Electronic Colloquium on Computational Complexity, Report No. 65 (2021)

One-way communication complexity and non-adaptive decision trees

Nikhil S. Mande
CWI, Amsterdam
nsm@cwi.nl

Swagato Sanyal
IIT Kharagpur
swagato@cse.iitkgp.ac.in

Abstract

We study the relationship between various one-way communication complexity measures of a composed function with the analogous decision tree complexity of the outer function. We consider two gadgets: the AND function on 2 inputs, and the Inner Product on a constant number of inputs. More generally, we show the following when the gadget is Inner Product on $2b$ input bits, denoted IP.

If f is a total Boolean function that depends on all of its inputs, then the bounded-error one-way quantum communication complexity of $f \circ \mathsf{IP}$ equals $\Omega(n(b-1))$.

If f is a *partial* Boolean function, then the deterministic one-way communication complexity of $f \circ \mathsf{IP}$ is at least $\Omega(b \cdot \mathsf{D}^{\rightarrow}_{\mathrm{dt}}(f))$, where $\mathsf{D}^{\rightarrow}_{\mathrm{dt}}(f)$ denotes the non-adaptive decision tree complexity of f.

To prove our quantum lower bound, we first show a lower bound on the VC-dimension of $f \circ \mathsf{IP}$. We then appeal to a result of Klauck [STOC'00], which immediately yields our quantum lower bound. Our deterministic lower bound relies on a combinatorial result due to Frankl and Tokushige [Comb.'99].

It is known due to a result of Montanaro and Osborne [arXiv'09] that the deterministic one-way communication complexity of $f \circ \mathsf{XOR}$ *equals* the non-adaptive parity decision tree complexity of f. In contrast, we show the following when the inner gadget is the AND function on 2 input bits.

There exists a function for which even the *randomized* non-adaptive AND decision tree complexity of f is exponentially large in the deterministic one-way communication complexity of $f \circ \mathsf{AND}$.

However, for symmetric functions f, the non-adaptive AND decision tree complexity of f is at most quadratic in the (even two-way) communication complexity of $f \circ \mathsf{AND}$.

In view of the first bullet, a lower bound on non-adaptive AND decision tree complexity of f *does not* lift to a lower bound on one-way communication complexity of $f \circ \mathsf{AND}$. The proof of the first bullet above uses the well-studied *Odd-Max-Bit* function. For the second bullet, we first observe a connection between the one-way communication complexity of f and the *Möbius sparsity* of f, and then use a known lower bound on the Möbius sparsity of symmetric functions. An upper bound on the non-adaptive AND decision tree complexity of symmetric functions follows implicitly from prior work on combinatorial group testing; for the sake of completeness, we include a proof of this result.

This paper includes and improves upon results from an earlier unpublished manuscript of one of the authors [San17].

N.S.M. is supported by the Dutch Research Council (NWO) through QuantERA project QuantAlgo 680-91-034.

S.S. is supported by an ISIRD Grant from Sponsored Research and Industrial Consultancy, IIT Kharagpur. Part of this work was done while S.S. was at NTU and CQT, and supported by the Singapore National Research Foundation under NRF RF Award No. NRF-NRFF2013-13.

ISSN 1433-8092

Contents

1 Introduction

Composed functions are important objects of study in analysis of Boolean functions and computational complexity. For Boolean functions $f : \{0,1\}^n \to \{0,1\}$ and $g : \{0,1\}^m \to \{0,1\}$, their composition $f \circ g : (\{0,1\}^m)^n \to \{0,1\}$ is defined as follows: $f \circ g(x_1, \ldots, x_n) := f(g(x_1), \ldots, g(x_n))$. In other words, $f \circ g$ is the function obtained by first computing g on n disjoint inputs of m bits each, and then computing f on the n resultant bits. Composed functions have been extensively looked at in the complexity theory literature, with respect to various complexity measures [BdW01, HLS07, Rei11, She12, She13, BT15, Tal13, Mon14, BK16, GJ16, AGJ+17, GLSS19, BB20].

Of particular interest to us is the case when g is a communication problem (also referred to as "gadget"). More precisely, let $g : \{0,1\}^b \times \{0,1\}^b \to \{0,1\}$ and $f : \{0,1\}^n \to \{0,1\}$ be Boolean functions. Consider the following communication problem: Alice has input $x = (x_1, \ldots, x_n)$ and Bob has input $y = (y_1 \ldots, y_n)$ where $x_i, y_i \in \{0,1\}^b$ for all $i \in [n]$. Their goal is to compute $f \circ g((x_1, y_1), \ldots, (x_n, y_n))$ using as little communication as possible. A natural protocol is the following: Alice and Bob jointly simulate an efficient query algorithm for f, using an optimal communication protocol for g to answer each query. Lifting theorems are statements that say this naive protocol is essentially optimal. Such theorems enable us to prove lower bounds on the rich model of communication complexity by proving feasibly easier-to-prove lower bounds in the query complexity (decision tree) model.

Various lifting theorems have been proved in the literature: lifting a query complexity measure to various one-sided zero communication complexity measures [GLM+16], lifting parallel decision tree complexity to round-constrained communication complexity [dRNV16], lifting deterministic query complexity to deterministic communication complexity [RM99, GPW18, CKLM19, WYY17], lifting DAG-like query complexity to DAG-like communication complexity [GGKS20], lifting randomized query complexity to randomized communication complexity [GPW20], lifting parity decision tree complexity to deterministic communication complexity using the XOR gadget [HHL18], lifting AND-decision tree complexity to deterministic communication complexity using the AND gadget [KLMY20], a deterministic lifting theorem for the Equality gadget [LM19], deterministic and randomized lifting theorems for gadgets with small discrepancy [CFK+21], etc.

In this work we are interested in the *one-way* communication complexity of composed functions. In this setting, a natural protocol is for Alice and Bob to simulate a *non-adaptive* decision tree for the outer function, using an optimal *one-way* communication protocol for the inner function. Thus, the one-way communication complexity of $f \circ g$ is at most the non-adaptive decision tree complexity of f times the one-way communication complexity of g.

Lifting theorems in the one-way model are less studied than in the two-way model. Montanaro and Osborne [MO09] observed that the deterministic one-way communication complexity of $f \circ \mathsf{XOR}$ *equals* the non-adaptive parity decision tree complexity of f. Thus, non-adaptive parity decision tree complexity lifts "perfectly" to deterministic communication complexity with the XOR gadget. Kannan et al. [KMSY18] showed that under uniformly distributed inputs, bounded-error non-adaptive parity decision tree complexity lifts to one-way bounded-error distributional communication complexity with the XOR gadget. Hosseini, Lovett and Yaroslavtsev [HLY19] showed that randomized non-adaptive parity decision tree complexity lifts to randomized communication complexity with the XOR gadget in the one-way broadcasting model with $\Theta(n)$ players.

We explore the tightness of the naive communication upper bound for two different choices of the gadget g: the Inner Product function, and the two-input AND function. For each choice of g, we compare the one-way communication complexity of $f \circ g$ with an appropriate type of non-adaptive decision tree complexity of f. Below, we motivate and state our results for each choice of the gadget.

1.1 Inner Product gadget

Let $\mathsf{Q}_{\mathrm{cc},\varepsilon}^{\rightarrow}(\cdot)$, $\mathsf{R}_{\mathrm{cc},\varepsilon}^{\rightarrow}(\cdot)$ and $\mathsf{D}_{\mathrm{cc}}^{\rightarrow}(\cdot)$ denote quantum ε-error, randomized ε-error and deterministic one-way communication complexity, respectively. When we allow the parties to share an arbitrary input-independent entangled state in the beginning of the protocol, denote the one-way quantum ε-error communication complexity by $\mathsf{Q}_{\mathrm{cc},\varepsilon}^{*,\rightarrow}(\cdot)$ (see Section 2.4 for formal definitions). Let $\mathsf{D}_{\mathrm{dt}}^{\rightarrow}(\cdot)$ denote deterministic non-adaptive decision tree complexity (see Section 2.3 for a formal definition). For an integer $b > 0$, let $\mathsf{IP} : \{0,1\}^b \times \{0,1\}^b \to \{0,1\}$ denote the *Inner Product Modulo 2* function, that outputs the parity of the bitwise AND of two b-bit input strings (see Definition 2.1). Let f denote the outer function, and f^{IP} denote the function $f \circ \mathsf{IP}$. Our first result shows that if f is a total function that depends on all of its input bits, the quantum (and hence, randomized) bounded-error one-way communication complexity of f^{IP} is $\Omega(n(b-1))$. Let $\mathbb{H}_{\mathsf{bin}}(\cdot)$ denote the binary entropy function.

Theorem 1.1. *Let $f : \{0,1\}^n \to \{0,1\}$ be a total Boolean function that depends on all its inputs (i.e., it is not a junta on a strict subset of its inputs), and let $\varepsilon \in (0,1/2)$. Let $\mathsf{IP} : \{0,1\}^b \times \{0,1\}^b \to \{0,1\}$ denote the Inner Product function on $2b$ input bits for $b \geq 1$. Then,*

$$\mathsf{Q}_{\mathrm{cc},\varepsilon}^{\rightarrow}(f^{\mathsf{IP}}) \geq (1 - \mathbb{H}_{\mathsf{bin}}(\varepsilon))n(b-1),$$
$$\mathsf{Q}_{\mathrm{cc},\varepsilon}^{*,\rightarrow}(f^{\mathsf{IP}}) \geq (1 - \mathbb{H}_{\mathsf{bin}}(\varepsilon))n(b-1)/2.$$

Remark 1.2. *In an earlier manuscript [San17], the second author proved a lower bound of $(1-\mathbb{H}_{\mathsf{bin}}(\varepsilon))n(b-1)$ on a weaker complexity measure, namely $\mathsf{R}_{\mathrm{cc},\varepsilon}^{\rightarrow}(F)$, via information-theoretic tools. Kundu [Kun17] subsequently observed that a quantum lower bound can also be obtained by additionally using Holevo's theorem.*

They also suggested to the second author via private communication that one might be able to recover these bounds using a result of Klauck [Kla00]. This is indeed the approach we take, and we thank them for suggesting this and pointing out the reference.

In order to prove Theorem 1.1, we appeal to a result of Klauck [Kla00, Theorem 3], who showed that the one-way ε-error quantum communication complexity of a function F is at least $(1 - \mathbb{H}_{\mathsf{bin}}(\varepsilon)) \cdot \mathsf{VC}(F)$, where $\mathsf{VC}(F)$ denotes the VC-dimension of F (see Definition 2.12). In the case when the parties can share an arbitrary entangled state in the beginning of a protocol, Klauck showed a lower bound of $(1 - \mathbb{H}_{\mathsf{bin}}(\varepsilon)) \cdot \mathsf{VC}(F)/2$. We exhibit a set of inputs that witnesses the fact that $\mathsf{VC}(f^{\mathsf{IP}}) \geq n(b-1)$.

Note that Theorem 1.1 is useful only when $b > 1$. Indeed, no non-trivial lifting statement is true for $b = 1$ when f is the AND function on n bits, since in this case, $f^{\mathsf{IP}} = \mathsf{AND}_{2n}$, whose one-way communication complexity is 1.

Our second result with the Inner Product gadget relates the deterministic one-way communication complexity of f^{IP} to the deterministic non-adaptive decision tree complexity of f, where f is an arbitrary *partial* Boolean function.

Theorem 1.3. *Let $\mathsf{S} \subseteq \{0,1\}^n$ be arbitrary, and $f : \mathsf{S} \to \{0,1\}$ be a* partial *Boolean function. Let $b \geq 4$. Then,*

$$\mathsf{D}_{\mathrm{cc}}^{\rightarrow}(f^{\mathsf{IP}}) = \Omega(b \cdot \mathsf{D}_{\mathrm{dt}}^{\rightarrow}(f)).$$

Given a protocol Π, our proof extracts a set of variables of cardinality linear in the complexity of Π, whose values always determine the value of f. The following claim which follows directly from the work of Frankl and Tokushige [FT99] is a crucial ingredient in our proof.

Theorem 1.4. *Let $q \geq 8$. Let $\mathcal{A} \subseteq [q]^n$ be such that for all $x^{(1)} = (x_1^{(1)}, \ldots, x_n^{(1)})$, $x^{(2)} = (x_1^{(2)}, \ldots, x_n^{(2)}) \in \mathcal{A}$, $|\{i \in [n] \mid x_i^{(1)} = x_i^{(2)}\}| \geq d$. Then, $|\mathcal{A}| < q^{n - \frac{d}{2}}$.*

We give the details of the derivation of Theorem 1.4 from the result of Frankl and Tokushige in Appendix C. Theorem 1.4 admits simple proofs when q is large compared to n. See [GMWW17] for a proof when q is a prime power, and $q \geq n$. Their proof is based on polynomials over finite fields. We give a different proof for all $q > (\frac{en}{d})^2$ in Appendix D.[1] However, such statements will only enable us to prove a lifting theorem for a gadget of size $b = \Omega(\log n)$. To prove Theorem 1.3 for constant-sized gadgets we need to set q to $O(1)$.

Remark 1.5. *An analogous lifting theorem for deterministic one-way protocols for total outer functions follows as a special case of both Theorem 1.1 and Theorem 1.3. However, the statement admits a simple and direct proof based on a fooling set argument.*

Theorem 1.1 and Theorem 1.3 give lower bounds even when the gadget is a constant-sized Inner Product function. It is worth mentioning here that prior works that consider lifting theorems with the Inner Product gadget [CKLM19, WYY17, CFK+21], albeit in the two-way model of communication complexity, require a super-constant gadget size.

1.2 AND gadget

Interactive communication complexity of functions of the form $f \circ \mathsf{AND}$ have gained a recent interest [KLMY20, Wu21]. In order to state and motivate our results regarding when the inner gadget is the 2-bit AND function, we first discuss some known results in the case when the inner gadget is the 2-bit XOR function.

Consider non-adaptive decision trees, where the trees are allowed to query arbitrary parities of the input variables. Denote the best cost of such a tree computing a Boolean function f, by $\mathsf{NAPDT}(f)$. An efficient non-adaptive parity decision tree for f can easily be simulated to obtain an efficient deterministic one-way communication protocol for $f \circ \mathsf{XOR}$. Thus, $\mathsf{D}_{\mathrm{cc}}^{\rightarrow}(f \circ \mathsf{XOR}) \leq \mathsf{NAPDT}(f)$. Montanaro and Osborne [MO09] observed that this inequality is, in fact, tight for all Boolean functions. More precisely,

[1]For every $\delta > 0$, the proof can be extended to work for all $q = \Omega(n/d)^{1+\delta}$.

Claim 1.6 ([MO09])**.** *For all Boolean functions $f : \{0,1\}^n \to \{0,1\}$,*

$$\mathsf{D}^{\to}_{\mathrm{cc}}(f \circ \mathsf{XOR}) = \mathsf{NAPDT}(f).$$

If the inner gadget were AND instead of XOR, then the natural analogous decision tree model to consider would be non-adaptive decision trees that have query access to arbitrary ANDs of subsets of inputs. Denote the best cost of such a tree computing a Boolean function f, by $\mathsf{NAADT}(f)$. Clearly, the one-way communication complexity of $f \circ \mathsf{AND}$ is bounded from above by $\mathsf{NAADT}(f)$, since a non-adaptive AND decision tree can be easily simulated to give a one-way communication protocol for $f \circ \mathsf{AND}$ of the same cost. Thus, $\mathsf{D}^{\to}_{\mathrm{cc}}(f \circ \mathsf{AND}) \leq \mathsf{NAADT}(f)$. On the other hand, one can show that $\mathsf{D}^{\to}_{\mathrm{cc}}(f \circ \mathsf{AND}) \geq \log(\mathsf{NAADT}(f))$ (see Claim 2.16). Thus we have

$$\log(\mathsf{NAADT}(f)) \leq \mathsf{D}^{\to}_{\mathrm{cc}}(f \circ \mathsf{AND}) \leq \mathsf{NAADT}(f). \tag{1}$$

We explore if an analogous statement to Claim 1.6 holds true if the inner function were AND instead of XOR. That is, is the second inequality in Equation (1) always tight?

We give a negative answer in a very strong sense and exhibit a function for which the first inequality is tight (up to an additive constant). We show that there is an exponential separation between these measures even if one allows randomization in the decision trees. We use $\mathsf{RNAADT}(f)$ to denote the randomized non-adaptive AND decision tree complexity of f.

Theorem 1.7. *There exists a function $f : \{0,1\}^n \to \{0,1\}$ such that*

$$\mathsf{RNAADT}(f) = \Omega(2^{\mathsf{D}^{\to}_{\mathrm{cc}}(f \circ \mathsf{AND})}).$$

The function f we use to witness the bound in Theorem 1.7 is a modification of the well-studied *Odd-Max-Bit* function, which we denote OMB_n. This function outputs 1 if and only if the maximum index of the input string that contains a 0, is odd (see Definition 2.2). A $\lceil \log(n+1) \rceil$-cost one-way communication protocol is easy to show, since Alice can simply send Bob the maximum index where her input is 0 (if it exists), and Bob can use this along with his input to conclude the parity of the maximum index where the bitwise AND of their inputs is 0. For a lower bound of $\Omega(n)$ on $\mathsf{RNAADT}(\mathsf{OMB}_n)$, we exhibit a hard distribution under which no low-cost deterministic non-adaptive AND decision tree can compute OMB_n with high accuracy.

Theorem 1.7 implies that, in contrast to the lifting theorem with the XOR gadget (Claim 1.6), the measure of non-adaptive AND decision tree complexity *does not* lift to a one-way communication lower bound for $f \circ \mathsf{AND}$. However we show that a statement analogous to Claim 1.6 does hold true for symmetric functions f, albeit with a quadratic factor, even when the measure is two-way communication complexity, denoted $\mathsf{D}_{\mathrm{cc}}(\cdot)$.

Theorem 1.8. *Let $f : \{0,1\}^n \to \{0,1\}$ be a symmetric function. Then,*

$$\mathsf{NAADT}(f) = O(\mathsf{D}_{\mathrm{cc}}(f \circ \mathsf{AND})^2).$$

In fact we prove a stronger bound in which $\mathsf{D}_{\mathrm{cc}}(f \circ \mathsf{AND})$ above is replaced by $\log \mathrm{rank}(M_{f\circ\mathsf{AND}})$, where $M_{f\circ\mathsf{AND}}$ denotes the communication matrix of $f \circ \mathsf{AND}$ (see Section 2.4). That is, we show for symmetric functions f that

$$\mathsf{NAADT}(f) = O(\log^2 \mathrm{rank}(M_{f\circ\mathsf{AND}})). \tag{2}$$

Since it is well known (see Equation (7)) that the communication complexity of a function is at least as large as the logarithm of the rank of its communication matrix, this implies Theorem 1.8. Among other things, Buhrman and de Wolf [BdW01] observed that the log-rank conjecture holds for symmetric functions composed with AND. In particular, they showed that if f is symmetric, then $\mathsf{D}_{\mathrm{cc}}(f \circ \mathsf{AND}) = O(\log \mathrm{rank}(M_{f\circ\mathsf{AND}}))$. This was also observed recently by Wu [Wu21], who also showed other results regarding the communication complexity of AND functions in connection with the log-rank conjecture. While we have a quadratically worse dependence in the RHS of Equation (2), our upper bound is on a complexity measure that can be exponentially larger than communication complexity in general, as demonstrated by Theorem 1.7.

Buhrman and de Wolf showed a lower bound on $\log \operatorname{rank}(M_{f\circ \mathsf{AND}})$ for symmetric functions f. An upper bound on $\mathsf{NAADT}(f)$ implicitly follows from prior work on group testing [DR83], but we provide a self-contained probabilistic proof for completeness. Combining these two results yields Equation (2), and hence Theorem 1.8.

Suitable analogues of Theorem 1.7 and Theorem 1.8 can be easily seen to hold when the inner gadget is OR instead of AND. In this case, the relevant decision tree model is non-adaptive OR decision trees. Interestingly, these decision trees are studied in the seemingly different context of *non-adaptive group testing algorithms*. Non-adaptive group testing is an active area of research (see, for, example, [CH08] and the references therein), and has additionally gained significant interest of late in view of the ongoing pandemic (see, for example, [ŽLG21]).

1.3 Organization

We introduce the necessary preliminaries in Section 2. In Section 3 we prove our results regarding the Inner Product gadget (Theorem 1.1 and Theorem 1.3, respectively). In Section 4 we prove our results regarding the AND gadget (Theorem 1.7 and Theorem 1.8). In Section A we show some results regarding the Addressing function, and we provide missing proofs from the main text in the remaining appendices.

2 Preliminaries

2.1 Notation

All logarithms in this paper are taken base 2. We use the notation $[n]$ to denote the set $\{1,\dots,n\}$. We often identify subsets of $[n]$ with their corresponding characteristic vectors in $\{0,1\}^n$. The view we take will be clear from context.

We now introduce function composition. Let $f:\{0,1\}^n\to\{0,1\}$ be a Boolean function and let $g:\{0,1\}^j\times\{0,1\}^j\to\{-1,1\}$ be a communication problem. Then $F=f\circ g:\{0,1\}^{nj}\times\{0,1\}^{nj}\to\{0,1\}$ denotes the function corresponding to the communication problem in which Alice is given input $x=(x_1,\dots,x_n)\in\{0,1\}^{nj}$, Bob is given input $y=(y_1,\dots,y_n)\in\{0,1\}^{nj}$, and their goal is to compute $F(X,Y)=f(g(x_1,y_1),\dots,g(x_n,y_n))$. We first define the Inner Product Modulo 2 function on $2b$ input bits, denoted IP (we drop the dependence of IP on b for convenience; the value of b will be clear from context).

Definition 2.1 (Inner Product Modulo 2)**.** *For an integer $b>0$, define the* Inner Product Modulo 2 *function, denoted* $\mathsf{IP}:\{0,1\}^b\times\{0,1\}^b\to\{0,1\}$ *by*

$$\mathsf{IP}(x_1,\dots,x_b,y_1,\dots,y_b)=\oplus_{i\in[b]}(\mathsf{AND}(x_i,y_i)).$$

Define $f^{\mathsf{IP}}=f\circ\mathsf{IP}$. If f is a partial function, so is f^{IP}.

Definition 2.2 (Odd-Max-Bit)**.** *Define the* Odd-Max-Bit *function, denoted* $\mathsf{OMB}_n:\{0,1\}^n\to\{0,1\}$*, by*

$$\mathsf{OMB}_n(x)=\begin{cases}1 & \text{if } \max\{i\in[n]:x_i=0\} \text{ is odd}\\ 0 & \text{otherwise.}\end{cases}\tag{3}$$

Define $\mathsf{OMB}_n(1^n)=0$.

Remark 2.3. *In the literature,* OMB_n *is typically defined with a 1 in the max of Equation (3) instead of 0. That function behaves very differently from our* OMB_n*. For example, it is known that even the* weakly unbounded-error *communication complexity of* $\mathsf{OMB}_n\circ\mathsf{AND}$ *(under the standard definition of* OMB_n*) is polynomially large in n [BVW07]. In contrast, it is easy to show that even the deterministic one-way communication complexity of* $\mathsf{OMB}_n\circ\mathsf{AND}$ *equals* $\lceil\log(n+1)\rceil$ *with our definition (see Theorem 4.4).*

Definition 2.4 (Binary entropy)**.** *For $p \in (0,1)$, the binary entropy of p, $\mathbb{H}_{\mathsf{bin}}(p)$, is defined to be the Shannon entropy of a random variable taking two distinct values with probabilities p and $1-p$.*

$$\mathbb{H}_{\mathsf{bin}}(p) := p \log \frac{1}{p} + (1-p) \log \frac{1}{1-p}.$$

In particular, if $\varepsilon = 1/2 - \Omega(1)$, then $1 - \mathbb{H}_{\mathsf{bin}}(\varepsilon) = \Omega(1)$. Let $\mathsf{S} \subseteq \{0,1\}^n$ be an arbitrary subset of the Boolean hypercube, and let $f : \mathsf{S} \to \{0,1\}$ be a partial Boolean function. If $\mathsf{S} = \{0,1\}^n$, then f is said to be a total Boolean function. When not explicitly mentioned otherwise, we assume Boolean functions to be total.

2.2 Möbius expansion of Boolean functions

Every Boolean function $f : \{0,1\}^n \to \{0,1\}$ has a unique expansion as

$$f = \sum_{S \subseteq [n]} \widetilde{f}(S) \mathsf{AND}_S, \tag{4}$$

where AND_S denotes the AND of the input variables in S and each $\widetilde{f}(S)$ is a real number. We refer to the functions AND_S as *monomials*, the expansion in Equation (4) as the *Möbius expansion* of f, and the real coefficients $\widetilde{f}(S)$ as the Möbius coefficients of f. It is known [Bei93] that the Möbius coefficients can be expressed as

$$\widetilde{f}(S) = \sum_{X \subseteq S} (-1)^{|S \setminus X|} f(X). \tag{5}$$

Define the *Möbius sparsity* of f, denoted $\mathrm{spar}(f)$, to be the number of Möbius coefficients of f that are non-zero. That is,

$$\mathrm{spar}(f) := \left| \left\{ S \subseteq [n] : \widetilde{f}(S) \neq 0 \right\} \right|. \tag{6}$$

We require the following definition, which captures the number of realizable 0/1-patterns of the monomials the Möbius support of f.

Definition 2.5 (Möbius pattern complexity)**.** *Let $f : \{0,1\}^n \to \{0,1\}$ be a Boolean function, and let $f = \sum_{S \in \mathcal{S}} \widetilde{f}(S) \mathsf{AND}_S$ be its Möbius expansion. For an input $x \in \{0,1\}^n$, define the* pattern of $f(x)$ *to be $(\mathsf{AND}_S(x))_{S \in \mathcal{S}}$. Define the* Möbius pattern complexity *of f, denoted $\mathsf{Pat}^{\mathsf{M}}(f)$, by*

$$\mathsf{Pat}^{\mathsf{M}}(f) = \left| \left\{ P \in \{0,1\}^{\mathcal{S}} : P = (\mathsf{AND}_S(x))_{S \in \mathcal{S}} \text{ for some } x \in \{0,1\}^n \right\} \right|.$$

When clear from context, we refer to the Möbius pattern complexity of f just as the pattern complexity of f.

2.3 Decision trees and their variants

For a partial Boolean function $f : \mathsf{S} \to \{0,1\}$, the deterministic non-adaptive query complexity (alternatively the non-adaptive decision tree complexity) $\mathsf{D}^{\rightarrow}_{\mathrm{dt}}(f)$ is the minimum integer k such that the following is true: there exist k indices $i_1, \ldots, i_k \in [n]$, such that for every Boolean assignment $a_{i_1}, \ldots, a_{i_k}$ to the input variables $x_{i_1}, \ldots, x_{i_k}$, f is constant on $\mathsf{S} \cap \{x \in \{0,1\}^n \mid \forall j = 1, \ldots, k, x_{i_j} = a_{i_j}\}$. It is easy to see that if f is a total function that depends on all input variables, then $\mathsf{D}^{\rightarrow}_{\mathrm{dt}}(f) = n$.

Definition 2.6 (Non-adaptive parity decision tree complexity)**.** *Define the* non-adaptive parity decision tree complexity *of $f : \{0,1\}^n \to \{0,1\}$, denoted by $\mathsf{NAPDT}(f)$, to be the minimum number of parities such that f can be expressed as a function of these parities. In other words, the non-adaptive parity decision tree complexity of f equals the minimal number k for which there exists $\mathcal{S} = \{\{S_1, \ldots, S_k\} : S_i \subseteq [n]$ for all $i \in [k]\}$ such that the function value $f(x)$ is determined by the values $\{\oplus_{j \in S_i} x_j : i \in [k]\}$ for all $x \in \{0,1\}^n$.*

Definition 2.7 (Non-adaptive AND decision tree complexity)**.** *Define the* non-adaptive AND decision tree complexity *of $f : \{0,1\}^n \to \{0,1\}$, denoted by $\mathsf{NAADT}(f)$, to be the minimum number of monomials such that f can be expressed as a function of these monomials. In other words, the non-adaptive AND decision tree complexity of f equals the minimal number k for which there exists $\mathcal{S} = \{\{S_1, \ldots, S_k\} : S_i \subseteq [n]$ for all $i \in [k]\}$ such that the function value $f(x)$ is determined by the values $\{\mathsf{AND}_{S_i}(x) : i \in [k]\}$ for all $x \in \{0,1\}^n$. We refer to such a set $\mathcal{S}$ as a* NAADT basis *for f.*

We also require a randomized variant of non-adaptive AND decision trees.

Definition 2.8 (Randomized non-adaptive AND decision tree complexity)**.** *A randomized non-adaptive AND decision tree T computing f is a distribution over non-adaptive AND decision trees with the property that $\Pr[T(x) = f(x) \geq 2/3]$ for all $x \in \{0,1\}^n$. The cost of T is the maximum depth of a non-adaptive AND decision tree in its support. Define the* randomized non-adaptive AND decision tree complexity *of $f : \{0,1\}^n \to \{0,1\}$, denoted by $\mathsf{RNAADT}(f)$, to be the minimum cost of a randomized non-adaptive AND decision tree that computes f.*

We first note some simple observations about the non-adaptive AND decision tree complexity of Boolean functions.

Claim 2.9. *Let $f : \{0,1\}^n \to \{0,1\}$ be a Boolean function and let $\mathcal{S} = \{S_1, \ldots, S_k\}$ be a NAADT basis for f. Then, every monomial in the Möbius support of f equals $\prod_{i \in T} \mathsf{AND}_{S_i}$, for some $T \subseteq [k]$.*

Proof. Since $\mathcal{S}$ is a NAADT basis for f, the values of $\{\mathsf{AND}_{S_i} : i \in [k]\}$ determine the value of f. That is, we can express f as

$$f = \sum_{T \subseteq [k]} b_T \prod_{i \in T} \mathsf{AND}_{S_i} \prod_{j \notin T} (1 - \mathsf{AND}_{S_j}),$$

for some values of $b_T \in \{0,1\}$. If a particular pattern of $\{\mathsf{AND}_{S_i} : i \in [k]\}$ is not attainable, we set $b_T = 0$ for the corresponding subset. Expanding this expression only yields monomials that are products of AND_{S_i}'s from $\mathcal{S}$. The claim now follows since the Möbius expansion of a Boolean function is unique. □

Claim 2.10. *Let $f : \{0,1\}^n \to \{0,1\}$ be a Boolean function with Möbius sparsity r. Then,*

$$\log r \leq \mathsf{NAADT}(f) \leq r.$$

Proof. The upper bound $\mathsf{NAADT}(f) \leq r$ follows from the fact that knowing the values of all ANDs in the Möbius support of f immediately yields the value of f by plugging these values in the Möbius expansion of f. That is, the Möbius support of f acts as a NAADT basis for f.

For the lower bound, let $\mathsf{NAADT}(f) = k$, and let $\mathcal{S} = \{S_1, \ldots, S_k\}$ be a NAADT basis for f. Claim 2.9 implies that every monomial in the Möbius expansion of f is a product of some of these AND_{S_i}'s. Thus, the Möbius sparsity of f is at most 2^k, yielding the required lower bound. □

Every Boolean function $f : \{-1,1\}^n \to \mathbb{R}$ can be uniquely written as $f = \sum_{S \subseteq [n]} \widehat{f}(S)(-1)^{\oplus_{j \in S} x_j}$. This representation is called the *Fourier expansion* of f and the real values $\widehat{f}(S)$ are called the Fourier coefficients of f. The Fourier sparsity of f is defined to be number of non-zero Fourier coefficients of f.

Sanyal [San19] showed the following relationship between non-adaptive parity decision complexity of a Boolean function and its Fourier sparsity.

Theorem 2.11 ([San19])**.** *Let $f : \{-1,1\}^n \to \{-1,1\}$ be a Boolean function with Fourier sparsity r. Then,*

$$\mathsf{NAPDT}(f) = O(\sqrt{r} \log r).$$

This theorem is tight up to the logarithmic factor, witnessed by the Addressing function.

2.4 Communication complexity

The standard model of two-party communication complexity was introduced by Yao [Yao79]. In this model, there are two parties, say Alice and Bob, each with inputs $x, y \in \{0,1\}^n$. They wish to jointly compute a function $F(x, y)$ of their inputs for some function $F : \mathcal{U} \to \{0,1\}$ that is known to them, where $\mathcal{U}$ is a subset of $\{0,1\}^n \times \{0,1\}^n$. They use a communication protocol agreed upon in advance. The cost of the protocol is the number of bits exchanged in the worst case (over all inputs). Alice and Bob are required to output the correct answer for all inputs $(x, y) \in \mathcal{U}$. The communication complexity of F is the best cost of a protocol that computes F, and we denote it by $\mathsf{D}_{\mathrm{cc}}(F)$. See, for example, [KN97], for an introduction to communication complexity.

In a deterministic one-way communication protocol, Alice sends a message $m(x)$ to Bob. Then Bob outputs a bit depending on $m(x)$ and y. The complexity of the protocol is the maximum number of bits a message contains for any input x to Alice. In a randomized one-way protocol, the parties share some common random bits $\mathcal{R}$. Alice's message is a function of x and $\mathcal{R}$. Bob's output is a function of $m(x), y$ and $\mathcal{R}$. The protocol Π is said to compute F with error $\varepsilon \in (0, 1/2)$ if for every $(x, y) \in \mathcal{U}$, the probability over $\mathcal{R}$ of the event that Bob's output equals $F(x, y)$ is at least $1 - \varepsilon$. The cost of the protocol is the maximum number of bits contained in Alice's message for any x and $\mathcal{R}$. In the one-way quantum model, Alice sends Bob a quantum message, after which Bob performs a projective measurement and outputs the measurement outcome. Depending on the model of interest, Alice and Bob may or may not share an arbitrary input-independent entangled state for free. As in the randomized setting, a protocol Π computes F with error ε if $\Pr[\Pi(x, y) \neq f(x, y)] \leq \varepsilon$ for all $(x, y) \in \mathcal{U}$.

The deterministic (ε-error randomized, ε-error quantum, ε-error quantum with entanglement, respectively) one-way communication complexity of F, denoted by $\mathsf{D}_{\mathrm{cc}}^{\rightarrow}(\cdot)$ ($\mathsf{R}_{\mathrm{cc},\varepsilon}^{\rightarrow}(\cdot)$, $\mathsf{Q}_{\mathrm{cc},\varepsilon}^{\rightarrow}(\cdot)$, $\mathsf{Q}_{\mathrm{cc},\varepsilon}^{*,\rightarrow}(\cdot)$, respectively), is the minimum cost of any deterministic (ε-error randomized, ε-error quantum, ε-error quantum with entanglement, respectively) one-way communication protocol for F.

Total functions F whose domain is $\{0,1\}^n \times \{0,1\}^n$ induce a communication matrix M_F whose rows and columns are indexed by strings in $\{0,1\}^n$, and the (x, y)'th entry equals $F(x, y)$. It is well known (see, for instance, [KN97]) that

$$\log \operatorname{rank}(M_F) \leq \mathsf{D}_{\mathrm{cc}}(F) \leq \operatorname{rank}(M_F), \tag{7}$$

where $\operatorname{rank}(\cdot)$ denotes real rank. One of the most famous conjectures in communication complexity is the log-rank conjecture, due to Lovász and Saks [LS88], that proposes that the communication complexity of any Boolean function is polylogarithmic in its rank, i.e. the first inequality in Equation (7) is always tight up to a polynomial dependence.

Buhrman and de Wolf [BdW01] observed that the Möbius sparsity of a Boolean function f equals the rank of the communication matrix of $f \circ \mathsf{AND}$. In view of the first inequality in Equation (7), this yields

$$\mathsf{D}_{\mathrm{cc}}(f \circ \mathsf{AND}) \geq \log(\operatorname{spar}(f)). \tag{8}$$

We require the definition of the Vapnik-Chervonenkis (VC) dimension [VC71].

Definition 2.12 (VC-dimension). *Consider a function $F : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$. A subset of columns C of M_F is said to be* shattered *if all of the $2^{|C|}$ patterns of 0's and 1's are attained by some row of M_F when restricted to the columns C. The* VC-dimension *of a function $F : \{0,1\}^n \times \{0,1\}^n$ is the maximum size of a shattered subset of columns of M_F.*

Klauck [Kla00] showed that the one-way quantum communication complexity of a function F is bounded below by the VC-dimension of F.

Theorem 2.13 ([Kla00, Theorem 3]). *Let $F : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ be a Boolean function. Then,*

$$\mathsf{Q}_{\mathrm{cc},\varepsilon}^{\rightarrow}(F) \geq (1 - \mathbb{H}_{\mathsf{bin}}(\varepsilon))\mathsf{VC}(f),$$
$$\mathsf{Q}_{\mathrm{cc},\varepsilon}^{*,\rightarrow}(F) \geq (1 - \mathbb{H}_{\mathsf{bin}}(\varepsilon))\mathsf{VC}(F)/2.$$

2.5 Pattern complexity and one-way communication complexity

In this section we observe that the logarithm of the pattern complexity, $\mathsf{Pat}^{\mathsf{M}}(f)$, of a Boolean function f equals the deterministic one-way communication complexity of $f \circ \mathsf{AND}$. We also give bounds on $\mathsf{NAADT}(f)$ in terms of $\mathsf{Pat}^{\mathsf{M}}(f)$. As a consequence we also show that $\mathsf{D}_{\mathsf{cc}}^{\rightarrow}(f \circ \mathsf{AND}) \geq \log(\mathsf{NAADT}(f))$.

Claim 2.14. *Let $f : \{0,1\}^n \to \{0,1\}$ be a Boolean function. Then,*

$$\mathsf{D}_{\mathsf{cc}}^{\rightarrow}(f \circ \mathsf{AND}) = \lceil \log(\mathsf{Pat}^{\mathsf{M}}(f)) \rceil.$$

It is well known that the one-way communication complexity of a function equals the logarithm of the number of distinct rows in its communication matrix. It is also not hard to show that the number of distinct rows in the communication matrix of $f \circ \mathsf{AND}$ equals the pattern complexity of f. Together these prove Claim 2.14. For completeness we provide a self-contained proof in Section E.

Next we show that the pattern complexity of f is bounded below by the non-adaptive AND decision tree complexity of f.

Claim 2.15. *Let $f : \{0,1\}^n \to \{0,1\}$ be a Boolean function. Then,*

$$\mathsf{Pat}^{\mathsf{M}}(f) \geq \mathsf{NAADT}(f).$$

Proof. Let $f = \sum_{S \subseteq [n]} \widetilde{f}(S)\mathsf{AND}_S$ and suppose $\mathsf{NAADT}(f) = k$. Let $\mathcal{S} = \{S_1, \ldots, S_k\}$ be a NAADT basis for f where $S_j \subseteq [n]$ for all $j \in [k]$. For each set $S_i \in \mathcal{S}$ we define the following string $X_i \in \{0,1\}^n$:

$$X_i(\ell) = \begin{cases} 1 & \ell \in S_i \\ 0 & \text{otherwise.} \end{cases}$$

Consider two indices $i \neq j \in [k]$. By definition $\mathsf{AND}_{S_i}(X_i) = 1$ and $\mathsf{AND}_{S_j}(X_j) = 1$. If $S_i \subsetneq S_j$, then $\mathsf{AND}_{S_j}(X_i) = 0$. Similarly if $S_j \subsetneq S_i$, then $\mathsf{AND}_{S_i}(X_j) = 0$. Thus, we have

$$(\mathsf{AND}_{S_i}(X_i), \mathsf{AND}_{S_j}(X_i)) \neq (\mathsf{AND}_{S_i}(X_j), \mathsf{AND}_{S_j}(X_j)).$$

Hence each of the strings X_i induces a different pattern for f, concluding the proof since the number of strings chosen equals $\mathsf{NAADT}(f)$. ☐

Combining Claim 2.14 and Claim 2.15, we have the following claim.

Claim 2.16. *Let $f : \{0,1\}^n \to \{0,1\}$ be a Boolean function. Then,*

$$\lceil \log(\mathsf{NAADT}(f)) \rceil \leq \mathsf{D}_{\mathsf{cc}}^{\rightarrow}(f \circ \mathsf{AND}) \leq \mathsf{NAADT}(f).$$

Proof. For the upper bound on $\mathsf{D}_{\mathsf{cc}}^{\rightarrow}(f \circ \mathsf{AND})$, let $\mathcal{S} = \{S_1, \ldots, S_k\}$ be a NAADT basis for f. By Claim 2.9, every monomial in the Möbius support of f is a product of some of these AND_{S_i}'s. Since there are at most 2^k possible values for $\{\mathsf{AND}_{S_i}(x) : i \in [k]\}$ and since these completely determine the pattern of $f(x)$ for any given $x \in \{0,1\}^n$, we have

$$\mathsf{Pat}^{\mathsf{M}}(f) \leq 2^{\mathsf{NAADT}(f)},$$

which proves the required upper bound in view of Claim 2.14.

The lower bound follows from Claim 2.14 and Claim 2.15 since we have

$$\mathsf{D}_{\mathsf{cc}}^{\rightarrow}(f \circ \mathsf{AND}) = \lceil \log(\mathsf{Pat}^{\mathsf{M}}(f)) \rceil \geq \lceil \log(\mathsf{NAADT}(f)) \rceil.$$

☐

3 Composition with Inner Product

In this section we prove Theorem 1.1 and Theorem 1.3, which are our results regarding the quantum and deterministic one-way communication complexities of functions composed with a small Inner Product gadget.

3.1 Quantum complexity

In this section, we prove Theorem 1.1 which gives a lower bound on the quantum one-way communication complexity of $f \circ \mathsf{IP}$ for total functions f that depend on all its inputs.

Proof of Theorem 1.1. By Theorem 2.13, it suffices to show that $\mathsf{VC}(f^{\mathsf{IP}}) \geq n(b-1)$. Since f is a function that depends on all its input variables, the following holds. For each index $i \in [n]$, there exist inputs

$$\begin{aligned} z^{(i,0)} &= z_1^{(i)}, \ldots, z_{i-1}^{(i)}, 0, z_{i+1}^{(i)}, \ldots, z_n^{(i)}, \\ z^{(i,1)} &= z_1^{(i)}, \ldots, z_{i-1}^{(i)}, 1, z_{i+1}^{(i)}, \ldots, z_n^{(i)} \end{aligned}$$

such that $f(z^{(i,0)}) = b_i$ and $f(z^{(i,1)}) = 1 - b_i$. That is, $z^{(i,0)}$ and $z^{(i,1)}$ have different function values, but differ only on the i'th bit.

For each $i \in [n]$ and $j \in \{2, 3, \ldots, b\}$, define a string $y^{(i,j)} \in \{0,1\}^{nb}$ as follows. For all $k \in [n]$ and $\ell \in [b]$,

$$y_{k,\ell}^{(i,j)} = \begin{cases} z_k^{(i)} & \text{if } k \neq i \text{ and } \ell = 1 \\ 1 & \text{if } k = i \text{ and } \ell = j \\ 0 & \text{otherwise.} \end{cases}$$

That is, for $k \neq i$, the k'th block of $y^{(i,j)}$ is $(z_k^{(i)}, 0^{b-1})$, and the i'th block of $y^{(i,j)}$ is $(0^{i-1}, 1, 0^{b-i})$. Consider the set of $n(b-1)$-many columns of $M_{f^{\mathsf{IP}}}$, one for each $y^{(i,j)}$. We now show that this set of columns is shattered. Consider an arbitrary string

$$c = c_{1,2}, \ldots, c_{1,b}, \ldots, c_{n,2}, \ldots, c_{n,b} \in \{0,1\}^{n(b-1)}.$$

We show below the existence of a row that yields this string on restriction to the columns described above. Define a string $x \in \{0,1\}^{nb}$ as follows. For all $i \in [n]$ and $j \in [b]$,

$$\begin{aligned} x_{i,1} &= 1, \\ x_{i,j} &= \begin{cases} c_{i,j} & \text{if } b_i = 0 \\ 1 - c_{i,j} & \text{if } b_i = 1. \end{cases} \end{aligned}$$

That is, the first element of each block of x is 1, and the remaining part of any block, say the i'th block, equals either the string $c_{i,2}, \ldots, c_{i,b}$ or its bitwise negation, depending on the value of b_i.

To complete the proof, we claim that the row of $M_{f^{\mathsf{IP}}}$ corresponding to this string x equals the string c when restricted to the columns $\{y^{(i,j)}\}_{i \in [n], j \in \{2,3,\ldots,b\}}$. To see this, fix $i \in [n]$ and $j \in \{2, 3, \ldots, b\}$ and consider $M_{f^{\mathsf{IP}}}(x, y^{(i,j)})$. Next, for each $k \in [n]$ with $k \neq i$, the inner product of the k'th block of x with the k'th block of y equals $z_k^{(i)}$, since $x_{k,1} = 1$ and the first element of the k'th block of $y^{(i,j)}$ equals $z_k^{(i)}$, and all other elements of the block are 0 by definition. In the i'th block of $y^{(i,j)}$, only the j'th element is non-zero, and equals 1 by definition. Moreover, $x_{i,j} = c_{i,j}$ if $b_i = 0$, and equals $1 - c_{i,j}$ otherwise. Hence, the inner products of the i'th blocks of x and $y^{(i,j)}$ equals $c_{i,j}$ if $b_i = 0$, and equals $1 - c_{i,j}$ otherwise. Thus, the string obtained on taking the block-wise inner product of x and $y^{(i,j)}$ equals

$$\begin{aligned} z_1^{(i)}, \ldots, z_{i-1}^{(i)}, c_{i,j}, z_{i+1}^{(i)}, \ldots, z_n^{(i)} \qquad & \text{if } b_i = 0 \\ z_1^{(i)}, \ldots, z_{i-1}^{(i)}, 1 - c_{i,j}, z_{i+1}^{(i)}, \ldots, z_n^{(i)} \qquad & \text{if } b_i = 1. \end{aligned}$$

By our definitions of $z^{(i,0)}, z^{(i,1)}$ and b_i for each $i \in [n]$, it follows that the value of f when applied to either of these inputs equals $c_{i,j}$. This concludes the proof. $\square$

3.2 Deterministic complexity

In this section we prove Theorem 1.3, which gives a lower bound on the deterministic one-way communication complexity of $f \circ \mathsf{IP}$ for even partial functions f. A crucial ingredient of our proof is Theorem 1.4. We derive Theorem 1.4 from a result of Frankl and Tokushige [FT99] in Appendix C. Now we proceed to the proof of Theorem 1.3.

Proof of Theorem 1.3. Let $q := 2^b - 1$ and let Π be an optimal one-way deterministic protocol for f^{IP} of complexity $\mathsf{D}_{\mathrm{cc}}^{\rightarrow}(f^{\mathsf{IP}}) =: c \log q$. Π induces a partition of $\{0,1\}^{nb}$ into at most q^c parts; each part corresponds to a distinct message. There are $(2^b-1)^n = q^n$ inputs $(x_1, \ldots, x_n)$ to Alice such that for each i, $x_i \neq 0^b$. Let $\mathcal{Z}$ be the set of those inputs. Identify $\mathcal{Z}$ with $[q]^n$. By the pigeon-hole principle there exists one part P in the partition induced by Π that contains at least q^{n-c} strings in $\mathcal{Z}$. Theorem 1.4 (which is applicable since the assumption $b \geq 4$ implies that $q \geq 8$) implies that there are two strings $x^{(1)} = (x_1^{(1)}, \ldots, x_n^{(1)}), x^{(2)} = (x_1^{(2)}, \ldots, x_n^{(2)}) \in \mathsf{P} \cap \mathcal{Z}$ such that $|\{i \in [n]\} \mid x_i^{(1)} = x_i^{(2)}\}| < 2c$. Let $\mathcal{I} := \{i \in [n]\} \mid x_i^{(1)} = x_i^{(2)}\}$. Let $z = (z_1, \ldots, z_n)$ denote a generic input to f. We claim that for each Boolean assignment $(a_i)_{i \in \mathcal{I}}$ to the variables in $\mathcal{I}$, f is constant on $\mathsf{S} \cap \{z : \forall i \in \mathcal{I}, z_i = a_i\}$. This will prove the theorem, since querying the variables $\{z_i \mid i \in \mathcal{I}\}$ determines f; thus $\mathsf{D}_{\mathrm{dt}}^{\rightarrow}(f) \leq |\mathcal{I}| < 2c$. Towards a contradiction, assume that there exist $z^{(1)}, z^{(2)} \in \mathsf{S} \cap \{z : \forall i \in \mathcal{I}, z_i = a_i\}$ such that $f(z^{(1)}) \neq f(z^{(2)})$. We will construct a string $y = (y_1, \ldots, y_n) \in \{0,1\}^{nb}$ in the following way:

$i \in \mathcal{I}$: Choose y_i such that $\mathsf{IP}(y_i, x_i^{(1)}) = \mathsf{IP}(y_i, x_i^{(2)}) = a_i$.

$i \notin \mathcal{I}$: Choose y_i such that $\mathsf{IP}(y_i, x_i^{(1)}) = z_i^{(1)}$ and $\mathsf{IP}(y_i, x_i^{(2)}) = z_i^{(2)}$.

Note that we can always choose a y as above since for each $i \in [n]$, $x_i^{(1)}, x_i^{(2)} \neq 0^b$, and for each $i \notin \mathcal{I}$, $x_i^{(1)} \neq x_i^{(2)}$. By the above construction, $f^{\mathsf{IP}}(x^{(1)}, y) = f(z^{(1)})$ and $f^{\mathsf{IP}}(x^{(2)}, y) = f(z^{(2)})$. Since by assumption $f(z^{(1)}) \neq f(z^{(2)})$, we have that $f^{\mathsf{IP}}(x^{(1)}, y) \neq f^{\mathsf{IP}}(x^{(2)}, y)$. But since Alice sends the same message on inputs $x^{(1)}$ and $x^{(2)}$, Π produces the same output on $(x^{(1)}, y)$ and $(x^{(2)}, y)$. This contradicts the correctness of Π. $\square$

4 Composition with AND

We first investigate the relationship between non-adaptive AND decision tree complexity and Möbius sparsity of Boolean functions. Recall that Claim 2.10 shows that for all Boolean functions $f : \{0,1\}^n \to \{0,1\}$,

$$\log \mathrm{spar}(f) \leq \mathsf{NAADT}(f) \leq \mathrm{spar}(f).$$

A natural question to ask is whether both of the bounds are tight, i.e. are there Boolean functions witnessing tightness of each bound? The first bound is trivially tight for any Boolean function with full Möbius sparsity, for example, the NOR function: querying all the input bits (which is querying n many ANDs) immediately yields the value of the function, and its Möbius sparsity can be shown to be 2^n.

One might expect that the upper bound is not tight in view of Theorem 2.11. The Addressing function witnesses tightness of the quadratic gap in Theorem 2.11. This gives rise to the natural question of whether an analogous bound holds true in the $\{0,1\}$-world: is it true for all Boolean functions f that $\mathsf{NAADT}(f) = \widetilde{O}(\sqrt{\mathrm{spar}(f)})$?

Interestingly we show in Section A that the Addressing function already gives a negative answer to this question. We observe in Section 4.1 that a stronger separation holds, and there exists a function (OMB_n) for which the second inequality in Claim 2.10 is in fact an equality. We use this same function to prove Theorem 1.7 in Section 4.2, which gives a maximal separation between $\mathsf{RNAADT}(f)$ and $\mathsf{D}_{\mathrm{cc}}^{\rightarrow}(f \circ \mathsf{AND}_2)$. Finally, we prove Theorem 1.8 in Section 4.3, which says that $\mathsf{NAADT}(f)$ is at most quadratically large in $\mathsf{D}_{\mathrm{cc}}(f \circ \mathsf{AND})$ for symmetric functions f.

4.1 Deterministic complexity

We prove in this section that the non-adaptive AND decision tree complexity of OMB_n is maximal whereas the one-way communication complexity of $\mathsf{OMB}_n \circ \mathsf{AND}$ is small.

Claim 4.1. *Let n be a positive integer. Then*

$$\mathsf{NAADT}(\mathsf{OMB}_n) = n,$$

and

$$\mathrm{spar}(\mathsf{OMB}_n) = \begin{cases} n & \text{if } n \text{ is even} \\ n+1 & \text{if } n \text{ is odd.} \end{cases}$$

Proof. Write the polynomial representation of OMB_n as

$$\mathsf{OMB}_n(x_1,\ldots,x_n) = (1-x_n)\cdot 0 + x_n(1-x_{n-1})\cdot 1 + x_n x_{n-1}\mathsf{OMB}_{n-2}(x_1,\ldots,x_{n-2}) \qquad \text{if } n \text{ is even} \quad (9)$$
$$\mathsf{OMB}_n(x_1,\ldots,x_n) = (1-x_n)\cdot 1 + x_n(1-x_{n-1})\cdot 0 + x_n x_{n-1}\mathsf{OMB}_{n-2}(x_1,\ldots,x_{n-2}) \qquad \text{if } n \text{ is odd.} \quad (10)$$

One can observe that the Möbius support of OMB_n equals $\{\{j,\ldots,n\} : j \le n\} \cup \{\emptyset\}$ if n is odd, and $\{\{j,\ldots,n\} : j \le n\}$ if n is even. Thus $\mathrm{spar}(\mathsf{OMB}_n) = n+1$ if n is odd, and equals n if n is even.

We now show that the $\mathsf{NAADT}(\mathsf{OMB}_n) = n$. Let $\mathcal{S}$ denote a NAADT basis for OMB_n. By Claim 2.9, any monomial in the Möbius expansion of OMB_n can be expressed as a product of some ANDs from $\mathcal{S}$. Thus, $\{n\}$ must participate in $\mathcal{S}$ since it appears in its Möbius support. Next, since $\{n-1,n\}$ appears in the support as well, either $\{n-1,n\}$ or $\{n-1\}$ must appear in $\mathcal{S}$. Continuing iteratively, we conclude that for all $i \in [n]$, there must exist a set in $\mathcal{S}$ that contains i, but does not contain any j for $j < i$. This immediately implies that $|\mathcal{S}| \ge n$. Equality holds since $\mathsf{NAADT}(f) \le n$ for any Boolean function $f : \{0,1\}^n \to \{0,1\}$. □

Remark 4.2. *The fact that the non-adaptive AND decision tree complexity (in fact even the non-adaptive* monotone *decision tree complexity, where the tree is allowed to make arbitrary monotone queries rather than just ANDs) of OMB_n equals $\Omega(n)$ already follows by a recent result of Amireddy, Jayasurya and Sarma [AJS20]. They show that any function with large alternating number (which is the largest number of switches of the function's value along a monotone path from 0^n to 1^n) must have large non-adaptive monotone decision tree complexity. We implicitly use the fact that OMB_n has large alternating number in the proof of Theorem 4.5, where we show that even the* randomized *non-adaptive AND decision tree complexity of OMB_n is $\Omega(n)$.*

Thus OMB_n witnesses that non-adaptive AND decision tree complexity can be as large as sparsity. We remark here that OMB_n admits a simple $O(\log n)$ depth (adaptive) AND-decision tree. This uses a binary search using AND-queries to determine the right-most index where a 0 is present.

One might expect that a result similar to Claim 1.6 holds when the inner function is AND instead of XOR. That is, it is plausible that the deterministic one-way communication complexity of $f \circ \mathsf{AND}$ equals the non-adaptive AND decision tree complexity of f. We show that this is not true, and exhibit an exponential separation between $\mathsf{D}_{\mathrm{cc}}^{\rightarrow}(\mathsf{OMB}_n \circ \mathsf{AND})$ and $\mathsf{NAADT}(\mathsf{OMB}_n)$.

Claim 4.3. *Let n be a positive integer. Then*

$$\mathsf{D}_{\mathrm{cc}}^{\rightarrow}(\mathsf{OMB}_n \circ \mathsf{AND}) = \lceil \log(n+1) \rceil.$$

Proof. From Equation (9) we conclude that the Möbius support of OMB_n equals

$$\mathcal{S} = \{\{n\}, \{n-1,n\}, \ldots, \{n,n-1,\ldots,1\}\} \qquad \text{if } n \text{ is even}$$
$$\mathcal{S} = \{\emptyset, \{n\}, \{n-1,n\}, \ldots, \{n,n-1,\ldots,1\}\} \qquad \text{if } n \text{ is odd.}$$

It is easy to verify that the only possible Möbius patterns attainable (ignoring the empty set since it always evaluates to 0) are $1^i 0^{n-i}$, for $i \in \{0, 1, \ldots, n\}$. Moreover, all of these patterns are attainable: the pattern $1^i 0^{n-i}$ is attained by the input string $0^{n-i} 1^i$. Thus,

$$\mathsf{Pat}^{\mathsf{M}}(\mathsf{OMB}_n) = n + 1.$$

Claim 2.14 implies $\mathsf{D}^{\rightarrow}_{\mathsf{cc}}(\mathsf{OMB}_n \circ \mathsf{AND}) = \lceil \log(n + 1) \rceil$. ☐

We readily obtain our main result of this section.

Theorem 4.4. *Let n be a positive integer. Then*

$$\mathsf{NAADT}(\mathsf{OMB}_n) = n,$$
$$\mathsf{D}^{\rightarrow}_{\mathsf{cc}}(\mathsf{OMB}_n \circ \mathsf{AND}) = \lceil \log(n + 1) \rceil.$$

Proof. It follows from Claim 4.1 and Claim 4.3. ☐

4.2 Randomized complexity

We prove that even the randomized non-adaptive AND decision tree complexity of OMB_n is $\Omega(n)$. In view of the small one-way communication complexity of $\mathsf{OMB}_n \circ \mathsf{AND}$ from Claim 4.3, Theorem 1.7 then follows.

Theorem 4.5. *Let n be a positive integer. Then,*

$$\mathsf{RNAADT}(\mathsf{OMB}_n) = \Omega(n).$$

Proof. We prove this by constructing a hard distribution μ on $\{0, 1\}^n$ such that any NAADT of small size computing OMB_n must have large error under this distribution of inputs. By the minimax principle, this would imply the required lower bound. Define $\mu : \{0, 1\}^n \to \mathbb{R}$ by

$$\mu(x) = \begin{cases} \frac{1}{n+1} & \text{if } x = 0^i 1^{n-i} \text{ for some } i \in \{0, 1, \ldots, n\} \\ 0 & \text{otherwise.} \end{cases} \tag{11}$$

Let $\mathcal{T}$ be a NAADT of cost at most cn (where c is a constant to be fixed later) computing OMB_n to error less than 0.48 under the distribution above. Let T be the set of ANDs queried by $\mathcal{T}$. We first argue that we may assume without loss of generality that every AND in T must have fan-in either 1 or 2. To see this, let $\mathsf{AND}(x_{i_1}, \ldots, x_{i_k})$ be an AND queried by T, where $k > 2$ and $i_1 < \cdots < i_k$. We may replace this by $\mathsf{AND}(x_{i_1}, x_{i_k})$ without affecting the output of the AND on any input in the support of μ. Let $J = \{j_1, j_2, \ldots, j_\ell\}$ be the set of indices involved in at least one of these ANDs. Order these indices such that $j_1 < j_2 < \cdots < j_\ell$. Since T was assumed to have size at most cn, we must have

$$\ell := |J| \leq 2cn. \tag{12}$$

For $m \in [\ell + 1]$, let I_m denote the set of indices in between, but not including, j_{m-1} and j_m. Here $j_0 := 0$ and $j_{\ell+1} := n + 1$. That is,

$$I_m = (j_{m-1}, j_m) \cap \mathbb{N}.$$

We remark that certain intervals I_k might be empty, and all of the I_k's are disjoint. However, we have

$$|I_1| + \cdots + |I_{\ell+1}| = n - \ell. \tag{13}$$

For each $m \in [\ell + 1]$, define the set $X_m \subseteq \{0, 1\}^n$ to be

$$X_m = \left\{ x \in \{0, 1\}^n : x = 0^i 1^{n-i} \text{ for some } i \in I_m \right\}.$$

Note that $\mathcal{T}$ cannot distinguish between inputs in X_m. Moreover each set X_m contains at least $\lfloor |I_m|/2 \rfloor$ many 0-inputs to OMB_n from the support of μ (see Equation (11)) and at least $\lfloor |I_m|/2 \rfloor$ many 1-inputs to OMB_n from the support of μ. Thus, by the definition of μ, $\mathcal{T}$ must make error at least

$$\begin{aligned}\frac{1}{n+1}\sum_{m=1}^{\ell+1}\left\lfloor\frac{|I_m|}{2}\right\rfloor &\geq \frac{1}{n+1}\sum_{m=1}^{\ell+1}\frac{|I_m|-1}{2} \geq \frac{1}{n+1}\cdot\frac{n-\ell-(\ell+1)}{2}\\ &= \frac{1}{2}\left(\frac{n-2\ell-1}{n+1}\right) = \frac{1}{2}-\frac{\ell+1}{n+1},\end{aligned}$$

where the second inequality holds by Equation (13). Set $c = 1/200$. Equation (12) implies that $\ell \leq n/100$. This implies that for sufficiently large values of n, any NAADT of cost at most $n/100$ computing OMB_n must make error at least $1/2 - 2\ell/n \geq 0.48$, concluding the proof. □

The proof of Theorem 1.7 now follows easily.

Proof of Theorem 1.7. It follows from Claim 4.3 and Theorem 4.5. □

4.3 Symmetric functions

In this section we show that symmetric functions f admit efficient non-adaptive AND decision trees in terms of the deterministic (even two-way) communication complexity of $f \circ \mathsf{AND}$. We require the following bounds on the Möbius sparsity of symmetric functions, due to Buhrman and de Wolf [BdW01]. For a non-constant symmetric function $f : \{0,1\}^n \to \{0,1\}$, define the following measure which captures the smallest Hamming weight inputs before which f is not a constant.

$$\mathsf{switch}(f) := \arg\min_k \{f \text{ is a constant on all } x : |x| < n-k\}.$$

Claim 4.6 ([BdW01, Lemma 5])**.** *Let n be sufficiently large, let $f : \{0,1\}^n \to \{0,1\}$ be a symmetric Boolean function, and let $k := \mathsf{switch}(f)$. Then,*

$$\log \mathrm{spar}(f) \geq \frac{1}{2}\log\left(\sum_{i=n-k}^{n}\binom{n}{i}\right).$$

Upper bounds on the non-adaptive AND decision tree complexity of symmetric functions follow from known results in the non-adaptive group testing literature. To the best of our knowledge, the following upper bounds were first shown (formulated differently) by Dyachkov and Rykov [DR83]. Also see [CH08] and the references therein.

Theorem 4.7. *Let $f : \{0,1\}^n \to \{0,1\}$ be a Boolean function with $\mathsf{switch}(f) = k < n/2$. Then*

$$\mathsf{NAADT}(f) = O\left(\log^2\binom{n}{k}\right).$$

We give a self-contained proof of Theorem 4.7 in Section B for clarity and completeness. We are now ready to prove Theorem 1.8.

Proof of Theorem 1.8. If $\mathsf{switch}(f) \geq n/2$, then Claim 4.6 implies that $\mathrm{spar}(f) = 2^{\Omega(n)}$. Equation (8) implies that $\mathsf{D}_{\mathrm{cc}}(f \circ \mathsf{AND}) = \Omega(n)$. Thus, a trivial NAADT of cost n witnesses $\mathsf{NAADT}(f) = O(\mathsf{D}_{\mathrm{cc}}(f \circ \mathsf{AND}))$ in this case.

Hence, we may assume $\mathsf{switch}(f) = k < n/2$. We have

$$\mathsf{NAADT}(f) = O\left(\log^2\binom{n}{k}\right) \qquad \text{by Theorem 4.7}$$

$$
\begin{aligned}
&= O(\log^2(\mathrm{spar}(f))) && \text{by Claim 4.6} \\
&= O(\mathsf{D}_{\mathrm{cc}}(f \circ \mathsf{AND})^2). && \text{by Equation (8)}
\end{aligned}
$$

$\square$

Acknowledgements S.S. thanks Prahladh Harsha and Jaikumar Radhakrishnan for pointing out the reference [FT99], and Srijita Kundu for pointing out the reference [Kla00]. N.S.M. thanks Ronald de Wolf for useful discussions.

References

[AGJ+17] Anurag Anshu, Dmitry Gavinsky, Rahul Jain, Srijita Kundu, Troy Lee, Priyanka Mukhopadhyay, Miklos Santha, and Swagato Sanyal. A composition theorem for randomized query complexity. In *37th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS)*, pages 10:1–10:13, 2017.

[AJS20] Prashanth Amireddy, Sai Jayasurya, and Jayalal Sarma. Power of decision trees with monotone queries. In *Proceedings od the Computing and Combinatorics - 26th International Conference (COCOON)*, pages 287–298, 2020.

[BB20] Shalev Ben-David and Eric Blais. A tight composition theorem for the randomized query complexity of partial functions: Extended abstract. In *61st IEEE Annual Symposium on Foundations of Computer Science, FOCS*, pages 240–246. IEEE, 2020.

[BdW01] Harry Buhrman and Ronald de Wolf. Communication complexity lower bounds by polynomials. In *Proceedings of the 16th Annual IEEE Conference on Computational Complexity (CCC)*, pages 120–130, 2001.

[Bei93] Richard Beigel. The polynomial method in circuit complexity. In *Proceedings of the Eighth Annual Structure in Complexity Theory Conference*, pages 82–95, 1993.

[BK16] Shalev Ben-David and Robin Kothari. Randomized query complexity of sabotaged and composed functions. In *43rd International Colloquium on Automata, Languages, and Programming (ICALP)*, pages 60:1–60:14, 2016.

[BT15] Mark Bun and Justin Thaler. Dual lower bounds for approximate degree and Markov-Bernstein inequalities. *Inf. Comput.*, 243:2–25, 2015. Earlier version in ICALP'13.

[BVW07] Harry Buhrman, Nikolai K. Vereshchagin, and Ronald de Wolf. On computation and communication with small bias. In *22nd Annual IEEE Conference on Computational Complexity (CCC)*, pages 24–32, 2007.

[CFK+21] Arkadev Chattopadhyay, Yuval Filmus, Sajin Koroth, Or Meir, and Toniann Pitassi. Query-to-communication lifting using low-discrepancy gadgets. *SIAM J. Comput.*, 50(1):171–210, 2021.

[CH08] Hong-Bin Chen and Frank K. Hwang. A survey on nonadaptive group testing algorithms through the angle of decoding. *J. Comb. Optim.*, 15(1):49–59, 2008.

[CKLM19] Arkadev Chattopadhyay, Michal Koucký, Bruno Loff, and Sagnik Mukhopadhyay. Simulation theorems via pseudo-random properties. *Comput. Complex.*, 28(4):617–659, 2019.

[DR83] Arkadii Georgievich Dyachkov and Vladimir Vasil'evich Rykov. A survey of superimposed code theory. *Problems of Control and Information Theory*, 12(4):1–13, 1983.

[dRNV16] Susanna F. de Rezende, Jakob Nordström, and Marc Vinyals. How limited interaction hinders real communication (and what it means for proof and circuit complexity). In *IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 295–304. IEEE Computer Society, 2016.

[FT99] Peter Frankl and Norihide Tokushige. The Erdos-Ko-Rado theorem for integer sequences. *Comb.*, 19(1):55–63, 1999.

[GGKS20] Ankit Garg, Mika Göös, Pritish Kamath, and Dmitry Sokolov. Monotone circuit lower bounds from resolution. *Theory Comput.*, 16:1–30, 2020.

[GJ16] Mika Göös and T. S. Jayram. A composition theorem for conical juntas. In *31st Conference on Computational Complexity (CCC)*, pages 5:1–5:16, 2016.

[GLM$^+$16] Mika Göös, Shachar Lovett, Raghu Meka, Thomas Watson, and David Zuckerman. Rectangles are nonnegative juntas. *SIAM J. Comput.*, 45(5):1835–1869, 2016.

[GLSS19] Dmitry Gavinsky, Troy Lee, Miklos Santha, and Swagato Sanyal. A composition theorem for randomized query complexity via max-conflict complexity. In *46th International Colloquium on Automata, Languages, and Programming (ICALP)*, volume 132, pages 64:1–64:13, 2019.

[GMWW17] Dmitry Gavinsky, Or Meir, Omri Weinstein, and Avi Wigderson. Toward better formula lower bounds: The composition of a function and a universal relation. *SIAM J. Comput.*, 46(1):114–131, 2017.

[GPW18] Mika Göös, Toniann Pitassi, and Thomas Watson. Deterministic communication vs. partition number. *SIAM J. Comput.*, 47(6):2435–2450, 2018.

[GPW20] Mika Göös, Toniann Pitassi, and Thomas Watson. Query-to-communication lifting for BPP. *SIAM J. Comput.*, 49(4), 2020. Earlier version in FOCS'17.

[HHL18] Hamed Hatami, Kaave Hosseini, and Shachar Lovett. Structure of protocols for XOR functions. *SIAM J. Comput.*, 47(1):208–217, 2018. Earlier version in FOCS'16.

[HLS07] Peter Høyer, Troy Lee, and Robert Spalek. Negative weights make adversaries stronger. In *Proceedings of the 39th Annual ACM Symposium on Theory of Computing (STOC)*, pages 526–535, 2007.

[HLY19] Kaave Hosseini, Shachar Lovett, and Grigory Yaroslavtsev. Optimality of linear sketching under modular updates. In Amir Shpilka, editor, *34th Computational Complexity (CCC)*, volume 137 of *LIPIcs*, pages 13:1–13:17. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019.

[Kla00] Hartmut Klauck. On quantum and probabilistic communication: Las vegas and one-way protocols. In *Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing (STOC)*, pages 644–651, 2000.

[KLMY20] Alexander Knop, Shachar Lovett, Sam McGuire, and Weiqiang Yuan. Log-rank and lifting for AND-functions. *CoRR*, abs/2010.08994, 2020. To appear in STOC'21.

[KMSY18] Sampath Kannan, Elchanan Mossel, Swagato Sanyal, and Grigory Yaroslavtsev. Linear sketching over f_2. In Rocco A. Servedio, editor, *33rd Computational Complexity Conference (CCC)*, volume 102 of *LIPIcs*, pages 8:1–8:37. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018.

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

[Kun17] Srijita Kundu. One-way quantum communication complexity with inner product gadget. *Electron. Colloquium Comput. Complex.*, 24:152, 2017. Comment #1 to TR17-152.

[LM19] Bruno Loff and Sagnik Mukhopadhyay. Lifting theorems for equality. In *36th International Symposium on Theoretical Aspects of Computer Science (STACS)*, volume 126 of *LIPIcs*, pages 50:1–50:19. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019.

[LS88] László Lovász and Michael E. Saks. Lattices, möbius functions and communication complexity. In *29th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 81–90, 1988.

[MO09] Ashley Montanaro and Tobias Osborne. On the communication complexity of XOR functions. *CoRR*, abs/0909.3392, 2009.

[Mon14] Ashley Montanaro. A composition theorem for decision tree complexity. *Chicago J. Theor. Comput. Sci.*, 2014, 2014.

[Rei11] Ben Reichardt. Reflections for quantum query algorithms. In *Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 560–569, 2011.

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

[San17] Swagato Sanyal. One-way communication and non-adaptive decision tree. *Electron. Colloquium Comput. Complex.*, 24:152, 2017.

[San19] Swagato Sanyal. Fourier sparsity and dimension. *Theory of Computing*, 15(11):1–13, 2019.

[She12] Alexander A. Sherstov. Making polynomials robust to noise. In *Proceedings of the 44th Symposium on Theory of Computing Conference (STOC)*, pages 747–758, 2012.

[She13] Alexander A. Sherstov. Approximating the AND-OR tree. *Theory Comput.*, 9:653–663, 2013.

[Tal13] Avishay Tal. Properties and applications of boolean function composition. In *Innovations in Theoretical Computer Science (ITCS)*, pages 441–454, 2013.

[VC71] VN Vapnik and A Ya Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability & Its Applications*, 16(2):264–280, 1971.

[Wu21] Hsin-Lung Wu. On the communication complexity of AND functions. *IEEE Transactions on Information Theory*, 2021.

[WYY17] Xiaodi Wu, Penghui Yao, and Henry S. Yuen. Raz-Mckenzie simulation with the inner product gadget. *Electron. Colloquium Comput. Complex.*, 24:10, 2017.

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

[ŽLG21] Julius Žilinskas, Algirdas Lančinskas, and Mario R Guarracino. Pooled testing with replication as a mass testing strategy for the covid-19 pandemics. *Scientific Reports*, 11(1):1–7, 2021.

A Addressing function

Recall that Theorem 2.11 shows that for all Boolean functions f, we have $\mathsf{NAPDT}(f) = O(\sqrt{r}\log r)$, where r denotes the Fourier sparsity of f. Moreover a quadratic separation is witnessed by the Addressing function. We showed in Claim 4.1 that such an upper bound does not hold in the $\{0,1\}$-world, and $\mathsf{NAADT}(\mathsf{OMB}_n) \in \{\mathrm{spar}(\mathsf{OMB}_n), \mathrm{spar}(\mathsf{OMB}_n) - 1\}$. In this section we show that the Addressing function already witnesses that a separation similar to that in Theorem 2.11 cannot hold in the $\{0,1\}$-world (even on allowing randomization in the decision trees). While the bound we obtain here is weaker than that in Claim 4.1, it is interesting that the Addressing function, which witnesses a quadratic separation in the $\{-1,1\}$-world, no longer does so in the $\{0,1\}$-world.

Definition A.1. *For an integer $n \geq 2$ that is a power of 2, define the Addressing function, denoted* $\mathsf{ADDR}_n : \{0,1\}^{\log n + n} \to \{0,1\}$, *by*

$$\mathsf{ADDR}_n(x, y) = y_{\mathsf{bin}(x)},$$

where $\mathsf{bin}(x)$ *denotes the integer in* $[n]$ *whose binary representation is* x. *We refer to the* x*-variables as* addressing variables *and the* y*-variables as* target variables.

The following is our main claim of this section.

Claim A.2. *Let $n \geq 2$ be a positive integer that is a power of 2. Then,*

$$\mathsf{RNAADT}(\mathsf{ADDR}_n) = \Theta(\mathrm{spar}(\mathsf{ADDR}_n)^{\frac{1}{\log 3}}).$$

We first show that even the randomized non-adaptive AND decision tree complexity of the Addressing function is large.

Claim A.3. *For an integer $n \geq 2$ that is a power of 2,*

$$\mathsf{RNAADT}(\mathsf{ADDR}_n) = \Theta(n).$$

Proof. Since ADDR_n is a function on $\log n + n$ variables, we have $\mathsf{RNAADT}(\mathsf{ADDR}_n) \leq \log n + n = O(n)$. In the remaining part of the proof we show that $\mathsf{RNAADT}(\mathsf{ADDR}_n) = \Omega(n)$. We exhibit a hard distribution on $\{0,1\}^{\log n + n}$ such that any NAADT of small cost that computes ADDR_n must have large error under this distribution. The required lower bound would then follow from the minimax principle. Define $\mu : \{0,1\}^{\log n + n} \to \mathbb{R}$ by

$$\mu(x, y) = \begin{cases} \frac{1}{2n} & y_{\mathsf{bin}(x)} = 0 \text{ and } y_z = 0 \text{ for all } z \neq \mathsf{bin}(x) \\ \frac{1}{2n} & y_{\mathsf{bin}(x)} = 1 \text{ and } y_z = 0 \text{ for all } z \neq \mathsf{bin}(x) \\ 0 & \text{otherwise.} \end{cases} \tag{14}$$

In other words, the distribution is obtained by picking an x uniformly at random from $\{0,1\}^{\log n}$, setting $y_z = 0$ for all $z \neq \mathsf{bin}(x)$, and setting $y_{\mathsf{bin}(x)}$ to 0 or 1 with equal probability. If $y_{\mathsf{bin}(x)}$ is set to 0, then this is a 0-input to ADDR_n and it is a 1-input if $y_{\mathsf{bin}(x)} = 1$.

Consider a deterministic non-adaptive AND decision tree T of cost $k < n/100$. Let $\mathcal{S} = \{S_1, \ldots, S_k\}$ denote the ANDs queried by T. By a simple counting, there must exist at least $99n/100$ target variables that do not appear as the only target variable in any of these sets. Denote this set of target variables by Y, and fix a $y_i \in Y$. Let $x \in \{0,1\}^{\log n}$ be such that $\mathsf{bin}(x) = i$. Define $w_0 = (x, y^0), w_1 = (x, y^1) \in \{0,1\}^{\log n + n}$ by

$$\begin{aligned} y_z^0 = y_z^1 &= 0 \quad \text{for all } z \neq i \\ y_i^0 &= 0 \\ y_i^1 &= 1. \end{aligned}$$

By our definition, the ANDs in $\mathcal{S}$ must output the same values on w_0 and w_1. Thus, T outputs the same value on these inputs, and hence must err on one of them. Since μ assigns $1/(2n)$ mass to both of these inputs, the total error of T is at least $|Y|/2n > 0.49$ since $|Y| \geq 99n/100$. $\square$

Thus $\mathsf{RNAADT}(\mathsf{ADDR}_n) = \Theta(n)$. However as we note in the following proof, its sparsity is $n^{\log 3}$, which is polynomially smaller that its sparsity in the $\{-1,1\}$-world which is n^2. This yields the bound in Claim A.2.

Proof of Claim A.2. Let $\mathbb{I}(E)$ denote the indicator function of E, that is, $\mathbb{I}(E) = 1$ if E is true, and 0 otherwise. From the expression in Definition A.1 we have

$$\mathsf{ADDR}_n(x, y) = \sum_{b \in \{0,1\}^{\log n}} y_b \mathbb{I}[x = b] = \sum_{b \in \{0,1\}^{\log n}} y_b \prod_{i \in [\log n] : b_i = 0} (1 - x_i) \prod_{i \in [\log n] : b_i = 1} x_i. \tag{15}$$

The monomials arising from each summand are disjoint, since monomials containing y_b only appear in the summand corresponding to b. For all $b \in \{0,1\}^{\log n}$, the number of monomials in $\prod_{i \in [\log n]: b_i = 0}(1 - x_i) \prod_{i \in [\log n]: b_i = 1} x_i$ equals $2^{\log n - |b|}$, where $|b|$ equals the Hamming weight (number of 1s) of b. From the expansion in Equation (15), we obtain

$$\mathrm{spar}(\mathsf{ADDR}_n) = \sum_{b \in \{0,1\}^{\log n}} 2^{\log n - |b|} = \sum_{j=0}^{\log n} \binom{\log n}{j} 2^j = 3^{\log n} = n^{\log 3}.$$

Since $\mathsf{RNAADT}(\mathsf{ADDR}_n) = \Theta(n)$ from Claim A.3, this concludes the proof of the claim. $\square$

B On non-adaptive AND decision trees for symmetric functions

Recall Theorem 4.7, restated below.

Theorem B.1 (Restatement of Theorem 4.7)**.** *Let $f : \{0,1\}^n \to \{0,1\}$ be a Boolean function with $\mathsf{switch}(f) = k < n/2$. Then*

$$\mathsf{NAADT}(f) = O\left(\log^2 \binom{n}{k}\right).$$

The proof is via the probabilistic method. We construct a random family of $O\left(\log^2 \binom{n}{k}\right)$ many ANDs and argue that with non-zero probability, their evaluations on any input determine the function's value.

We require the following intermediate claim.

Claim B.2. *Let n be a positive integer, and let $1 \leq k < n/2$ be an integer. Then, there exists a collection $\mathcal{X}$ of $O\left(\log^2 \binom{n}{k}\right)$ many subsets of $[n]$ satisfying the following.*

$$\forall i_1, \ldots, i_{k+1} \in [n], j \in [k+1], \exists X \in \mathcal{X} \text{ such that } i_j \in X, i_\ell \notin X \text{ for all } \ell \neq j. \tag{16}$$

Proof. Consider a random set $X \subseteq [n]$ chosen as follows: For each index $i \in [n]$ independently, include i in X with probability $1/(2k)$. Pick w many sets (where w is a parameter that we fix later) independently using the above sampling process, giving the multiset of sets $\mathcal{X} = \{X_1, \ldots, X_w\}$.

For any set $X \in \mathcal{X}$ and fixed $i_1, \ldots, i_{k+1} \in [n]$ and $j \in [k+1]$,

$$\Pr_X[i_j \in X \text{ and } i_\ell \notin X \text{ for all } \ell \neq j] = \frac{1}{2k} \cdot \left(1 - \frac{1}{2k}\right)^k \geq \frac{1}{2k \cdot e}, \tag{17}$$

where the last inequality uses the fact that $k \geq 1$ and the standard inequality that $1 - x \geq e^{-2x}$ for all $x \leq 1/2$. Thus Equation (17) implies that for fixed $i_1, \ldots, i_{k+1} \in [n]$ and $j \in [k+1]$,

$$\Pr_{\mathcal{X}}[\nexists X \in \mathcal{X} : i_j \in X \text{ and } i_\ell \notin X \text{ for all } \ell \neq j] \leq \left(1 - \frac{1}{2k \cdot e}\right)^w \leq \exp(-w/(2ke)). \tag{18}$$

By a union bound over these "bad events" for all $i_1, \ldots, i_{k+1} \in [n]$ and $j \in [k+1]$, we conclude that

$$\begin{aligned} &\Pr_{\mathcal{X}}[\forall i_1, \ldots, i_{k+1} \in [n] \text{ and } j \in [k+1],\ \exists X \in \mathcal{X} : i_j \in X \text{ and } i_\ell \notin X \text{ for all } \ell \neq j] \\ &\geq 1 - \binom{n}{k+1} \cdot (k+1) \cdot \exp(-w/(2ke)). \end{aligned} \tag{19}$$

We want to choose w such that this probability is greater than 0. Thus we require

$$1 > \binom{n}{k+1} \cdot (k+1) \cdot \exp(-w/(2ke))$$

$$\iff \exp(w/(2ke)) > (k+1)\cdot\binom{n}{k+1}$$

$$\iff w > 2ke\left(\log(k+1) + \log\binom{n}{k+1}\right). \qquad \text{taking logarithms and rearranging}$$

Since $\binom{n}{j+1} \geq n > j+1 \geq \log(j+1)$ for all $j \in \{1,2,3,\dots,n/2\}$ and $n > 2$, and since $\log\binom{n}{j} \geq j\log(n/j) \geq j$ for all $j \in \{1,2,\dots,n/2\}$, it suffices to choose

$$w \geq 2e\log\binom{n}{k}\left(2\log\binom{n}{k+1}\right). \tag{20}$$

By standard binomial inequalities we have $\log\binom{n}{k+1} \leq (k+1)\log(ne/(k+1))$, and $\log\binom{n}{k} > k\log\binom{n}{k}$. Next, since $k+1 \leq 2k$ for $k \geq 1$ and $ne/(k+1) < n^3/k^3$ for $k \in \{1,2,\dots,n/2\}$, Equation (20) implies that it suffices to choose

$$w \geq 2e\log\binom{n}{k}\left(12\log\binom{n}{k}\right).$$

For this choice of w, the RHS of Equation (19) is strictly positive. This proves the claim. $\square$

Proof of Theorem B.1. Let f be a symmetric function with $\mathsf{switch}(f) = k < n/2$, and let $\mathcal{X}$ be as in Claim B.2 with $|\mathcal{X}| = O\left(\log^2\binom{n}{k}\right)$. We now show how $\mathcal{X}$ yields a NAADT for f. Without loss of generality assume that $f(x) = 0$ for all $|x| < n-k$ (if not, output 1 in place of 0 in the **Output** step of Algorithm 1 below).

Algorithm 1: NAADT for f

Input: $x \in \{0,1\}^n$

1. Let $\mathcal{X}$ be as obtained from Claim B.2.
2. Query $\{\mathsf{AND}_X(x) : X \in \mathcal{X}\}$ to obtain a string $P_x \in \{0,1\}^{|\mathcal{X}|}$.

Output: $f(y)$ if $P_x = P_y$ for some y with $|y| \geq n-k$, and 0 otherwise.

We show below that the following holds: $P_x \neq P_y$ for all $x \neq y \in \{0,1\}^n$ such that $|y| \geq n-k$. This would show correctness of the algorithm and prove the theorem. Let $x \neq y \in \{0,1\}^n$ be two strings such that $|y| \geq n-k$. Without loss of generality assume $|y| \geq |x|$ (else swap the roles of x and y above). Let $I_x, I_y \subseteq [n]$ denote the sets of indices where x and y take value 0, respectively. By assumption, $x \neq y$ and $|I_x| \geq |I_y|$. Thus there exists an index $i_x \in I_x \setminus I_y$.

Since $|I_y| \leq k$, by Claim B.2 there exists $X \in \mathcal{X}$ such that $i_x \in X$ and $X \cap I_y = \emptyset$. Thus, for this X we have

$$\mathsf{AND}_X(x) = 0, \qquad \mathsf{AND}_X(y) = 1.$$

Hence $P_x \neq P_y$, which proves the correctness of the algorithm and yields the theorem. $\square$

Remark B.3. *The proof above in fact yields a NAADT of cost $O\left(\log^2\binom{n}{k}\right)$ for any function $f : \{0,1\}^n \to \{0,1\}$ for which f is a constant on inputs of Hamming weight less than $n-k$ for some $k < n/2$ (in particular, f need not be symmetric on inputs of larger Hamming weight).*

C Derivation of Theorem 1.4

Recall Theorem 1.4, restated below.

Theorem C.1 (Restatement of Theorem 1.4)**.** *Let $q \geq 8$. Let $\mathcal{A} \subseteq [q]^n$ be such that for all $x^{(1)} = (x_1^{(1)},\dots,x_n^{(1)})$, $x^{(2)} = (x_1^{(2)},\dots,x_n^{(2)}) \in \mathcal{A}$, $|\{i \in [n] \mid x_i^{(1)} = x_i^{(2)}\}| \geq d$. Then, $|\mathcal{A}| < q^{n-\frac{d}{2}}$.*

Let q be as in the statement of the theorem. For $x \in \{0,1\}^n$ and $\mathsf{S} \subseteq [n]$, let x_{S} denote the restriction of x to the indices in S. Let $|x|$ denote the Hamming weight of x, which is $|\{i \in [n] \mid x_i = 1\}|$.

For an arbitrary alphabet L, a set $\mathcal{H} \subseteq L^n$ is called d-intersecting if for each $x = (x_i)_{i\in[n]}, x' = (x'_i)_{i\in[n]} \in \mathcal{H}$, $|\{i \in [n] \mid x_i = x'_i\}| \geq d$. Let $\mathsf{agr}(d,q,n)$ denote the size of a largest d-intersecting set in $[q]^n$. Frankl and Tokushige determined $\mathsf{agr}(d,q,n)$ in their work.

For an integer $r \leq (n-d)/2$, Let $\mathcal{A}_r$ be the following d-intersecting family in $\{0,1\}^n$.

$$A_r := \{x \in \{0,1\}^n \mid |x_{\{1,\ldots,d+2r\}}| \geq d+r\}.$$

Now consider the following d-intersecting family $\mathcal{B}_r$ in $[q]^n$: A string $x \in [q]^n$ belongs to $\mathcal{B}_r$ iff there exists a string $z \in \mathcal{A}_r$ such that for each $i \in [n]$, $z_i = 1 \Rightarrow x_i = 1$. $\mathcal{B}_r$ is easily seen to be d-intersecting. Hence for each such r, $\mathsf{agr}(d,q,n) \geq |\mathcal{B}_r|$.

Frankl and Tokushige showed that in fact there is a choice of r for which $\mathsf{agr}(d,q,n) = |\mathcal{B}_r|$. In other words, there exists a choice of r such that $\mathcal{B}_r$ is a largest d-intersecting family in $[q]^n$.

Theorem C.2 (Theorem 2 in [FT99])**.** *Let $q \geq 3, r = \lfloor \frac{d-1}{q-2} \rfloor$ and $n \geq d+2r$. Then,* $\mathsf{agr}(d,q,n) = |\mathcal{B}_r|$.

Proving Theorem 1.4 now amounts to estimating $|\mathcal{B}_r|$. A string in $\mathcal{B}_r$ can be generated as follows.

Choose a subset $T \in [d+2r]$ of size $d+r$.

For each $i \in T$, set $x_i = 1$.

For each $i \notin T$, set x_i arbitrarily.

There are $\binom{d+2r}{d+r}$ choices of T. For each choice of T, there are q^{n-d-r} ways of assigning variables with indices outside T. We thus have,

$$\begin{aligned}
|\mathcal{B}_r| &\leq \binom{d+2r}{d+r} \cdot q^{n-d-r} \\
&\leq \left(\frac{e(d+2r)}{d+r}\right)^{d+r} \cdot q^{n-d-r} \\
&= e^{d+r} \cdot \left(1 + \frac{r}{d+r}\right)^{d+r} \cdot q^{n-d-r} \\
&\leq e^{d+2r} \cdot q^{n-d-r} \quad \text{(Since } 1+z \leq e^z \text{ for all real } z\text{)} \\
&= q^{n-d(1-\frac{1}{\log_e q})-r(1-\frac{2}{\log_e q})}
\end{aligned} \tag{21}$$

By the assumption $q \geq 8$, we have that $1 - \frac{2}{\log_e q} > 0$ and $1 - \frac{1}{\log_e q} > \frac{1}{2}$. Thus from (21) we have,

$$|\mathcal{B}_r| < q^{n-\frac{d}{2}}.$$

D A proof of Theorem 1.4 for $q = \Omega(\frac{n}{d})^2$

In this section we give a self-complete and simple proof of the statement of Theorem 1.4 for the special case of $q > (en/d)^2$ (with a worse constant).

Let $\mathcal{X} \subseteq [q]^n$ be such that every $x, x' \in \mathcal{X}$ agree in at least d locations. Observe that each pair (x,x') can be uniquely specified by,

A set $P_{x,x'} \subseteq [n]$ of indices of size d such that $x_i = x'_i$ for each $i \in P_{x,x'}$.

A vector $\mathbf{a} = (a_i)_{i\in P_{x,x'}} \in [q]^d$. $\mathbf{a}$ represents $x_{P_{x,x'}} = x'_{P_{x,x'}}$.

Vectors $x_{\overline{P_{x,x'}}}$ and $x'_{\overline{P_{x,x'}}}$.

Thus the number of pairs (x,x') is at most the number of such representations, which is upper bounded by $\binom{n}{d} \cdot q^d \cdot q^{2(n-d)} \leq (en/d)^d \cdot q^{2n-d} < q^{2n-\frac{d}{2}}$ (since $q > (\frac{en}{d})^2$). Thus $|\mathcal{X}|^2 < q^{2n-\frac{d}{2}} \Rightarrow |\mathcal{X}| < q^{n-\frac{d}{4}}$.

E Communication complexity of AND functions and pattern complexity

Proof of Claim 2.14. Let $\mathcal{S}$ denote the Möbius support of f, and say $\mathsf{Pat}^{\mathsf{M}}(f) = k$. Write the Möbius expansion of f as

$$f = \sum_{S \in \mathcal{S}} \widetilde{f}(S)\mathsf{AND}_S. \tag{22}$$

We first show that $\mathsf{D}_{\mathrm{cc}}^{\rightarrow}(f \circ \mathsf{AND}) \leq \lceil \log k \rceil$ by exhibiting a one-way protocol of cost $\lceil \log k \rceil$. Alice computes the pattern of x and sends Bob the pattern using $\lceil \log k \rceil$ bits of communication. Bob now knows the values of $\{\mathsf{AND}_S(x) : S \in \mathcal{S}\}$. Since Bob can compute $\{\mathsf{AND}_S(y) : S \in \mathcal{S}\}$ without any communication, he can now compute the value of $f \circ \mathsf{AND}(x, y)$ using the formula

$$(f \circ \mathsf{AND})(x, y) = \sum_{S \in \mathcal{S}} \widetilde{f}(S)\mathsf{AND}_S(x)\mathsf{AND}_S(y).$$

It remains to show that $\mathsf{D}_{\mathrm{cc}}^{\rightarrow}(f \circ \mathsf{AND}) \geq \lceil \log k \rceil$. Let $\mathsf{D}_{\mathrm{cc}}^{\rightarrow}(f \circ \mathsf{AND}) = d$. Thus there are at most 2^d messages that Alice can send Bob. We show that any two inputs $x, x' \in \{0,1\}^n$ for which Alice sends the same message have the same pattern, which would prove $2^d \geq k$, and prove the claim since d must be an integer.

Let x, x' be 2 inputs to Alice for which her message to Bob is m. We have

$$f(x \wedge y) = \sum_{S \in \mathcal{S}} \widetilde{f}(S)\mathsf{AND}_S(x)\mathsf{AND}_S(y)$$
$$f(x' \wedge y) = \sum_{S \in \mathcal{S}} \widetilde{f}(S)\mathsf{AND}_S(x')\mathsf{AND}_S(y)$$

Since m and y completely determine the value of the function, we must have

$$\sum_{S \in \mathcal{S}} \widetilde{f}(S)\mathsf{AND}_S(x)\mathsf{AND}_S(y) = \sum_{S \in \mathcal{S}} \widetilde{f}(S)\mathsf{AND}_S(x')\mathsf{AND}_S(y) \qquad \text{for all } y \in \{0,1\}^n.$$

Define the functions $g_x, g_{x'} : \{0,1\}^n \to \{0,1\}$ by

$$g_x(y) = \sum_{S \in \mathcal{S}} \widetilde{f}(S)\mathsf{AND}_S(x)\mathsf{AND}_S(y)$$
$$g_{x'}(y) = \sum_{S \in \mathcal{S}} \widetilde{f}(S)\mathsf{AND}_S(x')\mathsf{AND}_S(y).$$

Thus by uniqueness of the Möbius expansion of Boolean functions, $g_x = g_{x'}$ as functions of y. This implies $\widetilde{g_x}(S) = \widetilde{g_{x'}}(S)$ for all $S \in \mathcal{S}$. Since $\widetilde{g_x}(S) = \widetilde{f}(S)\mathsf{AND}_S(x)$ and $\widetilde{g_{x'}}(S) = \widetilde{f}(S)\mathsf{AND}_S(x')$ for all $S \in \mathcal{S}$,

$$\mathsf{AND}_S(x) = \mathsf{AND}_S(x') \qquad \text{for all } S \in \mathcal{S},$$

This shows that the pattern induced by x and the pattern induced by x' are the same, concluding the proof. $\square$

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il