i_1+\cdots+i_p\le k} (y_{i_1,\ldots,i_p} + y'_{i_1,\ldots,i_p}) \ge \sum_{i_1+\cdots+i_p\le k} |d_{i_1,\ldots,i_p}|$$

Thus, we proved that Equations (3) under constraints (4) has a non-zero integral solution. It implies that Equation (1) under constraints (2) cannot have more that one integral solution if Theorem 26 holds. □

Theorem 26 is proved by induction on p. The base case ($p = 1$) has already been established in [BGKL04] (see the proof of Proposition 21). The induction step is detailed in Appendix C. This new result leads to the following simultaneous protocol for composed functions:

Theorem 27. *Let M be a $k \times n$ matrix over $\mathbb{F}_{p+1}$ with $k > 1 + 5^p \log n - p$. For $0 \le i_1 + \cdots + i_p \le k$, let $y_{i_1,\ldots,i_p}$ be the number of columns of M such that each $s \in \{1,\ldots,p\}$ occurs exactly i_s times in M. For $j = 1,\ldots,k$, let player j see all of M except row j. Then there exists a simultaneous multiparty protocol in which each player sends $\mathcal{O}((k+p)^p \log n)$ bits to the referee, after which the referee can calculate $(y_{i_1,\ldots,i_p})_{i_1+\cdots+i_p\le k}$.*

Proof. As in [BGKL04] and the proof of Proposition 21, player j sends for all $i_1+\cdots+i_p \le k-1$ the number $a_j(i_1,\ldots,i_p)$ of columns she sees which contain exactly i_s occurrences of the element $s \in \mathbb{F}_{p+1} \setminus \{0\}$. Then, the referee computes $b_{i_1,\ldots,i_p} = \sum_{j=1}^{k} a_j(i_1,\ldots,i_p)$ (for all $i_1+\cdots+i_p \le k-1$) and considers the associated equations defined in Theorem 25. It is easy to see that the $y_{i_1,\ldots,i_p}$'s must verify these equations. Since they admit exactly one integral solution (according to Theorem 25), the referee can compute it and recover the $y_{i_1,\ldots,i_p}$'s.

Note that the total number of variables $(y_{i_1,\ldots,i_p})_{i_1+\cdots+i_p\le k}$ is $\mathcal{O}((k+p)^p)$, hence the cost of the protocol. □

The total cost of the previous protocol is $\mathcal{O}(k(k+p)^p \log n)$. Thus, it is efficient for any function in $\text{SYM} \circ \text{SYM}_p$ when k is polylog n and p is constant. We now generalize the notion of compressibility introduced in [BGKL04] to handle larger k:

Definition 28. Let $X = \{x_1,\ldots,x_k\}$ be a set of variables over $\mathbb{F}_p$, and $f : \mathbb{F}_p^k \to \{0,1\}$. For any partition $A\dot{\cup}B$ of X, let denote by $C_{A\to B}(f)$ the (one-way) communication complexity of the following two party problem:

- Alice sees A and Bob sees B.
- Alice sends a message to Bob.
- Bob deduces $f(x_1, \ldots, x_k)$.

The function f is said to be $c-$*compressible* (for some constant c) if for any partition $A\dot{\cup}B$ of X, we have:

$$C_{A\to B}(f) = c\log|B|$$

We call $\textsc{Comp}_p$ the set of all compressible symmetric functions over $\mathbb{F}_p$. Our previous protocol applies to $\textsc{Sym} \circ \textsc{Comp}_p$, whenever $k \geq 5^p \log n$ and p is constant:

Theorem 29. *Let $n, p, k \geq 2$ such that p is a constant and $k \geq 5^p \log n$. If $f : \{0,1\}^n \to \{0,1\}$ and $g : \mathbb{F}_p^k \to \{0,1\}$ are symmetric functions, then:*

$$\mathrm{D}_k^{||}(f \circ g) = \mathcal{O}\left((k+1)^p \log n\right)$$

Moreover, if g is $c-$compressible then:

$$\mathrm{D}_k^{||}(f \circ g) = \mathcal{O}\left(\log^{1+c+p} n\right)$$

Proof. The first point directly stems from Theorem 27.

We prove the second one. Let's consider the $k \times n$ matrix M over $\mathbb{F}_p$ representing the input of $f \circ g$. We define $\ell = 5^{p+1} \log n$, so that only the first ℓ players are going to speak. We also let $u_i \in \mathbb{F}_p^k$ be the content of column i, and $v_i \in \mathbb{F}_p^\ell$, $w_i \in \mathbb{F}_p^{k-\ell}$ such that $u_i = v_i \cdot w_i$ (u_i is the vector appearing from row 1 to ℓ, and w_i is the remaining values).

Since g is compressible, $g(u_i)$ is determined by v_i and a message m_i of size at most $c\log\ell$ that only depends on w_i (thus m_i is known by players 1 to ℓ). The set of all possible messages m_i's has size $r = 2^{c\log\ell} = 5^{c(p+1)} \log^c n$.

Players 1 to ℓ now form r new matrices $M_1, \ldots, M_r$ where each M_j groups all the columns u_i's of M that lead to a same message m_i. Once again, this step does not require any communication. We then discard rows $\ell+1$ to k in each M_j, and apply separately the protocol from Theorem 27 to the first ℓ rows of each M_j. Since f and g are symmetric, and the message associated to each M_j is known, the referee can recover $f \circ g(x_1, \ldots, x_k)$. □

The $\textsc{Eval}_{\mathbb{F}_p}$ function is compressible since it suffices for Alice to send $\sum_{x\in A} x \mod p$ to Bob. This is also the case of the MAJ_t function:

Lemma 30. *The MAJ_t function is 2-compressible.*

Proof. The proof goes as in [BGKL04]. Let's consider a partition $A\dot{\cup}B = X$ of the input $X = (x_1, \ldots, x_k) \in \mathbb{F}_{2^t}^k$. We have $\mathrm{MAJ}_t(x_1, \ldots, x_k) = 1$ if and only if $\sum_{i=1}^k |x_i| \geq kt/2$ (where $|x_i|$ is the Hamming weight of the binary representation of x_i over t bits). If $\sum_{x\in A} |x_i| < kt/2 - t|B|$, then Alice already knows $\mathrm{MAJ}_t(x_1, \ldots, x_k) = 0$. On the other hand, if $\sum_{x\in A} |x_i| \geq kt/2$ then she knows $\mathrm{MAJ}_t(x_1, \ldots, x_k) = 1$. Finally, if $\sum_{x\in A} |x_i|$ is between $kt/2 - t|B|$ and $kt/2$ then $\mathrm{MAJ}_t(x_1, \ldots, x_k)$ also depends on what Bob sees. Thus, it is enough for Alice to send one of the $2 + t|B|$ messages that describe the previous situations. This requires $\log(2 + t|B|)$ bits. □

Consequently, we obtain the first efficient simultaneous protocol for $\mathrm{MAJ} \circ \mathrm{MAJ}_t$ when $t > 1$ is constant:

Proposition 31. *For all constant t and $k \geq 5^{2^t} \log n$, we have:*

$$\mathrm{D}_k^{||}(\mathrm{MAJ} \circ \mathrm{MAJ}_t) = \mathcal{O}\left(\log^{3+2^t} n\right)$$

The same reasoning shows that the threshold function Th_s is also compressible (where $Th_s(x_1, \ldots, x_k) = 1$ if and only if $\sum_{i=1}^k x_i \geq s$). This solves open problem 1.(a) formulated in Section 8 of [BGKL04] when the block-width t is constant.

3.3 Future work

We detailed the first efficient *simultaneous* protocol for composed functions of constant block-width in $\textsc{Sym} \circ \textsc{Comp}_p$. Removing the compressibility condition would be an improvement to this result. However, the technique used in [ACFN15] for $\textsc{Sym} \circ \textsc{Comp}_2$ does not generalized to $p > 2$. On the other hand, the protocol from [CS14] works for $\textsc{Sym} \circ \overrightarrow{\textsc{Any}}_p$ and p up to polylog n, but is not simultaneous. It is remarkable that the only known simultaneous protocols for large families of composed functions ([BGKL04], [ACFN15] and Theorem 29) are always derived from the equations introduced in [BGKL04].

The biggest open problem remains to find a function that breaks the $\log n$ barrier. We proved that such a function cannot be in $\textsc{Sym} \circ \textsc{Comp}_p$, but other composed functions are still conjectured to be hard for more than $\log n$ players. This is for instance the case of the $\mathrm{MAJ} \circ \mathrm{MAJ}_{\sqrt{n}}$ function. The $\textsc{Eval}_G$ function described in Section 2 is also believed to break the barrier (for well-chosen G), but the connection with Ramsey theory makes it even harder to prove. Finally, many matrix related problems are also considered to be of great interest. For instance, Raz [Raz00] showed an $\Omega(n/2^k)$ lower bound for deciding the top-left entry of the multiplication of k $n \times n$ matrices over $\mathbb{F}_2$. More recently, Gowers and Viola [GV15] studied the interleaved group products, where each player receives a tuple $(x_{i,1}, \ldots, x_{i,n})$ in $G = SL(2, q)$, with the promise that $\prod_{i=1}^n x_{1,i} \cdots x_{k,i} = g$ *or* h. Finding which is the case has cost $\Omega(n \log |G|)$ when $k = 2$, and it is conjectured to remain hard for larger k.

To conclude, one of the difficulties to break the $\log n$ barrier is the lack of methods to produce lower bounds in the NOF model. The *discrepancy method* is the only one that generalizes from the two player case, but it is hard to use and it applies in fact directly to randomized communication. Finding a lower bound technique that works specifically for deterministic multiparty communication complexity is an open challenge.

4 Decision tree complexity and log-rank conjecture

This last section addresses another major unsolved problem in communication complexity: the log-rank conjecture, and its links to decision tree complexity.

Given a function $f : \{0,1\}^n \to \{0,1\}$ and an unknown input $x \in \{0,1\}^n$, the decision tree model characterizes the amount of information that have to be queried on x in order to compute $f(x)$. It turns out that decision tree complexity is a convenient way to upper bound the communication complexity of XOR and AND functions, and possibly prove the log-rank conjecture for them.

Here, we present different models of decision tree complexity and their relations to communication complexity. We then focus on the decision tree complexities of symmetric functions.

4.1 Definitions and links to communication complexity

A query is a boolean function $q : \{0,1\}^t \to \{0,1\}$ that operates on a particular subset of the n input variables $x = x_1 \dots x_n \in \{0,1\}^n$. The decision tree model depends on the type of queries that are allowed. We first provide a general definition, and then three specific kinds of decision trees.

Definition 32. A *deterministic decision tree* is an ordered binary tree, where each internal node is labeled with a query, and each leaf is labeled with 0 or 1. Given $x \in \{0,1\}^n$, the tree is recursively evaluated by starting at the root and going in the left subtree if the query of the current node evaluates to 0 on x, or the right subtree if it evaluates to 1.

The *deterministic decision tree complexity* of a boolean function $f : \{0,1\}^n \to \{0,1\}$ is the smallest depth of a decision tree that computes $f(x)$ for all x. The (error-bounded) *randomized decision tree complexity* is defined similarly, with the extra possibility of choosing the queries at random. We also refer the reader to [HŠ05] for a description of the (error-bounded) *quantum decision tree complexity*, that will be briefly used later.

We now define three different set of queries and the corresponding decision tree models:

- *Regular query:* returns the value of one of the variables (e.g. x_2). The corresponding model is the *regular decision tree* model (or just *decision tree* model). The deterministic, randomized and quantum decision tree complexities of a function f will be denoted respectively by $\mathrm{DT}(f)$, $\mathrm{RDT}(f)$ and $\mathrm{QDT}(f)$.
- *Parity query:* returns the parity of a subset of the variables (e.g. $x_1 \oplus x_4 \oplus x_7$). The corresponding model is the *parity decision tree* model, and the complexities are denoted by $\mathrm{DT}^{\oplus}(f)$, $\mathrm{RDT}^{\oplus}(f)$ and $\mathrm{QDT}^{\oplus}(f)$.
- *Conjunctive query:* returns the conjunction of a subset of the variables (e.g. $x_2 \wedge x_3$). The corresponding model is the *conjunctive decision tree* model, and the complexities are denoted by $\mathrm{DT}^{\wedge}(f)$, $\mathrm{RDT}^{\wedge}(f)$ and $\mathrm{QDT}^{\wedge}(f)$.

See Figure 4.1 for an example of a decision tree computing the MAJ_3 function.

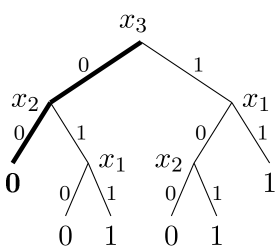

Figure 4: A (regular) deterministic decision tree computing the majority function on 3 bits (each query is next to its node). The computation on input $x = 100$ is shown in bold.

The regular decision tree complexity (also called *query complexity*) is a well-studied subject. For instance, it is known that $\mathrm{DT}(f)$ is polynomially related to $\deg(f)$ ([NS92]) and $\log \mathrm{DT}(f)$ is the time needed to compute f on a CREW PRAM ([Nis89]). In a breakthrough result, Grover [Gro96] also proved that $\mathrm{QDT}(\mathrm{OR}) = \Theta(\sqrt{n})$, whereas $\mathrm{RDT}(\mathrm{OR}) = \Omega(n)$.

Parity and conjunctive decision trees can be much more efficient than regular ones (for instance $\mathrm{DT}(\mathrm{AND}) = n$, whereas $\mathrm{DT}^{\wedge}(\mathrm{AND}) = 1$). They are also intricately related to the communication complexity of XOR and AND functions:

Proposition 33 (Folklore)**.** *For any XOR function $F(x,y) = f(x \oplus y)$ we have:*

$$\mathrm{D}_2(F) \leq 2 \cdot \mathrm{DT}^{\oplus}(f)$$

Similarly, for any AND function $F(x,y) = f(x \wedge y)$:

$$\mathrm{D}_2(F) \leq 2 \cdot \mathrm{DT}^{\wedge}(f)$$

These results also hold in the randomized and quantum frameworks.

Proof. Let's consider a parity decision tree T computing f. Alice and Bob want to compute $F(x,y) = f(x \oplus y)$. They simulate T on input $x \oplus y$: for each parity query $(x_{i_1} \oplus y_{i_1}) \oplus \cdots \oplus (x_{i_t} \oplus y_{i_t})$, Alice sends $x_{i_1} \oplus \cdots \oplus x_{i_t} \in \{0,1\}$ to Bob who computes $(x_{i_1} \oplus \cdots \oplus x_{i_t}) \oplus (y_{i_1} \oplus \cdots \oplus y_{i_t}) = (x_{i_1} \oplus y_{i_1}) \oplus \cdots \oplus (y_{i_t} \oplus y_{i_t}) \in \{0,1\}$ and sends back the result to Alice. The total cost of the protocol is $2 \cdot \mathrm{DT}^{\oplus}(f)$.

The proof is similar for AND functions. □

These relationships between communication and decision tree complexities provide a new framework to prove the log-rank conjecture for XOR and AND functions. Indeed, according to Propositions 8, 9 and 33, it is now enough to show $\mathrm{DT}^{\oplus}(f) \leq \log^c \mathrm{mon}(f)$ and $\mathrm{DT}^{\wedge}(f) \leq \log^c \mathrm{mon}^{\star}(f)$. In practice, this approach has already been used to prove the log-rank conjecture for XOR functions with constant $\deg_2(f)$ over $\mathbb{F}_2$ (see [TWXZ13]).

However, the gap in the inequalities from Proposition 33 could be so important that the log-rank conjecture holds for communication complexity and not for decision trees. In other words, it would be comforting to know whether these complexities are polynomially related or not:

Conjecture 34. *For any XOR function $F(x,y) = f(x \oplus y)$:*

$$\mathrm{D}_2(F) =_{poly} \mathrm{DT}^{\oplus}(f),\ \mathrm{R}_2(F) =_{poly} \mathrm{RDT}^{\oplus}(f) \text{ and } \mathrm{Q}_2(F) =_{poly} \mathrm{QDT}^{\oplus}(f)$$

For any AND function $F(x,y) = f(x \wedge y)$:

$$\mathrm{D}_2(F) =_{poly} \mathrm{DT}^{\wedge}(f),\ \mathrm{R}_2(F) =_{poly} \mathrm{RDT}^{\wedge}(f) \text{ and } \mathrm{Q}_2(F) =_{poly} \mathrm{QDT}^{\wedge}(f)$$

The first case has been closed very recently by proving $\mathrm{D}_2(F) \leq \mathcal{O}\left(\mathrm{DT}^{\oplus}(f)^6\right)$ (see [HL16]). The other ones remain widely open. In next section, we solve them for symmetric functions.

4.2 Decision tree complexities of symmetric functions

Recall that a function $f : \{0,1\}^n \to \{0,1\}$ is symmetric if $f(x)$ only depends on the Hamming weight $|x|$ of x. We often use $f : \{0,\ldots,n\} \to \{0,1\}$ instead, with the understanding that $f(|x|) = f(x)$. Symmetric functions are commonly studied in complexity theory, because of their simplicity and the basic measures associated with them (e.g. $r(f)$, $\ell(f)$ and $t(f)$ defined in Section 1.3). Note that $\mathrm{AND}, \mathrm{OR}, \mathrm{MAJ}$ and MOD_m are all symmetric.

Several properties of the Fourier spectrum of symmetric functions are already known. There exist characterizations of the approximate degree [Pat92], minimal degree [KLM+09], spectral norm [AFH12], etc. The communication complexity of symmetric XOR and AND functions is also well studied ($F(x,y) = f(x \oplus y)$ is said to be symmetric if f is symmetric). It is already known that the log-rank conjecture holds for them, both in the deterministic ([ZS09, BdW01]), randomized ([ZS09, BdW01]) and quantum ([ZS09, Raz03]) frameworks. More precisely, here are the communication complexities established in the previous papers

($h(n) \in \Theta^\star(c(n))$ means $\Omega(c(n)) \le h(n) \le \mathcal{O}(c(n)\cdot\log n)$, and $h(n) \in \Theta^\dagger(c(n))$ means $\Omega(c(n)/\log n) \le h(n) \le \mathcal{O}(c(n))$):

	XOR functions	AND functions
Deterministic	$\Theta(n)$	$\Theta\left((n-t(f))\left(1+\log\frac{n}{n-t(f)}\right)\right)$
Randomized	$\Theta(r(f))$	$\Theta^\dagger\left((n-t(f))\left(1+\log\frac{n}{n-t(f)}\right)\right)$
Quantum	$\Theta(r(f))$	$\Theta^\star\left(\sqrt{n\cdot\ell_0(f)}+\ell_1(f)\right)$

Figure 5: Communication complexities of (nontrivial[1]) symmetric XOR and AND functions.

The regular decision tree complexity of symmetric functions is also known [BBC+01, BdW02]. On the other hand, Ada et al. [AFH12] proved that the smallest size of a parity decision tree computing f is $2^{\Theta(r(f)\log(n/r(f)))}$, and Aspnes et al. [ABD+10] obtained a tight characterization in terms of $\ell(f)$ for k^+ decision trees (a model for which each node has $k+1$ branching options). Moreover, Proposition 33 and Figure 5 already provide lower bounds on the parity and conjunction decision tree complexities of symmetric functions. In the following, we will prove the matching upper bounds and obtain the next results:

	Regular	Parity	Conjunctive
Deterministic	$\Theta(n)$	$\Theta(n)$	$\Theta\left((n-t(f))\left(1+\log\frac{n}{n-t(f)}\right)\right)$
Randomized	$\Theta(n)$	$\Theta(r(f))$	$\Theta^\dagger\left((n-t(f))\left(1+\log\frac{n}{n-t(f)}\right)\right)$
Quantum	$\Theta\left(\sqrt{n\cdot\ell(f)}\right)$	$\Theta(r(f))$	$\Theta^\star\left(\sqrt{n\cdot\ell_0(f)}+\ell_1(f)\right)$

Figure 6: Decision tree complexities of (nontrivial[2]) symmetric functions.

The lower bounds for $\mathrm{DT}(f)$ and $\mathrm{RDT}(f)$ can be found in [BdW02]. Using the quantity $\Gamma(f)=\min\{|2k-n+1| : f(k)\neq f(k+1)\}$, it has already been proved in [BBC+01] that $\mathrm{QDT}(f)=\Theta\left(\sqrt{n\cdot(n-\Gamma(f))}\right)$. It is easy to see that $n-\Gamma(f)$ is in fact $\approx 2\ell(f)$.

We now prove the first missing upper bound:

Theorem 35. *Let $f:\{0,1\}^n\to\{0,1\}$ a symmetric function. We have:*

$$\mathrm{RDT}^\oplus(f)=\mathcal{O}(r(f))$$

Proof. Using results from [Yao03, GKdW04, HSZZ06] (see also [BBG14]), Leung et al. [LLZ11] built a (public coin) randomized communication protocol of cost $\mathcal{O}(r(f))$ for computing any symmetric XOR function. The only information exchanged by the two players on input (x,y) during their protocol are $x\cdot r$ and $y\cdot r$, for $\mathcal{O}(r(f))$ random $r\in\{0,1\}^n$ (recall that $x\cdot r = x_1r_1\oplus\cdots\oplus x_nr_n\in\{0,1\}$). This is equivalent to performing $\mathcal{O}(r(f))$ parity queries on random subsets of x and y.

[1] The trivial XOR functions $F(x,y)=f(x\oplus y)$ are the constant functions and the two parity functions ($f(x)=|x| \mod 2$ or $f(x)=1-|x| \mod 2$). They all have $\mathcal{O}(1)$ complexity.

[2] The trivial functions in the regular model are the two constant functions. The trivial functions in the parity model are the constant functions and the two parity functions. They all have $\mathcal{O}(1)$ complexity.

Thus, for any fixed y, we can simulate the previous protocol on a randomized parity decision tree in order to compute $F(x,y)$. In particular, for $y=0$, we can compute $F(x,0) = f(x)$ for any x. $\square$

Next, we build a conjunctive decision tree protocol for $\mathrm{DT}^{\wedge}(f)$ that matches the previous known lower bound. In fact, our algorithm also applies to non-symmetric functions (the definition of $t(f)$ has to be slightly changed in this case).

Theorem 36. *Let $f : \{0,1\}^n \to \{0,1\}$ a symmetric function. We have:*

$$\mathrm{DT}^{\wedge}(f) = \mathcal{O}\left((n-t(f))\left(1+\log\frac{n}{n-t(f)}\right)\right)$$

with the convention that it is 0 if $t(f)=n$.

Proof. In the following, we use t instead of $t(f)$. If $t \le n/2$, then $(n-t)\left(1+\log\frac{n}{n-t}\right) = \Omega(n)$. Similarly, if $t=n$, then we trivially have $\mathrm{DT}^{\wedge}(f) = \mathcal{O}(1)$. Thus, we will only be interested in $n/2 < t < n$.

For $B \subseteq \{1,\dots,n\}$, we denote by $\wedge^B x$ the conjunctive query performed on the subset of x indexed by B. For instance, $\wedge^{\{1,3\}}1010 = 1$, whereas $\wedge^{\{1,2\}}1010 = 0$.

Input : $x \in \{0,1\}^n$ and $f : \{0,\dots,n\} \to \{0,1\}$ with $n/2 < t(f) < n$
Output: $f(|x|)$
1 Let $T = \{1,\dots,n\}$
2 **while** $|T| \ge n-t$ **do**
3   Take a partition $\dot{\bigcup} B_i$ of T into $2(n-t)$ sets of size $\approx \frac{|T|}{2(n-t)}$
4   Compute $S = \{i : \wedge^{B_i} x = 0\}$
5   **if** $|S| > n-t$ **then**
6   Return $f(0)$
7   **else**
8   Update $T = \bigcup_{i\in S} B_i$
9 Query separately all the x_i's for $i \in T$
10 Define $y \in \{0,1\}^n$ such that $y_i = x_i$ if $i \in T$, and $y_i = 1$ otherwise
11 Return $f(|y|)$

At each step of the algorithm, if $i \notin T$ then $x_i = 1$. Thus, $y = x$ at the end, and the algorithm correctly returns $f(|x|)$ on line 11. On the other hand, if line 6 is reached then it implies $|x| < t$ (since there exists more than $n-t$ disjoint conjunctive queries on which x evaluates to 0). Thus, $f(|x|) = f(0)$ by definition of t. The algorithm is always correct.

Whenever line 8 is reached, the size of T is divided at least by 2. Moreover, the **while** loop stops if $|T| < n-t$. Thus, line 2 is executed $\mathcal{O}\left(\log\frac{n}{n-t}\right)$ times. Finally, each operation from lines 3 to 11 has complexity $\mathcal{O}(n-t)$. Thus, the total complexity of the algorithm is $\mathcal{O}\left((n-t)\left(1+\log\frac{n}{n-t}\right)\right)$. $\square$

Finally, in order to prove the last missing upper bound, we use a reasoning introduced in [Raz03] for quantum communication complexity of AND-functions.

Theorem 37. *Let $f : \{0,1\}^n \to \{0,1\}$ a symmetric function. We have:*

$$\mathrm{QDT}^{\wedge}(f) = \mathcal{O}\left(\sqrt{n\cdot\ell_0(f)} + \ell_1(f)\cdot\log\left(\frac{n}{l_1(f)}\right)\right)$$

Proof. We assume, without loss of generality, that $f = 0$ in $[\ell_0, n - \ell_1]$. We also define $f_0, f_1 : \{0, \dots, n\} \to \{0, 1\}$ such that $f = f_0 \vee f_1$ where $f_0^{-1}(1) \subseteq [0, \ell_0 - 1]$ and $f_1^{-1}(1) \subseteq [n - \ell_1 + 1, n]$. We have $\mathrm{QDT}^\wedge(f) \leq \mathrm{QDT}^\wedge(f_0) + \mathrm{QDT}^\wedge(f_1)$.

We remark that $\ell(f_0) = \ell_0(f)$. Thus, using the upper bound known for $\mathrm{QDT}(f_0)$, we obtain $\mathrm{QDT}^\wedge(f_0) \leq \mathrm{QDT}(f_0) \leq \mathcal{O}\left(\sqrt{n \cdot \ell_0(f)}\right)$. On the other hand, since $t(f_1) = n - \ell_1(f) + 1$, we have $\mathrm{QDT}^\wedge(f_1) \leq \mathrm{DT}^\wedge(f_1) \leq \mathcal{O}\left(\ell_1(f) \cdot \log\left(\frac{n}{l_1(f)}\right)\right)$. $\square$

All these results, summarized in Figure 6, provide a better understanding of the regular, parity and conjunctive decision tree complexities of symmetric functions. In particular, it confirms that Conjecture 34 is true for symmetric functions.

4.3 Future work

The study of symmetric functions in the decision tree model is not completed. For instance, is it possible to extend the characterization from [AFH12] of the parity decision tree size to the conjunctive model?

Regarding the log-rank conjecture for XOR and AND functions, we provide new evidence that communication and decision tree complexities are polynomially related. The proof of Conjecture 34 in the general case probably requires very advanced tools (the result $\mathrm{D}_2(F) =_{poly} \mathrm{DT}^\oplus(f)$ from [HL16] relies on additive combinatoric), but it could be easier to study it first for other restricted families of boolean functions (monotone, bounded-degree, etc.).

5 Conclusion

Three of the main open questions in communication complexity were addressed in this report. We first studied the $\textsc{Eval}_G$ function and its links to Ramsey theory. We proposed the first construction of a large corner-free set over $\mathbb{F}_p^n$. The $\textsc{Eval}_G$ function gathers several of the biggest challenges in communication complexity, but the associated Ramsey number are still poorly understood.

We then described the $\log n$ barrier problem, and proved that it cannot be solved by composed functions in $\textsc{Sym} \circ \textsc{Comp}_p$ for constant p. In particular, our result applies to $\mathrm{MAJ} \circ \mathrm{MAJ}_t$, which is the first time that an efficient simultaneous protocol is found for $t > 1$. Recall that breaking the barrier would help to close a major conjecture about ACC^0, but the current lower bound techniques do not seem to be powerful enough for such a result. On the other hand, we think that other strong upper bounds can be obtained for larger families of composed functions. We particularly seek to remove the compressibility condition in our result.

Finally, we gave a full characterization of the regular, parity and conjunctive decision tree complexities of symmetric functions. These results strengthen the conjecture that communication and decision tree complexities are polynomially related. It also provides a better understanding of decision tree complexities, which could be used to solve the log-rank conjecture for XOR and AND functions. Besides that, the study of symmetric functions is interesting in its own right and we hope to further characterize the related complexity measures.

References

[ABD$^+$10] James Aspnes, Eric Blais, Murat Demirbas, Ryan O'Donnell, Atri Rudra, and Steve Uurtamo. K+ decision trees. In *Proceedings of the 6th International Conference on Algorithms for Sensor Systems, Wireless Adhoc Networks, and Autonomous Mobile Entities*, ALGOSENSORS'10, pages 74–88, Berlin, Heidelberg, 2010. Springer-Verlag.

[ACFN15] Anil Ada, Arkadev Chattopadhyay, Omar Fawzi, and Phuong Nguyen. The NOF multiparty communication complexity of composed functions. *Computational Complexity*, 24(3):645–694, 2015.

[Ada14] Anil Ada. *Communication Complexity*. PhD thesis, McGill University, 2014.

[AFH12] Anil Ada, Omar Fawzi, and Hamed Hatami. Spectral norm of symmetric functions. *CoRR*, abs/1205.5282, 2012.

[AMS96] Noga Alon, Yossi Matias, and Mario Szegedy. The space complexity of approximating the frequency moments. In *Proceedings of the Twenty-eighth Annual ACM Symposium on Theory of Computing*, STOC '96, pages 20–29, New York, NY, USA, 1996. ACM.

[BBC$^+$01] Robert Beals, Harry Buhrman, Richard Cleve, Michele Mosca, and Ronald de Wolf. Quantum lower bounds by polynomials. *J. ACM*, 48(4):778–797, July 2001.

[BBG14] Eric Blais, Joshua Brody, and Badih Ghazi. The information complexity of Hamming distance. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2014, September 4-6, 2014, Barcelona, Spain*, pages 465–489, 2014.

[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.

[BdW01] Harry Buhrman and Ronald de Wolf. Communication complexity lower bounds by polynomials. In *Proceedings of the 16th Annual Conference on Computational Complexity*, CCC '01, pages 120–, Washington, DC, USA, 2001. IEEE Computer Society.

[BdW02] Harry Buhrman and Ronald de Wolf. Complexity measures and decision tree complexity: A survey. *Theor. Comput. Sci.*, 288(1):21–43, October 2002.

[Beh46] F. A Behrend. On sets of integers which contain no three terms in arithmetical progression. In *Proceedings of the National Academy of Sciences of the United States of America*, volume 32.12, pages 331–332, 1946.

[BGKL04] László Babai, Anna Gál, Peter G. Kimmel, and Satyanarayana V. Lokam. Communication complexity of simultaneous messages. *SIAM J. Comput.*, 33(1):137–166, January 2004.

[BKL95] László Babai, Peter G. Kimmel, and Satyanarayana V. Lokam. *Simultaneous messages vs. communication*, pages 361–372. Springer Berlin Heidelberg, Berlin, Heidelberg, 1995.

[BNS92] László Babai, Noam Nisan, and Márió Szegedy. Multiparty protocols, pseudorandom generators for logspace, and time-space trade-offs. *J. Comput. Syst. Sci.*, 45(2):204–232, October 1992.

[Bou99] J. Bourgain. On triples in arithmetic progression. *Geometric & Functional Analysis GAFA*, 9(5):968–984, 1999.

[BPS07] Paul Beame, Toniann Pitassi, and Nathan Segerlind. Lower bounds for Lovász–Schrijver systems and beyond follow from multiparty communication complexity. *SIAM J. Comput.*, 37(3):845–869, 2007.

[BT94] Richard Beigel and Jun Tarui. On ACC. *Computational Complexity*, 4(4):350–366, 1994.

[CFL83] Ashok K. Chandra, Merrick L. Furst, and Richard J. Lipton. Multi-party protocols. In *Proceedings of the Fifteenth Annual ACM Symposium on Theory of Computing*, STOC '83, pages 94–99, New York, NY, USA, 1983. ACM.

[CS14] Arkadev Chattopadhyay and Michael E. Saks. The power of super-logarithmic number of players. In Klaus Jansen, José D. P. Rolim, Nikhil R. Devanur, and Cristopher Moore, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2014)*, volume 28 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 596–603, Dagstuhl, Germany, 2014. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

[CT93] Fan R. K. Chung and Prasad Tetali. Communication complexity and quasi randomness. *SIAMJDiscreteMath*, 6(1):110–123, 1993.

[FK78] H. Furstenberg and Y. Katznelson. An ergodic Szemerédi theorem for commuting transformations. *Journal d'Analyse Mathématique*, 34(1):275–291, 1978.

[GKdW04] Dmitry Gavinsky, Julia Kempe, and Ronald de Wolf. Quantum communication cannot simulate a public coin. *CoRR*, quant-ph/0411051, 2004.

[Gow07] W. T. Gowers. Hypergraph regularity and the multidimensional Szemerédi theorem. *Annals of Mathematics*, 166(3):897–946, 2007.

[Gre05] Ben Green. Finite field models in additive combinatorics. In Bridget S. Webb, editor, *Surveys in Combinatorics 2005*, pages 1–28. Cambridge University Press, 2005. Cambridge Books Online.

[Gro94] Vince Grolmusz. The BNS lower bound for multi-party protocols is nearly optimal. *Information and Computation*, 112:51–54, 1994.

[Gro96] Lov K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-eighth Annual ACM Symposium on Theory of Computing*, STOC '96, pages 212–219, New York, NY, USA, 1996. ACM.

[GV15] Timothy Gowers and Emanuele Viola. The communication complexity of interleaved group products. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing*, STOC '15, pages 351–360, New York, NY, USA, 2015. ACM.

[HG91] Johan Håstad and Mikael Goldmann. On the power of small-depth threshold circuits. *Computational Complexity*, 1(2):113–129, 1991.

[HL16] Kaave Hosseini and Shachar Lovett. Structure of protocols for XOR functions. *Electronic Colloquium on Computational Complexity (ECCC)*, 23:44, 2016.

[HŠ05] P. Høyer and R. Špalek. Lower bounds on quantum query complexity. *EATCS Bulletin*, 87:78–103, October, 2005.

[HSZZ06] Wei Huang, Yaoyun Shi, Shengyu Zhang, and Yufan Zhu. The communication complexity of the Hamming distance problem. *Information Processing Letters*, 2006.

[KLM+09] Mihail N. Kolountzakis, Richard J. Lipton, Evangelos Markakis, Aranyak Mehta, and Nisheeth K. Vishnoi. On the Fourier spectrum of symmetric boolean functions. *Combinatorica*, 29(3):363–387, 2009.

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

[LLZ11] Ming Lam Leung, Yang Li, and Shengyu Zhang. Tight bounds on the randomized communication complexity of symmetric XOR functions in one-way and SMP models. *CoRR*, abs/1101.4555, 2011.

[LM07] Michael T. Lacey and William McClain. On an argument of Shkredov on two-dimensional corners. *Online Journal of Analytic Combinatorics*, 2007.

[Lov14] Shachar Lovett. Recent advances on the log-rank conjecture in communication complexity. *CoRR*, abs/1403.8106, 2014.

[LS88] L. Lovasz and M. Saks. Lattices, Mobius functions and communications complexity. In *Proceedings of the 29th Annual Symposium on Foundations of Computer Science*, SFCS '88, pages 81–90, Washington, DC, USA, 1988. IEEE Computer Society.

[LS09] Troy Lee and Adi Shraibman. Lower bounds in communication complexity. *Foundations and Trends in Theoretical Computer Science*, 3(4):263–399, 2009.

[MS82] Kurt Mehlhorn and Erik M. Schmidt. Las Vegas is better than determinism in VLSI and distributed computing. In *Proceedings of the Fourteenth Annual ACM Symposium on Theory of Computing*, STOC '82, pages 330–337, New York, NY, USA, 1982. ACM.

[Nis89] N. Nisan. CREW PRAMS and decision trees. In *Proceedings of the Twenty-first Annual ACM Symposium on Theory of Computing*, STOC '89, pages 327–335, New York, NY, USA, 1989. ACM.

[NS92] Noam Nisan and Mario Szegedy. On the degree of boolean functions as real polynomials. In *Proceedings of the Twenty-fourth Annual ACM Symposium on Theory of Computing*, STOC '92, pages 462–467, New York, NY, USA, 1992. ACM.

[O'D14] Ryan O'Donnell. *Analysis of Boolean Functions*. Cambridge University Press, New York, NY, USA, 2014.

[Pat92] Ramamohan Paturi. On the degree of polynomials that approximate symmetric boolean functions (preliminary version). In *Proceedings of the Twenty-fourth Annual ACM Symposium on Theory of Computing*, STOC '92, pages 468–474, New York, NY, USA, 1992. ACM.

[Raz00] Ran Raz. The BNS-Chung criterion for multi-party communication complexity. *Computational Complexity*, 9:2000, 2000.

[Raz03] A A Razborov. Quantum communication complexity of symmetric predicates. *Izvestiya: Mathematics*, 67(1):145, 2003.

[RY15] Anup Rao and Amir Yehudayoff. Simplified lower bounds on the multiparty communication complexity of disjointness. In *30th Conference on Computational Complexity, CCC 2015, June 17-19, 2015, Portland, Oregon, USA*, pages 88–101, 2015.

[Smo87] R. Smolensky. Algebraic methods in the theory of lower bounds for boolean circuit complexity. In *Proceedings of the Nineteenth Annual ACM Symposium on Theory of Computing*, STOC '87, pages 77–82, New York, NY, USA, 1987. ACM.

[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 2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, FOCS '13, pages 658–667, Washington, DC, USA, 2013. IEEE Computer Society.

[Yao79] Andrew Chi-Chih Yao. Some complexity questions related to distributive computing. In *Proceedings of the Eleventh Annual ACM Symposium on Theory of Computing*, STOC '79, pages 209–213, New York, NY, USA, 1979. ACM.

[Yao90] Andrew Chi-Chih Yao. On ACC and threshold circuits. In *31st Annual Symposium on Foundations of Computer Science, St. Louis, Missouri, USA, October 22-24, 1990, Volume II*, pages 619–627, 1990.

[Yao03] Andrew Chi-Chih Yao. On the power of quantum fingerprinting. In *Proceedings of the Thirty-fifth Annual ACM Symposium on Theory of Computing*, STOC '03, pages 77–81, New York, NY, USA, 2003. ACM.

[ZS09] Zhiqiang Zhang and Yaoyun Shi. Communication complexities of symmetric XOR functions. *Quantum Info. Comput.*, 9(3):255–263, March 2009.

Appendices

A Proof of Proposition 10

We first prove the upper bound $\mathrm{D}^{||}_{k+1}(\textsc{Eval}_G) \leq k \cdot \log(c^{\angle}_k(G))$. Let's consider a valid coloring of G^k with $c^{\angle}_k(G)$ colors. We build a protocol for $\textsc{Eval}_G$ on input $(x_1, \dots, x_{k+1})$ as follow:

- Player $k+1$ sends the color of $(x_1, \dots, x_k)$ to the referee.
- For all $1 \leq i \leq k$, player i computes $x'_i = -\sum\limits_{j \neq i} x_j$ and sends the color of $(x_1, \dots, x_{i'}, \dots, x_k)$ to the referee.

- The referee outputs 1 (i.e. $\sum x_i = 0$) if and only if all the colors she received are the same.

If $\sum x_i = 0$ then $x'_i = x_i$ for all i, and all the colors are indeed the same. On the other hand, if $\sum x_i = -\lambda \neq 0$ then:

- Player 1 sent the color of $(x_1 + \lambda, x_2, \dots, x_k)$.
- Player 2 sent the color of $(x_1, x_2 + \lambda, x_3, \dots, x_k)$.
- $\dots$
- Player k sent the color of $(x_1, \dots, x_{k-1}, x_k + \lambda)$.
- Player $k+1$ sent the color of $(x_1, \dots, x_k)$.

In other words, the players sent the colors of a corner into G^k. Since the coloring is valid, the corner is not monochromatic and at least two colors are different. The referee will correctly output 0.

Note that only one player needs to send her color to the other ones if we do not care of simultaneity. Thus $\mathrm{D}_{k+1}(\textsc{Eval}_G) \leq k + \log(c_k^{\angle}(G))$.

We now prove the lower bound $\log(c_k^{\angle}(G)) \leq \mathrm{D}_{k+1}(\textsc{Eval}_G)$. To this end, we make use of two of the most basic objects in the NOF model: stars and cylinder intersections (see [KN97] for a reminder of what they are). Let's consider an optimal protocol for $\textsc{Eval}_G$ of cost $c = \mathrm{D}_{k+1}(\textsc{Eval}_G)$. It partitions G^{k+1} into at most 2^c cylinder intersections. Recall that the protocol has the same value on each of these cylinder intersections. We then color each $(x_1, \dots, x_k) \in G^k$ by the label of the cylinder intersection that contains $(x_1, \dots, x_k, -\sum_{i=1}^{k} x_i)$. We want to show that this coloring is valid. Let's assume that it is not the case, and consider a monochromatic corner:

$$(x_1, x_2, \dots, x_k), (x_1 + \lambda, x_2, \dots, x_k), (x_1, x_2 + \lambda, \dots, x_k), \dots, (x_1, x_2, \dots, x_k + \lambda)$$

It implies that the following values belong to the same cylinder intersection:

$$\begin{array}{c} (x_1, x_2, \dots, x_k, -\sum x_i - \lambda + \lambda) \\ (x_1 + \lambda, x_2, \dots, x_k, -\sum x_i - \lambda) \\ \vdots \\ (x_1, x_2, \dots, x_k + \lambda, -\sum x_i - \lambda) \end{array}$$

Note that they all sum to 0, thus the protocol outputs 1 on the cylinder intersection they belong to. Moreover, they form a star whose center is $(x_1, x_2, \dots, x_k, -\sum x_i - \lambda)$. Since the center must be in the same cylinder intersection, it implies that the protocol outputs 1 on input $(x_1, x_2, \dots, x_k, -\sum x_i - \lambda)$, which is false since the sum is not 0.

Thus, the previous coloring is valid and has size at most $2^c = 2^{\mathrm{D}_{k+1}(\textsc{Eval}_G)}$.

B Proof of Theorem 17

We want to estimate the size of the set $S_c^k = \{M \in (\mathbb{F}_p^n)^k : \forall i \in \{0, \dots, k\}, n_{i,c}(M) = N_i\}$ when

$$k \geq \left\lceil \frac{\log n}{\log\left(1 + \frac{1}{p-1}\right)} \right\rceil$$

and:

$$\begin{cases} N_i = \left\lfloor \binom{k}{i} \frac{(p-1)^i}{p^k} n \right\rfloor, \ 1 \le i \le k-1 \\ N_0 = n - \sum\limits_{i=1}^{k-1} N_i \\ N_k = 0 \end{cases}$$

For all $0 \le i \le k-1$, we denote by α_i the real number such that $N_i = \alpha_i \frac{(p-1)^i}{p^k} n$. We obtain the following inequalities:

Lemma 38. *If* $k \ge \left\lceil \frac{\log n}{\log\left(1+\frac{1}{p-1}\right)} \right\rceil$, *then:*

1. $N_0 \le 1 + k$
2. $\alpha_0^{N_0} \le e^{k+k^2} p^{k+k^2}$
3. $N_0 \cdots N_{k-1} \le (1+k)2^{k^2}$

Proof. Let's denote $\lambda = \frac{\log n}{\log\left(1+\frac{1}{p-1}\right)}$ (such that $\left(\frac{p}{p-1}\right)^\lambda = n$). We have $\lambda \le k$ and $\left(\frac{p-1}{p}\right)^k \le \frac{1}{n}$. We now prove the three points of the lemma:

Point 1: Since $N_i \ge \binom{k}{i} \frac{(p-1)^i}{p^k} n - 1$ for all $1 \le i \le k-1$, we have:

$$\begin{aligned} N_0 &= n - \sum_{i=1}^{k-1} N_i \\ &\le n + (k-1) + \frac{n}{p^k} + \frac{(p-1)^k n}{p^k} - \sum_{i=0}^{k} \binom{k}{i} \frac{(p-1)^i}{p^k} n \\ &\le k + \left(\frac{p-1}{p}\right)^k n \qquad \text{since } n/p^k \le 1 \\ &\le 1 + k \end{aligned}$$

Point 2: Recall that $N_0 = \alpha_0 \frac{(p-1)^0}{p^k} n$. Thus:

$$\begin{aligned} \alpha_0^{N_0} &\le \left(N_0 \cdot p^k\right)^{N_0} \\ &\le (1+k)^{1+k} p^{k+k^2} \qquad \text{using Point 1 above} \\ &\le e^{k+k^2} p^{k+k^2} \end{aligned}$$

Point 3:

$$
\begin{aligned}
N_0 \cdots N_{k-1} &\leq (1+k) \prod_{i=1}^{k} \binom{k}{i} \frac{(p-1)^i}{p^k} n \\
&\leq \frac{(1+k)2^{k^2}(p-1)^{\frac{k(k+1)}{2}} n^k}{p^{k^2}} \\
&\leq \frac{(1+k)2^{k^2}(p-1)^{k^2} n^k}{p^{k^2}} \\
&\leq (1+k)2^{k^2} n^k \left(\left(\frac{p-1}{p} \right)^k \right)^k \\
&\leq (1+k)2^{k^2} n^k \left(\frac{1}{n} \right)^k \\
&\leq (1+k)2^{k^2}
\end{aligned}
$$

□

Using the inequalities established in the previous Lemma and the Stirling's formula, we estimate the size of S_c^k as follow:

$$
\begin{aligned}
|S_c^k| &= \binom{n}{N_0 \quad N_1 \quad \cdots \quad N_{k-1}} \cdot \left((p-1)^0 \binom{k}{0} \right)^{N_0} \cdots \left((p-1)^{k-1} \binom{k}{k-1} \right)^{N_{k-1}} \\
&= \frac{n!}{N_0! \cdots N_{k-1}!} \cdot \left((p-1)^0 \binom{k}{0} \right)^{N_0} \cdots \left((p-1)^{k-1} \binom{k}{k-1} \right)^{N_{k-1}} \\
&\geq \frac{(ne^{-1})^n \sqrt{2\pi n} \cdot \left((p-1)^0 \binom{k}{0} \right)^{N_0} \cdots \left((p-1)^{k-1} \binom{k}{k-1} \right)^{N_{k-1}}}{e^k (N_0 e^{-1})^{N_0} \cdots (N_{k-1} e^{-1})^{N_{k-1}} \sqrt{(2\pi)^k N_0 \cdots N_{k-1}}} \\
&\geq \frac{(ne^{-1})^n \sqrt{2\pi n} \cdot \left((p-1)^0 \binom{k}{0} \right)^{N_0} \cdots \left((p-1)^{k-1} \binom{k}{k-1} \right)^{N_{k-1}}}{e^k \left(\alpha_0 \frac{(p-1)^0}{p^k} n e^{-1} \right)^{N_0} \cdots \left(\alpha_{k-1} \frac{p^{k-1}}{p^k} n e^{-1} \right)^{N_{k-1}} \sqrt{(2\pi)^k N_0 \cdots N_{k-1}}} \\
&\geq \frac{(ne^{-1})^n \sqrt{2\pi n} \cdot \binom{k}{0}^{N_0} \cdots \binom{k}{k-1}^{N_{k-1}}}{e^k (ne^{-1})^{\sum_{i=0}^{k-1} N_i} p^{-k \sum_{i=0}^{k-1} N_i} \alpha_0^{N_0} \cdots \alpha_{k-1}^{N_{k-1}} \sqrt{(2\pi)^k N_0 \cdots N_{k-1}}} \\
&\geq \frac{p^{nk} \sqrt{2\pi n} \cdot \binom{k}{0}^{N_0} \cdots \binom{k}{k-1}^{N_{k-1}}}{e^k \alpha_0^{N_0} \cdots \alpha_{k-1}^{N_{k-1}} \sqrt{(2\pi)^k N_0 \cdots N_{k-1}}} \\
&\geq \frac{p^{nk} \sqrt{2\pi n}}{e^k \alpha_0^{N_0} \sqrt{(2\pi)^k N_0 \cdots N_{k-1}}} \qquad \text{since } \alpha_i \leq \binom{k}{i} \text{ when } i > 0 \\
&\geq \frac{p^{nk} \sqrt{2\pi n}}{e^{2k+k^2} p^{k+k^2} \sqrt{(2\pi)^k (1+k) 2^{k^2}}} \qquad \text{according to Lemma 38} \\
&\geq \frac{p^{nk}}{C^{k^2} p^{k+k^2}}
\end{aligned}
$$

where C is a constant such that $C^{k^2} > e^{2k+k^2} \sqrt{(2\pi)^k (1+k) 2^{k^2}}$.

C Proof of Theorem 26

We prove that Theorem 26 holds for p, assuming that it is true for $p-1$.

Suppose by contradiction that Equations (3) under constraints (4) has a non-zero integral solution $d = (d_{i_1,\dots,i_p})_{0 \le i_1+\dots+i_p \le k}$. We define:

$$u = \max\{t \le k : \forall i_1 + \dots + i_p \le t, d_{i_1,\dots,i_p} = 0\}$$

(if the maximum does not exist, i.e. $d_{0,\dots,0} \neq 0$, then we take $u = 0$). Since at least one $d_{i_1,\dots,i_p}$ is non-zero, we must have $u < k$. In fact, we obtain the following stronger bound:

Lemma 39. *We have:*

$$u + 1 \le 1 + 5^{p-1} \log n - (p-1)$$

Proof. We assume that $u > 0$ (otherwise the result is trivial). According to Equations 5, for all $i_1 + \dots + i_p = u$:

$$(k-u)d_{i_1,\dots,i_p} + \sum_{j=1}^{p} (i_j + 1) d_{i_1,\dots,i_{j-1},i_j+1,i_{j+1},\dots,i_p} = 0$$

Since $d_{i_1,\dots,i_p} = 0$ whenever $i_1 + \dots + i_p \le u$, it can be rewritten as:

$$((u+1) - (i_1 + \dots + i_{p-1}))d_{i_1,\dots,i_{p-1},i_p+1} + \sum_{j=1}^{p-1} (i_j + 1) d_{i_1,\dots,i_{j-1},i_j+1,i_{j+1},\dots,i_p} = 0$$

We now define $d' = (d'_{i_1,\dots,i_{p-1}})_{0 \le i_1+\dots+i_{p-1} \le u+1}$ such that $d'_{i_1,\dots,i_{p-1}} = d_{i_1,\dots,i_{p-1},u+1-(i_1+\dots+i_{p-1})}$. The previous equations imply:

$$\begin{cases} ((u+1) - (i_1 + \dots + i_{p-1}))d'_{i_1,\dots,i_{p-1}} + \sum\limits_{j=1}^{p-1} (i_j + 1) d'_{i_1,\dots,i_{j-1},i_j+1,i_{j+1},\dots,i_{p-1}} = 0 \\ 0 \le i_1 + \dots + i_{p-1} \le u \end{cases}$$

We also have $\sum\limits_{i_1+\dots+i_{p-1} \le u+1} |d'_{i_1,\dots,i_{p-1}}| \le 2n$. However, there exists $i_1 + \dots + i_p = u+1$ such that $d_{i_1,\dots,i_p} \neq 0$ (by definition of u), i.e. $d'_{i_1,\dots,i_{p-1}} \neq 0$. Consequently, applying our induction hypothesis to d' (at rank $p-1$), we must have $u + 1 \le 1 + 5^{p-1} \log n - (p-1)$ (otherwise d' would contradict Theorem 26 at rank $p-1$). ☐

Next, for all $u + 1 \le t \le k$, we define:

$$m_t = \max_{i_1+\dots+i_p=t} |d_{i_1,\dots,i_p}|$$

By definition of u (and the fact that $u < k$), we must have $m_{u+1} \ge 1$. We obtain the following lower bounds on the m_t's:

Lemma 40. *For all $t \ge u+1$, we have:*

$$m_t \ge \binom{k+p-1}{t+p-1} \binom{k+p-1}{u+p}^{-1}$$

Proof. For all $i_1 + \cdots + i_p \leq k$, we have:

$$0 = (k - (i_1 + \cdots + i_p))d_{i_1,\ldots,i_p} + \sum_{j=1}^{p}(i_j + 1)d_{i_1,\ldots,i_{j-1},i_j+1,i_{j+1},\ldots,i_p}$$

So:

$$(k - (i_1 + \cdots + i_p))|d_{i_1,\ldots,i_p}| \leq \sum_{j=1}^{p}(i_j + 1)|d_{i_1,\ldots,i_{j-1},i_j+1,i_{j+1},\ldots,i_p}|$$

In particular, for all $i_1 + \cdots + i_p \leq k$ such that $m_{i_1+\cdots+i_p} = |d_{i_1,\ldots,i_p}|$, we obtain:

$$\begin{aligned}(k - (i_1 + \cdots + i_p))m_{i_1+\cdots+i_p} &\leq \sum_{j=1}^{p}(i_j + 1)|d_{i_1,\ldots,i_{j-1},i_j+1,i_{j+1},\ldots,i_p}| \\ &\leq \sum_{j=1}^{p}(i_j + 1)m_{i_1+\cdots+i_p+1} \\ &\leq (i_1 + \cdots + i_p + p)m_{i_1+\cdots+i_p+1}\end{aligned}$$

Thus, for all $u + 1 \leq t < k$:

$$\frac{k-t}{t+p}m_t \leq m_{t+1}$$

Finally, it is easy to see that it implies $m_t \geq \binom{k+p-1}{t+p-1}\binom{k+p-1}{u+p-1}^{-1}m_{u+1} \geq \binom{k+p-1}{t+p-1}\binom{k+p-1}{u+p-1}^{-1}$. ☐

The last step is to sum up over all the m_t's, for $t \geq u + 1$:

$$\begin{aligned}\sum_{t=u+1}^{k} m_t &\geq \binom{k+p-1}{u+p-1}^{-1} \sum_{t=u+1}^{k}\binom{k+p-1}{t+p-1} \\ &\geq \binom{k+p-1}{u+p-1}^{-1}\left(\sum_{t=0}^{k+p-1}\binom{k+p-1}{t} - \sum_{t=0}^{u+p-1}\binom{k+p-1}{t}\right)\end{aligned}$$

Since $u + p - 1 \leq 5^{p-1}\log n \leq (k + p - 1)/2$ (according to Lemma 39), we have $\sum_{t=0}^{u+p-1}\binom{k+p-1}{t} \leq (u + p)\binom{k+p-1}{u+p-1}$ and $\binom{k+p-1}{u+p-1}^{-1} \geq \binom{k+p-1}{5^{p-1}\log n}^{-1}$. Thus:

$$\begin{aligned}\sum_{t=u+1}^{k} m_t &\geq \binom{k+p-1}{u+p-1}^{-1}2^{k+p-1} - (u+p)\binom{k+p-1}{u+p-1}^{-1}\binom{k+p-1}{u+p-1} \\ &\geq \binom{k+p-1}{5^{p-1}\log n}^{-1}2^{k+p-1} - (u+p)\end{aligned}$$

Moreover, since $k > 1+5^p\log n-p$, we can define $k' \geq 1$ such that $k'5^p\log n \leq k+p-1 < (k'+1)5^p\log n$. Using the well-known bound $\binom{n}{m} \leq (ne/m)^m$, we obtain:

$$\begin{aligned}\sum_{t=u+1}^{k} m_t &\geq \binom{(k'+1)5^p\log n}{5^{p-1}\log n}^{-1}2^{k+p-1} - (u+p) \\ &\geq \left(\frac{1}{5e(k'+1)}\right)^{5^{p-1}\log n}2^{k'5^p\log n} - (u+p) \\ &\geq n^{5^{p-1}(5k'-\log(5e(k'+1)))} - (u+p) \\ &\geq n^{5^{p-1}(4k'-\log(5e))} - (u+p)\end{aligned}$$

Finally, $k' \geq 1$, $\log(5e) \approx 3.8$ and $u + p \leq 5^{p-1} \log n$. Thus, we have $\sum_{t=u+1}^{k} m_t > 2n$ (for n large enough). However, $\sum_{t=u+1}^{k} m_t \leq \sum_{i_1+\cdots+i_p \leq k} |d_{i_1,\ldots,i_p}| \leq 2n$. This is a contradiction.

D Comments on the internship

This report was produced during my internship at Carnegie Mellon University in Pittsburgh (United States of America), which took place from February 1 to June 17, 2016 under the supervision of Anil ADA. It was carried out during the last year of a Master degree in Theoretical Computer Science from the Ecole Normale Supérieure de Lyon (France).

The first part of the internship was dedicated to familiarize myself with communication complexity and the recent literature on the subject. We then tried to improve the existing protocols for composed functions. We did not succeed to make the result from [CS14] simultaneous, but we extended the construction of [BGKL04] to $\textsc{Sym} \circ \textsc{Comp}_p$ functions. Many tools from Fourier analysis were tried to remove the compressibility condition in the latter result, but they eventually failed. We then turned our attention to other functions likely to break the $\log n$ barrier ($\textsc{Eval}_G$, interleaved group products, $\mathrm{MAJ} \circ \mathrm{MAJ}_{\sqrt{n}}$) and we established the corner-free set construction over $\mathbb{F}_p^n$. During the last part of the internship, we studied the log-rank conjecture in the context of XOR and AND functions. In particular, we discovered the recent paper [HL16] that links communication and parity decision tree complexities, and we proved a similar full characterization for symmetric functions.

Finally, I would thank Anil for having offered me the opportunity to do this internship. I really appreciated the advice he gave me throughout my stay in Pittsburgh, and the knowledge he shared with me. Carnegie Mellon University was also a great place to work, and I was very pleased to attend some of the numerous and diverse seminars ran by the computer science department.