One-Way Communication Complexity and Non-Adaptive Decision Trees

Nikhil S. Mande ✉ 🏠
CWI, Amsterdam, The Netherlands

Swagato Sanyal ✉
Indian Institute of Technology, Kharagpur, India

Suhail Sherif ✉
Vector Institute, Toronto, Canada

──── **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 for all $b \geq 2$, denoted IP.

- If f is a total Boolean function that depends on all of its n input bits, 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 $\Omega(b \cdot \mathsf{D}_{\mathrm{dt}}^{\rightarrow}(f))$, where $\mathsf{D}_{\mathrm{dt}}^{\rightarrow}(f)$ denotes 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 independently proven by Ahlswede and Khachatrian [Adv. Appl. Math.'98], and 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 *quantum* 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 give a 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.

It is well known that the rank of the communication matrix of a function F is an upper bound on its deterministic one-way communication complexity. This bound is known to be tight for some F. However, in our final result we show that this is not the case when $F = f \circ \mathsf{AND}$. More precisely we show that for all f, the deterministic one-way communication complexity of $F = f \circ \mathsf{AND}$ is at most $(\mathrm{rank}(M_F))(1 - \Omega(1))$, where M_F denotes the communication matrix of F.

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 [8, 24, 39, 42, 43, 9, 44, 35, 5, 18, 2, 17, 4].

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 [19, 13, 38, 20, 11, 48, 16, 21, 22, 27, 30, 10].

In this work we are interested in the *one-way* communication complexity of composed functions. Here, 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 [36] 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. [25] 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 [23] 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. Formal definitions of the measures considered in this section can be found in Section 2 and Appendix A.

Inner Product Gadget

Let $\mathsf{Q}^{\rightarrow}_{cc,\varepsilon}(\cdot)$, $\mathsf{R}^{\rightarrow}_{cc,\varepsilon}(\cdot)$ and $\mathsf{D}^{\rightarrow}_{cc}(\cdot)$ 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 $\mathsf{Q}^{*,\rightarrow}_{cc,\varepsilon}(\cdot)$. Let $\mathsf{Q}^{\rightarrow}_{dt}(\cdot)$ and $\mathsf{D}^{\rightarrow}_{dt}(\cdot)$ denote bounded-error quantum non-adaptive decision tree complexity and deterministic non-adaptive decision tree complexity, respectively. 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. 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 $f \circ \mathsf{IP}$ is $\Omega(n(b-1))$. Let $\mathbb{H}_{\mathsf{bin}}(\cdot)$ denote the binary entropy function. If $\varepsilon = 1/2 - \Omega(1)$, then $1 - \mathbb{H}_{\mathsf{bin}}(\varepsilon) = \Omega(1)$.

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

▶ Remark 1.2. In an earlier manuscript [40], the second author proved a lower bound of $(1 - \mathbb{H}_{\mathsf{bin}}(\varepsilon))n(b-1)$ on a weaker complexity measure, namely $\mathsf{R}^{\rightarrow}_{cc,\varepsilon}(F)$, via information-theoretic tools. Kundu [28] 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 [26]. 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 [26, 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 2.7). 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 $\mathsf{VC}(f \circ \mathsf{IP}) \geq n(b-1)$. 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, $f \circ \mathsf{IP} = \mathsf{AND}_{2n}$, whose one-way communication complexity is 1.

Our second result with the Inner Product gadget relates the deterministic one-way communication complexity of $f \circ \mathsf{IP}$ 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 $b \geq 2$ and $\mathsf{IP} : \{0,1\}^b \times \{0,1\}^b \to \{0,1\}$. Then $\mathsf{D}^{\rightarrow}_{cc}(f \circ \mathsf{IP}) = \Omega(b \cdot \mathsf{D}^{\rightarrow}_{dt}(f))$.*

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 a result due to Ahlswede and Khachatrian [1] and independently Frankl and Tokushige [15], is a crucial ingredient in our proof.

▶ **Theorem 1.4.** *Let $q \geq 3$ and $1 \leq d \leq n/3$. 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, $|\mathcal{A}| < q^{n-\frac{d}{10}}$.*

We refer the reader to the full version of our paper [33] for a proof.

▶ 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 the Inner Product function on 4 input bits (and lower bounds do not hold for the Inner Product gadget with fewer inputs). It is worth mentioning here that prior works that consider lifting theorems with the Inner Product gadget [11, 48, 10], albeit in the two-way model of communication complexity, require a super-constant gadget size.

AND Gadget

Interactive communication complexity of functions of the form $f \circ \mathsf{AND}$ have gained a recent interest [27, 47]. 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 minimum cost (number of parity queries) 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 [36] observed that this inequality is, in fact, tight for all Boolean functions. More precisely,

▷ Claim 1.6 ([36]).   For all Boolean functions $f : \{0,1\}^n \rightarrow \{0,1\}$, $\mathsf{D}_{\mathrm{cc}}^{\rightarrow}(f \circ \mathsf{XOR}) = \mathsf{NAPDT}(f)$.

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 minimum cost (number of AND queries) of such a tree computing a Boolean function f by $\mathsf{NAADT}(f)$. Clearly, $\mathsf{D}_{\mathrm{cc}}^{\rightarrow}(f \circ \mathsf{AND})$ 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 complexity. Thus, $\mathsf{D}_{\mathrm{cc}}^{\rightarrow}(f \circ \mathsf{AND}) \leq \mathsf{NAADT}(f)$. On the other hand, one can show that $\mathsf{D}_{\mathrm{cc}}^{\rightarrow}(f \circ \mathsf{AND}) \geq \log(\mathsf{NAADT}(f))$ (see Claim 4.3). Thus

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

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 the decision trees to have *quantum* query access to ANDs of subsets of input variables. It is worth noting that, in contrast, if one is given quantum query access to *parities* (in place of ANDs) of subsets of input variables, then one can completely recover an n-bit string using just 1 query [6], rendering this model trivial. Let $\mathsf{QNAADT}(f)$ denote the bounded-error quantum non-adaptive AND decision tree complexity of f.

▶ **Theorem 1.7.** *There exists a function* $f : \{0,1\}^n \to \{0,1\}$ *such that* $\mathsf{QNAADT}(f) = \Omega(2^{\mathsf{D}_{cc}^{\to}(f \circ \mathsf{AND})})$.

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 2.3). 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. A crucial property that we use to show a lower bound of $\Omega(n)$ on $\mathsf{QNAADT}(\mathsf{OMB}_n)$ is that OMB_n has large *alternating number*, that is, there is a monotone path on the Boolean hypercube from 0^n to 1^n on which the value of OMB_n flips many times.

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 $\mathsf{D}_{cc}(\cdot)$.

▶ **Theorem 1.8.** *Let* $f : \{0,1\}^n \to \{0,1\}$ *be a symmetric function. Then* $\mathsf{NAADT}(f) = O(\mathsf{D}_{cc}(f \circ \mathsf{AND})^2)$.

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

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

Since it is well known (Equation (6)) 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. There have been multiple works (see, for example, [8, 47, 27] and the references therein) studying the communication complexity of AND functions in connection with the log-rank conjecture [31] which states that the communication complexity is bounded from above by a polynomial in the logarithm of the rank of the communication matrix. Among other things, Buhrman and de Wolf [8] observed that the log-rank conjecture holds for symmetric functions composed with AND. In particular, they showed that if f is symmetric, then $\mathsf{D}_{cc}(f \circ \mathsf{AND}) = O(\log \mathrm{rank}(M_{f \circ \mathsf{AND}}))$. Most recently, Knop et al. [27] showed that $\mathsf{D}_{cc}(f \circ \mathsf{AND}) = O(\mathrm{poly}(\log \mathrm{rank}(M_{f \circ \mathsf{AND}}), \log n)$ for all Boolean functions $f : \{0,1\}^n \to \{0,1\}$, nearly resolving the log-rank conjecture for AND functions.

While we have a quadratically worse dependence in the RHS of Equation (2) as compared to the above-mentioned bound for symmetric functions due to Buhrman and de Wolf, our upper bound is on a complexity measure that can be exponentially larger than communication complexity in general (Theorem 1.7).

Buhrman and de Wolf showed a lower bound on $\log \mathrm{rank}(M_{f \circ \mathsf{AND}})$ for symmetric functions f. An upper bound on $\mathsf{NAADT}(f)$ implicitly follows from prior work on group testing [14], 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, [12] and the references therein), and has additionally gained significant interest of late in view of the ongoing pandemic (see, for example, [50]).

Our final result regarding the AND gadget deals with the relationship between one-way communication complexity and rank of the underlying communication matrix. It is easy to show that for functions $F : \{0,1\}^m \times \{0,1\}^n \to \{0,1\}$,

$$\log \text{rank}(M_F) \leq \mathsf{D}_{cc}^{\to}(F) \leq \text{rank}(M_F), \tag{3}$$

where M_F denotes the communication matrix of F and is defined by $M_F(x,y) = F(x,y)$, and $\text{rank}(\cdot)$ denotes real rank. The first bound can be seen to be tight for functions with maximal rank, for example the Equality function. The second inequality is tight, for example, for the Addressing function on $(\log n + n)$ input bits (see Definition A.1) where Alice receives n target bits and Bob receives $\log n$ addressing bits. Sanyal [41] showed that the upper bound can be improved for functions of the form $F = f \circ \mathsf{XOR}$. More precisely they showed that for all Boolean functions $f : \{0,1\}^n \to \{0,1\}$,

$$\mathsf{D}_{cc}^{\to}(f \circ \mathsf{XOR}) \leq O\left(\sqrt{\text{rank}(M_{f \circ \mathsf{XOR}})} \log \text{rank}(M_{f \circ \mathsf{XOR}})\right), \tag{4}$$

and moreover this bound is tight up to the logarithmic factor on the RHS, when f is the Addressing function. We show that the same bound does not hold when the XOR gadget is replaced by AND. We show that (see [33, Corollary A.5]) when f is the Addressing function, then

$$\mathsf{D}_{cc}^{\to}(f \circ \mathsf{AND}) \geq \text{rank}(M_{f \circ \mathsf{AND}})^{\log_3 2} \approx \text{rank}(M_{f \circ \mathsf{AND}})^{0.63}, \tag{5}$$

Thus it is plausible that the upper bound in terms of rank from Equation (3) might be tight for some function of the form $f \circ \mathsf{AND}$. We show that this is not the case.

▶ **Theorem 1.9.** *Let $f : \{0,1\}^n : \{0,1\}$ be a Boolean function. Then,*

$$\mathsf{D}_{cc}^{\to}(f \circ \mathsf{AND}) \leq (\text{rank}(M_{f \circ \mathsf{AND}}))(1 - \Omega(1)).$$

We show that $\mathsf{D}_{cc}^{\to}(f \circ \mathsf{AND})$ is equal to the logarithm of a measure that we define in this work: the *Möbius pattern complexity* of f, which is the total number of distinct evaluations of the monomials in the Möbius expansion of f (see Section 2 for a formal definition of Möbius expansion).

▶ **Definition 1.10** (Möbius pattern complexity). *Let $f : \{0,1\}^n \to \{0,1\}$ be a Boolean function, and let $f = \sum_{S \in \mathcal{S}_f} \widetilde{f}(S)\mathsf{AND}_S$ be its Möbius expansion. For an input $x \in \{0,1\}^n$, define the* pattern of x to be $(\mathsf{AND}_S(x))_{S \in \mathcal{S}_f}$. *Define the* Möbius pattern complexity *of f, denoted* $\mathsf{Pat}^{\mathsf{M}}(f)$, *by* $\mathsf{Pat}^{\mathsf{M}}(f) := \left|\left\{P \in \{0,1\}^{\mathcal{S}_f} : P = (\mathsf{AND}_S(x))_{S \in \mathcal{S}_f} \text{ for some } x \in \{0,1\}^n\right\}\right|$.

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

All of our results involving bounds for $\mathsf{D}_{cc}^{\to}(f \circ \mathsf{AND})$ use the above-mentioned equivalence between it and $\log(\mathsf{Pat}^{\mathsf{M}}(f))$ (see Claim 4.1). We unravel interesting mathematical structure in the Möbius supports of Boolean functions, and use them to bound their pattern complexity. We hope that pattern complexity will prove useful in future research.

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). In Section 4 we prove our results regarding the AND gadget (Theorem 1.7 and Theorem 1.8). We provide remaining preliminaries and missing proofs from the main text in the remaining appendices. Due to space constraints, some proofs are deferred to the full version of our paper [33].

2 Preliminaries

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

▶ **Definition 2.1** (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 $1 - p$.*

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

We now 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 2.2** (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* $\mathsf{IP}(x_1, \ldots, x_b, y_1, \ldots, y_b) = \oplus_{i \in [b]}(\mathsf{AND}(x_i, y_i))$.

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

▶ **Definition 2.3** (Odd-Max-Bit). *Define the* Odd-Max-Bit *function,[1] denoted* $\mathsf{OMB}_n : \{0, 1\}^n \to \{0, 1\}$, *by* $\mathsf{OMB}_n(x) = 1$ *if* $\max\{i \in [n] : x_i = 0\}$ *is odd, and* $\mathsf{OMB}_n(x) = 0$ *otherwise. Define* $\mathsf{OMB}_n(1^n) = 0$.

Möbius Expansion of Boolean Functions

Every Boolean function $f : \{0, 1\}^n \to \{0, 1\}$ has a unique expansion as $f = \sum_{S \subseteq [n]} \widetilde{f}(S) \mathsf{AND}_S$, 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 as the *Möbius expansion* of f, and the real coefficients $\widetilde{f}(S)$ as the Möbius coefficients of f. It is known [3] that the Möbius coefficients can be expressed as $\widetilde{f}(S) = \sum_{X \subseteq S}(-1)^{|S \setminus X|} f(X)$. Define the *Möbius support* of f, denoted $\mathcal{S}_f$, to be the set $\mathcal{S}_f := \left\{ S \subseteq [n] : \widetilde{f}(S) \neq 0 \right\}$. Define the *Möbius sparsity* of f, denoted $\mathrm{spar}(f)$, to be $\mathrm{spar}(f) := |\mathcal{S}_f|$.

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 $\mathsf{S} \cap \{x \in \{0, 1\}^n \mid \forall j = 1, \ldots, k, x_{i_j} = a_{i_j}\}$. Equivalently $\mathsf{D}^{\rightarrow}_{\mathrm{dt}}(f)$ is the minimum number of variables such that f can be expressed as a function of these variables. It is easy to see that if f is a total function that depends on all input variables, then $\mathsf{D}^{\rightarrow}_{\mathrm{dt}}(f) = n$.

---

[1] In the literature, OMB_n is typically defined with a 1 in the max 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 [7]. 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 4.8).

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. 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. Any set of monomials $\mathcal{S}$ whose evaluations determine f is called an *NAADT basis* for f. We also require the natural randomized and quantum analogues of non-adaptive AND decision tree complexity, denoted $\mathsf{RNAADT}(\cdot)$ and $\mathsf{QNAADT}(\cdot)$, respectively. Formal definitions of these measures can be found in Appendix A. We first note some simple observations about the non-adaptive AND decision tree complexity of Boolean functions.

▷ **Claim 2.4.** 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 an 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 $b_T \in \{0,1\}$. 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 2.5.** 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 an NAADT basis for f.

For the lower bound, let $\mathsf{NAADT}(f) = k$, and let $\mathcal{S} = \{S_1, \ldots, S_k\}$ be an NAADT basis for f. Claim 2.4 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 $f : \{0,1\}^n \to \mathbb{R}$ 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 [41] showed the following relationship between non-adaptive parity decision complexity of a Boolean function and its Fourier sparsity.

▶ **Theorem 2.6** ([41]). *Let $f : \{0,1\}^n \to \{-1,1\}$ 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.

Communication Complexity

The standard model of two-party communication complexity was introduced by Yao [49]. 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 $\mathsf{D}_{cc}(F)$. See, for example, [29], 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. We refer the reader to [46] for an introduction to quantum communication complexity. 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 $\mathsf{D}_{cc}^{\to}(\cdot)$ ($\mathsf{R}_{cc,\varepsilon}^{\to}(\cdot)$, $\mathsf{Q}_{cc,\varepsilon}^{\to}(\cdot)$, $\mathsf{Q}_{cc,\varepsilon}^{*,\to}(\cdot)$, 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 known that

$$\log \operatorname{rank}(M_F) \leq \mathsf{D}_{cc}(F) \leq O(\sqrt{\operatorname{rank}(M_F)} \log \operatorname{rank}(M_F)), \tag{6}$$

where $\operatorname{rank}(\cdot)$ denotes real rank. The first inequality is well known (see, for instance [29]), and the second inequality was shown by Lovett [32]. One of the most famous conjectures in communication complexity is the log-rank conjecture, due to Lovász and Saks [31], that proposes that the communication complexity of any Boolean function is polylogarithmic in its rank, i.e. the first inequality in Equation (6) is always tight up to a polynomial dependence.

Buhrman and de Wolf [8] observed that the Möbius sparsity of a Boolean function f equals the rank of the communication matrix of $f \circ \mathsf{AND}$. That is, for all Boolean functions $f : \{0, 1\}^n \to \{0, 1\}$,

$$\operatorname{spar}(f) = \operatorname{rank}(M_{f \circ \mathsf{AND}}). \tag{7}$$

In view of the first inequality in Equation (6), this yields

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

We require the definition of the Vapnik-Chervonenkis (VC) dimension [45].

▶ **Definition 2.7** (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 $F : \{0,1\}^n \times \{0,1\}^n$, denoted $\mathsf{VC}(F)$, is the maximum size of a shattered subset of columns of M_F.*

Klauck [26] showed that the one-way quantum communication complexity of a function F is bounded below by the VC-dimension of F.

▶ **Theorem 2.8** ([26, Theorem 3]). *Let $F : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ be a Boolean function. Then, $\mathsf{Q}^{\to}_{\mathsf{cc},\varepsilon}(F) \geq (1 - \mathbb{H}_{\mathsf{bin}}(\varepsilon))\mathsf{VC}(F)$ and $\mathsf{Q}^{*,\to}_{\mathsf{cc},\varepsilon}(F) \geq (1 - \mathbb{H}_{\mathsf{bin}}(\varepsilon))\mathsf{VC}(F)/2$.*

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, respectively, of functions composed with a small Inner Product gadget.

Quantum Complexity

Proof of Theorem 1.1. By Theorem 2.8, it suffices to show that $\mathsf{VC}(f \circ \mathsf{IP}) \geq n(b-1)$. Since f is a function that depends on all its input variables, the following holds. For each index $i \in [n]$, there exist inputs $z^{(i,0)} = z_1^{(i)}, \ldots, z_{i-1}^{(i)}, 0, z_{i+1}^{(i)}, \ldots, z_n^{(i)}$ and $z^{(i,1)} = z_1^{(i)}, \ldots, z_{i-1}^{(i)}, 1, z_{i+1}^{(i)}, \ldots, z_n^{(i)}$ such that $f(z^{(i,0)}) = v_i$ and $f(z^{(i,1)}) = 1 - v_i$. 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 $(0^{j-1}, 1, 0^{b-j})$. Consider the set of $n(b-1)$-many columns of $M_{f \circ \mathsf{IP}}$, 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 now show 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 $j \in [b]$, $x_{i,1} = 1$ and

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

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 v_i.

To complete the proof, we claim that the row of $M_{f \circ \mathsf{IP}}$ 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 $M_{f \circ \mathsf{IP}}(x, y^{(i,j)})$. 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 $v_i = 0$, 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 $v_i = 0$, 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 $z_1^{(i)}, \ldots, z_{i-1}^{(i)}, c_{i,j}, z_{i+1}^{(i)}, \ldots, z_n^{(i)}$ if $v_i = 0$ and $z_1^{(i)}, \ldots, z_{i-1}^{(i)}, 1 - c_{i,j}, z_{i+1}^{(i)}, \ldots, z_n^{(i)}$ if $v_i = 1$. By our definitions of $z^{(i,0)}, z^{(i,1)}$ and v_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. ◄

Deterministic Complexity

We now prove Theorem 1.3, which gives a lower bound on the deterministic one-way communication complexity of $f \circ \mathsf{IP}$ for partial functions f. A crucial ingredient of our proof is Theorem 1.4. 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 $f \circ \mathsf{IP}$ of complexity $\mathsf{D}_{cc}^{\rightarrow}(f \circ \mathsf{IP}) =: c \log q$. The theorem is trivially true if $c \geq n/30$ since $\mathsf{D}_{dt}^{\rightarrow}(f) \leq n$. In the remainder of the proof we assume that $c < n/30$. 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 $\mathcal{Z}$. We now invoke Theorem 1.4 with d set to $10c$. This is applicable since $d \leq n/3$ and the assumption $b \geq 2$ implies that $q \geq 3$. Theorem 1.4 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 $|\{i \in [n] \mid x_i^{(1)} = x_i^{(2)}\}| < 10c$. 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 $\mathsf{D}_{dt}^{\rightarrow}(f) \leq |\mathcal{I}| < 10c$. 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, $f \circ \mathsf{IP}(x^{(1)}, y) = f(z^{(1)})$ and $f \circ \mathsf{IP}(x^{(2)}, y) = f(z^{(2)})$. Since by assumption $f(z^{(1)}) \neq f(z^{(2)})$, we have that $f \circ \mathsf{IP}(x^{(1)}, y) \neq f \circ \mathsf{IP}(x^{(2)}, y)$. 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. ◄

▶ **Remark 3.1.** It can be seen that the proof of Theorem 1.3 also works when the inner gadget $g : \{0,1\}^{b_1} \times \{0,1\}^{b_2} \to \{0,1\}$ satisfies the following general property: There exists a subset X of $\{0,1\}^{b_1}$ (Alice's input in the gadget) such that:

- $|X| \geq 3$,
- for all $x_1 \neq x_2 \in X$ and all $b_1, b_2 \in \{0,1\}$, there exists $y \in \{0,1\}^{b_2}$ such that $g(x_1, y) = b_1$ and $g(x_2, y) = b_2$.

This is satisfied, for example, for the Addressing function on $\{0,1\}^{\log b + b}$ when $b \geq 4$ (see Definition A.1). For $g = \mathsf{IP}_b$, the set X equals $\{0,1\}^b \setminus \{0^b\}$.

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 2.5 shows that for all Boolean functions $f : \{0,1\}^n \to \{0,1\}$, $\log \text{spar}(f) \leq \text{NAADT}(f) \leq \text{spar}(f)$. 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 2.6. The Addressing function witnesses tightness of the quadratic gap in Theorem 2.6. This gives rise to the natural question of whether an analogous bound holds true in the Möbius-world: is it true for all Boolean functions f that $\text{NAADT}(f) = \widetilde{O}(\sqrt{\text{spar}(f)})$? Interestingly we show (see [33, Appendix A]) that the Addressing function already gives a negative answer to this question. In Claim 4.6 we observe that the function OMB_n witnesses tightness of the second inequality in Claim 2.5, that is, $\text{NAADT}(\text{OMB}_n) = \text{spar}(f)$ for even n (and $\text{NAADT}(\text{OMB}_n) = \text{spar}(f) - 1$ for odd n). We then use this same function to prove Theorem 1.7, which gives a maximal separation between $\text{QNAADT}(f)$ and $\text{D}_{\text{cc}}^{\rightarrow}(f \circ \text{AND}_2)$. Finally, we prove Theorem 1.8, which says that $\text{NAADT}(f)$ is at most quadratically large in $\text{D}_{\text{cc}}(f \circ \text{AND})$ for symmetric f.

Pattern Complexity and One-Way Communication Complexity

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

▷ **Claim 4.1.**   Let $f : \{0,1\}^n \to \{0,1\}$ be a Boolean function. Then $\text{D}_{\text{cc}}^{\rightarrow}(f \circ \text{AND}) = \lceil \log(\text{Pat}^{\text{M}}(f)) \rceil$.

Proof. Write the Möbius expansion of f as

$$f = \sum_{S \in \mathcal{S}_f} \widetilde{f}(S) \text{AND}_S. \tag{9}$$

Say $\text{Pat}^{\text{M}}(f) = k$. We first show that $\text{D}_{\text{cc}}^{\rightarrow}(f \circ \text{AND}) \leq \lceil \log k \rceil$ 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 $\{\text{AND}_S(x) : S \in \mathcal{S}_f\}$. Since Bob can compute $\{\text{AND}_S(y) : S \in \mathcal{S}_f\}$ without any communication, he can now compute the value of $f \circ \text{AND}(x, y)$ using the formula

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

It remains to show that $\text{D}_{\text{cc}}^{\rightarrow}(f \circ \text{AND}) \geq \lceil \log k \rceil$. Let $\text{D}_{\text{cc}}^{\rightarrow}(f \circ \text{AND}) = d$. 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

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

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

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

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

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

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

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

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 $S \in \mathcal{S}_f$. 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 $S \in \mathcal{S}_f$,

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

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

Next we show that the pattern complexity of f is bounded below by the Möbius sparsity of f.

▷ **Claim 4.2.** Let $f : \{0, 1\}^n \to \{0, 1\}$ be a Boolean function. Then $\mathsf{Pat}^{\mathsf{M}}(f) \geq \mathrm{spar}(f)$.

Proof. Recall that $\mathcal{S}_f$ denotes the Möbius support of f. For each $S \in \mathcal{S}_f$, define the input x^S to be the n-bit characteristic vector of the set S. We now show that each of these inputs induces a different pattern for f. Let $S_1 \neq S_2 \in \mathcal{S}_f$, with $|S_1| \geq |S_2|$. Since they are different sets, there must be an index $j \in S_1$ such that $j \notin S_2$. Note that $\mathsf{AND}_{S_1}(x^{S_1}) = 1$. On the other hand $x_j^{S_2} = 0$ implies $\mathsf{AND}_{S_1}(x^{S_2}) = 0$. Hence x^{S_1} and x^{S_2} induce different patterns. Since $\mathrm{spar}(f) = |\mathcal{S}_f|$, this completes the proof. ◁

From Claim 2.5 we know that $\mathrm{spar}(f) \geq \mathsf{NAADT}(f)$ and from Claim 4.1 we know that $\mathsf{D}_{\mathrm{cc}}^{\to}(f \circ \mathsf{AND}) = \lceil \log(\mathsf{Pat}^{\mathsf{M}}(f)) \rceil$. Along with Claim 4.2, these imply the following claim.

▷ **Claim 4.3.** Let $f : \{0, 1\}^n \to \{0, 1\}$ be a Boolean function. Then $\lceil \log(\mathsf{NAADT}(f)) \rceil \leq \mathsf{D}_{\mathrm{cc}}^{\to}(f \circ \mathsf{AND}) \leq \mathsf{NAADT}(f)$.

Proof. For the upper bound on $\mathsf{D}_{\mathrm{cc}}^{\to}(f \circ \mathsf{AND})$, let $\mathcal{S} = \{S_1, \dots, S_k\}$ be an NAADT basis for f. By Claim 2.4, 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 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 4.1.

For the lower bound, we have

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

where the equality follows from Claim 4.1, the first inequality follows from Claim 4.2 and the last inequality follows from Claim 2.5. ◁

The pattern complexity of f is trivially at most $2^{\mathrm{spar}(f)}$ since each pattern is a $\mathrm{spar}(f)$-bit string. Interestingly we show that there is no function for which this bound is tight.

▷ **Claim 4.4.** Let $f : \{0,1\}^n \rightarrow \{0,1\}$ be a Boolean function. Then $\mathsf{Pat}^{\mathsf{M}}(f) \leq 2^{(1-\Omega(1))\mathrm{spar}(f)}$.

We prove Claim 4.4 in Appendix B. Its proof proceeds by analyzing the identity $f^2 = f$ and using it to deduce "dependencies" between monomials in the Möbius support of f. The analogous relation in the Fourier-world has been nearly determined by Sanyal [41]; their main result (Theorem 2.6) essentially shows that the Fourier analog of pattern complexity of a Boolean function is at most exponential in the square root of its Fourier sparsity. This is a stronger bound than that in Claim 4.4, but the same bound cannot hold in the Möbius-world since the Addressing function witnesses $\mathsf{Pat}^{\mathsf{M}}(\mathsf{ADDR}_n) \geq 2^{\mathrm{spar}(\mathsf{ADDR}_n)^{\log_3 2}}$ (see [33, Appendix A]). Nevertheless we conjecture that a stronger bound than that of Claim 4.4 is possible.

▶ **Conjecture 4.5.** *Let* $f : \{0,1\}^n \rightarrow \{0,1\}$ *be a Boolean function. Then* $\mathsf{Pat}^{\mathsf{M}}(f) \leq 2^{\left(\mathrm{spar}(f)^{1-\Omega(1)}\right)}$.

Conjecture 4.5 would strengthen Theorem 1.9, showing that $\mathsf{D}_{cc}^{\rightarrow}(f \circ \mathsf{AND}) = \mathrm{rank}(M_{f \circ \mathsf{AND}})^{1-\Omega(1)}$.

Proof of Theorem 1.9. We have

$$\mathsf{D}_{cc}^{\rightarrow}(f \circ \mathsf{AND}) = \lceil \log(\mathsf{Pat}^{\mathsf{M}}(f)) \rceil \leq (1 - \Omega(1))\mathrm{spar}(f) \leq (1 - \Omega(1))\mathrm{rank}(M_{f \circ \mathsf{AND}}),$$

where the equality holds by Claim 4.1, the first inequality follows from Claim 4.4 and the last inequality holds by Equation (7). ◀

Our results regarding the one-way communication complexity of $f \circ \mathsf{AND}$ use the Booleanness of f to bring out mathematical insights about the dependencies of monomials in the Möbius support of f. These dependencies enable us to establish interesting bounds on the pattern complexity of f. We hope that pattern complexity will find more use in future research.

Deterministic AND 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 4.6.** Let n be a positive integer. Then $\mathsf{NAADT}(\mathsf{OMB}_n) = n$. Moreover, $\mathrm{spar}(\mathsf{OMB}_n) = n$ if n is even, and $\mathrm{spar}(\mathsf{OMB}_n) = n + 1$ if n is odd.

Proof of Claim 4.6. Write the polynomial representation of OMB_n as $\mathsf{OMB}_n(x) =$

$$(1 - x_n) \cdot 0 + x_n(1 - x_{n-1}) \cdot 1 + x_n x_{n-1} \mathsf{OMB}_{n-2}(x_1, \ldots, x_{n-2}) \quad \text{if } n \text{ is even, or} \quad (10)$$

$$(1 - x_n) \cdot 1 + x_n(1 - x_{n-1}) \cdot 0 + x_n x_{n-1} \mathsf{OMB}_{n-2}(x_1, \ldots, x_{n-2}) \quad \text{if } n \text{ is odd.} \quad (11)$$

The Möbius support of OMB_n equals $\{\{j,\ldots,n\} : j \leq n\} \cup \{\emptyset\}$ if n is odd, and $\{\{j,\ldots,n\} : j \leq n\}$ 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 2.4, 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 implies that $|\mathcal{S}| \geq n$. Equality holds since $\mathsf{NAADT}(f) \leq n$ for any Boolean function $f : \{0,1\}^n \to \{0,1\}$. ◁

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 (adaptive) AND-decision tree that makes $O(\log n)$ AND-queries in the worst case. 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 $\mathsf{D}_{\mathrm{cc}}^{\to}(\mathsf{OMB}_n \circ \mathsf{AND})$ and $\mathsf{NAADT}(\mathsf{OMB}_n)$.

▷ **Claim 4.7.** Let n be a positive integer. Then $\mathsf{D}_{\mathrm{cc}}^{\to}(\mathsf{OMB}_n \circ \mathsf{AND}) = \lceil \log(n+1) \rceil$.

Proof. From Equation (10) we have that the Möbius support of OMB_n equals the set $\mathcal{S} = \{\{n\}, \{n-1,n\}, \ldots, \{n, n-1, \ldots, 1\}\}$ if n is an even integer, and equals the set $\mathcal{S} = \{\emptyset, \{n\}, \{n-1,n\}, \ldots, \{n, n-1, \ldots, 1\}\}$ if n is an odd integer. It is easy to verify that the only possible Möbius patterns attainable (ignoring the empty set since it always evaluates to 1) 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 4.1 implies $\mathsf{D}_{\mathrm{cc}}^{\to}(\mathsf{OMB}_n \circ \mathsf{AND}) = \lceil \log(n+1) \rceil$. ◁

We obtain our main result of this section, which follows from Claim 4.6 and Claim 4.7.

▶ **Theorem 4.8.** *Let n be a positive integer. Then $\mathsf{NAADT}(\mathsf{OMB}_n) = n$ and $\mathsf{D}_{\mathrm{cc}}^{\to}(\mathsf{OMB}_n \circ \mathsf{AND}) = \lceil \log(n+1) \rceil$.*

Quantum Complexity

We prove that even the quantum non-adaptive AND decision tree complexity of OMB_n is $\Omega(n)$. We refer the reader to Section A for necessary preliminaries of quantum computing. In view of the small one-way communication complexity of $\mathsf{OMB}_n \circ \mathsf{AND}$ from Claim 4.7, Theorem 1.7 then follows.

▶ **Theorem 4.9.** *Let n be a positive integer. Then $\mathsf{QNAADT}(\mathsf{OMB}_n) = \Omega(n)$.*

Before we prove this theorem, we introduce an auxiliary function and state some properties of it that are of use to us.

▶ **Definition 4.10.** *Let n be a positive integer. Define the set $\mathsf{S} \subset \{0,1\}^n$ to be $\mathsf{S} = \left\{ x \in \{0,1\}^n : x = 0^i 1^{n-i} \text{ for some } i \in [n] \right\}$. Define the partial function $\mathsf{OMB}'_n : \mathsf{S} \to \{0,1\}$ by $\mathsf{OMB}'_n(x) = \mathsf{OMB}_n(x)$.*

▷ **Claim 4.11.** Let n be a positive integer. Then $\mathsf{RNAADT}(\mathsf{OMB}'_n) = \mathsf{R}_{\mathrm{dt}}^{\to}(\mathsf{OMB}'_n)$ and $\mathsf{QNAADT}(\mathsf{OMB}'_n) = \mathsf{Q}_{\mathrm{dt}}^{\to}(\mathsf{OMB}'_n)$.

We require the following result, which follows implicitly from a result of Montanaro [34].

▶ **Theorem 4.12.** *Let* $\mathsf{S} \subseteq \{0,1\}^n$, $I \subseteq [n]$ *and* $f : \mathsf{S} \to \{0,1\}$ *be such that for all* $i \in I$ *there exists* $x \in \mathsf{S}$ *such that* $f(x \oplus e_i) = 1 - f(x)$. *Then* $\mathsf{Q}_{\mathrm{dt}}^{\to}(f) = \Omega(|I|)$.

We defer the proofs of Claim 4.11 and Theorem 4.12 to the full version of our paper [33, Section 4.3].

Proof of Theorem 4.9. Clearly $\mathsf{QNAADT}(\mathsf{OMB}_n) \geq \mathsf{QNAADT}(\mathsf{OMB}_n')$. Claim 4.11 implies that $\mathsf{QNAADT}(\mathsf{OMB}_n') = \mathsf{Q}_{\mathrm{dt}}^{\to}(\mathsf{OMB}_n')$. Recall that the domain of OMB_n' equals $S = \{x \in \{0,1\}^n : x = 0^i 1^{n-i} \text{ for some } i \in [n]\}$. By definition, $\mathsf{OMB}_n'(0^i 1^{n-i}) \neq \mathsf{OMB}_n'(0^{i-1} 1^{n-i+1})$ for all $i \in [n]$. Thus Theorem 4.12 is applicable with $I = [n]$ and $f = \mathsf{OMB}_n'$. Combining the above, we have $\mathsf{QNAADT}(\mathsf{OMB}_n) \geq \mathsf{QNAADT}(\mathsf{OMB}_n') = \mathsf{Q}_{\mathrm{dt}}^{\to}(\mathsf{OMB}_n') = \Omega(n)$. ◀

Proof of Theorem 1.7. It follows from Claim 4.7 and Theorem 4.9. ◀

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 [8]. 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: $\mathsf{switch}(f) := \min \{k : f \text{ is a constant on all } x \text{ such that } |x| < n - k\}$.

▷ **Claim 4.13 ([8, 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 [14]. Also see [12] and the references therein.

▶ **Theorem 4.14.** *Let* $f : \{0,1\}^n \to \{0,1\}$ *be a symmetric 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 4.14 in Appendix C 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 4.13 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

$$\mathsf{NAADT}(f) = O\left(\log^2 \binom{n}{k}\right) = O(\log^2(\mathrm{spar}(f))) = O(\mathsf{D}_{\mathrm{cc}}(f \circ \mathsf{AND})^2),$$

where the first equality follows from Theorem 4.14, the second from Claim 4.13, and the third from Equation (8). ◀

References

1   Rudolf Ahlswede and Levon H Khachatrian. The diametric theorem in hamming spaces—optimal anticodes. *Advances in Applied mathematics*, 20(4):429–449, 1998.

2   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.

3   Richard Beigel. The polynomial method in circuit complexity. In *Proceedings of the Eighth Annual Structure in Complexity Theory Conference*, pages 82–95, 1993. `doi:10.1109/SCT.1993.336538`.

4   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.

5   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.

6   Ethan Bernstein and Umesh V. Vazirani. Quantum complexity theory. *SIAM J. Comput.*, 26(5):1411–1473, 1997. `doi:10.1137/S0097539796300921`.

7   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. `doi:10.1109/CCC.2007.18`.

8   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. `doi:10.1109/CCC.2001.933879`.

9   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. `doi:10.1016/j.ic.2014.12.003`.

10 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. `doi:10.1137/19M1310153`.

11 Arkadev Chattopadhyay, Michal Koucký, Bruno Loff, and Sagnik Mukhopadhyay. Simulation theorems via pseudo-random properties. *Comput. Complex.*, 28(4):617–659, 2019. `doi:10.1007/s00037-019-00190-7`.

12 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. `doi:10.1007/s10878-007-9083-3`.

13 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.

14 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.

15 Peter Frankl and Norihide Tokushige. The Erdos-Ko-Rado theorem for integer sequences. *Comb.*, 19(1):55–63, 1999. `doi:10.1007/s004930050045`.

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

17 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.

18 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. `doi:10.4230/LIPIcs.CCC.2016.5`.

19   Mika Göös, Shachar Lovett, Raghu Meka, Thomas Watson, and David Zuckerman. Rectangles are nonnegative juntas. *SIAM J. Comput.*, 45(5):1835–1869, 2016.

20   Mika Göös, Toniann Pitassi, and Thomas Watson. Deterministic communication vs. partition number. *SIAM J. Comput.*, 47(6):2435–2450, 2018. `doi:10.1137/16M1059369`.

21   Mika Göös, Toniann Pitassi, and Thomas Watson. Query-to-communication lifting for BPP. *SIAM J. Comput.*, 49(4), 2020. Earlier version in FOCS'17. `doi:10.1137/17M115339X`.

22   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. `doi:10.1137/17M1136869`.

23   Kaave Hosseini, Shachar Lovett, and Grigory Yaroslavtsev. Optimality of linear sketching under modular updates. In Amir Shpilka, editor, *34th Computational Complexity Conference, CCC 2019, July 18-20, 2019, New Brunswick, NJ, USA*, volume 137 of *LIPIcs*, pages 13:1–13:17. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019.

24   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.

25   Sampath Kannan, Elchanan Mossel, Swagato Sanyal, and Grigory Yaroslavtsev. Linear sketching over f_2. In Rocco A. Servedio, editor, *33rd Computational Complexity Conference, CCC 2018, June 22-24, 2018, San Diego, CA, USA*, volume 102 of *LIPIcs*, pages 8:1–8:37. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018.

26   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. `doi:10.1145/335305.335396`.

27   Alexander Knop, Shachar Lovett, Sam McGuire, and Weiqiang Yuan. Log-rank and lifting for and-functions. In *Proceedings of 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 197–208. ACM, 2021. `doi:10.1145/3406325.3450999`.

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

29   Eyal Kushilevitz and Noam Nisan. *Communication complexity*. Cambridge University Press, 1997.

30   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.

31   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. `doi:10.1109/SFCS.1988.21924`.

32   Shachar Lovett. Communication is bounded by root of rank. *J. ACM*, 63(1):1:1–1:9, 2016. `doi:10.1145/2724704`.

33   Nikhil S. Mande, Suhail Sherif, and Swagato Sanyal. One-way communication complexity and non-adaptive decision trees. *CoRR*, abs/2105.01963, 2021. `arXiv:2105.01963`.

34   Ashley Montanaro. Nonadaptive quantum query complexity. *Inf. Process. Lett.*, 110(24):1110–1113, 2010. `doi:10.1016/j.ipl.2010.09.009`.

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

36   Ashley Montanaro and Tobias Osborne. On the communication complexity of XOR functions. *CoRR*, abs/0909.3392, 2009. `arXiv:0909.3392`.

37   Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2000. URL: `https://books.google.com/books?id=-s4DEy7o-a0C`.

38   Ran Raz and Pierre McKenzie. Separation of the monotone NC hierarchy. *Comb.*, 19(3):403–435, 1999. `doi:10.1007/s004930050062`.

39   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.

40 Swagato Sanyal. One-way communication and non-adaptive decision tree. *Electron. Colloquium Comput. Complex.*, 24:152, 2017. URL: `https://eccc.weizmann.ac.il/report/2017/152`.

41 Swagato Sanyal. Fourier sparsity and dimension. *Theory of Computing*, 15(11):1–13, 2019. `doi:10.4086/toc.2019.v015a011`.

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

43 Alexander A. Sherstov. Approximating the AND-OR tree. *Theory Comput.*, 9:653–663, 2013. `doi:10.4086/toc.2013.v009a020`.

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

45 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.

46 Ronald de Wolf. Quantum communication and complexity. *Theoretical Computer Science*, 287(1):337–353, 2002. `doi:10.1016/S0304-3975(02)00377-8`.

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

48 Xiaodi Wu, Penghui Yao, and Henry S. Yuen. Raz-mckenzie simulation with the inner product gadget. *Electron. Colloquium Comput. Complex.*, 24:10, 2017. URL: `https://eccc.weizmann.ac.il/report/2017/010`.

49 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. `doi:10.1145/800135.804414`.

50 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 Preliminaries

▶ **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.

▶ **Definition A.2** (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* $\mathcal{S} = \{\{S_1, \ldots, S_k\} : S_i \subseteq [n] \text{ for all } i \in [k]\}$ *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 A.3** (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* $\mathcal{S} = \{\{S_1, \ldots, S_k\} : S_i \subseteq [n] \text{ for all } i \in [k]\}$ *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 an* NAADT basis *for* f.

▶ **Definition A.4** (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 cost 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 refer the reader to [37] for the basics of quantum computing.

▶ **Definition A.5** (Quantum non-adaptive AND decision tree complexity). *A quantum non-adaptive AND decision tree of cost c is a query algorithm that works with a state space $|S_1, \ldots, S_c\rangle|b\rangle|w\rangle$, where each $S_j \subseteq [n]$, $b \in \{0,1\}^c$ and the last register captures a workspace of an arbitrary dimension. It is specified by a starting state $|\psi\rangle$ and a projective measurement $\{\Pi, I - \Pi\}$. For an input $x \in \{0,1\}^n$, the action of the non-adaptive query oracle $O_x^{\otimes c}$ is captured by its action on the basis states, described below.*

$$O_x^{\otimes c}|S_1, \ldots, S_c\rangle|b_1, \ldots, b_c\rangle|w\rangle \mapsto |S_1, \ldots, S_c\rangle|b_1 \oplus \mathsf{AND}_{S_1}(x), \ldots, b_c \oplus \mathsf{AND}_{S_c}(x)\rangle|w\rangle.$$

We use O_x to refer to this oracle since c is already unambiguously determined by the state space. The algorithm accepts x with probability $\|\Pi O_x|\psi\rangle\|^2$.

Define the quantum non-adaptive AND decision tree complexity *of $f : \{0,1\}^n \to \{0,1\}$, denoted by $\mathsf{QNAADT}(f)$, to be the minimum cost of a quantum non-adaptive AND decision tree that outputs the correct value of $f(x)$ with probability at least $2/3$ for all $x \in \{0,1\}^n$.*

The quantum non-adaptive query complexity, denoted $\mathsf{Q}_{\mathrm{dt}}^{\rightarrow}$, is defined similarly, the only difference being that the sets $S_1, \ldots, S_c$ are restricted to be singletons. Montanaro [34] observed that $\mathsf{Q}_{\mathrm{dt}}^{\rightarrow}(f) = \Omega(n)$ for all total Boolean functions $f : \{0,1\}^n \to \{0,1\}$ that depend on all input bits. Our proof of Theorem 4.9 uses ideas from their proof.

B   Proof of Claim 4.4

In this section we prove Claim 4.4, restated below.

▷ **Claim B.1** (Restatement of Claim 4.4). Let $f : \{0,1\}^n \to \{0,1\}$ be a Boolean function. Then $\mathsf{Pat}^{\mathsf{M}}(f) \leq 2^{(1-\Omega(1))\mathrm{spar}(f)}$.

Our proof of Claim 4.4 relies on the following observation about the structure of the Möbius support of any Boolean function.

▷ **Claim B.2.** Let $f : \{0,1\}^n \to \{0,1\}$ be a Boolean function with Möbius support $\mathcal{S}_f$. For any two distinct sets $S, T \in \mathcal{S}_f$ there exists a set of "partners" $p(\{S,T\}) \subseteq \mathcal{S}_f$ such that
- $p(\{S,T\}) \neq \{S,T\}$,
- $|p(\{S,T\})| = 2$ if $S \cup T \notin \mathcal{S}_f$ and $|p(\{S,T\})| = 1$ if $S \cup T \in \mathcal{S}_f$, and
- $\bigcup_{U \in p(\{S,T\})} U = S \cup T$.

Proof. Let $\sum_{S \in \mathcal{S}_f} \widetilde{f}(S)\mathsf{AND}_S$ be the Möbius expansion of f. Since f has range $\{0,1\}$, we know that $f = f^2$. However,

$$f^2 = \left(\sum_{S \in \mathcal{S}_f} \widetilde{f}(S)\mathsf{AND}_S\right)\left(\sum_{T \in \mathcal{S}_f} \widetilde{f}(T)\mathsf{AND}_T\right) = \sum_{W \subseteq [n]}\left(\sum_{S,T \subseteq [n]: S \cup T = W} \widetilde{f}(S)\widetilde{f}(T)\right)\mathsf{AND}_W.$$

Since the Möbius expansion of f is unique, we can compare the two expansions to see that for all sets $W \subseteq [n]$,

$$\widetilde{f}(W) = \sum_{S,T \subseteq [n]: S \cup T = W} \widetilde{f}(S)\widetilde{f}(T). \tag{12}$$

As a consequence we have the following structure. Let $S \neq T \in \mathcal{S}_f$ such that $S \cup T \notin \mathcal{S}_f$. Since $\widetilde{f}(S \cup T) = 0$, the summation corresponding to $W = S \cup T$ in Equation (12) must have at least one non-zero summand apart from $\widetilde{f}(S)\widetilde{f}(T)$. Hence there must exist $U \neq V \in \mathcal{S}_f$ such that $\{S,T\} \neq \{U,V\}$ and $U \cup V = S \cup T$. We choose an arbitrary such pair $\{U,V\}$ and define $p(\{S,T\}) = \{U,V\}$. For $S,T \in \mathcal{S}_f$ such that $S \cup T \in \mathcal{S}_f$, let $p(\{S,T\})$ be defined as $\{S \cup T\}$. It clearly satisfies the necessary conditions. $\lhd$

▶ **Observation B.3.** *Let $f : \{0,1\}^n \to \{0,1\}$ be a Boolean function with Möbius support $\mathcal{S}_f$. For any two distinct sets $S,T \in \mathcal{S}_f$, let $p(\{S,T\}) \subseteq \mathcal{S}_f$ be as in Claim B.2. Then for any pattern $P \in \{0,1\}^{\mathcal{S}_f}$,*

$$P_S \cdot P_T = \prod_{W \in p(\{S,T\})} P_W.$$

Proof. Let P be a pattern in $\{0,1\}^{\mathcal{S}_f}$. There must exist an $x \in \{0,1\}^n$ such that for all sets $W \in \mathcal{S}_f$, $P_W = \mathsf{AND}_W(x)$. Since $S \cup T = \bigcup_{W \in p(\{S,T\})} W$, we have $P_S \cdot P_T = \mathsf{AND}_{S \cup T}(x) = \prod_{W \in p(\{S,T\})} P_W$. ◀

Proof of Claim 4.4. We analyze the pattern complexity of f in iterations. To define these iterations, we define a sequence of subsets of $\mathcal{S}_f$, described in Algorithm 1.

■ **Algorithm 1** Defining the Iterations.

Initialize $\mathcal{T}_0 \leftarrow \emptyset, i \leftarrow 0$.
while $|\mathcal{T}_i| \leq \mathrm{spar}(f) - 2$ **do**
  Choose S,T with $S \neq T$ from $\mathcal{S}_f \setminus \mathcal{T}_i$.
  Set $\mathcal{T}_{i+1} \leftarrow \mathcal{T}_i \cup \{S,T\} \cup p(\{S,T\})$.
  Set $i \leftarrow i + 1$.
end
Set $\mathsf{num\_iterations} \leftarrow i$.
Set $\mathcal{T}_{\mathsf{num\_iterations}+1} \leftarrow \mathcal{S}_f$.

For $i \in \{0, \ldots, \mathsf{num\_iterations} + 1\}$, define the partial patterns

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

We now show that

$$\forall j \in \{0, \ldots, \mathsf{num\_iterations}\}, \ |\mathcal{P}_j| \leq \left(\frac{15}{16}\right)^j 2^{|\mathcal{T}_j|}. \tag{13}$$

We prove this by induction. Equation (13) is true when $j = 0$ since both sides are 1. Now let $i > 0$ and assume as our induction hypothesis that Equation (13) is true when $j = i - 1$. As our inductive step, we will prove that for every partial pattern $P \in \mathcal{P}_{i-1}$, the number of partial patterns $Q \in \mathcal{P}_i$ that extend P (in the sense that Q restricted to indices in $\mathcal{T}_{i-1}$ is equal to P) is at most $(15/16)2^{|\mathcal{T}_i|-|\mathcal{T}_{i-1}|}$. Since every partial pattern in $\mathcal{P}_i$ is an extension of a partial pattern in $\mathcal{P}_{i-1}$, this would imply that $|\mathcal{P}_i| \leq (15/16)2^{|\mathcal{T}_i|-|\mathcal{T}_{i-1}|}|\mathcal{P}_{i-1}|$. Along with our induction hypothesis, this will prove Equation (13) for $j = i$, and hence for all j.

To prove the inductive step, consider any partial pattern $P \in \mathcal{P}_{i-1}$. Let S, T be the sets chosen when constructing $\mathcal{T}_i$ from $\mathcal{T}_{i-1}$. We know from Observation B.3 that any partial pattern $Q \in \mathcal{P}_i$ must satisfy $Q_S \cdot Q_T = \prod_{W \in p(\{S,T\})} Q_W$. Consider the extension Q' of P that sets $Q'_W = 1$ for all $W \in p(\{S,T\})$ and $Q'_W = 0$ for all $W \in \{S,T\} \setminus p(\{S,T\})$. Clearly such a Q' does not satisfy $Q'_S \cdot Q'_T = \prod_{W \in p(\{S,T\})} Q'_W$. Hence of the $2^{|\mathcal{T}_i| - |\mathcal{T}_{i-1}|}$ possible extensions of P, at most $2^{|\mathcal{T}_i| - |\mathcal{T}_{i-1}|} - 1$ will be in $\mathcal{P}_i$. Since $|\mathcal{T}_i| - |\mathcal{T}_{i-1}| \leq 4$, we can conclude that

$$|\mathcal{P}_i| \leq (2^{|\mathcal{T}_i| - |\mathcal{T}_{i-1}|} - 1)|\mathcal{P}_{i-1}| \leq (15/16)2^{|\mathcal{T}_i| - |\mathcal{T}_{i-1}|}|\mathcal{P}_{i-1}|.$$

This proves Equation (13).

Finally, note that the while loop in Algorithm 1 quits when $|\mathcal{T}_i| \geq \mathrm{spar}(f) - 1$. Hence num_iterations $\geq (\mathrm{spar}(f) - 1)/4$. If it quits with $|\mathcal{T}_i| = \mathrm{spar}(f)$, then Equation (13) implies that $\mathsf{Pat}^{\mathsf{M}}(f) \leq (15/16)^{(\mathrm{spar}(f)-1)/4} 2^{\mathrm{spar}(f)} \leq 1.02 \cdot 2^{0.98\mathrm{spar}(f)}$. If it quits with $|\mathcal{T}_i| = \mathrm{spar}(f) - 1$, then each of the partial patterns in $\mathcal{P}_i$ can have at most two extensions to actual patterns of f. Hence even in this case $\mathsf{Pat}^{\mathsf{M}}(f) \leq 2.04 \cdot 2^{0.98\mathrm{spar}(f)}$. ◁

In fact with a more careful analysis (see [33, Proof of Claim 4.4]) we obtain an upper bound of $\mathsf{Pat}^{\mathsf{M}}(f) \leq 2^{((\log 6)/3)\mathrm{spar}(f)+1} \approx 2^{0.86\mathrm{spar}(f)+1}$.

C   On Non-Adaptive AND Decision Trees for Symmetric Functions

Recall Theorem 4.14, restated below.

▶ **Theorem C.1** (Restatement of Theorem 4.14). *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 C.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.

$$\forall i_1, \ldots, i_{k+1} \in [n], j \in [k+1], \exists X \in \mathcal{X} \text{ such that } i_j \in X, i_\ell \notin X \text{ for all } \ell \neq j. \tag{14}$$

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 fixed $i_1, \ldots, i_{k+1} \in [n]$, $j \in [k+1]$ and $t \in [w]$,

$$\Pr_{X_t}[i_j \in X_t \text{ and } i_\ell \notin X_t \text{ for all } \ell \neq j] = \frac{1}{2k} \cdot \left(1 - \frac{1}{2k}\right)^k \geq \frac{1}{2k \cdot e}, \tag{15}$$

where the last inequality uses the fact that $k \geq 1$ and the standard inequality that $1 - x \geq e^{-2x}$ for all $0 \leq x \leq 1/2$. Thus Equation (15) implies that for fixed $i_1, \ldots, i_{k+1} \in [n]$ and $j \in [k+1]$,

$$\Pr_{\mathcal{X}}[\nexists X \in \mathcal{X} : i_j \in X \text{ and } i_\ell \notin X \text{ for all } \ell \neq j] \leq \left(1 - \frac{1}{2k \cdot e}\right)^w \leq \exp(-w/(2ke)). \tag{16}$$

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

$$\Pr_{\mathcal{X}}[\forall i_1, \ldots, i_{k+1} \in [n] \text{ and } j \in [k+1], \exists X \in \mathcal{X} : i_j \in X \text{ and } i_\ell \notin X \text{ for all } \ell \neq j]$$

$$\geq 1 - \binom{n}{k+1} \cdot (k+1) \cdot \exp(-w/(2ke)). \tag{17}$$

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

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

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

$$\iff w > 2ke \left(\log(k+1) + \log \binom{n}{k+1} \right).$$

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

$$w \geq 2e \log \binom{n}{k} \left(2 \log \binom{n}{k+1} \right). \tag{18}$$

By standard binomial inequalities we have $\log \binom{n}{k+1} \leq (k+1) \log(ne/(k+1))$, and $\log \binom{n}{k} > k \log(n/k)$. Next, since $k+1 \leq 2k$ for $k \geq 1$ and $ne/(k+1) < n^3/k^3$ for $k \in \{1, 2, \ldots, n/2\}$, Equation (18) 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 (17) is strictly positive. This proves the claim.

$\triangleleft$

Proof of Theorem 4.14. Let f be a symmetric function with $\mathsf{switch}(f) = k < n/2$, and let $\mathcal{X}$ be as in Claim C.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 2 below).

■ **Algorithm 2** NAADT for f.

Input: $x \in \{0,1\}^n$
1. Let $\mathcal{X}$ be as obtained from Claim C.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 as follows:

- If $P_x = P_y$ for some $|y| \geq n - k$, then x must equal y by the above. In this case we output the correct value since we have learned x.
- If $P_x \neq P_y$ for any $|y| \geq n - k$, then $|x| < n - k$. Since f evaluates to 0 on all such inputs, we output the correct value in this case.

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 C.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. ◀

▶ Remark C.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).