INFORMATION THEORY AND THE RANDOMIZED COMMUNICATION COMPLEXITY OF FUNCTIONS

Nikos Leonardos

Supervisor: Stathis Zachos

$\mu\prod\lambda\forall$

July 7, 2010

Στους γονείς μου,
Πάνο και Ελένη.

To my parents,
Panos and Eleni.

Abstract

Communication complexity is now an established research area of Theoretical Computer Science. It has many applications and it has proved to be a versatile tool in proving lower bounds for many different models of computation. In this dissertation we focus on information-theoretic methods for proving lower bounds in the randomized communication complexity of functions. In particular, we show how these methods can be used to prove lower bounds on the randomized two-party communication complexity of functions that arise from read-once boolean formulae.

A read-once boolean formula is a formula in propositional logic with the property that every variable appears exactly once. Such a formula can be represented by a tree, where the leaves correspond to variables, and the internal nodes are labeled by binary connectives. Under certain assumptions, this representation is unique. Thus, one can define the depth of a formula as the depth of the tree that represents it.

The complexity of the evaluation of general read-once formulae has attracted interest mainly in the decision tree model. In the communication complexity model many interesting results deal with specific read-once formulae, such as DISJOINTNESS and TRIBES. In this paper we use information theory methods to prove lower bounds that hold for any read-once formula. Our lower bounds are of the form $n(f)/c^{d(f)}$, where $n(f)$ is the number of variables and $d(f)$ is the depth of the formula, and they are optimal up to the constant in the base of the denominator.

Acknowledgments

In this part of the dissertation I have the opportunity to thank those people that defined my course as a graduate student.

First, I would like to thank my adviser Stathis Zachos. He was the person that guided me through my first steps in Theoretical Computer Science. I recall that the first book he advised me to read was Enderton's book on logic. It was an interesting read, but only later I realized how rightful an advise this was. Stathis is also memorable for his friendly attitude towards students. He runs the lab named CoReLab; a nice environment for undergraduate and graduate students, of which I still consider myself a member.

A notable member of CoReLab is Aris Pagourtzis, now a lecturer, whom I would like to thank for his guidance during my years at CoReLab. I remember Aris for his breadth of knowledge and his patience in explaining technical things.

I thank also Dimitris Fotakis for being a member of my dissertation committee and showing interest in my work.

Prof. Yannis Moschovakis instilled in me the rigorous reasoning needed in mathematical proofs, through his courses Recursion and Set Theory. With his course Arithmetical Complexity, he shared with students his work on proving lower bounds for the GCD function and other problems.

Prof. Constantinos Dimitracopoulos completed my knowledge of mathematical logic, which, together with set theory, constitute the foundation on which most of my background in mathematics rests. I also need to thank him for running MPLA for most of the time I was part of it.

I also thank Prof. Elias Koutsoupias that taught me Algorithms. He is memorable for his incredible way of passing knowledge, accomplishing to reveal the intuition behind everything he taught.

Finally, I would like to thank my adviser in Rutgers University, Michael Saks. Mike has been great in teaching me how to do research. He provided me with the guidance I needed to produce my first result, and showed me how to follow my intuition when trying to come up with a proof. On the same time he makes sure I have enough freedom to mature as a PhD student. The results presented in this dissertation have been accomplished with his collaboration.

Besides all these great teachers that I was lucky to have, I would like to thank all the friends that I made in CoReLab and MPLA. Although I'll probably forget to mention someone, I'll try to list my closest ones. Panagiotis Cheilaris, Petros Potikas, Katerina Potika, Andreas Göbel, Kostas Bimpikis, Christina Apergi, Costis Georgiou, Aggelina Vidali, Vasilis Paschalis, Giorgos Poulios.

Contents

Chapter 1

Introduction

A landmark result in the theory of two-party communication complexity is the linear lower bound on the randomized communication complexity of set-disjointness proved by [12]. [16] gave a simplified proof, and [3] gave an elegant information theory proof, building on the informational complexity framework of [5]. The first application of information-theoretic methods in communication complexity lower bounds can be traced to [1].

Let us define a *two-party boolean function* to be a boolean function f together with a partition of its variables into two parts. We usually refer to the variables in the two classes as x and y and write $f(x, y)$ for the function. A two-party function is associated with the following communication problem: Given that Alice gets x and Bob gets y, compute $f(x, y)$.

If f is any n-variate boolean function and g is a 2-variate boolean function, we define f^g to be the two-party function taking two n bit strings x and y and defined to be $f^g(x, y) = f(g(x_1, y_1), \ldots, g(x_n, y_n))$. The disjointness communication problem can be reformulated as a boolean function computation problem: Alice gets $x \in \{0,1\}^n$, Bob gets $y \in \{0,1\}^n$ and they want to compute $(\mathrm{OR}_n)^\wedge(x, y)$, where OR_n is the n-wise OR function.

[11], extended the techniques for disjointness in order to prove a linear lower bound for the randomized complexity on the function $(\mathrm{TRIBES}_{s,t})^\wedge$ where $\mathrm{TRIBES}_{s,t}$ is the function taking input $(z_{i,j} : 1 \leq i \leq s, 1 \leq j \leq t)$ and equal to $\mathrm{TRIBES}_{s,t}(z) = \bigwedge_{i=1}^{s} \bigvee_{j=1}^{t} z_{i,j}$.

The functions OR_n and $\mathrm{TRIBES}_{s,t}$ are both examples of *read-once boolean functions*. These are functions that can be represented by boolean formulae involving $\vee$ and $\wedge$, in which each variable appears (possibly negated) at most once. Such a formula can be represented by a rooted ordered tree, with nodes labeled by $\vee$ and $\wedge$, and the leaves labeled by variables. It is well known (see e.g. [9]) that for any read-once function f, f has a unique representation (which we call the *canonical representation* of f) as a tree in which the labels of nodes on each root-to-leaf path alternate between $\wedge$ and $\vee$. The depth of f, $d(f)$, is defined to be the maximum depth of a leaf in the canonical representation, and

$n(f)$ is the number of variables.

We want to consider communication problems derived from arbitrary read-once formulae. Based on the examples of OR_n and $\mathrm{TRIBES}_{s,t}$ mentioned above it seems natural to consider the function $f^\wedge$, but in the case that f is the n-wise AND, $f^\wedge$ trivializes (and can be computed with a two-bit protocol), and the more interesting function to consider is $f^\vee$.

Denote by $R_\delta(f)$ the δ-error randomized communication complexity of f (see the paragraph on "communication complexity" in 1.1 for more details). We prove that for any read-once function f, at least one of the functions $f^\vee$ and $f^\wedge$ has high δ-error communication complexity.

Theorem 1. *For any read-once function f with $d(f) \geq 1$,*

$$\max\{R_\delta(f^\wedge), R_\delta(f^\vee)\} \geq (1 - 2\sqrt{\delta}) \cdot \frac{n(f)}{8^{d(f)}}.$$

This result is, in some sense, best possible (up to the constant 8 in the base of $d(f)$). That is, there is a constant $c > 1$, such that if f is given by a t-uniform tree of depth d (in which each non-leaf node has t children and all leaves are at the same depth, and so $n = t^d$), then $f^\wedge$ and $f^\vee$ both have randomized communication protocols using $O(n(f)/c^{d(f)})$ bits. This follows from the fact (see [18]) that f has a randomized decision tree algorithm using an expected number $O(n(f)/c^{d(f)})$ of queries, and any decision tree algorithm for f is easily converted to a communication protocol for $f^\vee$ or $f^\wedge$ having comparable complexity. In fact, for t-uniform trees, we can improve the lower bound.

Theorem 2. *For any read-once function f that can be represented by a t-uniform* AND/OR *tree of depth $d \geq 1$,*

$$\max\{R_\delta(f^\wedge), R_\delta(f^\vee)\} \geq (1 - 2\sqrt{\delta}) \cdot \frac{t(t-1)^{d-1}}{4^d}.$$

Independently, [10], also using the informational complexity approach, obtained the weaker bound $(1 - 2\sqrt{\delta}) \cdot n(f)/\left(d(f)!16^{d(f)}\right)$.

As a simple corollary of 1 we obtain a similar lower bound for the more general class of *read-once threshold functions*. Recall that a *t-out-of-k threshold gate* is the boolean function with k inputs that is one if the sum of the inputs is at least t. A threshold tree is a rooted tree whose internal nodes are labeled by threshold gates and whose leaves are labeled by distinct variables (or their negations). A read-once threshold function is a function representable by a threshold tree. We prove the following bound.

Theorem 3. *For any read-once threshold function f with $d(f) \geq 1$,*

$$\max\{R_\delta(f^\wedge), R_\delta(f^\vee)\} \geq (1 - 2\sqrt{\delta}) \cdot \frac{n(f)}{16^{d(f)}}.$$

This result should be compared with the result of [9] that every read-once threshold function f has randomized decision tree complexity at least $n(f)/2^{d(f)}$. A lower bound on communication complexity of $f^\vee$ or $f^\wedge$ gives the same lower bound on decision tree complexity for f, however, the implication goes only one way, since communication protocols for $f^\vee$ and $f^\wedge$ do not have to come from a decision tree algorithm for f, and can be much faster. (For example, $(\mathrm{AND}_n)^n$ is equal to AND_{2n} that has randomized decision tree complexity $\Theta(n)$ but communication complexity 2.) Thus, up to the constant in the base of the denominator, our result can be viewed as a strengthening of the decision tree lower bound.

Our results are interesting only for formulae of small depth. For example, for f that is represented by a binary uniform tree $n(f)/8^{d(f)} < 1$, while there is a simple $\sqrt{n(f)}$ lower bound that follows by embedding either a $\sqrt{n(f)}$-wise OR or a $\sqrt{n(f)}$-wise AND. Binary uniform trees require $\Omega(\sqrt{n(f)})$ communication even for quantum protocols. This is because $\sqrt{n(f)}$-wise PARITY can be embedded in such a tree (see [8]), and then the bound follows from the lower bound for the generalized inner product function (see [6] and [13]). This can also be shown by methods of [15], which seem more promising towards a lower bound on the quantum communication complexity of arbitrary AND/OR trees.

Finally, we consider the more general setting, where $f(x, y)$ is a two-party read-once formula with its variables partitioned arbitrarily between Alice and Bob. This situation includes the case where the function is of the form $f^\vee$ or $f^\wedge$ and the variable partition is the natural one indicated earlier. As the case $f = \mathrm{AND}_n$ shows, we don't have a lower bound on $R_\delta(f)$ of the form $n(f)/c^{d(f)}$. However we can get an interesting general lower bound.

Consider the deterministic simultaneous message model, which is perhaps the weakest non-trivial communication complexity model. In this model Alice and Bob are trying to communicate $f(x, y)$ to a third party, the referee. Alice announces some function value $m_A(x)$ and simultaneously Bob announces a function value $m_B(y)$, and together $m_A(x)$ and $m_B(y)$ are enough for the referee to determine $f(x, y)$. The deterministic simultaneous message complexity, denoted $D^{||}(f)$, is the minimum number of bits (in worst case) that must be sent by Alice and Bob so that the referee can evaluate f. As a consequence of 8 we prove the following.

Theorem 4. *For any two-party read-once function f with $d(f) \geq 1$,*

$$R_\delta(f) \geq (1 - 2\sqrt{\delta}) \cdot \frac{D^{||}(f)}{d(f) \cdot 8^{d(f)-1}}.$$

1.1 Notation, terminology, and preliminaries

In this section we establish notation and terms that we will use to describe the basic objects that we will be dealing with. We list standard definitions and

state some basic inequalities in information theory. We discuss communication complexity and set up its connection with information theory.

Definitions pertaining to rooted trees. All trees in this paper are rooted. For a tree T we write V_T for the set of vertices, L_T for the set of leaves, $N_T = |L_T|$ for the number of leaves, and d_T for the depth of T. For a vertex u, path(u) is the set of vertices on a path from u to the root (including both the root and u).

We write $T = T_1 \circ \cdots \circ T_k$ when, for each $j \in \{1, \ldots, k\}$, T_j is the subtree rooted at the j-th child of the root of T.

A tree is called *t-uniform* if all its leaves are at the same depth d, and every non-leaf node has exactly t children.

A tree is in *standard form* if there are no nodes with exactly one child. For example, a standard binary tree is one where every internal node has exactly two children.

A *full binary subtree* of a tree T is a binary tree in standard form that is contained in T, contains the root of T, and whose leaf-set is a subset of the leaf-set of T. Denote by FBS_T the set of full binary subtrees of T.

Definitions pertaining to boolean functions. We denote by $[n]$ the set $\{1, \ldots, n\}$ of integers. Let $f : \mathcal{S}_1 \times \cdots \times \mathcal{S}_n \to \mathbb{R}$ be a function and suppose that, for $i \in [n]$, $h_i : \mathcal{Z}_i \to \mathcal{S}_i$. For $\mathcal{H} = \langle h_1, \ldots, h_n \rangle$, let $f^{\mathcal{H}} : \mathcal{Z}_1 \times \cdots \times \mathcal{Z}_n \to \mathbb{R}$ denote the function defined by $f^{\mathcal{H}}(z_1, \ldots, z_n) = f(h_1(z_1), \ldots, h_n(z_n))$. When $h_j = h$ for all $j \in [n]$, we write $f^h = f^{\mathcal{H}}$.

A *tree circuit* is a rooted tree in which every leaf corresponds to an input variable (or its negation), and each gate comes from the set $\{\mathrm{AND}, \mathrm{OR}, \mathrm{NAND}, \mathrm{NOR}\}$. We write f_C for the function represented by a tree circuit C. An AND/OR tree is a tree circuit with gates AND and OR. The tree circuit is *read-once* if the variables occurring at leaves are distinct; all tree circuits in this paper are assumed to be read-once. A Boolean function f is read-once if it can be represented by a read-once tree circuit. The depth of a read-once function f, denoted $d(f)$, is the minimum depth of a read-once tree circuit that computes it. As mentioned in the introduction, it is well-known that every read-once function f has a unique representation, called the *canonical representation of f*, whose tree is in standard form and such that the gates along any root to leaf path alternate between $\wedge$ and $\vee$. It is easy to show that the depth of the canonical representation is $d(f)$, that is, the canonical representation has minimum depth over all read-once tree circuits that represent f.

If T is any rooted tree, we write f_T for the boolean function obtained by associating a distinct variable x_j to each leaf j and labeling each gate by a NAND gate. We use symbol '$\overline{\wedge}$' for NAND.

Chapter 2

Information theory

2.1 Random variables and distributions.

We consider discrete probability spaces (Ω, ζ), where Ω is a finite set and ζ is a nonnegative valued function on Ω summing to 1. If $(\Omega_1, \zeta_1), \ldots, (\Omega_n, \zeta_n)$ are such spaces, their product is the space (Λ, ν), where $\Lambda = \Omega_1 \times \cdots \times \Omega_n$ is the Cartesian product of sets, and for $\omega = (\omega_1, \ldots, \omega_n) \in \Lambda$, $\nu(\omega) = \prod_{j=1}^{n} \zeta_j(\omega_j)$. In the case that all of the (Ω_i, ζ_i) are equal to a common space (Ω, ζ) we write $\Lambda = \Omega^n$ and $\nu = \zeta^n$.

We use uppercase for random variables, as in $X, Y, \mathbf{D}$, and write in bold those that represent vectors of random variables. For a variable X with range $\mathcal{X}$ that is distributed according to a probability distribution μ, i.e. $\Pr[X = x] = \mu(x)$, we write $X \sim \mu$. If X is uniformly distributed in $\mathcal{X}$, we write $X \in_R \mathcal{X}$.

Unless otherwise stated, all random variables take on values from finite sets.

2.2 Entropy and Mutual Information.

Let X, Y, Z be random variables on a common probability space, taking on values, respectively, from finite sets $\mathcal{X}, \mathcal{Y}, \mathcal{Z}$. Let A be any event. The *entropy* of X, the *conditional entropy of X given A*, and the *conditional entropy of X given Y* are respectively (we use log for $\log_2$)

$$\begin{aligned}
\mathrm{H}(X) &= -\sum_{x \in \mathcal{X}} \Pr[X = x] \cdot \log \Pr[X = x], \\
\mathrm{H}(X \mid A) &= -\sum_{x \in \mathcal{X}} \Pr[X = x \mid A] \cdot \log \Pr[X = x \mid A], \\
\mathrm{H}(X \mid Y) &= \sum_{y \in \mathcal{X}} \Pr[Y = y] \cdot \mathrm{H}(X \mid Y = y).
\end{aligned}$$

The *mutual information* between X and Y is

$$\mathrm{I}(X\,;\,Y) = \mathrm{H}(X) - \mathrm{H}(X \mid Y) = \mathrm{H}(Y) - \mathrm{H}(Y \mid X)$$

and the *conditional mutual information* of X and Y given Z is

$$\begin{aligned}\mathrm{I}(X\,;\,Y\,|\,Z) &= \mathrm{H}(X\,|\,Z) - \mathrm{H}(X\,|\,Y,Z)\\ &= \mathrm{H}(Y\,|\,Z) - \mathrm{H}(Y\,|\,X,Z)\\ &= \sum_{z\in\mathcal{Z}} \Pr[Z=z]\cdot \mathrm{I}(X\,;\,Y\,|\,Z=z).\end{aligned}$$

We will need the following facts about the entropy. (See [7, Chapter 2], for proofs and more details.)

Proposition 5. *Let X, Y, Z be random variables.*

1. $\mathrm{H}(X) \geq \mathrm{H}(X\,|\,Y) \geq 0$.
2. *If $\mathcal{X}$ is the range of X, then* $\mathrm{H}(X) \leq \log|\mathcal{X}|$.
3. $\mathrm{H}(X,Y) \leq \mathrm{H}(X) + \mathrm{H}(Y)$ *with equality if and only if X and Y are independent. This holds for conditional entropy as well.* $\mathrm{H}(X,Y\,|\,Z) \leq \mathrm{H}(X\,|\,Z) + \mathrm{H}(Y\,|\,Z)$ *with equality if and only if X and Y are independent given Z.*

The following proposition makes mutual information useful in proving direct-sum theorems.

Proposition 6 ([3])**.** *Let $\mathbf{Z} = \langle \mathbf{Z}_1, \ldots, \mathbf{Z}_n\rangle, \Pi, \mathbf{D}$ be random variables. If the $\mathbf{Z}_j$'s are independent given $\mathbf{D}$, then* $\mathrm{I}(\mathbf{Z}\,;\,\Pi\,|\,\mathbf{D}) \geq \sum_{j=1}^{n} \mathrm{I}(\mathbf{Z}_j\,;\,\Pi\,|\,\mathbf{D})$.

Proof. By definition $\mathrm{I}(\mathbf{Z}\,;\,\Pi\,|\,\mathbf{D}) = \mathrm{H}(\mathbf{Z}\,|\,\mathbf{D}) - \mathrm{H}(\mathbf{Z}\,|\,\Pi,\mathbf{D})$. By 3, $\mathrm{H}(\mathbf{Z}\,|\,\mathbf{D}) = \sum_j \mathrm{H}(\mathbf{Z}_j\,|\,\mathbf{D})$ and $\mathrm{H}(\mathbf{Z}\,|\,\Pi,\mathbf{D}) \leq \sum_j \mathrm{H}(\mathbf{Z}_j\,|\,\Pi,\mathbf{D})$. The result follows. □

Chapter 3

Communication complexity

In this work we will be dealing with the two-party private-coin randomized communication model, introduced by [19]. Alice is given $x \in \mathcal{X}$ and Bob $y \in \mathcal{Y}$. They wish to compute a function $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ by exchanging messages according to a protocol Π. Let the random variable $\Pi(x,y)$ denote the transcript of the communication on input $\langle x,y\rangle$ (where the probability is over the random coins of Alice and Bob) and $\Pi_{\text{out}}(x,y)$ the outcome of the protocol. We call Π a *δ-error protocol for f* if, for all $\langle x,y\rangle$, $\Pr[\Pi_{\text{out}}(x,y) = f(x,y)] \geq 1-\delta$. The *communication cost of* Π is $\max|\Pi(x,y)|$, where the maximum is over all input pairs $\langle x,y\rangle$ and over all coin tosses of Alice and Bob. The *δ-error randomized communication complexity of f*, denoted $R_\delta(f)$, is the cost of the best δ-error protocol for f. (See [14] for more details.)

3.1 Communication problems associated with boolean functions.

If f is an arbitrary n-variate boolean function, and g is a 2-variate boolean function, we denote by f^g the two-party boolean function given by $f^g(x,y) = f(g(x_1,y_1),\ldots,g(x_n,y_n))$. Our goal is to prove Theorems 1 and 2, which say that for any read-once boolean function f, either $f^\vee$ or $f^\wedge$ has high randomized communication cost. To do this it will be more convenient to consider $f^{\bar{\wedge}}$ for functions f that come from trees using only NAND gates. We first prove the following lemma.

For $f_1, f_2 : \{0,1\}^n \to \{0,1\}$, we write $f_1 \equiv f_2$ when $(\exists\sigma \in \{0,1\}^n)(\forall x \in \{0,1\}^n)(f_1(x) = f_2(\sigma \oplus x))$, where $\sigma \oplus x$ is the bitwise XOR of σ and x.

Lemma 7. *Let C be an* AND/OR *tree in canonical form and let T be the underlying tree. Then, $f_C \equiv f_T$ when the root of C is labeled by an* OR *gate, and $f_C \equiv \neg f_T$ when the root of C is labeled by an* AND *gate.*

Proof. We proceed by induction on d_T. When $d_T = 1$, the case with an AND at the root is trivial. For OR we observe that $f_C(x) = \bigvee_j x_j = \neg\bigwedge_j^n \neg x_j = f_T(\neg x)$.

Now suppose $d_T > 1$. Let $C = C_1 \wedge \cdots \wedge C_k$ and recall that C is in canonical form; thus, each C_j has an OR at the root. It follows by induction that $f_C(x) \equiv \bigwedge_j f_{T_j} = \neg f_T(x)$. If $C = C_1 \vee \cdots \vee C_k$, then we have $f_C = \bigvee_j f_{C_j} = \neg \bigwedge_j \neg f_{C_j} \equiv \neg \bigwedge_j f_{T_j} = f_T$. □

Our lower bounds follow from the following main theorem.

Theorem 8. *1. Let T be a tree in standard form with $d_T \geq 1$.*

$$R_\delta(f_T^{\bar{\wedge}}) \geq (2 - 4\sqrt{\delta}) \cdot \frac{N_T}{8^{d_T}}.$$

2. If T is, in addition, a t-uniform tree of depth $d_T \geq 1$, then

$$R_\delta(f_T^{\bar{\wedge}}) \geq (1 - 2\sqrt{\delta}) \cdot \frac{t(t-1)^{d_T-1}}{4^{d_T}}.$$

To deduce Theorems 1 and 2 we use the following proposition.

Proposition 9. *Let f be a read-once formula. Then there is a tree T in standard form such that (1) $R_\delta(f_T^{\bar{\wedge}}) \leq \max\{R_\delta(f^\wedge), R_\delta(f^\vee)\}$, (2) $N_T \geq n(f)/2$, (3) $d_T \leq d(f)$. Moreover, if the canonical representation of f is a uniform tree, $N_T = n(f)$.*

Proof. Let C be the representation of f in canonical form. Define tree circuits C_1 and C_2 as follows. To obtain C_1 delete all leaves that feed into $\wedge$ gates, and introduce a new variable for any node that becomes a leaf. Let C_1 be the canonical form of the resulting tree. Let C_2 be obtained similarly by deleting all leaves that feed into $\vee$ gates. Let f_1 and f_2, respectively, be the functions computed by C_1 and C_2. Let T_1 and T_2 be the trees underlying C_1 and C_2 respectively. We take T to be whichever of T_1 and T_2 has more leaves. Clearly conditions (2) and (3) above will hold. If the underlying tree of C is uniform, then one of C_1, C_2 will have $n(f)$ leaves; so in the uniform case we have $N_T = n(f)$. Condition (1) follows immediately from the following claim.

Claim 10. *(1) $R_\delta(f^\wedge) \geq R_\delta(f_1^\wedge)$. (2) $R_\delta(f_1^\wedge) = R_\delta(f_{T_1}^{\bar{\wedge}})$. (3) $R_\delta(f^\vee) \geq R_\delta(f_2^\vee)$. (4) $R_\delta(f_2^\vee) = R_\delta(f_{T_2}^{\bar{\wedge}})$.*

To prove the first part of the claim, it suffices to observe that any communication protocol for $f^\wedge$ can be used as a protocol for $f_1^\wedge$. In particular, given an input (x, y) to $f_1^\wedge$ Alice and Bob can—without any communication—construct input (x', y') to $f^\wedge$ such that $f^\wedge(x', y') = f_1^\wedge(x, y)$. This is done as follows. If j is a leaf of C that is also a leaf of C_1, then Alice sets $x'_j = x_j$ and Bob sets $y'_j = y_j$. Suppose j is a leaf of C that is not a leaf of C_1. If the parent $p(j)$ of j is a leaf of C_1, then Alice sets $x'_j = x_{p(j)}$ and Bob sets $y'_j = y_{p(j)}$. If $p(j)$ is not a leaf of C_1, then Alice sets $x'_j = 1$ and Bob sets $y'_j = 1$. It is easy to verify that $f^\wedge(x', y') = f_1^\wedge(x, y)$. The second part of the claim follows from 7. The proofs of parts 3 and 4 follow similarly. □

3.2 Communication complexity lower bounds via information theory.

The informational complexity paradigm, introduced by [5], and used in [17, 2, 4, 3, 11], provides a way to prove lower bounds on communication complexity via information theory. We are given a two-party function f and we want to show that any δ-error randomized communication protocol Π for f requires high communication. We introduce a probability distribution over the inputs to Alice and Bob. We then analyze the behavior of Π when run on inputs chosen randomly according to the distribution. The informational complexity is the mutual information of the string of communicated bits (the *transcript* of Π) with Alice and Bob's inputs, and provides a lower bound on the amount of communication.

More precisely, let $\Omega = (\Omega, \zeta)$ be a probability space over which are defined random variables $\mathbf{X} = \langle X_1, \ldots, X_n \rangle$ and $\mathbf{Y} = \langle Y_1, \ldots, Y_n \rangle$ representing Alice and Bob's inputs. The *information cost* of a protocol Π with respect to ζ is defined to be $\mathrm{I}(\mathbf{X}, \mathbf{Y}\,;\, \Pi(\mathbf{X}, \mathbf{Y}))$, where $\Pi(\mathbf{X}, \mathbf{Y})$ is a random variable following the distribution of the communication transcripts when the protocol Π runs on input $\langle \mathbf{X}, \mathbf{Y} \rangle \sim \zeta$. The *$\delta$-error informational complexity* of f with respect to ζ, denoted $\mathrm{IC}_{\zeta,\delta}(f)$, is $\min_\Pi \mathrm{I}(\mathbf{X}, \mathbf{Y}\,;\, \Pi(\mathbf{X}, \mathbf{Y}))$, where the minimum is over all δ-error randomized protocols for f.

Mutual information may be easier to handle if one conditions on the appropriate random variables. To that end, [3] introduced the notion of *conditional information cost* of a protocol Π with respect to an auxiliary random variable. Let (Ω, ζ) be as above, and let $\mathbf{D}$ be an additional random variable defined on Ω. The *conditional information cost* of Π conditioned on $\mathbf{D}$ with respect to ζ is defined to be $\mathrm{I}(\mathbf{X}, \mathbf{Y}\,;\, \Pi(\mathbf{X}, \mathbf{Y}) \,|\, \mathbf{D})$, where $\Pi(\mathbf{X}, \mathbf{Y})$ is as above and $(\langle \mathbf{X}, \mathbf{Y} \rangle, \mathbf{D}) \sim \zeta$. The *$\delta$-error conditional informational complexity* of f conditioned on $\mathbf{D}$ with respect to ζ, denoted $\mathrm{IC}_{\zeta,\delta}(f \,|\, \mathbf{D})$, is $\min_\Pi \mathrm{I}(\mathbf{X}, \mathbf{Y}\,;\, \Pi(\mathbf{X}, \mathbf{Y}) \,|\, \mathbf{D})$, where the minimum is over all δ-error randomized protocols for f.

Conditional informational complexity provides a lower bound on randomized communication complexity, as shown by the following calculation. By definition of mutual information $\mathrm{I}(\mathbf{X}, \mathbf{Y}\,;\, \Pi(\mathbf{X}, \mathbf{Y}) \,|\, \mathbf{D}) = \mathrm{H}(\Pi(\mathbf{X}, \mathbf{Y}) \,|\, \mathbf{D}) - \mathrm{H}(\Pi(\mathbf{X}, \mathbf{Y}) \,|\, \mathbf{X}, \mathbf{Y}, \mathbf{D})$. Applying in turn parts (i) and (ii) of 5 gives that, for any δ-error protocol Π, $\mathrm{I}(\mathbf{X}, \mathbf{Y}\,;\, \Pi(\mathbf{X}, \mathbf{Y}) \,|\, \mathbf{D}) \leq \mathrm{H}(\Pi(\mathbf{X}, \mathbf{Y})) \leq R_\delta(f)$.

3.3 The methods of Jayram et al. [3]

[3] introduced new techniques for proving lower bounds on information cost. In this section we summarize their method and list the results and definitions from [3] that we will use.

Their methodology has two main parts. In the first part they make use of 6 to obtain a direct-sum theorem for the informational complexity of the function. This works particularly well with functions of the form $f^h(\mathbf{x}, \mathbf{y}) =$

$f(h(x_1, y_1), \ldots, h(x_n, y_n))$. Before stating the direct-sum theorem, we need some definitions.

Definition 11 (Sensitive input)**.** *Consider $f : \mathcal{S}_1 \times \cdots \times \mathcal{S}_n \to \mathbb{R}$, a family of functions $\mathcal{H} = \langle h_j : \mathcal{Z}_j \to \mathcal{S}_j \rangle_{j \in [n]}$, and $\mathbf{z} = \langle z_1, \ldots, z_n \rangle \in \mathcal{Z}_1 \times \cdots \times \mathcal{Z}_n$. For $j \in [n]$, $u \in \mathcal{Z}_j$, let $\mathbf{z}[j, u] = \langle z_1, \ldots, z_{j-1}, u, z_{j+1}, \ldots, z_n \rangle$. We say that $\mathbf{z}$ is* sensitive *for $f^{\mathcal{H}}$ if $(\forall j \in [n])(\forall u \in \mathcal{Z}_j)(f^{\mathcal{H}}(\mathbf{z}[j, u]) = h_j(u))$.*

For an example, consider the function $\mathrm{DISJ}_n(\mathbf{x}, \mathbf{y}) = \bigvee_{j=1}^{n}(x_j \wedge y_j)$. Any input $\langle \mathbf{x}, \mathbf{y} \rangle$ such that, for all $j \in [n]$, $x_j \wedge y_j = 0$, is sensitive.

Definition 12 (Collapsing distribution, [3])**.** *Let $f, \mathcal{H}$ be as in 11. Call a distribution μ over $\mathcal{Z}_1 \times \cdots \times \mathcal{Z}_n$* collapsing *for $f^{\mathcal{H}}$, if every $\mathbf{z}$ in the support of μ is sensitive.*

Theorem 13 ([3])**.** *Let $f : \mathcal{S}^n \to \{0, 1\}$, and $h : \mathcal{X} \times \mathcal{Y} \to \mathcal{S}$. Consider random variables $\mathbf{X} = \langle X_1, \ldots, X_n \rangle \in \mathcal{X}^n, \mathbf{Y} = \langle Y_1, \ldots, Y_n \rangle \in \mathcal{Y}^n, \mathbf{D} = \langle D_1, \ldots, D_n \rangle$, and $\mathbf{Z} = \langle Z_1, \ldots, Z_n \rangle$, where $Z_j = \langle X_j, Y_j, D_j \rangle$ for $j \in [n]$.*

Assume that $\{Z_j\}_{j \in [n]}$ is a set of mutually independent variables, and $Z_j \sim \zeta$ for all $j \in [n]$ (thus, $\mathbf{Z} \sim \zeta^n$). If, for all $j \in [n]$, X_j and Y_j are independent given D_j, and the marginal distribution of $(\mathbf{X}, \mathbf{Y})$ is a collapsing distribution for f^h, then $\mathrm{IC}_{\zeta^n, \delta}(f^h \,|\, \mathbf{D}) \geq n \cdot \mathrm{IC}_{\zeta, \delta}(h \,|\, D)$.

Defining a distribution ζ satisfying the two requirements asked in 13, moves the attention from $\mathrm{IC}_{\zeta^n, \delta}(f^h \,|\, \mathbf{D})$ to $\mathrm{IC}_{\zeta, \delta}(h \,|\, D)$. For example, in [3] it is shown how to define ζ when f^h is $\mathrm{DISJ}_n(\mathbf{x}, \mathbf{y}) = \bigvee_{j=1}^{n}(x_j \wedge y_j)$. Then one only has to deal with $\mathrm{IC}_{\zeta, \delta}(h \,|\, D)$, where $h(x, y) = x \wedge y$.

The second part of the method is a framework for proving lower bounds on information cost. The first step consists of a passage from mutual information to Hellinger distance.

Definition 14. *(Hellinger distance.) The* Hellinger distance *between probability distributions P and Q on a domain Ω is defined by*

$$\mathrm{h}(P, Q) = \sqrt{\tfrac{1}{2} \textstyle\sum_{\omega \in \Omega} \left(\sqrt{P_\omega} - \sqrt{Q_\omega}\right)^2}.$$

We write $\mathrm{h}^2(P, Q)$ for $(\mathrm{h}(P, Q))^2$.

Lemma 15 ([3])**.** *Let $\Phi(z_1)$, $\Phi(z_2)$, and $Z \in_R \{z_1, z_2\}$ be random variables. If $\Phi(z)$ is independent of Z for each $z \in \{z_1, z_2\}$, then $\mathrm{I}(Z\,;\, \Phi(Z)) \geq \mathrm{h}^2(\Phi(z_1), \Phi(z_2))$.*

The following proposition states useful properties of Hellinger distance. They reveal why Hellinger distance is better to work with than mutual information.

Proposition 16 (Properties of Hellinger distance, [3])**.**

1. (Triangle inequality.) *Let P, Q, and R be probability distributions over domain Ω; then* $\mathrm{h}(P,Q) + \mathrm{h}(Q,R) \geq \mathrm{h}(P,R)$. *It follows that the square of the Hellinger distance satisfies a weak triangle inequality:*

$$\mathrm{h}^2(P,Q) + \mathrm{h}^2(Q,R) \geq \tfrac{1}{2}\,\mathrm{h}^2(P,R).$$

2. (Cut-and-paste property.) *For any randomized protocol Π and for any $x, x' \in \mathcal{X}$ and $y, y' \in \mathcal{Y}$,*

$$\mathrm{h}(\Pi(x,y), \Pi(x',y')) = \mathrm{h}(\Pi(x,y'), \Pi(x',y)).$$

3. (Pythagorean property.) *For any randomized protocol Π and for any $x, x' \in \mathcal{X}$ and $y, y' \in \mathcal{Y}$,*

$$\mathrm{h}^2(\Pi(x,y), \Pi(x',y)) + \mathrm{h}^2(\Pi(x,y'), \Pi(x',y')) \leq 2\,\mathrm{h}^2(\Pi(x,y), \Pi(x',y')).$$

4. *For any δ-error randomized protocol Π for a function f, and for any two input pairs (x,y) and (x',y') for which $f(x,y) \neq f(x',y')$,*

$$\mathrm{h}^2(\Pi(x,y), \Pi(x',y')) \geq 1 - 2\sqrt{\delta}.$$

After an application of 15 we are left with a sum of Hellinger distance terms, which we need to lower bound. Applying properties (i)–(iii) several times we can arrive at a sum of terms different than the ones we started with. To obtain a lower bound we would like the final terms to include terms to which Property 4 can be applied.

Chapter 4

Read-once boolean formulae

Let $T = T_1 \circ \cdots \circ T_n$ be a tree in standard form computing a function f_T. A first step towards simplifying the informational complexity of $f_T^{\bar{\wedge}}$ would be to apply the following straightforward generalization of 13.

Theorem 17. *Consider a function $f : \mathcal{S}_1 \times \cdots \times \mathcal{S}_n \to \{0,1\}$, a family of functions $\mathcal{H} = \langle h_j : \mathcal{X}_j \times \mathcal{Y}_j \to \mathcal{S}_j \rangle_{j \in [n]}$, random variables $\mathbf{X} = \langle X_1, \ldots, X_n \rangle \in \mathcal{X}_1 \times \cdots \times \mathcal{X}_n$, $\mathbf{Y} = \langle Y_1, \ldots, Y_n \rangle \in \mathcal{Y}_1 \times \cdots \times \mathcal{Y}_n$, $\mathbf{D} = \langle D_1, \ldots, D_n \rangle$, and $\mathbf{Z} = \langle Z_1, \ldots, Z_n \rangle$, where $Z_j = \langle X_j, Y_j, D_j \rangle$ for $j \in [n]$.*

Assume that $\{Z_j\}_{j \in [n]}$ is a set of mutually independent variables, and $Z_j \sim \zeta_j$ for all $j \in [n]$ (thus, $\mathbf{Z} \sim \zeta_1 \cdots \zeta_n$). If, for all $j \in [n]$, X_j and Y_j are independent given D_j, and the marginal distribution of $(\mathbf{X}, \mathbf{Y})$ is a collapsing distribution for $f^{\mathcal{H}}$, then $\mathrm{IC}_{\zeta_1 \cdots \zeta_n, \delta}(f^{\mathcal{H}} \,|\, \mathbf{D}) \geq \sum_{j=1}^{n} \mathrm{IC}_{\zeta_j, \delta}(h_j \,|\, D_j)$.

One can apply 17 to the function $f_T^{\bar{\wedge}}$, with f the n-bit NAND and $h_j = f_{T_j}$, for $j \in [n]$. However, this won't take us very far. The problem is that if μ—the marginal distribution of $\langle \mathbf{X}, \mathbf{Y} \rangle$—is collapsing for f_T, then the support of μ is a subset of $(f^{\mathcal{H}})^{-1}(0)$. Therefore, we will inherit for each subtree a distribution μ_j with a support inside $h_j^{-1}(1)$. But the support of a collapsing distribution should lie inside $h_j^{-1}(0)$. This means that we cannot apply 17 repeatedly. This problem arose in [11] when studying the function $\mathrm{TRIBES}_{m,n}(\mathbf{x}, \mathbf{y}) = \bigwedge_{k=1}^{m} \mathrm{DISJ}_n(\mathbf{x}_k, \mathbf{y}_k) = \bigwedge_{k=1}^{m} \bigvee_{j=1}^{n} (x_{kj} \wedge y_{kj})$. [11] managed to overcome this problem by proving a more complicated direct-sum theorem for a non-collapsing distribution for DISJ. Inspired by their idea, we show how to do the same for arbitrary read-once boolean functions.

The information cost of a protocol Π that we will employ for our proof will have the form $\mathrm{I}(\mathbf{X}, \mathbf{Y} \,;\, \Pi(\mathbf{X}, \mathbf{Y}) \,|\, \Gamma, \mathbf{D})$, where random variables Γ and $\mathbf{D}$ are auxiliary variables that will be used to define the distribution over the inputs.

4.1 Further definitions on trees

We proceed with definitions of objects that will be needed to finally define a distribution ζ for $(\langle \mathbf{X}, \mathbf{Y} \rangle, \langle \Gamma, \mathbf{D} \rangle)$, which will give meaning to $\mathrm{IC}_{\zeta,\delta}(\underset{\sim}{f}^{\wedge}_T \,|\, \Gamma, \mathbf{D}) = \min_{\Pi} \mathrm{I}(\mathbf{X}, \mathbf{Y}\,; \Pi(\mathbf{X}, \mathbf{Y}) \,|\, \Gamma, \mathbf{D})$.

Definition 18. *(Valid coloring.) For our purposes, a* coloring *of a tree T is a partition of V_T into two sets $\gamma = \langle \mathrm{W}_\gamma, \mathrm{R}_\gamma \rangle$. The vertices of W_γ are said to be* white *and the vertices of R_γ are said to be* red*. A coloring is* valid *if it satisfies the following conditions.*

1. *The root is white.*
2. *A white node is either a leaf or exactly one of its children is red.*
3. *A red node is either a leaf or exactly two of its children are red.*

Example. For a standard binary tree, a valid coloring paints all nodes on some root-to-leaf path white and all the rest red. Thus, the number of valid colorings equals the number of leaves.

Consider now a t-uniform tree T, colored properly by γ. Each white node has exactly one red child that is the root of a red binary subtree. For $t > 2$ there will be two kinds of white leaves: those that have no red nodes on the path that connects them to the root, and those that have at least one red node on that path. Notice that the union of a white leaf of the first kind, the corresponding root-to-leaf path, and the red binary subtrees that are "hanging" from the white nodes on the path, form a full binary subtree S of T. Furthermore, the restriction of γ on S, denoted γ_S, is a valid coloring for S.

Definitions related to colorings. We note some properties of valid colorings and give further definitions of related objects. Consider a tree T and a valid coloring $\gamma = \langle \mathrm{W}_\gamma, \mathrm{R}_\gamma \rangle$.

(1) The red nodes induce a forest of binary trees in standard form called the *red forest*.

(2) We can define a one-to-one correspondence between the trees in the red forest and internal white nodes of T as follows. For each white node w, its unique red child is the root of one of the full binary trees. We let $\mathrm{RT}(w) = \mathrm{RT}_{\gamma,T}(w)$ denote the set of vertices in the red binary tree rooted at the red child of w. (For convenience, if w is a leaf, $\mathrm{RT}(w)$ is empty.)

(3) The *principal component* of γ is the set of white nodes whose path to the root consists only of white nodes. A *principal leaf* of γ is a leaf belonging to the principal component. Let $\mathrm{PL}_T(\gamma)$ denote the set of principal leaves of γ.

(4) A full binary subtree S of T (i.e. $S \in \mathrm{FBS}_T$) is said to be *compatible* with γ, written $S \propto \gamma$, if S has exactly one white leaf. (Notice that, since γ is valid, this leaf would have to be a principal leaf. Thus, $S \propto \gamma$ is equivalent to saying that the restriction of γ on V_S is a valid coloring for S.)

(5) Define $\mathrm{FBS}_T(\gamma) = \{S \in \mathrm{FBS}_T \mid S \propto \gamma\}$. This set is in one-to-one correspondence with the set $\mathrm{PL}_T(\gamma)$ of principal leaves. If u is a principal leaf, then the set $\mathrm{path}(u) \cup \bigcup_{w \in \mathrm{path}(u)} \mathrm{RT}(w)$ induces a tree $\mathrm{F}_\gamma(u)$ that belongs to $\mathrm{FBS}_T(\gamma)$, and conversely if S is in $\mathrm{FBS}_T(\gamma)$, then its unique white leaf u is principal and $S = \mathrm{F}_\gamma(u)$.

(6) Define the positive integers $m_{\gamma,T} = |\mathrm{FBS}_T(\gamma)| = |\mathrm{PL}_T(\gamma)|$, $m_T = \sum_\gamma m_{\gamma,T}$, and $\rho_T = \min_\gamma m_{\gamma,T}$, where the min is over all valid colorings γ. (Notice that, if $T = T_1 \circ \cdots \circ T_n$, then $\rho_T = \sum_j \rho_{T_j} - \max_j \rho_{T_j}$.)

On notation. Consider a tree T, $u \in V_T$, and a coloring γ of T. We write T_u for the subtree of T rooted at u. Consider a vector $\mathbf{z} \in \Sigma^{N_T}$, where each coordinate corresponds to a leaf. We write $\mathbf{z}_u$ for the part of $\mathbf{z}$ that corresponds to the leaves of T_u. For $S \in \mathrm{FBS}_T$ we write $\mathbf{z}_S$ for the part of $\mathbf{z}$ that corresponds to the leaves of S. We treat colorings similarly. For example, γ_S stands for $\langle \mathrm{W}_\gamma \cap V_S, \mathrm{R}_\gamma \cap V_S \rangle$.

4.2 The input distribution

Our proof will have two main components, analogous to the ones in [11]. The distribution over the inputs that we shall define is carefully chosen so that each component of the proof can be carried out.

In the first part (4.3) we prove a direct-sum theorem for arbitrary trees. Given an arbitrary tree T in standard form, we show how the information cost of a protocol for $f_T^{\barwedge}$ can be decomposed into a sum of information costs that correspond to full binary subtrees of T. In the second part of the proof (4.4) we provide a lower bound on the informational complexity of $f_S^{\barwedge}$, where S is an arbitrary binary tree in standard form.

For a uniform binary tree with N_S leaves, there is a natural distribution for which one can prove an $\Omega(\sqrt{N_S})$ lower bound on information cost. However, this distribution is not useful for us because it does not seem to be compatible with the first part of the proof. It turns out that for our purposes it is sufficient to prove a much weaker lower bound on the information cost for binary trees, of the form $\Omega(1/c^d)$ for some fixed $c > 0$, which will be enough to give a lower bound of $\Omega(n/c^d)$ on the communication complexity for general trees. The distribution for binary trees that we choose gives such a bound and is also compatible with the first part of the proof. This allows us to show that the information cost of a tree of depth d is at least $\frac{n}{2^d} B(d)$, where $B(d)$ is a lower bound on the information cost of (a communication protocol on) a depth-d binary tree.

Given an arbitrary tree T in standard form, we now define a distribution over inputs to Alice and Bob for $f_T^{\barwedge}$.

First, we associate to each standard binary tree S a special input $\langle \alpha_S, \beta_S \rangle$. We will be interested in the value $f_S^{\barwedge}(\alpha_S, \beta_S)$. These inputs, which now seem arbitrary, introduce structure in the final distribution. This structure is crucial for the effectiveness of the second part of our proof.

Definition 19. *We define input* $\langle \alpha_S, \beta_S \rangle$ *to* $f_S^{\bar{\wedge}}$ *for a standard binary tree* S*. The definition is recursive on the depth* d_S *of the tree.*

$$\langle \alpha_S, \beta_S \rangle = \begin{cases} \langle 1, 1 \rangle & \text{if } d_S = 0, \\ \langle \alpha_{S_1} \overline{\alpha}_{S_2}, \overline{\beta}_{S_1} \beta_{S_2} \rangle & \text{if } S = S_1 \circ S_2. \end{cases}$$

We will need the following property of $\langle \alpha_S, \beta_S \rangle$.

Proposition 20. *For a standard binary tree* S *with* $d_S > 0$, $f_S^{\bar{\wedge}}(\alpha_S, \beta_S) = f_S^{\bar{\wedge}}(\overline{\alpha}_S, \overline{\beta}_S) = 0$ *and* $f_S^{\bar{\wedge}}(\alpha_S, \overline{\beta}_S) = f_S^{\bar{\wedge}}(\overline{\alpha}_S, \beta_S) = 1$.

Proof. The proof is by induction on d_S.

For $d_S = 1$ the (unique) tree results in the function $f_S^{\bar{\wedge}}(x_1x_2, y_1y_2) = (x_1 \mathbin{\bar{\wedge}} y_1) \mathbin{\bar{\wedge}} (x_2 \mathbin{\bar{\wedge}} y_2)$. Clearly,

$$f_S^{\bar{\wedge}}(\alpha_S, \beta_S) = f_S^{\bar{\wedge}}(10, 01) = 0, \quad f_S^{\bar{\wedge}}(\overline{\alpha}_S, \overline{\beta}_S) = f_S^{\bar{\wedge}}(01, 10) = 0\,;$$
$$f_S^{\bar{\wedge}}(\alpha_S, \overline{\beta}_S) = f_S^{\bar{\wedge}}(10, 10) = 1, \quad f_S^{\bar{\wedge}}(\overline{\alpha}_S, \beta_S) = f_S^{\bar{\wedge}}(01, 01) = 1.$$

Suppose $d_S > 1$ and let $S = S_1 \circ S_2$. We have $f_S(\alpha_S, \beta_S) = f_{S_1}^{\bar{\wedge}}(\alpha_{S_1}, \overline{\beta}_{S_1}) \mathbin{\bar{\wedge}} f_{S_2}^{\bar{\wedge}}(\overline{\alpha}_{S_2}, \beta_{S_2}) = 1 \mathbin{\bar{\wedge}} 1 = 0$ (where we applied the inductive hypothesis on S_1 and S_2). The other cases can be verified in a similar manner. □

An input will be determined by three independent random variables $\Gamma, \mathbf{D}, \mathbf{R}$, which are defined as follows.

(i) Γ ranges over valid colorings γ for T, according to a distribution that weights each γ by the number of principal leaves it has. More precisely

$$\Pr[\Gamma = \gamma] = m_{\gamma,T}/m_T.$$

(ii) $\mathbf{D} = \langle D_1, \ldots, D_N \rangle \in_R \{\textsc{Alice},\textsc{Bob}\}^N$. So, for any $\mathbf{d} \in \{\textsc{Alice},\textsc{Bob}\}^N$, $\Pr[\mathbf{D} = \mathbf{d}] = 2^{-N}$.

(iii) $\mathbf{R} = \langle R_1, \ldots, R_N \rangle \in_R \{0,1\}^N$. So, for any $\mathbf{r} \in \{0,1\}^N$, $\Pr[\mathbf{R} = \mathbf{r}] = 2^{-N}$.

The inputs $\mathbf{X} = \langle X_1, \ldots, X_N \rangle$ and $\mathbf{Y} = \langle Y_1, \ldots, Y_N \rangle$ are determined by values γ, $\mathbf{d} = \langle d_1, \ldots, d_N \rangle$, and $\mathbf{r} = \langle r_1, \ldots, r_N \rangle$ for Γ, $\mathbf{D}$, and $\mathbf{R}$ as follows.

(i) Let $F_1, \ldots, F_k$ be the trees in the red forest determined by γ. The input to F_j, for $j \in [k]$, is $\langle \alpha_{F_j}, \beta_{F_j} \rangle$.

(ii) For a white leaf j, the corresponding input $\langle X_j, Y_j \rangle$ is determined as follows. If $d_j = \textsc{Alice}$, set $\langle X_j, Y_j \rangle = \langle 0, r_j \rangle$. If $d_j = \textsc{Bob}$, set $\langle X_j, Y_j \rangle = \langle r_j, 0 \rangle$.

The reader may think of the random variables $\mathbf{D}$ and $\mathbf{R}$ as labeling the leaves of the tree T. For a leaf $j \in [N]$, the corresponding variable D_j chooses the player whose j-th bit will be fixed to 0. The j-th bit of the other player is then set to be equal to the random bit R_j.

Example. At this point it might be useful for the reader to see how the input for a binary tree S is distributed. As remarked earlier, a coloring γ for S paints a root-to-leaf path white and all the other nodes red. For any such γ we have $\Pr[\Gamma = \gamma] = 1/N_S$. All the other input bits, besides the ones that correspond to the single white leaf, are fixed according to 19 and the red forest determined by γ. Thus, the only entropy in the input (given a coloring γ) comes from the single white leaf. The mutual information of the transcript and this leaf is what we lower bound in 4.4.

Let ζ_T be the resulting distribution on $(\langle \mathbf{X}, \mathbf{Y} \rangle, \langle \Gamma, \mathbf{D} \rangle)$. Let μ_T (resp. ν_T) be the marginal distribution of $\langle \mathbf{X}, \mathbf{Y} \rangle$ (resp. $\langle \Gamma, \mathbf{D} \rangle$). We often drop subscript T and write ζ, μ, and ν.

Proposition 21. *Consider a tree T and let $\langle \mathbf{x}, \mathbf{y}, \gamma, \mathbf{d} \rangle$ be in the support of ζ. If u is a red node with a white parent, then $f^{\bar{\wedge}}_{T_u}(\mathbf{x}_u, \mathbf{y}_u) = 0$. If u is a white node, then $f^{\bar{\wedge}}_{T_u}(\mathbf{x}_u, \mathbf{y}_u) = 1$.*

Proof. The proof is by induction on d_{T_u}.

When $d_{T_u} = 0$, u is a leaf. If u is red and its parent is white, then T_u is a (one-vertex) tree in the red forest determined by γ. 19 then implies that $\langle \mathbf{x}_u, \mathbf{y}_u \rangle = \langle 1, 1 \rangle$ and so $f^{\bar{\wedge}}_{T_u}(\mathbf{x}_u, \mathbf{y}_u) = 0$. If u is white, notice that either $\mathbf{x}_u = 0$ or $\mathbf{y}_u = 0$ (see item (ii) above).

When $d_{T_u} > 0$ and u is white, then u has a red child v. By induction $f^{\bar{\wedge}}_{T_v}(\mathbf{x}_v, \mathbf{y}_v) = 0$, and it follows that $f^{\bar{\wedge}}_{T_u}(\mathbf{x}_u, \mathbf{y}_u) = 1$. If u is red and its parent is white, then there is a tree F rooted at u in the red forest. We claim that $f^{\bar{\wedge}}_{T_u}(\mathbf{x}_u, \mathbf{y}_u) = f^{\bar{\wedge}}_F(\mathbf{x}_F, \mathbf{y}_F)$. The statement then follows by 20, because, according to the definition of ζ_T, $\langle \mathbf{x}_F, \mathbf{y}_F \rangle = \langle \alpha_F, \beta_F \rangle$. The claim holds because every $v \in V_F$ has only white children outside F, and—by the induction hypothesis—their values do not affect the value of v (since the inputs to a $\bar{\wedge}$-gate that are equal to '1' are, in some sense, irrelevant to the output). □

4.3 A direct-sum theorem for read-once boolean formulae

Let T be an arbitrary tree in standard form and $S \in \mathrm{FBS}_T$. Suppose we have a communication protocol Π for $f^{\bar{\wedge}}_T$ and we want a protocol for $f^{\bar{\wedge}}_S$. One natural way to do this is to have Alice extend her input $\mathbf{x}_S$ for S to an input $\mathbf{x}$ for T and Bob extend his input $\mathbf{y}_S$ for S to an input $\mathbf{y}$ for T, in such a way that $f^{\bar{\wedge}}_T(\mathbf{x}, \mathbf{y}) = f^{\bar{\wedge}}_S(\mathbf{x}_S, \mathbf{y}_S)$. Then by running Π on $\langle \mathbf{x}, \mathbf{y} \rangle$ they obtain the desired output.

Let Π be any protocol for $f^{\bar{\wedge}}_T$. For any $S \in \mathrm{FBS}_T$ we will construct a family of protocols for $f^{\bar{\wedge}}_S$. Each protocol in the family will be specified by a pair $\langle \gamma, \mathbf{d} \rangle$ where γ is a valid coloring of T that is compatible with S, and $\mathbf{d} \in \{\textsc{Alice}, \textsc{Bob}\}^{N_T}$

Alice and Bob plug their inputs in T, exactly where S is embedded. To generate the rest of the input bits for T, they first use γ to paint the nodes of T not in S. For a red leaf j, the values of X_j and Y_j are determined by the coloring γ, so Alice and Bob can each determine x_j and y_j without communication. For a white leaf j outside S, they have to look at the value of d_j. If $d_j = \textsc{Alice}$, Alice sets $x_j = 0$, and Bob uses a random bit of his own to (independently) set his input bit y_j. If $d_j = \textsc{Bob}$, Bob sets $y_j = 0$, and Alice uses a random bit to set x_j. After this preprocessing, they simulate Π. Denote this protocol by $\Pi_S[\gamma, \mathbf{d}]$.

To argue the correctness of $\Pi_S[\gamma, \mathbf{d}]$ for any S, γ, and $\mathbf{d}$, notice that any node in S has only white children outside S (this follows from the conditions that a coloring satisfies). From 21 we know that a white node does not affect the value of its parent.

We now define a distribution over the triples $\langle S, \gamma, \mathbf{d}\rangle$ so that the average of the information cost of $\Pi_S[\gamma, \mathbf{d}]$ will be related to the information cost of Π. Recall that N_T is the number of leaves, and that m_T and ρ_T are integers related to the tree T defined in part (6) of the paragraph on "definitions related to colorings" in 4.1. The distribution ξ_T for triples $\langle S, \gamma, \mathbf{d}\rangle$ is as follows,

$$\xi_T(S, \gamma, \mathbf{d}) = \begin{cases} \frac{1}{m_T 2^{N_T}} & \text{if } S \propto \gamma, \\ 0 & \text{otherwise.} \end{cases}$$

This is indeed a distribution since

$$\sum_{S,\gamma,\mathbf{d}} \xi_T(S, \gamma, \mathbf{d}) = \sum_{S \propto \gamma} \sum_{\mathbf{d}} \tfrac{1}{m_T 2^{N_T}} = \sum_{S \propto \gamma} \tfrac{1}{m_T} = 1.$$

Lemma 22. *Consider any protocol Π for a tree T. Let $(\langle \mathbf{X}, \mathbf{Y}\rangle, \langle \Gamma, \mathbf{D}\rangle) \sim \zeta_T$ and $(\langle \mathbf{X}', \mathbf{Y}'\rangle, \langle \Gamma', \mathbf{D}'\rangle) \sim \zeta_S$; then*

$$\mathrm{I}(\mathbf{X}, \mathbf{Y}\,;\, \Pi \,|\, \Gamma, \mathbf{D}) \geq \rho_T \cdot \mathbf{E}_{\langle S,\gamma,\mathbf{d}\rangle \sim \xi_T}[\mathrm{I}(\mathbf{X}', \mathbf{Y}'\,;\, \Pi_S[\gamma, \mathbf{d}] \,|\, \Gamma', \mathbf{D}')].$$

Proof. We start by evaluating the right-hand side. (Recall that for γ and $\mathbf{d}$ we write γ_S and $\mathbf{d}_S$ for their restrictions in $S \in \mathrm{FBS}_T$.)

$$\begin{aligned}
&\mathbf{E}_{\langle S,\gamma,\mathbf{d}\rangle \sim \xi_T}[\mathrm{I}(\mathbf{X}', \mathbf{Y}'\,;\, \Pi_S[\gamma, \mathbf{d}] \,|\, \Gamma', \mathbf{D}')] \\
&= \sum_{S,\gamma,\mathbf{d}} \xi_T(S, \gamma, \mathbf{d}) \sum_{\gamma', \mathbf{d}'} \nu_S(\gamma', \mathbf{d}') \cdot \mathrm{I}(\mathbf{X}', \mathbf{Y}'\,;\, \Pi_S[\gamma, \mathbf{d}] \,|\, \Gamma' = \gamma', \mathbf{D}' = \mathbf{d}')] \\
&= \sum_{S,\gamma',\mathbf{d}'} \sum_{\gamma : S \propto \gamma} \sum_{\mathbf{d}} \tfrac{1}{m_T 2^{N_T}} \cdot \tfrac{1}{N_S 2^{N_S}} \cdot \mathrm{I}(\mathbf{X}', \mathbf{Y}'\,;\, \Pi_S[\gamma, \mathbf{d}] \,|\, \Gamma' = \gamma', \mathbf{D}' = \mathbf{d}')] \qquad (4.1) \\
&= \sum_{\substack{S,\gamma: \\ S \propto \gamma}} \sum_{\mathbf{d}} \tfrac{1}{m_{\gamma,T}} \cdot \tfrac{m_{\gamma,T}}{m_T 2^{N_T}} \cdot \mathrm{I}(\mathbf{X}', \mathbf{Y}'\,;\, \Pi_S[\gamma, \mathbf{d}] \,|\, \Gamma' = \gamma_S, \mathbf{D}' = \mathbf{d}_S). \qquad (4.2)
\end{aligned}$$

The transition from 4.1 to 4.2 needs to be justified. Look first at equation 4.2. Fix values $\widehat{S}, \widehat{\gamma}$, and $\widehat{\mathbf{d}}$ for the summation indices S, γ, and $\mathbf{d}$ respectively. Consider

the corresponding term $A = \mathrm{I}(\mathbf{X}', \mathbf{Y}' ; \Pi_{\widehat{S}}[\widehat{\gamma}, \widehat{\mathbf{d}}] \,|\, \Gamma' = \widehat{\gamma}_S, \mathbf{D}' = \widehat{\mathbf{d}}_S)$ in the sum. Now look at 4.1. Fix indices S, γ', and $\mathbf{d}'$ to $\widehat{S}$, $\widehat{\gamma}_S$, and $\widehat{\mathbf{d}}_S$ respectively. We claim that there are $N_S 2^{N_S}$ values $\langle \gamma, \mathbf{d} \rangle$, such that $\mathrm{I}(\mathbf{X}', \mathbf{Y}' ; \Pi_{\widehat{S}}[\gamma, \mathbf{d}] \,|\, \Gamma' = \widehat{\gamma}_S, \mathbf{D}' = \widehat{\mathbf{d}}_S) = A$. Indeed, any $\langle \gamma, \mathbf{d} \rangle$ such that γ agrees with $\widehat{\gamma}$ outside S, and $\mathbf{d}$ agrees with $\widehat{\mathbf{d}}$ outside S, contributes A to the sum in equation 4.1. There are N_S such γ and 2^{N_S} such $\mathbf{d}$.

Let us define $j(\gamma, S)$ to be the white leaf of S which is colored white by γ. Recalling the definition of ρ_T (4.1), the last equation gives

$$\begin{aligned} &\mathbf{E}_{\langle S,\gamma,\mathbf{d}\rangle \sim \xi_T}[\mathrm{I}(\mathbf{X}', \mathbf{Y}' ; \Pi_S[\gamma, \mathbf{d}] \,|\, \Gamma', \mathbf{D}')] \\ &\leq \tfrac{1}{\rho_T} \sum_{\substack{S,\gamma: \\ S \propto \gamma}} \sum_{\mathbf{d}} \tfrac{m_{\gamma,T}}{m_T 2^{N_T}} \cdot \mathrm{I}(X'_{j(\gamma,S)}, Y'_{j(\gamma,S)} ; \Pi_S[\gamma, \mathbf{d}] \,|\, \Gamma' = \gamma_S, \mathbf{D}' = \mathbf{d}_S). \end{aligned} \tag{4.3}$$

For the left-hand side we have

$$\begin{aligned} &\mathrm{I}(\mathbf{X}, \mathbf{Y} ; \Pi \,|\, \Gamma, \mathbf{D}) \\ &= \sum_{\gamma,\mathbf{d}} \nu_T(\gamma, \mathbf{d}) \cdot \mathrm{I}(\mathbf{X}, \mathbf{Y} ; \Pi \,|\, \Gamma = \gamma, \mathbf{D} = \mathbf{d}) \\ &\geq \sum_{\gamma,\mathbf{d}} \tfrac{m_{\gamma,T}}{m_T 2^{N_T}} \sum_{j \in \mathrm{PL}_T(\gamma)} \mathrm{I}(X_j, Y_j ; \Pi \,|\, \Gamma = \gamma, \mathbf{D} = \mathbf{d}) \\ &= \sum_{\substack{S,\gamma: \\ S \propto \gamma}} \sum_{\mathbf{d}} \tfrac{m_{\gamma,T}}{m_T 2^{N_T}} \cdot \mathrm{I}(X_{j(\gamma,S)}, Y_{j(\gamma,S)} ; \Pi \,|\, \Gamma = \gamma, \mathbf{D} = \mathbf{d}). \end{aligned} \tag{4.4}$$

The inequality follows from 6, ignoring terms that correspond to nonprincipal leaves. The last equality follows from the bijection between $\mathrm{FBS}_T(\gamma)$ and $\mathrm{PL}_T(\gamma)$ as discussed in 4.1.

In view of 4.3 and 4.4, to finish the proof one only needs to verify that the two distributions $(X'_{j(\gamma,S)}, Y'_{j(\gamma,S)}, \Pi_S[\gamma, \mathbf{d}] \,|\, \Gamma' = \gamma_S, \mathbf{D}' = \mathbf{d}_S)$ and $(X_{j(\gamma,S)}, Y_{j(\gamma,S)}, \Pi \,|\, \Gamma = \gamma, \mathbf{D} = \mathbf{d})$ are identical. To see this, notice first that $\Pr[X'_{j(\gamma,S)} = b_x \,|\, \Gamma' = \gamma_S, \mathbf{D}' = \mathbf{d}_S] = \Pr[X_{j(\gamma,S)} = b_x \,|\, \Gamma = \gamma, \mathbf{D} = \mathbf{d}]$, because S is colored the same in both cases and $j(\gamma, S)$ is the white leaf of S. Similarly for $Y'_{j(\gamma,S)}$ and $Y_{j(\gamma,S)}$. Finally, it follows immediately from the definition of $\Pi_S[\gamma, \mathbf{d}]$, that $\Pr[\Pi_S[\gamma, \mathbf{d}](\mathbf{X}', \mathbf{Y}') = \tau \,|\, X'_{j(\gamma,S)} = b_x, Y'_{j(\gamma,S)} = b_y, \Gamma' = \gamma_S, \mathbf{D}' = \mathbf{d}_S] = \Pr[\Pi(\mathbf{X}, \mathbf{Y}) = \tau \,|\, X_{j(\gamma,S)} = b_x, Y_{j(\gamma,S)} = b_y, \Gamma = \gamma, \mathbf{D} = \mathbf{d}]$. □

To obtain a lower bound from this lemma, we want to lower bound ρ_T and the informational complexity of standard binary trees. The later is done in the next section. The following lemma shows that we can assume $\rho_T \geq N_T / 2^{d_T}$.

Lemma 23. *For any tree T with N leaves and depth d, there is a tree $\widehat{T}$ with the following properties. (1) $\widehat{T}$ is in standard form, (2) $R_\delta(f_T^{\bar{\wedge}}) \geq R_\delta(f_{\widehat{T}}^{\bar{\wedge}})$, (3) $\rho_{\widehat{T}} \geq N/2^d$.*

Proof. First, we describe the procedure which applied on T produces $\widehat{T}$. If T is a single node we set $\widehat{T} = T$. Otherwise, assume $T = T_1 \circ \cdots \circ T_n$ and denote N_j the number of leaves in each T_j. We consider two cases.

A. If there is a j such that $N_j \geq N/2$, then we apply the procedure to T_j to obtain $\widehat{T}_j$, set $\widehat{T} = \widehat{T}_j$, and remove the remaining subtrees.

B. Otherwise, for each $j \in [n]$ apply the procedure on T_j to get $\widehat{T}_j$, and set $\widehat{T} = \widehat{T}_1 \circ \cdots \circ \widehat{T}_n$.

Now we prove by induction on d that $\widehat{T}$ has properties (1) and (3). When $d = 0$ and T is a single node, $\rho_T = 1$ and all properties are easily seen to be true. Otherwise, if $\widehat{T}$ is created as in case A, then clearly property (1) holds. For property (3) assume $\widehat{T} = \widehat{T}_j$. By induction, $\rho_{\widehat{T}_j} \geq N_j/2^{d-1}$. It follows that $\rho_{\widehat{T}} = \rho_{\widehat{T}_j} \geq N/2^d$ (since $N_j \geq N/2$). Now suppose case B applies and $\widehat{T}$ is created from $\widehat{T}_1, \ldots, \widehat{T}_n$. The restructuring described in case B preserves property (1). For property (3) assume—without loss of generality—that $\rho_{\widehat{T}_1} \leq \cdots \leq \rho_{\widehat{T}_n}$. By the definition of ρ_T (4.1, part (6) in "definitions related to colorings"),

$$\rho_{\widehat{T}} = \sum_{j=1}^{n-1} \rho_{\widehat{T}_j} \geq \sum_{j=1}^{n-1} N_j/2^{d-1} = (N - N_n)/2^{d-1} > (N - N/2)/2^{d-1} = N/2^d.$$

Finally, property (2) is true because Alice and Bob can simulate the protocol for f_T after they set their bits below a truncated tree to '1'. □

4.4 Bounding the informational complexity of binary trees

In this section we concentrate on standard binary trees. Our goal is to prove a lower bound of the form $\mathrm{I}(\mathbf{X}, \mathbf{Y}\,;\, \Pi \,|\, \Gamma, \mathbf{D}) \geq 2^{-\Theta(d_T)}$. We prove such an inequality using induction on d_T. The following statement provides the needed strengthening for the inductive hypothesis.

Proposition 24. *Let T be a standard binary tree, and let T_u be a subtree rooted at an internal node u of T. Assume that $(\langle \mathbf{X}_u, \mathbf{Y}_u \rangle, \langle \Gamma_u, \mathbf{D}_u \rangle) \sim \zeta_{T_u}$ and $\langle \mathbf{X}, \mathbf{Y} \rangle = \langle a\mathbf{X}_u b, c\mathbf{Y}_u d \rangle$, where a, b, c, d are fixed bit-strings. Then, for any protocol Π, we have*

$$\mathrm{I}(\mathbf{X}_u, \mathbf{Y}_u\,;\, \Pi(\mathbf{X}, \mathbf{Y}) \,|\, \Gamma_u, \mathbf{D}_u) \geq \frac{\mathrm{h}^2(\Pi(a\alpha_{T_u} b, c\overline{\beta}_{T_u} d), \Pi(a\overline{\alpha}_{T_u} b, c\beta_{T_u} d))}{2N_{T_u} 2^{d_{T_u}+1}}.$$

Proof. The proof is by induction on the depth d_{T_u} of T_u.

When $d_{T_u} = 0$ we have $f_{T_u}(x, y) = x \overline{\wedge} y$. This case was shown in [3, Section 6], but we redo it here for completeness. First, notice that Γ_u is constant and thus

the left-hand side simplifies to $\mathrm{I}(X_u, Y_u \,;\, \Pi(X,Y) \,|\, D_u)$. Expanding on values of D_u this is equal to

$$\tfrac{1}{2}\big(\mathrm{I}(Y_u \,;\, \Pi(a0b, cY_ud) \,|\, D_u = \textsc{Alice}) + \mathrm{I}(X_u \,;\, \Pi(aX_ub, c0d) \,|\, D_u = \textsc{Bob})\big),$$

because given $D_u = \textsc{Alice}$ we have $X_u = 0$ and given $D_u = \textsc{Bob}$ we have $Y_u = 0$. Also, given $D_u = \textsc{Alice}$ we have $Y_u \in_R \{0,1\}$ and thus the first term in the expression above can be written as $\mathrm{I}(Z \,;\, \Pi(a0b, cZd))$, where $Z \in_R \{0,1\}$. Now we apply 15 to bound this from below by $\mathrm{h}^2(\Pi(a0b, c0d), \Pi(a0b, c1d))$. Bounding the other term similarly and putting it all together we get

$$\begin{aligned}
&\mathrm{I}(X_u, Y_u \,;\, \Pi(X,Y) \,|\, D_u) \\
&\quad \geq \tfrac{1}{2}\big(\mathrm{h}^2(\Pi(a0b, c0d), \Pi(a0b, c1d)) + \mathrm{h}^2(\Pi(a0b, c0d), \Pi(a1b, c0d))\big) \\
&\quad \geq \tfrac{1}{4} \cdot \mathrm{h}^2(\Pi(a0b, c1d), \Pi(a1b, c0d)).
\end{aligned}$$

For the last inequality we used the triangle inequality of Hellinger distance (16). Since $\langle \alpha_{T_u}, \beta_{T_u} \rangle = \langle 1, 1 \rangle$ this is the desired result.

Now suppose $d_{T_u} > 0$ and let $T_u = T_{u_1} \circ T_{u_2}$. Either $u_1 \in \mathrm{W}_{\Gamma_u}$ (i.e. u_1 is white), or $u_2 \in \mathrm{W}_{\Gamma_u}$. Thus, expanding on Γ_u, the left-hand side can be written as follows.

$$\begin{aligned}
&\tfrac{N_{T_{u_1}}}{N_{T_u}} \cdot \mathrm{I}(\mathbf{X}_u, \mathbf{Y}_u \,;\, \Pi(a\mathbf{X}_ub, c\mathbf{Y}_ud) \,|\, \Gamma_u, u_1 \in \mathrm{W}_{\Gamma_u}, \mathbf{D}_u) \\
&\qquad + \tfrac{N_{T_{u_2}}}{N_{T_u}} \cdot \mathrm{I}(\mathbf{X}_u, \mathbf{Y}_u \,;\, \Pi(a\mathbf{X}_ub, c\mathbf{Y}_ud) \,|\, \Gamma_u, u_2 \in \mathrm{W}_{\Gamma_u}, \mathbf{D}_u).
\end{aligned}$$

When u_1 is white, $\langle \mathbf{X}_{u_2}, \mathbf{Y}_{u_2} \rangle = \langle \alpha_{T_{u_2}}, \beta_{T_{u_2}} \rangle$, and $(\langle \mathbf{X}_{u_1}, \mathbf{Y}_{u_1} \rangle, \langle \Gamma_{u_1}, \mathbf{D}_{u_1} \rangle)$ is distributed according to $\zeta_{T_{u_1}}$. Similarly, given that u_2 is white, $\langle \mathbf{X}_{u_1}, \mathbf{Y}_{u_1} \rangle = \langle \alpha_{T_{u_1}}, \beta_{T_{u_1}} \rangle$, and $(\langle \mathbf{X}_{u_2}, \mathbf{Y}_{u_2} \rangle, \langle \Gamma_{u_2}, \mathbf{D}_{u_2} \rangle)$ is distributed according to $\zeta_{T_{u_2}}$. Thus, the above sum simplifies to

$$\begin{aligned}
&\tfrac{N_{T_{u_1}}}{N_{T_u}} \cdot \mathrm{I}(\mathbf{X}_{u_1}, \mathbf{Y}_{u_1}; \Pi(a\mathbf{X}_{u_1}\alpha_{T_{u_2}}b, c\mathbf{Y}_{u_1}\beta_{T_{u_2}}d) | \Gamma_{u_1}, \mathbf{D}_{u_1}) \\
&\qquad + \tfrac{N_{T_{u_2}}}{N_{T_u}} \cdot \mathrm{I}(\mathbf{X}_{u_2}, \mathbf{Y}_{u_2}; \Pi(a\alpha_{T_{u_1}}\mathbf{X}_{u_2}b, c\beta_{T_{u_1}}\mathbf{Y}_{u_2}d) | \Gamma_{u_2}, \mathbf{D}_{u_2}).
\end{aligned}$$

By induction, this is bounded from below by

$$\begin{aligned}
&\tfrac{N_{T_{u_1}}}{N_{T_u} \cdot 2N_{T_{u_1}} 2^{d_{T_u}}} \cdot \mathrm{h}^2(\Pi(a\alpha_{T_{u_1}}\alpha_{T_{u_2}}b, c\overline{\beta}_{T_{u_1}}\beta_{T_{u_2}}d), \Pi(a\overline{\alpha}_{T_{u_1}}\alpha_{T_{u_2}}b, c\beta_{T_{u_1}}\beta_{T_{u_2}}d)) \\
&+ \tfrac{N_{T_{u_2}}}{N_{T_u} \cdot 2N_{T_{u_2}} 2^{d_{T_u}}} \cdot \mathrm{h}^2(\Pi(a\alpha_{T_{u_1}}\alpha_{T_{u_2}}b, c\beta_{T_{u_1}}\overline{\beta}_{T_{u_2}}d), \Pi(a\alpha_{T_{u_1}}\overline{\alpha}_{T_{u_2}}b, c\beta_{T_{u_1}}\beta_{T_{u_2}}d)).
\end{aligned}$$

Applying the cut-and-paste property (16) of Hellinger distance this becomes

$$\begin{aligned}
&\tfrac{N_{T_{u_1}}}{N_{T_u} \cdot 2N_{T_{u_1}} 2^{d_{T_u}}} \cdot \mathrm{h}^2(\Pi(a\alpha_{T_{u_1}}\alpha_{T_{u_2}}b, c\beta_{T_{u_1}}\beta_{T_{u_2}}d), \Pi(a\overline{\alpha}_{T_{u_1}}\alpha_{T_{u_2}}b, c\overline{\beta}_{T_{u_1}}\beta_{T_{u_2}}d)) \\
&+ \tfrac{N_{T_{u_2}}}{N_{T_u} \cdot 2N_{T_{u_2}} 2^{d_{T_u}}} \cdot \mathrm{h}^2(\Pi(a\alpha_{T_{u_1}}\alpha_{T_{u_2}}b, c\beta_{T_{u_1}}\beta_{T_{u_2}}d), \Pi(a\alpha_{T_{u_1}}\overline{\alpha}_{T_{u_2}}b, c\beta_{T_{u_1}}\overline{\beta}_{T_{u_2}}d)).
\end{aligned}$$

Now, since the square of Hellinger distance satisfies the (weak) triangle inequality (see 16), we have

$$\geq \tfrac{1}{2N_{T_u}2^{d_{T_u}+1}} \cdot \mathrm{h}^2(\Pi(a\alpha_{T_{u_1}}\overline{\alpha}_{T_{u_2}}b, c\beta_{T_{u_1}}\overline{\beta}_{T_{u_2}}d), \Pi(a\overline{\alpha}_{T_{u_1}}\alpha_{T_{u_2}}b, c\overline{\beta}_{T_{u_1}}\beta_{T_{u_2}}d)).$$

Recalling the 19 of $\langle \alpha_T, \beta_T \rangle$ we get

$$= \tfrac{1}{2N_{T_u}2^{d_{T_u}+1}} \cdot \mathrm{h}^2(\Pi(a\alpha_T b, c\overline{\beta}_T d), \Pi(a\overline{\alpha}_T b, c\beta_T d)).$$

This completes the inductive proof. □

Corollary 25. *For any binary tree T in standard form*

$$\mathrm{IC}_{\zeta_T,\delta}(f_T^{\overline{\wedge}} \,|\, \Gamma, \mathbf{D}) \geq (1 - 2\sqrt{\delta}) \cdot \frac{1}{4^{d_T+1}}.$$

Proof. First apply 24 with the root of T as u and empty a, b, c, d.

$$\begin{aligned}
\mathrm{IC}_{\zeta_T,\delta}(f_T^{\overline{\wedge}} \,|\, \Gamma, \mathbf{D}) &\geq \tfrac{1}{4^{d_T+1}} \cdot \mathrm{h}^2(\Pi(\alpha_T, \overline{\beta}_T), \Pi(\overline{\alpha}_T, \beta_T)) \\
&\geq \tfrac{1}{4^{d_T+1}} \cdot \left(\tfrac{1}{2}\,\mathrm{h}^2(\Pi(\alpha_T, \overline{\beta}_T), \Pi(\overline{\alpha}_T, \overline{\beta}_T)) + \tfrac{1}{2}\,\mathrm{h}^2(\Pi(\alpha_T, \beta_T), \Pi(\overline{\alpha}_T, \beta_T))\right) \\
&\geq \tfrac{1}{4^{d_T+1}} \cdot (1 - 2\sqrt{\delta}).
\end{aligned}$$

The second inequality is an application of the Pythagorean property of Hellinger distance—3. The last inequality follows from 20 and 4. □

4.5 Lower bounds for read-once boolean functions

In this section we use the main lemmas we have proved to obtain bounds for read-once boolean functions.

Corollary 26. *1. For any tree T in standard form,*

$$\mathrm{IC}_{\zeta_T,\delta}(f_T^{\overline{\wedge}} \,|\, \Gamma, \mathbf{D}) \geq (1 - 2\sqrt{\delta}) \cdot \frac{\rho_T}{4^{d_T+1}}.$$

2. If, in addition, T is t-uniform,

$$\mathrm{IC}_{\zeta_T,\delta}(f_T^{\overline{\wedge}} \,|\, \Gamma, \mathbf{D}) \geq (1 - 2\sqrt{\delta}) \cdot \frac{(t-1)^{d_T}}{4^{d_T+1}}.$$

Proof. Let Π be a δ-error protocol for $f_T^{\overline{\wedge}}$. 22 holds for any Π, therefore

$$\mathrm{IC}_{\zeta_T,\delta}(f_T^{\overline{\wedge}} \,|\, \Gamma, \mathbf{D}) \geq \rho_T \cdot \min_{S \in \mathrm{FBS}_T} \mathrm{IC}_{\zeta_S,\delta}(f_S^{\overline{\wedge}} \,|\, \Gamma, \mathbf{D}).$$

We now use the bound from 25 to obtain (i). For (ii), we can compute ρ_T exactly to be $(t-1)^{d_T}$. □

Corollary 27. *1. For any tree T in standard form,*

$$R_\delta(f_T^{\bar{\wedge}}) \geq (2 - 4\sqrt{\delta}) \cdot \frac{N_T}{8^{d_T+1}}.$$

2. If, in addition, T is t-uniform,

$$R_\delta(f_T^{\bar{\wedge}}) \geq (1 - 2\sqrt{\delta}) \cdot \frac{(t-1)^{d_T}}{4^{d_T+1}}.$$

Proof. Recalling that informational complexity is a lower bound for randomized complexity, (ii) is immediate from 2. For (i), we apply 1 to $f_{\widehat{T}}$, where $\widehat{T}$ is as in 23. □

The constants do not match the ones in 8. Let $T = T_1 \circ \cdots \circ T_t$. The slight improvements can be obtained by applying 17 with f being the t-variate NAND, and, for each $j \in [t]$, h_j and ζ_j being f_{T_j} and ζ_{T_j}, respectively. Applying 1 to each of the trees T_j gives 1; similarly for 2.

4.6 Lower bound for read-once threshold functions

In this section we prove 3, stated in the introduction.

A *threshold* gate, denoted T_k^n for $n > 1$ and $1 \leq k \leq n$, receives n boolean inputs and outputs '1' if and only if at least k of them are '1'. A *threshold tree* is a rooted tree in which every leaf corresponds to a distinct input variable and every gate is a threshold gate. A *read-once threshold function* f_E is a function that can be represented by a threshold tree E. As before, we define $f_E^\wedge$ and $f_E^\vee$ and we want to lower bound $\max\{R_\delta(f_E^\wedge), R_\delta(f_E^\vee)\}$. The following proposition shows that Alice and Bob can reduce a problem defined by an AND/OR tree to one defined by a threshold tree. 3 will then follow as a corollary of 1.

Proposition 28. *For any threshold tree E, there is an* AND/OR *tree T such that, for $g \in \{\wedge, \vee\}$, (1) $R_\delta(f_T^g) \leq R_\delta(f_E^g)$, (2) $N_T \geq N_E/2^{d_E}$, and (3) $d_T = d_E$.*

Proof. We define T by recursion on d_E. When $d_E = 0$ we set $T = E$. Otherwise, let $E = E_1 \circ \cdots \circ E_n$, and assume $N_{E_1} \geq \cdots \geq N_{E_n}$. Suppose the gate on the root is T_k^n. We consider cases on k. (1) If $1 < k \leq n/2$, build $T_1, \ldots, T_{n-k+1}$ recursively, set $T = T_1 \circ \cdots \circ T_{n-k+1}$, and put an $\vee$-gate on the root. (2) If $n/2 < k < n$, build $T_1, \ldots, T_k$ recursively, set $T = T_1 \circ \cdots \circ T_k$, and put an $\wedge$-gate on the root. (3) Otherwise, if $k = 1$ or $k = n$, the threshold gate is equivalent to an $\vee$ or $\wedge$-gate respectively. We build $T_1, \ldots, T_n$ recursively and we set $T = T_1 \circ \cdots \circ T_n$. The gate on the root remains as is.

Properties (2) and (3) are easily seen to hold. For (1), it is not hard to show that a protocol for f_E^g can be used to compute f_T^g. Alice and Bob need only to fix appropriately their inputs in the subtrees that where cut of from E. If an input bit belongs to a subtree T_j that was cut of in case (1), then Alice and Bob set their inputs in T_j to '0'. If T_j was cut of in case (2), then Alice and Bob set their inputs in T_j to '1'. Afterwards, they simulate the protocol for f_E^g. □

The tree T in the above proposition may not be a canonical representation of some function. However, transforming to the canonical representation will only decrease its depth, and thus strengthen our lower bound. Thus, by this Proposition and 1 we obtain 3 as a corollary.

4.7 General form of main theorem

The lower bounds we obtained apply to functions of the (restricted) form $f^{\wedge}$ and $f^{\vee}$. In this section we consider arbitrary two-party read-once functions, and prove 4, stated in the introduction. Theorems 1 and 2 are deduced from our main result 8. We also use 8 to deduce communication complexity lower bounds for two-party read-once functions.

Consider an AND/OR tree-circuit C in canonical form, and suppose that its leaf-set is partitioned into two sets $\mathcal{X}_C = \{x_1, \ldots, x_s\}$ and $\mathcal{Y}_C = \{y_1, \ldots, y_t\}$ (thus, f_C is a two-party read-once function). We show that C can be transformed to a tree T in standard form, such that Alice and Bob can decide the value of f_T using any protocol for f_C. (The reader may have expected $f_T^{\widehat{\wedge}}$ in the place of f_T. To avoid confusion we note that f_T will already be a two-party read-once function. In particular, for some tree T' with $d_{T'} = d_T - 1$ and $N_{T'} = N_T/2$, $f_T = f_{T'}^{\widehat{\wedge}}$.)

Lemma 29. *For any two-party read-once function f, there is a tree T in standard form, such that (1) $R_\delta(f_T) \leq R_\delta(f)$, (2) $N_T \geq D^{||}(f)/d(f)$, and (3) $d_T \leq d(f)$.*

Proof. We use notation from the paragraph before the statement of the lemma. The transformation of C proceeds in three stages.

In the first stage we collapse subtrees to single variables. For a node w let $A_w = \{u \in V_C \mid u \text{ is a child of } w \text{ and } L_{C_u} \subseteq \mathcal{X}_C\}$. Define B_w with $\mathcal{Y}$ in the place of $\mathcal{X}$. Let $W_{\mathcal{X}} = \{w \in V_C \mid L_{C_w} \nsubseteq \mathcal{X}_C \text{ and } A_w \neq \emptyset\}$. Define $W_{\mathcal{Y}}$ similarly. For each $w \in W_{\mathcal{X}}$, collapse $\{C_u \mid u \in A_w\}$ to a single variable x_w. That is, we remove all C_u with $u \in A_w$ from the tree, and add a new leaf x_w as a child of w. Similarly with $\mathcal{Y}$ in the place of $\mathcal{X}$ and B_w in the place of A_w. Name the resulting tree C_1. We claim that $R_\delta(f_C) = R_\delta(f_{C_1})$ and $D^{||}(f_C) = D^{||}(f_{C_1})$. It is easy to see that $R_\delta(f_C) \geq R_\delta(f_{C_1})$ and $D^{||}(f_C) \geq D^{||}(f_{C_1})$. Alice, for each $w \in W_{\mathcal{X}}$, can set each $x \in \mathcal{X}_{A_w}$ equal to x_w. Bob, for each $w \in W_{\mathcal{Y}}$, can set each $y \in \mathcal{Y}_{B_w}$ equal to y_w. After this preprocessing that requires no communication, they run a protocol for f_C. For the other direction, suppose $w \in W_{\mathcal{X}}$ is labeled by an AND gate. Alice sets x_w equal to $\bigwedge_{u \in A_w} f_{C_u}(\mathbf{x}_u)$ (for an OR gate, replace $\bigwedge$ with $\bigvee$). Bob acts similarly and afterwords they run a protocol for f_{C_1}. Clearly, $N_C \geq N_{C_1}$ and $d_C \geq d_{C_1}$. Notice also that in C_1 each node has at most one leaf in $\mathcal{X}_{C_1}$ and at most one in $\mathcal{Y}_{C_1}$ (where the partition for L_{C_1} is the obvious one).

In the second stage, we remove every leaf of C_1 that has a non-leaf sibling. If after these two stages some nodes are left with only one child, we collapse them with their unique child and label the new node with the gate of the child. Name

the resulting tree C_2. We have $R_\delta(f_{C_1}) \geq R_\delta(f_{C_2})$ and $D^{||}(f_{C_1}) \geq D^{||}(f_{C_2})$, since Alice and Bob can generate values ('1'/'0') for the truncated leaves according to the gate of the parent (AND/OR). Clearly, $d_{C_1} \geq d_{C_2}$. Observe also that $N_{C_2} \geq N_{C_1}/d_{C_1}$. This is because for every pair of leaves in C_1 that remain in C_2, there can be at most $2(d_{C_1} - 1)$ leaves that will be removed—one pair for each of the $d_{C_1} - 1$ nodes along the path to the root (see last sentence of previous paragraph).

For the final stage, let T be the tree-circuit that is otherwise identical to C_2, but every gate of C_2 has been replaced by a NAND gate. It follows from 7 that $f_T \equiv f_{C_2}$ or $f_T \equiv \neg f_{C_2}$. Thus, for the models of interest, the complexity of f_{C_2} is equal to that of f_T. Also, $N_T = N_{C_2}$ and $d_T = d_{C_2}$.

For part (2), observe that $D^{||}(f_{C_1}) \leq N_{C_1}$. Tracing the inequalities from each stage,

$$N_T = N_{C_2} \geq N_{C_1}/d_{C_1} \geq D^{||}(f_{C_1})/d_{C_1} = D^{||}(f)/d_{C_1} \geq D^{||}(f)/d(f).$$

Parts (1) and (3) are immediate. □

The tree-circuit T is in standard form, and 8 can be applied, yielding $R_\delta(f_T) \geq 4(2 - 4\sqrt{\delta}) \cdot N_T/8^{d_T}$. (For the constants involved, recall the parenthetic remark before the statement of the lemma.) Then, 4,

$$R_\delta(f) \geq (8 - 16\sqrt{\delta}) \cdot \frac{D^{||}(f)}{d(f) \cdot 8^{d(f)}},$$

follows from the lemma.

Bibliography

[1] Farid M. Ablayev. Lower bounds for one-way probabilistic communication complexity and their application to space complexity. *Theor. Comput. Sci.*, 157(2):139–159, 1996.

[2] Ziv Bar-Yossef, T. S. Jayram, Ravi Kumar, and D. Sivakumar. Information theory methods in communication complexity. In *IEEE Conference on Computational Complexity*, pages 93–102, 2002.

[3] Ziv Bar-Yossef, T. S. Jayram, Ravi Kumar, and D. Sivakumar. An information statistics approach to data stream and communication complexity. *J. Comput. Syst. Sci.*, 68(4):702–732, 2004.

[4] Amit Chakrabarti, Subhash Khot, and Xiaodong Sun. Near-optimal lower bounds on the multi-party communication complexity of set disjointness. In *IEEE Conference on Computational Complexity*, pages 107–117. IEEE Computer Society, 2003.

[5] Amit Chakrabarti, Yaoyun Shi, Anthony Wirth, and Andrew Yao. Informational complexity and the direct sum problem for simultaneous message complexity. In *In Proceedings of the 42nd Annual IEEE Symposium on Foundations of Computer Science*, pages 270–278, 2001.

[6] Richard Cleve, Wim van Dam, Michael Nielsen, and Alain Tapp. Quantum entanglement and the communication complexity of the inner product function. In *QCQC '98: Selected papers from the First NASA International Conference on Quantum Computing and Quantum Communications*, pages 61–74, London, UK, 1998. Springer-Verlag.

[7] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory.* Wiley-Interscience, 2006.

[8] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. A quantum algorithm for the hamiltonian nand tree. *Theory of Computing*, 4(1):169–190, 2008.

[9] Rafi Heiman, Ilan Newman, and Avi Wigderson. On read-once threshold formulae and their randomized decision tree complexity. *Theor. Comput. Sci.*, 107(1):63–76, 1993.

[10] T. S. Jayram, Swastik Kopparty, and Prasad Raghavendra. On the communication complexity of read-once AC^0 formulae. In *IEEE Conference on Computational Complexity*, pages 329–340, 2009.

[11] T. S. Jayram, Ravi Kumar, and D. Sivakumar. Two applications of information complexity. In *STOC*, pages 673–682. ACM, 2003.

[12] Bala Kalyanasundaram and Georg Schnitger. The probabilistic communication complexity of set intersection. *SIAM J. Discret. Math.*, 5(4):545–557, 1992.

[13] Ilan Kremer. Quantum communication. Master's thesis, Computer Science Department, Hebrew University, 1995.

[14] Eyal Kushilevitz and Noam Nisan. *Communication Complexity*. Cambridge University Press, New York, NY, USA, 2006.

[15] Troy Lee, Adi Shraibman, and Shengyu Zhang. Personal communication, 2009.

[16] A. A. Razborov. On the distributional complexity of disjointness. *Theor. Comput. Sci.*, 106(2):385–390, 1992.

[17] Michael Saks and Xiaodong Sun. Space lower bounds for distance approximation in the data stream model. In *STOC '02: Proceedings of the thirty-fourth annual ACM symposium on Theory of computing*, pages 360–369, New York, NY, USA, 2002. ACM.

[18] Michael Saks and Avi Wigderson. Probabilistic boolean decision trees and the complexity of evaluating game trees. In *SFCS '86: Proceedings of the 27th Annual Symposium on Foundations of Computer Science (sfcs 1986)*, pages 29–38, Washington, DC, USA, 1986. IEEE Computer Society.

[19] Andrew Chi-Chih Yao. Some complexity questions related to distributive computing(preliminary report). In *STOC '79: Proceedings of the eleventh annual ACM symposium on Theory of computing*, pages 209–213, New York, NY, USA, 1979. ACM.