Electronic Colloquium on Computational Complexity, Report No. 101 (2025)

Exact versus Approximate Representations of Boolean Functions in the De Morgan Basis

Arkadev Chattopadhyay * Yogesh Dahiya † Shachar Lovett ‡

Abstract

A seminal result of Nisan and Szegedy (STOC, 1992) shows that for any total Boolean function, the degree of the real polynomial that computes the function, and the minimal degree of a real polynomial that point-wise approximates the function, are at most polynomially separated. Extending this result from degree to other complexity measures like sparsity of the polynomial representation, or total weight of the coefficients, remains poorly understood.

In this work, we consider this problem in the De Morgan basis, and prove an analogous result for the sparsity of the polynomials at a logarithmic scale. Our result further implies that the exact ℓ_1 norm and its approximate variant are also similarly related to each other at a log scale. This is in contrast to the Fourier basis, where the analog of our results are known to be false.

Our proof is based on a novel random restriction method. Unlike most existing random restriction methods used in complexity theory, our random restriction process is adaptive and is based on how various complexity measures simplify during the restriction process.

1 Introduction

Polynomial representations of Boolean functions have been invaluable in theoretical computer science and discrete mathematics. While the representation could use any field, in this work we consider only polynomials over the reals. Two bases are particularly prominent.

The first arises naturally by viewing the domain of Boolean functions as $\{0,1\}^n$, and hence, every multilinear monomial just represents the Boolean AND of a subset of variables. This is known as the *De Morgan basis*. The other basis comes about by viewing the domain as $\{1,-1\}^n$ which is a simple linear transformation of $\{0,1\}^n$ that maps $0 \mapsto 1$ and $1 \mapsto -1$. In this basis, called the *Fourier basis*, each monomial represents the Boolean parity of a subset of variables.

Two natural complexity measures show up in either basis: the degree and sparsity of the representation. As every Boolean function has a unique representation in either basis, it is usually quite straightforward to determine the degree and sparsity of the unique representation for a function f, which we denote by $\deg(f)$ and $\mathrm{spar}(f)$ in the De Morgan basis, and by $\deg^{\oplus}(f)$, and $\mathrm{spar}^{\oplus}(f)$ in the Fourier basis. The linear invertible mapping from one basis to the other ensures that for every f, $\deg(f) = \deg^{\oplus}(f)$. But sparsity can be very sensitive to the basis chosen. For example, the n-bit AND function has sparsity 1 in the De Morgan basis and 2^n in the Fourier basis; and the n-bit Parity function has sparsity 1 in the Fourier basis and 2^n in the De Morgan basis. We understand reasonably satisfactorily exact polynomial representations of Boolean functions. However, when we turn to approximations, the picture becomes subtler.

Classical approximation theory deals with polynomials that point-wise approximate functions. In an influential work, Nisan and Szegedy [44] introduced this notion to the study of Boolean functions. In particular, they defined the complexity measure of approximate degree of a Boolean function f, denoted by $\widetilde{\deg}(f)$, to be the smallest degree needed by a polynomial to point-wise approximate f to within a constant distance that is, by default, taken to be $1/3$. Observe that the same reasoning as

*TIFR, Mumbai. Email: arkadev.c@tifr.res.in. Supported by the Department of Atomic Energy, Govmt. of India, under project #RTI4001 and by a Google India Faculty Award.

†TIFR, Mumbai. Email: yogesh.dahiya@tifr.res.in. Supported by the Department of Atomic Energy, Govmt. of India, under project #RTI4001 and the R. Narasimhan postdoctoral fellowship.

‡UC San Diego. Email: slovett@ucsd.edu. Supported by Simons Investigator Award #929894 and NSF award CCF-2425349.

ISSN 1433-8092

applied above for exact degree implies that approximate degree of a function is also a measure that is independent of the basis. The notion of approximate degree has had tremendous impact in computer science as it is related to many other complexity measures including the randomized and quantum query complexity of f [2,5,15–17] and the quantum and classical communication complexity of appropriately lifted functions [6,14,19,42,46,48–50]. It has also found applications in learning theory [36,38], differential privacy [18,51], secret sharing [10,11], and many other areas. Unlike exact degree, getting tight bounds on approximate degree often turns out to be challenging. However, Nisan and Szegedy proved a remarkable structural result that for every total Boolean function f, the approximate and exact degree of f are polynomially related to each other. One of the striking applications of this result is the polynomial equivalence of quantum and classical query models for total Boolean functions, first derived by Beals et al. [5]. In a much more recent work, building upon Huang's breakthrough proof [33] of the sensitivity conjecture, Aaronson et al. [1] finally gave a tight relationship between the two measures by showing that $\deg(f) = O(\widetilde{\deg}(f))^2$. The tightness is witnessed by the Boolean AND and OR functions.

Given Nisan and Szegedy's result, one naturally wonders if approximation could reduce *sparsity* needed for total functions. In the Fourier basis, it is known that approximation does reduce sparsity exponentially. For instance, the Fourier sparsity of the n-bit AND function is 2^n. However, it can be shown that its Fourier approximate-sparsity is $O(n^2)$ (implicit in Bruck and Smolesnky [12, appendix] and explicit in [23, Lemma 2.8]). Surprisingly, the question if approximation helps significantly in the De Morgan basis remained unaddressed. Very recently, Knop et al. [39] conjectured that in the De Morgan basis, approximation should not significantly reduce sparsity for any total Boolean function. Our main result, stated below, confirms this conjecture. We denote by $\widetilde{\mathrm{spar}}(f)$, the approximate-sparsity of f in the De Morgan basis.

Theorem 1.1 (Main Theorem). *For every total Boolean function $f : \{0,1\}^n \to \{0,1\}$, we have*

$$\log(spar(f)) = O\big(\log(\widetilde{spar}(f))^2 \cdot \log n\big).$$

Before we continue, let us remark on the tightness of this result.

Remark 1.2. *The n-bit* OR *function has sparsity $2^n - 1$ and approximate sparsity $2^{O(\sqrt{n}\log n)}$, showing that exponential gaps may exist between the two measures in the absolute scale. It is thus necessary to consider the log scale as done in Theorem 1.1 for seeking polynomial relationship between the two measures. The example of* OR *also demonstrates the tightness of our bound up to poly-logarithmic factors. Finally, the appearance of the ambient dimension n in our result is unavoidable. Consider the function $\mathrm{THR}^n_{n-1} : \{0,1\}^n \to \{0,1\}$ defined by*

$$\mathrm{THR}^n_{n-1}(x) = 1 \quad \textit{if and only if} \quad |x| \geqslant n-1,$$

namely, the function evaluates to 1 if the input has at most one zero. It's simple to verify that its exact sparsity is $n+1$, and we show in section 3.3 that its approximate sparsity is $O(\log n)$, implying that an additive $O(\log n)$ or multiplicative $O\left(\frac{\log n}{\log\log n}\right)$ factor is necessary in Theorem 1.1.

One of the motivations of Nisan and Szegedy to study approximate degree was to relate this measure with decision tree complexity. Let $\mathrm{D}^{dt}(f)$ and $\mathrm{R}^{dt}(f)$ denote the deterministic and randomized bounded-error decision tree complexities of f respectively. Their result, along with the recent improvement of [1] yields the following relationship.

Theorem 1.3 (Nisan-Szegedy [44, Theorem 1.5] + Aaronson et al. [1]). *For every total Boolean function f, the following holds:*

$$\widetilde{deg}(f) \leqslant R^{dt}(f) \leqslant D^{dt}(f) \leqslant O(\widetilde{deg}(f)^4).$$

Just as $\widetilde{\deg}(f)$ lower bounds $\mathrm{R}^{dt}(f)$, it is straightforward to verify that $\log(\widetilde{\mathrm{spar}}(f))$, up to an additive $\log n$ term, lower bounds the randomized AND-decision tree (ADT) complexity of f. In an ADT, each internal node queries the AND of a subset of variables. ADT's have connections to combinatorial group testing algorithms and have also been the subject of several recent works [9,20,40,43]. We denote the deterministic and randomized ADT complexities of f by $\mathrm{D}^{\wedge dt}(f)$ and $\mathrm{R}^{\wedge dt}(f)$ respectively.

Combining our main result with the recent result of Knop et al. [40] yields the following ADT analog of Theorem 1.3.

Theorem 1.4. *For every total Boolean function $f : \{0,1\}^n \to \{0,1\}$, the following holds:*

$$\Omega(\log(\widetilde{spar}(f)) - \log n) \overset{(1)}{=} R^{\wedge dt}(f) \leqslant D^{\wedge dt}(f) \overset{(2)}{=} O\left((\log \widetilde{spar}(f))^6 \cdot \log n\right).$$

It is worth noting that one could go one step further in the chain of inequalities to show that $\mathrm{D}^{\wedge dt}(f)$ is upper bounded by $O(\mathrm{R}^{\wedge dt}(f)^6)$ upto polylog factors, thus concluding that randomization doesn't yield more than polynomial savings over the cost of deterministic ADT algorithms. Such a conclusion, in fact with a better polynomial bound, was first derived recently by Chattopadhyay, Dahiya, Mande, Radhakrishnan and Sanyal [20]. But our current technique is quite different and the previous result could not give an upper bound on ADT complexity in terms of approximate sparsity as we do here.

Apart from degree and sparsity, there is a third complexity measure that has been well investigated in the Fourier basis. This is the Fourier ℓ_1 norm, also called the spectral norm of a Boolean function f. Denoted by $\|\hat{f}\|_1$, it is defined as the sum of the magnitude of the Fourier coefficients of f. It appeared in the context of additive combinatorics [28], communication complexity of XOR functions [21, 23, 24] and analysis of Boolean functions [3, 12, 22]. One naturally defines its ε-approximate version, denoted by $\|\hat{f}\|_{1,\varepsilon}$, to be the amount of Fourier ℓ_1 mass needed by any real-valued function g to point-wise approximate f within ε. Can approximation reduce significantly the needed ℓ_1 mass? Very recently, Cheung, Hatami, Hosseini, Nikolov, Pitassi and Shirley [24], constructed a Boolean function f such that $\log(\|\hat{f}\|_{1,1/3})$ is exponentially smaller than $\log(\|\hat{f}\|_1)$, which implies that the Fourier basis yields exponential advantage to approximation even with respect to the spectral norm.

In contrast, the proof method that we develop to establish Theorem 1.1 shows that approximation does not significantly reduce even the ℓ_1 mass of a total Boolean function in the De Morgan basis. More precisely, let $\mathrm{wt}(f)$ and $\widetilde{\mathrm{wt}}_\varepsilon(f)$ represent the exact and ε-approximate ℓ_1 norm of f in the De Morgan basis (we write $\widetilde{\mathrm{wt}}(f) := \widetilde{\mathrm{wt}}_{1/3}(f)$ when $\varepsilon = 1/3$).

Theorem 1.5. *For every total Boolean function $f : \{0,1\}^n \to \{0,1\}$, we have*

$$\log wt(f) = O\left((\log \widetilde{wt}(f))^2 \cdot \log n\right).$$

General representations: A natural question emerges from our results. Let $\mathcal{F}$ be a family of elementary real-valued functions defined over the n-ary Boolean cube B_n, such that $\mathcal{F}$ spans the vector space $\mathbb{R}^{B_n}$ of all real-valued functions. The *sparsity* (resp., *weight*) of a (Boolean) function f wrt $\mathcal{F}$ is defined as the minimum integer (resp., non-negative real number) $k \geqslant 0$ such that f can be written as a linear combination of at most k functions from $\mathcal{F}$ (resp., with total absolute coefficient sum at most k)[1]. Denote these complexity measures by $\mathrm{spar}_\mathcal{F}(f)$ and $\mathrm{wt}_\mathcal{F}(f)$ respectively. For example, when $\mathcal{F}$ is the family of all AND functions, these measures correspond to the De Morgan sparsity and ℓ_1 norm of f, and when $\mathcal{F}$ corresponds to all parities, these correspond to the Fourier sparsity and the spectral norm of f. Likewise, one defines the approximate sparsity and weight of f with respect to $\mathcal{F}$, denoting them by $\widetilde{\mathrm{spar}}_\mathcal{F}(f)$ and $\widetilde{\mathrm{wt}}_\mathcal{F}(f)$ respectively.

Question 1.6. *What properties of $\mathcal{F}$ ensure that approximation doesn't help reduce sparsity or the weight of a Boolean function, i.e. do there exist constants α and β such that for all Boolean functions f, $\log(spar_\mathcal{F}(f)) = O((\log(\widetilde{spar}_\mathcal{F}(f))^\alpha)$ and/or $\log(wt_\mathcal{F}(f)) = O((\log(\widetilde{wt}_\mathcal{F}(f))^\beta)$?*

This question is quite broad. For instance, if one views the input domain of the functions as the set of $m \times m$ Boolean matrices, and $\mathcal{F}$ be the set of all rank one matrices, then Question 1.6 specializes to asking if log of the rank of a Boolean matrix is always at most a fixed polynomial of the log of its approximate-rank. In general, this is false. For example, the identity matrix has rank m but its approximate rank is $(\log m)^{O(1)}$. However, it is unknown for special classes of Boolean matrices like those that are the truth table of AND-functions (i.e., functions composed with 2-bit AND gadgets). Understanding the power of approximation for such special classes of matrices is of significant interest, given its connection to quantum communication complexity. We talk more about this aspect in Section 1.2.

Summarizing what we have seen, if $\mathcal{F}$ is the set of all parities, i.e. the Fourier monomials, then approximation can significantly help, and reduce sparsity exponentially. We showed that if $\mathcal{F}$ is the set of all monotone Boolean AND functions, i.e. the De Morgan basis, then approximations do not help, and

[1] Note that there may be more than one way of doing that.

reduce sparsity by at most a polynomial factor on the log scale. In fact our main result gives us slightly more: let $[n]$ be partitioned into two sets, the set of positive literals denoted by $\mathcal{P}$ and the set of negated literals $\mathcal{N}$. Each such partition defines a *shifted* De Morgan basis, where a shifted monomial is given by a pair of sets $P \subseteq \mathcal{P}$ and $N \subseteq \mathcal{N}$, and corresponds to the Boolean function $M_{P,N} := \prod_{i \in P} x_i \prod_{j \in N}(1-x_j)$. Observe that while Or has full De Morgan sparsity, it has sparsity just 2 in the fully shifted De Morgan basis, i.e., the basis that corresponds to $\mathcal{P} = \varnothing$ ans $\mathcal{N} = [n]$. There are 2^n such shifted bases, and it is straightforward to verify that our main results—Theorem 1.1 and Theorem 1.5—imply that, in each shifted basis, the approximate sparsity and approximate ℓ_1-norm are polynomially related to the exact sparsity and exact ℓ_1-norm. A natural generalization of the case when $\mathcal{F}$ is just a shifted De Morgan basis, is the case when we populate the set $\mathcal{F}$ with all shifted monomials. More precisely, consider $\mathcal{F} := \{M_{P,N} : P, N \subseteq [n], P \cap N = \varnothing\}$, where each $M_{P,N} := \prod_{i \in P} x_i \prod_{j \in N}(1-x_j)$ is called a *generalized monomial*. Observe that any shifted De Morgan basis is a strict subset of $\mathcal{F}$, the set of generalized monomials, whose size is 3^n. The following concrete question, which is a special case of Question 1.6, remains intriguingly open!

Question 1.7. *Does there exist a total Boolean function f for which the approximate generalized-monomial sparsity (approximate generalized-monomial weight), denoted by $\widetilde{gspar}(f)$ ($\widetilde{gwt}(f)$), is super-polynomially smaller in the log scale than its exact generalized-monomial sparsity (generalized-monomial weight), denoted by $gspar(f)$ ($gwt(f)$)?*

As expected, generalized monomials can significantly reduce sparsity compared to any shifted De Morgan basis. For instance, consider the following function that mixes two shifted OR's by a monotone addressing scheme: let $f_{\text{mixed}} : \{0,1\}^2 \times \{0,1\}^n \to \{0,1\}$, where $f_{\text{mixed}}(x, y)$ outputs 0 if $x = 00$, outputs 1 if $x = 11$, computes the Boolean Or of y if $x = 10$, and computes the Boolean And of y if $x = 01$. It is easy to verify that $\text{gspar}(f_{\text{mixed}}) = O(1)$, while the sparsity of f_{mixed} in any shifted De Morgan basis is $2^{\Omega(n)}$. Observing that f_{mixed} is a monotone function, it becomes interesting to answer Question 1.7 for monotone functions. We provide a negative answer below.

Theorem 1.8. *For every monotone Boolean function $f : \{0,1\}^n \to \{0,1\}$, the following hold:*

(a) $\log gspar(f) = O\left((\log \widetilde{gspar}(f))^4 \cdot (\log n)^3\right)$,

(b) $\log gwt(f) = O\left((\log \widetilde{gwt}(f))^4 \cdot (\log n)^3\right)$.

The above result, in fact, yields the following more detailed picture about query complexity. Recall that the size of a decision tree is the number of leaves in it. The deterministic (randomized) decision tree size complexity of a function f, denoted by $\text{DSize}^{dt}(f)$ ($\text{RSize}^{dt}(f)$), is the smallest size needed by an ordinary deterministic (randomized) decision tree to compute f. We can use Theorem 1.8 and other standard results to get the following.

Corollary 1.9. *For every monotone Boolean function $f : \{0,1\}^n \to \{0,1\}$, the following hold:*

(a) $\Omega(\log(\widetilde{gspar}(f)) - \log n) \overset{(1)}{=} \log RSize^{dt}(f) \leqslant \log DSize^{dt}(f) \overset{(2)}{=} O\left((\log \widetilde{gspar}(f))^4 \cdot (\log n)^3\right)$.

(b) $\Omega(\log(\widetilde{gwt}(f))) \overset{(1)}{=} \log RSize^{dt}(f) \leqslant \log DSize^{dt}(f) \overset{(2)}{=} O\left((\log \widetilde{gwt}(f))^4 \cdot (\log n)^3\right)$.

Remark 1.10. *In particular, this yields the fact that deterministic and randomized decision tree size measure of a monotone function are, upto poly-log(n) factors, polynomially related in the log scale. Such a relationship was recently proven to be true even for general functions in Chattopadhyay et. al. [20]. However, the tighter relationship that we prove here via approximate generalized sparsity and weight for monotone functions, is known to be false for general functions as witnessed by the Sink function[2]. The Sink function has $\binom{n}{2}$ input bits, corresponding to the edges of a complete graph on n vertices. Each Boolean assignment orients the edges. Sink outputs 1 iff there exists a Sink vertex in the resulting directed graph. Its generalized sparsity and weight is just n, whereas $RSize^{dt}(Sink)$ is $2^{\Omega(n)}$.*

[2] Sink was used to construct a counter-example to the Log-Approximate-Rank Conjecture in [23]

1.1 Our Method

Lower bounds on approximate sparsity were known for specific functions such as OR_n and PARITY_n (these are folklore results), typically established using random restrictions and approximate degree lower bounds. However, these results apply only to specific functions or restricted classes of functions. The general idea is to apply a random restriction ρ, which selects a random subset of variables and fixes each to 0, with the goal of eliminating high-degree monomials from a candidate sparse polynomial P that approximates f, such that $f|_\rho$ still has large approximate degree while $P|_\rho$ has degree that is too small, yielding a contradiction. This is illustrated by considering the n-bit OR function. Let P be any sparse polynomial approximating OR_n. Consider a random restriction ρ that, independently for each of the n variables, fixes it to 0 with probability 1/2 and leaves it free with probability 1/2. With high probability, at least $n/3$ variables are left free. On the other hand, any monomial of degree larger than $\sqrt{n}$ survives (i.e., none of its variables are set to 0) with probability at most $2^{-\sqrt{n}}$. If the number of monomials in P is s, then the probability that $P|_\rho$ contains a monomial of degree larger than $\sqrt{n}$ is at most $s \cdot 2^{-\sqrt{n}}$, which is less than 1/2 if $s < 2^{\sqrt{n}}/2$. With high probability, $(\text{OR}_n)|_\rho$ is an r-bit OR function with $r \geqslant n/3$. Since $\text{OR}_n|_\rho$ is still approximated by $P|_\rho$ (for every ρ), and recalling that the approximate degree of OR_r is $\Omega(\sqrt{r})$, we conclude that $s = 2^{\Omega(\sqrt{n})}$.

While this works for OR function, there are functions which are very different from OR and yet have large sparsity in De Morgan basis. An example of that is $\text{AND}_n \circ \text{OR}_2$, where the bottom ORs are 2-bit functions. It is simple to verify that this function has sparsity $2^{\Omega(n)}$. But there is no way to induce a large OR in this function. If one tried to apply a random restriction like the one that worked for OR, one concludes easily that it won't work as with high probability one of the bottom OR_2 will have both its input variables fixed to 0, thereby killing the entire function. One way to fix this is to consider a slightly more careful restriction. For each of the bottom ORs, one selects one of its two input variables at random and fixes it to 0, leaving the other variable free. It is not hard to show that in this case the restricted function is always the AND over the remaining n free variables, and if the approximating polynomial for the $\text{AND}_n \circ \text{OR}_2$ had sparsity $2^{o(\sqrt{n})}$, then with nonzero probability, the restricted polynomial would give an $o(\sqrt{n})$-degree approximation to AND_n, contradicting known lower bounds. The important thing to note here is that our random restriction is no longer done independently for each variable, as the restrictions on the two variables in each OR_2 block are correlated.

Our approach generalizes this idea. We design a random restriction process that works for all functions with large exact sparsity. It's not a-priori clear what useful combinatorial structural information can be extracted from just knowing that a function has large sparsity. This is precisely the main technical contribution of our work. We devise a random restriction procedure for such functions with large sparsity and it differs from the two simple cases that we considered above in the following two ways: (i) the method is adaptive in the sense that the next bit that is fixed or left free depends on what happened in the previous step[3] (ii) variables are not exclusively fixed to 0, and some may get fixed to 1 as well.

As some of the variables may get fixed to 1, we can no longer argue that all high degree monomials of the approximating polynomial are 'zeroed' out as was happening for the two cases discussed earlier. Instead, we argue that if the unrestricted approximating polynomial was sparse to begin with, then with high probability, the restricted polynomial will not have any monomial that has large degree in the free variables. Note that a non-zeroed out monomial could either completely collapse to a constant value, in which case it has degree zero with respect to the free variables, or otherwise it could be nonzero but low degree on the free variables.

It is known, via a result generally attributed to Grolmusz [29], that lower bounds on approximate sparsity yields lower bounds on approximate weight (ℓ_1 norm) in any basis and even wrt any arbitrary set $\mathcal{F}$ of basic functions. Hence, our lower bound on the sparsity of approximating polynomials when their exact sparsity is large, in fact, yields a lower bound on the weight of an approximating polynomial as well. To relate exact ℓ_1 norm and approximate ℓ_1 norm, we need to assume that the exact ℓ_1 norm, instead of exact sparsity, of f is large. This is indeed a weaker assumption as in the De Morgan basis, all non-zero coefficients in the unique polynomial representation of f are integer-valued. This is precisely what our random restriction procedure MAXDEGREERESTRICTION, described in Section 3.1, does by taking a bit more care.

We are able to adapt our restriction technique to deal with generalized polynomials as long as the

[3]The restriction that we used for $\text{AND}_n \circ \text{OR}_2$ is still non-adaptive as the fixing in each of the OR_2 block is independent of the other blocks and can therefore be done all at once.

function has an additional property: it has what we call a *separating set* of inputs. A separating set of inputs for f is a special *fooling* set with respect to subcubes. Given such a separating set, we're able to modify our random restriction technique to reduce the degree of generalized monomials with respect to free variables while ensuring that the restricted function retains high degree. Finally, we observe that monotone functions with large generalized sparsity have such large separating sets and this helps us prove that generalized sparsity (generalized weight) and approximate generalized sparsity (approximate generalized weight) are polynomially related in the log scale for monotone functions.

1.2 Other Related Work

Random restrictions have been used to obtain lower bounds for (approximate) sparsity before, say for the OR_n function (folklore) or for specific functions [4, 22]. Similarly, random restrictions have been famously quite successful in other parts of complexity theory like circuit complexity [27, 30] and proof complexity [7,41,45]. Designing new random restrictions in these two areas remains an active, technically challenging theme of current research [31,32]. In these works, random restrictions are applied to a target functions (in circuits) or CNF formulas (in proof complexity) that have explicit convenient combinatorial properties. On the other hand, we design a generic random restriction scheme that can be applied on any function that has the algebraic property of large sparsity in the De Morgan basis. Using this new scheme we derive a general structural result applicable to *all* functions. Such an application of random restriction seems rare to us. It vaguely reminds us of two results, both in proof complexity, where *deterministic* greedy restrictions were used to obtain results that are applicable to all formulas: the first is by Impagliazzo, Pudlak and Sgall [34] showing that degree lower bounds are enough to prove size (read sparsity) lower bounds for proofs in polynomial calculus. The second is by Ben-Sasson and Wigderson [8] who also used a greedy restriction method to show that width of resolution proofs always translate to size of proofs.

Buhrman and de Wolf [14] were interested, among other things, in characterizing the bounded-error quantum communication complexity of AND functions, i.e 2-party communication functions of the form $f_n \circ \mathrm{AND}_2$, where f_n is an arbitrary n-bit Boolean function. This problem has remained open. There was a breakthrough made by Razborov [46] who showed that for symmetric f_n, quantum protocols offer no advantage over classical randomized protocols. Whether there exists some f_n for which the quantum and classical randomized communication complexities are widely separated for $f_n \circ \mathrm{AND}_2$ remains open despite several efforts [14, 37, 47, 50]. Our main result provides a conditional answer to that question as discussed below.

Razborov proved his lower bound by showing a lower bound on the log of the approximate rank of the communication matrix. Buhrman and de Wolf [14], observing that the exact sparsity of any function f in the De Morgan basis is equal to the rank of the communication matrix of $f \circ \mathrm{AND}_2$, raised informally the following interesting question:

Question 1.11. *Is it true that for every function f, (logarithm of) the approximate sparsity of f and (logarithm of) the approximate rank of the communication matrix of $f \circ \mathrm{AND}_2$ are within a polynomial of each other?*

Remark 1.12. *Buhrman and de Wolf didn't quite phrase this question in the manner we do. Their discussion didn't put any quantitative bounds, nor do they talk about relating things in the log scale. Our version, therefore, may be a significant weakening of what they had in mind.*

Our result provides a fresh impetus to seek an answer to the above question for the following reason. Assuming the answer is positive, Theorem 1.1, combined with the recent resolution (up to a $\log n$ factor) of the log-rank conjecture for AND-functions by Knop, Lovett, McGuire, and Yuan [40], implies that the deterministic classical zero error communication complexity of every function $f \circ \mathrm{AND}_2$ is at most a fixed polynomial of its quantum bounded-error communication complexity, ignoring poly-logarithmic factors.

2 Preliminaries

In this section, we collect notation, definitions, and known results that will be used throughout the paper. All functions considered are defined on the Boolean hypercube $\{0, 1\}^n$, and all polynomials are assumed to be multilinear real polynomials.

Definition 2.1 (Multilinear Polynomial Representation). *A polynomial $Q \in \mathbb{R}[x_1, x_2, \ldots, x_n]$ is called* multilinear *if each variable appears with degree at most one in every monomial. Over the Boolean domain, every function $f : \{0,1\}^n \to \mathbb{R}$ admits a unique multilinear polynomial representation. That is, there exists a unique multilinear polynomial $Q \in \mathbb{R}[x_1, x_2, \ldots, x_n]$ such that $Q(x) = f(x)$ for all $x \in \{0,1\}^n$.*

Definition 2.2 (Support, Degree, Sparsity, and Norm of a Polynomial). *Let $Q \in \mathbb{R}[x_1, \ldots, x_n]$ be a multilinear polynomial written as*

$$Q(x) = \sum_{S \subseteq [n]} a_S \prod_{i \in S} x_i.$$

- *The* support *of Q, denoted $\mathsf{Vars}(Q)$, is the set of variables that appear in some monomial with a nonzero coefficient.*
- *The* degree *of Q, denoted $\deg(Q)$, is $\max\{|S| \mid a_S \neq 0\}$.*
- *The* sparsity *of Q, denoted $spar(Q)$, is the number of nonzero coefficients a_S.*
- *The ℓ_1-norm of Q, denoted $wt(Q)$, is given by $\sum_{S \subseteq [n]} |a_S|$.*

Definition 2.3 (Complexity Measures for Functions via Polynomials). *Let $f : \{0,1\}^n \to \mathbb{R}$ be a function, and let $\mathcal{P}(f)$ denote its unique multilinear polynomial representation. We define the following complexity measures:*

$$\deg(f) := \deg(\mathcal{P}(f)), \quad spar(f) := spar(\mathcal{P}(f)), \quad wt(f) := wt(\mathcal{P}(f)).$$

Remark 2.4. *For Boolean functions $f : \{0,1\}^n \to \{0,1\}$, the coefficients in $\mathcal{P}(f)$ are integers (see for example [35, Chapter 2]), and hence $spar(f) \leqslant wt(f)$.*

Definition 2.5 (Complexity Measures for Functions via Approximating Polynomials). *Let $f : \{0,1\}^n \to \mathbb{R}$ and let $\varepsilon > 0$. We define:*

$$\begin{aligned} \widetilde{deg}_\varepsilon(f) &:= \min\{\deg(Q) \mid Q \text{ satisfies } |Q(x) - f(x)| \leqslant \varepsilon \text{ for all } x \in \{0,1\}^n\}, \\ \widetilde{spar}_\varepsilon(f) &:= \min\{spar(Q) \mid Q \text{ satisfies } |Q(x) - f(x)| \leqslant \varepsilon \text{ for all } x \in \{0,1\}^n\}, \\ \widetilde{wt}_\varepsilon(f) &:= \min\{wt(Q) \mid Q \text{ satisfies } |Q(x) - f(x)| \leqslant \varepsilon \text{ for all } x \in \{0,1\}^n\}. \end{aligned}$$

When $\varepsilon = 1/3$, we write $\widetilde{deg}(f) := \widetilde{deg}_{1/3}(f)$, $\widetilde{spar}(f) := \widetilde{spar}_{1/3}(f)$, and $\widetilde{wt}(f) := \widetilde{wt}_{1/3}(f)$.

Theorem 2.6 ([25, Claim 4.3]). *Let $f : \{0,1\}^n \to \{0,1\}$ be a Boolean function. Then for any $0 < \varepsilon < \frac{1}{2}$,*

$$\widetilde{deg}_\varepsilon(f) = O\left(\widetilde{deg}_{1/3}(f) \cdot \log(1/\varepsilon)\right).$$

Theorem 2.7 ([1, Theorem 4]). *For every Boolean function $f : \{0,1\}^n \to \{0,1\}$,*

$$\deg(f) = O(\widetilde{deg}(f)^2).$$

Remark 2.8. *The bound in Theorem 2.7 is tight; for example, the OR_n function satisfies $\deg(\mathrm{OR}_n) = n$ and $\widetilde{deg}(\mathrm{OR}_n) = \Theta(\sqrt{n})$.*

Definition 2.9 (Restrictions). *A* restriction *ρ on a set of variables $V \subseteq \{x_1, \ldots, x_n\}$ is a partial assignment*

$$\rho : V \to \{0, 1, *\},$$

where for $x_i \in V$, $\rho(x_i) \in \{0,1\}$ indicates that x_i is fixed, and $\rho(x_i) = *$ means x_i is left free. We define:

$$\begin{aligned} \mathsf{SetVars}(\rho) &:= \{x_i \in V \mid \rho(x_i) \in \{0,1\}\}, \\ \mathsf{FreeVars}(\rho) &:= \{x_i \in V \mid \rho(x_i) = *\}. \end{aligned}$$

The size *of ρ, denoted $|\rho|_*$, is the number of free variables:*

$$|\rho|_* := |\mathsf{FreeVars}(\rho)|.$$

Let $Q \in \mathbb{R}[x_1, \ldots, x_n]$ be a polynomial and $V \subseteq \{x_1, \ldots, x_n\}$. For a restriction ρ on V, we write $Q|_\rho$ for the polynomial obtained by substituting $x_i = \rho(x_i)$ for all $x_i \in \mathsf{SetVars}(\rho)$.

For an input $w \in \{0,1\}^V$ and a subset $T \subseteq V$, we write $w|_T \in \{0,1\}^T$ to denote the projection of w to the coordinates in T. For singleton sets, we simply write w_i for $w|_{\{x_i\}}$.

Let $F \subseteq \{0,1\}^V$ be a set of Boolean assignments and ρ a restriction on V. The restriction of F under ρ, denoted $F|_\rho$, is defined as

$$F|_\rho := \left\{ w|_{\mathsf{FreeVars}(\rho)} \,\middle|\, w \in F,\ \forall x_i \in \mathsf{SetVars}(\rho),\ w_i = \rho(x_i) \right\}.$$

3 Sparsity vs. Approximate Sparsity

In this section, we show that for Boolean functions $f : \{0,1\}^n \to \{0,1\}$, the exact and approximate sparsity are polynomially related on the logarithmic scale. Rather than assuming large sparsity and arguing for large approximate sparsity, we start with the weaker assumption of a large exact ℓ_1-norm (see Remark 2.4) and argue large approximate sparsity. This approach yields both Theorem 1.1 and Theorem 1.5 in one go, showing that the logarithms of the exact and approximate sparsity, as well as of the exact and approximate ℓ_1-norm, are polynomially related up to a $\log n$ factor.

Proof Overview. Let f be a Boolean function with large exact ℓ_1-norm. We aim to show that any polynomial approximating f within error $1/3$ must also have large sparsity. The argument proceeds via a carefully constructed random restriction ρ, sampled using Algorithm 1, which satisfies the following properties:

1. With high probability, ρ leaves $\ell = \Omega(\log \mathrm{wt}(f)/\log n)$ variables free.
2. The restricted function $f|_\rho$ has full degree on the variables left free.
3. For any monomial M and any $t \geqslant 1$, the probability that $\deg(M|_\rho) \geqslant t$ is at most 2^{-t}.

With the above properties of ρ, the reason why the approximate sparsity of f must be large becomes evident. Let ℓ (roughly $\log \mathrm{wt}(f)/\log n$) denote the number of variables left free by ρ. Suppose there exists a polynomial Q approximating f having sparsity less than $2^{\sqrt{\ell/c}}$, for some constant $c > 0$ (to be chosen appropriately). Then, by property (3) above and using a probabilistic argument, there exists a restriction ρ that eliminates all monomials of degree at least $\sqrt{\ell/c}$ in Q. Consequently, the restricted polynomial $Q|_\rho$ has degree strictly less than $\sqrt{\ell/c}$.

On the other hand, by property (2), the restricted function $f|_\rho$ has degree ℓ. Therefore, $Q|_\rho$ approximates $f|_\rho$, a Boolean function of degree ℓ, using a polynomial of degree less than $\sqrt{\ell/c}$. This contradicts the known relationship between degree and approximate degree for Boolean functions—specifically, that $\deg(f) \leqslant c \cdot \widetilde{\deg}(f)^2$ for some universal constant c [1].

We conclude that any polynomial approximating f must have sparsity at least $2^{\Omega(\sqrt{\ell})}$. Since $\ell = \Theta(\log \mathrm{wt}(f)/\log n)$, it follows that the logarithms of exact ℓ_1-norm and approximate sparsity are related quadratically (up to a $\log n$ factor).

The novelty of our proof lies in the method of sampling random restrictions that satisfy the key properties outlined above. In contrast to the non-adaptive restrictions commonly used in circuit complexity and related areas, our sampling procedure is adaptive—it takes into account the effects of previous random choices on the hardness measure (in our case, the sparsity of the restricted function). We believe this adaptive approach to sampling restrictions may have applications beyond the present context.

We now abstract the above idea into a general notion of hardness:

Definition 3.1 (ℓ-Variable Max-Degree Distribution)**.** *Let $f : \{0,1\}^n \to \{0,1\}$, and let $\mathcal{D}$ be a distribution over restrictions $\rho : \{x_1, \ldots, x_n\} \to \{0,1,*\}$. We say that $\mathcal{D}$ is an ℓ-*variable max-degree distribution *for f if:*

1. *With probability at least 0.9, ρ leaves at least ℓ variables free.*
2. *For every ρ in the support of $\mathcal{D}$, we have $\deg(f|_\rho) = |\rho|_*$.*

3. For any monomial M and any $t \in \mathbb{N}$, $\Pr_{\rho\sim\mathcal{D}}[\deg(M|_\rho) \geqslant t] \leqslant 2^{-t}$.

We will show that a large exact ℓ_1-norm implies the existence of such a distribution, which in turn implies that any polynomial approximating f must have large sparsity—thereby connecting exact ℓ_1-norm and approximate sparsity.

Organization of this section. In Section 3.1, we show how to construct a max-degree distribution when f has large exact ℓ_1-norm. In Section 3.2, we use this to prove Theorems 1.1 and 1.5. In Section 3.3, we discuss the tightness of our bounds. Finally, in Section 3.4, we explore implications for the AND query model.

3.1 The Restriction Process and Its Properties

Algorithm 1 MAXDEGREERESTRICTION

1: **Input:** Non-zero multilinear polynomial $Q \in \mathbb{R}[x_1,\ldots,x_n]$; set $V \subseteq \{x_1,\ldots,x_n\}$ with $\mathrm{vars}(Q) \subseteq V$.
2: **Output:** A restriction $\rho : V \to \{0,1,*\}$.
3: **if** $|V| = 0$ **then**
4:     **return** empty ρ
5: **else**
6:     **if** there exists $x_i \in V$, $u \in \{0,1\}$ such that $\mathrm{wt}(Q|_{x_i=u}) \geqslant \left(1-\frac{1}{n}\right)\cdot \mathrm{wt}(Q)$ **then**
7:         $\rho' \leftarrow$ MAXDEGREERESTRICTION$(Q|_{x_i=u}, V\backslash\{x_i\})$
8:         Set $\rho(x_i) \leftarrow u$, and for all $x_j \in V\backslash\{x_i\}$, set $\rho(x_j) \leftarrow \rho'(x_j)$
9:     **else**
10:         Choose $x_i \in V$ arbitrarily
11:         Express Q as $Q = R_1 \cdot x_i + R_0$
12:         With probability $1/2$:
13:             $\rho_0 \leftarrow$ MAXDEGREERESTRICTION$(R_0, V\backslash\{x_i\})$
14:             Set $\rho(x_i) \leftarrow 0$, and for all $x_j \in V\backslash\{x_i\}$, set $\rho(x_j) \leftarrow \rho_0(x_j)$
15:         Otherwise:
16:             $\rho_* \leftarrow$ MAXDEGREERESTRICTION$(R_1, V\backslash\{x_i\})$
17:             Set $\rho(x_i) \leftarrow *$, and for all $x_j \in V\backslash\{x_i\}$, set $\rho(x_j) \leftarrow \rho_*(x_j)$
18:     **end if**
19:     **return** ρ
20: **end if**

Algorithm 1 describes a procedure for sampling random restrictions for a given input polynomial Q. When applied to the unique multilinear polynomial Q that exactly represents a Boolean function f, we will show that the resulting distribution over restrictions is ℓ-variable max-degree for f, where $\ell = \Omega\left(\frac{\log \mathrm{wt}(f)}{\log n}\right)$. In this subsection, we establish the properties of the distribution induced by this process.

We begin with some observations about Algorithm 1. First, we claim that if a call to MAXDEGREERESTRICTION on a polynomial Q reaches line 9, then Q is balanced with respect to each of its variables. That is, for every $x_i \in \mathsf{Vars}(Q)$, if we write $Q = R_1x_i + R_0$, then both $\mathrm{wt}(R_0)$ and $\mathrm{wt}(R_1)$ are at least a $(1/2n)$-fraction of $\mathrm{wt}(Q)$. Formally:

Claim 3.2. *Suppose Algorithm 1 reaches the `else` branch at line 9 on input polynomial Q. Then for every $x_i \in \mathsf{Vars}(Q)$, writing $Q = R_1x_i + R_0$, we have*

$$wt(R_0) \geqslant \frac{1}{2n} wt(Q) \quad \text{and} \quad wt(R_1) \geqslant \frac{1}{2n} wt(Q).$$

Proof. If Algorithm 1 reaches the `else` branch at line 9 on input polynomial Q, then by the condition of that line, we have:

$$\forall x_i \in \mathsf{Vars}(Q),\ \forall u \in \{0,1\}, \quad \mathrm{wt}(Q|_{x_i=u}) < (1-\tfrac{1}{n})\mathrm{wt}(Q).$$

Fix a variable $x_i \in \mathsf{Vars}(Q)$, and write $Q = R_1x_i + R_0$. We claim that both $\mathrm{wt}(R_0) \geqslant \frac{1}{2n}\mathrm{wt}(Q)$ and $\mathrm{wt}(R_1) \geqslant \frac{1}{2n}\mathrm{wt}(Q)$. Suppose, for contradiction, that one of these inequalities fails.

- **Case 1:** $\mathrm{wt}(R_1) < \frac{1}{2n}\mathrm{wt}(Q)$. Since $Q|_{x_i=0} = R_0$, we get:

$$\mathrm{wt}(Q) = \mathrm{wt}(R_0) + \mathrm{wt}(R_1) = \mathrm{wt}(Q|_{x_i=0}) + \mathrm{wt}(R_1) < (1 - \tfrac{1}{n})\mathrm{wt}(Q) + \tfrac{1}{2n}\mathrm{wt}(Q) < \mathrm{wt}(Q),$$

which is a contradiction.

- **Case 2:** $\mathrm{wt}(R_0) < \frac{1}{2n}\mathrm{wt}(Q)$. Since $Q|_{x_i=1} = R_1 + R_0$, cancellations could occur between monomials in R_1 and R_0, but even in the worst case we have: $\mathrm{wt}(Q|_{x_i=1}) \geqslant \mathrm{wt}(R_1) - \mathrm{wt}(R_0)$.

Therefore,

$$\mathrm{wt}(Q|_{x_i=1}) \geqslant \mathrm{wt}(R_1) - \mathrm{wt}(R_0) = \mathrm{wt}(Q) - 2\mathrm{wt}(R_0) > \left(1 - \tfrac{1}{n}\right)\mathrm{wt}(Q),$$

which contradicts the assumption that $\mathrm{wt}(Q|_{x_i=1}) < \left(1 - \frac{1}{n}\right)\mathrm{wt}(Q)$.

Hence, both $\mathrm{wt}(R_0) \geqslant \frac{1}{2n}\mathrm{wt}(Q)$ and $\mathrm{wt}(R_1) \geqslant \frac{1}{2n}\mathrm{wt}(Q)$ must hold for every $x_i \in \mathsf{Vars}(Q)$. □

Observation 3.3. *If the input polynomial Q is initially non-zero, then by Claim 3.2, all recursive calls in Algorithm 1 continue to operate on non-zero polynomials. Furthermore, if at any point during the recursion the sparsity of the input polynomial becomes 1—i.e., the polynomial consists of a single monomial M—then, due to line 6, the algorithm deterministically sets all remaining variables in the final restriction ρ according to the support of M. Specifically, for each $x_i \in V$, we set $\rho(x_i) \leftarrow 1$ if $x_i \in \mathrm{vars}(M)$, and $\rho(x_i) \leftarrow 0$ otherwise.*

We next observe how the range of the input polynomial over $\{0,1\}^V$ evolves during recursion. Let Q be the input at some stage. The following cases arise in the recursion:

1. If the recursion proceeds via line 7, the next polynomial is $Q|_{x_i=u}$ for some $x_i \in V$, $u \in \{0,1\}$.
2. If via line 13, we recurse on $R_0 = Q|_{x_i=0}$, where x_i is chosen in line 10.
3. If via line 16, we write $Q = R_1 \cdot x_i + R_0$, where x_i is chosen in line 10, and recurse on $R_1 = \partial_{x_i} Q$.

In cases (1) and (2), the recursive call uses a restriction of Q, so the range of values on Boolean inputs does not increase. In case (3), the derivative R_1 may have a larger range. However, as a discrete derivative, its range is controlled—it lies within twice the range of Q.

Claim 3.4. *Let $Q \in \mathbb{R}[x_1, \ldots, x_n]$ with $\mathsf{Vars}(Q) \subseteq V$, and suppose $Q = R_1 \cdot x_i + R_0$ for some $x_i \in V$, such that the range of Q over $\{0,1\}^V$ is contained in $[a, b]$. Then the range of R_1 over $\{0,1\}^{V \setminus \{x_i\}}$ is contained in $[-(b-a), b-a]$.*

Proof. Fix any $w \in \{0,1\}^{V \setminus \{x_i\}}$, and let $w_0, w_1 \in \{0,1\}^V$ be its extensions with $x_i = 0$ and $x_i = 1$, respectively. Then $R_1(w) = Q(w_1) - R_0(w) = Q(w_1) - Q(w_0)$. Since both $Q(w_0)$ and $Q(w_1)$ lie in $[a, b]$, their difference lies in $[-(b-a), b-a]$, as claimed. □

In particular, if we start with a polynomial Q representing a Boolean function and take k successive derivatives, the range of the resulting polynomial is contained in $[-2^{k-1}, 2^{k-1}]$ by repeated application of the claim.

We now show that the restriction ρ returned by Algorithm 1 leaves a significant number of variables free. The algorithm proceeds recursively and follows one of three possible execution paths:

- If $|V| = 0$, the recursion terminates and the algorithm backtracks.
- If the condition on line 6 holds, the algorithm makes a single recursive call (line 7).
- Otherwise, the condition on line 9 holds, and the algorithm makes one of two recursive calls (lines 13, 16) depending on the outcome of a coin toss.

The recursion continues while $|V| \geqslant 1$, and halts when $|V| = 0$, after which the final restriction is assembled by backtracking.

We classify recursive calls into two types: a call is *passive* if the condition on line 6 is satisfied, and *active* if the condition on line 9 is satisfied.

We argue that any execution of the algorithm must involve a substantial number of active recursive calls. In each such call, a variable is left free with probability 1/2. Therefore, if the algorithm makes ℓ active calls, the expected number of variables left free in the final restriction is $\ell/2$.

Moreover, since the decision to leave a variable free (i.e., assign it the value $*$) in an active call is independent of the choices made in previous calls, the number of free variables in the final restriction is tightly concentrated around its expectation. By a standard Chernoff bound, with high probability, at least a constant fraction of these ℓ active calls will indeed result in variables being left free.

This leads to the following formal statement:

Claim 3.5. *Let Q be the multilinear polynomial representing a Boolean function with $wt(Q) \geqslant 10(4n)^{40}$, and let $\rho = \textsc{MaxDegreeRestriction}(Q, \{x_1, \ldots, x_n\})$ be the restriction output by Algorithm 1. Then, with probability at least 0.9, the restriction ρ leaves at least $\Omega\left(\frac{\log wt(Q)}{\log n}\right)$ variables free.*

Proof. We begin by showing that any execution of the algorithm must involve a substantial number of *active* recursive calls. Let the algorithm make t total recursive calls, of which ℓ are active. Since the size of V decreases by 1 in each call and the recursion terminates when $|V| = 0$, we have $t \leqslant n$.

Now, observe how the ℓ_1-norm and the range of the polynomial (when evaluated on inputs in $\{0,1\}^V$) evolve during the recursion:

- In a *passive* call (i.e., when line 6 is satisfied), the polynomial in the next step is of the form $Q|_{x_i=u}$, whose ℓ_1-norm is at least $(1 - 1/n) \cdot \mathrm{wt}(Q)$, and which takes the same range of values on Boolean inputs as Q.
- In an *active* call (i.e., when line 9 is satisfied), the ℓ_1-norm of the next polynomial drops by at most a factor $1/(2n)$, by Claim 3.2. If recursion proceeds via line 13, the value set over Boolean inputs does not increase. If via line 16, the next polynomial is the discrete derivative $\partial_{x_i} Q$, whose range on Boolean inputs increases by at most a factor of 2 (Claim 3.4).

The recursion terminates with $|V| = 0$ and a nonzero constant polynomial. Since each active call can at most double the range of values on Boolean inputs, and the initial polynomial Q takes values in $\{0,1\}$, the final constant must lie in $[-2^\ell, 2^\ell]$. Therefore, the total shrinkage in the ℓ_1-norm over the course of the recursion satisfies:

$$\begin{aligned} 2^\ell &\geqslant (1 - 1/n)^{t-\ell} \cdot (1/2n)^\ell \cdot \mathrm{wt}(Q) \\ &\geqslant (1 - 1/n)^n \cdot (1/2n)^\ell \cdot \mathrm{wt}(Q) && \text{(since } t - \ell \leqslant n) \\ &\geqslant \frac{1}{10} \cdot \left(\frac{1}{2n}\right)^\ell \cdot \mathrm{wt}(Q) && \text{(using } (1-1/n)^n \geqslant 1/10 \text{ for } n \geqslant 2). \end{aligned}$$

Taking logarithms and rearranging, we obtain:

$$\ell \geqslant \frac{\log(\mathrm{wt}(Q)/10)}{\log(4n)}.$$

Define $\ell^* := \frac{\log(\mathrm{wt}(Q)/10)}{\log(4n)}$. Thus, every run of the algorithm contains at least ℓ^* active recursive calls.

Let $X_1, X_2, \ldots, X_{\ell^*}$ be indicator random variables, where $X_i = 1$ if the variable chosen in the i-th active call is left free (which occurs with probability 1/2), and 0 otherwise. These variables are independent by construction, and $\mathbb{E}\left[\sum_{i=1}^{\ell^*} X_i\right] = \frac{\ell^*}{2}$. By a standard Chernoff bound, we get:

$$\Pr\left(\sum_{i=1}^{\ell^*} X_i \leqslant \frac{\ell^*}{4}\right) \leqslant e^{-\ell^*/16} \leqslant 0.1,$$

where the last inequality follows from the assumption $\mathrm{wt}(Q) \geqslant 10(4n)^{40}$.

Since the number of variables left free in the final restriction is at least $\sum_{i=1}^{\ell^*} X_i$, we conclude that with probability at least 0.9, the algorithm leaves at least $\ell^*/4 = \Omega\left(\frac{\log \mathrm{wt}(Q)}{\log n}\right)$ variables free. □

Finally, we show that for any input polynomial Q, the restriction ρ returned by Algorithm 1 has the following properties: the restricted polynomial $Q|_\rho$ has full degree—that is, its degree equals the number of variables left free by ρ; and for any monomial M, the degree of $M|_\rho$ exhibits exponential tail decay: the probability that $\deg(M|_\rho) \geqslant t$ is at most 2^{-t}.

Claim 3.6. *Let $Q \in \mathbb{R}[x_1, \dots, x_n]$ be a non-zero polynomial with $\mathsf{Vars}(Q) \subseteq V$. Let ρ be the restriction returned by* MaxDegreeRestriction(Q, V)*, as described in Algorithm 1. Then:*

(a) The restricted polynomial $Q|_\rho$ is non-zero and has full degree; that is, $\deg(Q|_\rho) = |\rho|_*$.

(b) For any monomial M with $\mathsf{Vars}(M) \subseteq V$, and any $t \in \mathbb{N}$, the degree of the restricted monomial satisfies:

$$\Pr_\rho\left(\deg(M|_\rho) \geqslant t\right) \leqslant 2^{-t}.$$

Proof. Claim (b) is trivial when $t = 0$, so assume $t > 0$. We prove both parts simultaneously by induction on $|V|$.

Base Case ($|V| = 0$): Here, Q must be a non-zero constant polynomial. The claim holds trivially.

Inductive Step ($|V| \geqslant 1$): We consider the two possible branches of the algorithm, depending on which condition is satisfied at runtime (line 6 or line 9):

1. **Case where the "if" condition (line 6) is satisfied:** Suppose the condition is satisfied for some variable $x_i \in V$ and some value $u \in \{0, 1\}$. The algorithm then returns the restriction $\rho = \rho' \cup \{x_i \leftarrow u\}$, where ρ' is obtained from a recursive call with a strictly smaller support set. By the induction hypothesis, the restricted polynomial $(Q|_{x_i=u})|_{\rho'}$ is non-zero and has full degree. Hence,

$$\deg(Q|_\rho) = \deg\left((Q|_{x_i=u})|_{\rho'}\right) \overset{(1)}{=} |\rho'|_* \overset{(2)}{=} |\rho|_*,$$

where (1) follows from the induction hypothesis, and (2) holds because ρ and ρ' leave the same number of variables free. Therefore, $Q|_\rho$ is non-zero and has full degree.

Moreover, for any monomial M, we have:

$$\Pr_\rho\left(\deg(M|_\rho) \geqslant t\right) \leqslant \Pr_{\rho'}\left(\deg(M|_{\rho'}) \geqslant t\right) \leqslant 2^{-t},$$

where the final inequality follows by the induction hypothesis.

2. **Case where the else clause at line 9 is executed:** Let $x_i \in V$ be the variable chosen in line 10. Then with probability $1/2$, the algorithm sets $x_i \leftarrow 0$, and returns $\rho = \rho_0 \cup \{x_i \leftarrow 0\}$; with the remaining probability $1/2$, it leaves x_i free (denoted by $*$) and returns $\rho = \rho_* \cup \{x_i \leftarrow *\}$.

   - **(a) Degree of $Q|_\rho$:** If x_i is set to 0 in ρ, then by induction

$$\deg(Q|_\rho) = \deg\left((Q|_{x_i=0})|_{\rho_0}\right) = |\rho_0|_* = |\rho|_*.$$

   If x_i is left free, then $Q|_\rho = x_i \cdot (R_1|_{\rho_*}) + (R_0|_{\rho_*})$, where $Q = x_i R_1 + R_0$. Hence:

$$\deg(Q|_\rho) \overset{(1)}{=} \max\left(1 + \deg(R_1|_{\rho_*}), \deg(R_0|_{\rho_*})\right) \geqslant 1 + \deg(R_1|_{\rho_*}) \overset{(2)}{=} 1 + |\rho_*|_* = |\rho|_*,$$

   where (1) uses the fact that $R_1|_{\rho_*}$ is non-zero, and (2) applies the induction hypothesis to R_1. Since $\deg(Q|_\rho)$ cannot exceed $|\rho|_*$, the inequality above must in fact be an equality. Moreover, since both $(Q|_{x_i=0})|_{\rho_0}$ and $R_1|_{\rho_*}$ are non-zero by the induction hypothesis, it follows that $Q|_\rho$ is also non-zero.

   - **(b) Degree of any monomial under restriction:** Let M be any monomial.

– If $x_i \notin \mathsf{Vars}(M)$, then:

$$\Pr_{\rho}(\deg(M|_\rho) \geqslant t) = \frac{1}{2}\Pr_{\rho_0}(\deg(M|_{\rho_0}) \geqslant t) + \frac{1}{2}\Pr_{\rho_*}(\deg(M|_{\rho_*}) \geqslant t) \leqslant \frac{2^{-t}}{2} + \frac{2^{-t}}{2} = 2^{-t},$$

by the inductive hypothesis applied to both ρ_0 and ρ_*.

– If $x_i \in \mathsf{Vars}(M)$, then with probability $1/2$, $x_i \leftarrow 0$, so $M|_\rho = 0$. With the remaining probability $1/2$, x_i remains free, and $M|_\rho = x_i \cdot M'|_{\rho_*}$, where $M' = M/x_i$. Hence:

$$\Pr_{\rho}(\deg(M|_\rho) \geqslant t) = \tfrac{1}{2}\Pr_{\rho_*}(\deg(M'|_{\rho_*}) \geqslant t-1) \leqslant \tfrac{1}{2} \cdot 2^{-(t-1)} = 2^{-t}.$$

again using the inductive hypothesis.

This completes the proof. $\square$

Applying Algorithm 1 to the polynomial Q representing a Boolean function f, and combining Claim 3.5 with Claim 3.6, we conclude that a large ℓ_1-norm implies the existence of a max-degree distribution.

Theorem 3.7. *Let $f : \{0,1\}^n \to \{0,1\}$ be a Boolean function with $wt(f) \geqslant 10(4n)^{40}$. Then there exists an ℓ-variable max-degree distribution for f, where $\ell = \Omega\left(\frac{\log wt(f)}{\log n}\right)$.*

Proof. Let Q be the unique multilinear real polynomial that exactly computes f, and let $V = \{x_1, \ldots, x_n\}$. Consider the distribution over restrictions $\rho \sim \textsc{MaxDegreeRestriction}(Q, V)$ generated by Algorithm 1. By Claim 3.5 and Claim 3.6, this distribution satisfies all three conditions of an ℓ-variable max-degree distribution for $\ell = \Omega\left(\frac{\log \mathrm{wt}(f)}{\log n}\right)$. $\square$

3.2 Putting Everything Together

We begin by showing that the existence of a max-degree distribution for a Boolean function f implies that any polynomial approximating f must have large sparsity and large ℓ_1-norm. Combined with Theorem 3.7, which guarantees such a distribution when $\mathrm{wt}(f)$ is large, this implies the following: on a logarithmic scale, the exact sparsity and exact ℓ_1-norm are at most quadratically larger than their approximate counterparts, up to a $\log n$ factor—proving Theorem 1.1 and Theorem 1.5.

Claim 3.8. *Let $f : \{0,1\}^n \to \{0,1\}$, and suppose there exists an ℓ-variable max-degree distribution $\mathcal{D}$ for f. Then,*

$$\log \widetilde{spar}(f) = \Omega(\sqrt{\ell}).$$

Proof. Let $\mathcal{D}$ be an ℓ-variable max-degree distribution for f. Suppose, for the sake of contradiction, that the claim does not hold. Let $k = \sqrt{\ell/c}$, where $c > 0$ is a constant to be chosen later. Assume there exists a real polynomial Q that $1/3$-approximates f and has sparsity

$$\mathrm{spar}(Q) \leqslant \frac{1}{10} \cdot 2^k.$$

We will argue that such a polynomial cannot exist, thereby proving the claim.

Sample a restriction $\rho \sim \mathcal{D}$, and consider the restricted polynomial $Q|_\rho$. By property (3) of $\mathcal{D}$, the probability that any fixed monomial in Q has degree at least k under ρ is at most 2^{-k}. Applying a union bound over all monomials in Q, we have

$$\Pr_{\rho}\left(\deg(Q|_\rho) \geqslant k\right) \leqslant \mathrm{spar}(Q) \cdot 2^{-k} \leqslant \frac{1}{10}.$$

By property (1) of $\mathcal{D}$, with probability at least 0.9, ρ leaves at least ℓ variables free. Thus, with probability at least 0.8, both of the following hold:

$$|\rho|_* \geqslant \ell \quad \text{and} \quad \deg(Q|_\rho) < k.$$

Fix such a restriction ρ. Then $Q|_\rho$ is a polynomial of degree less than k that 1/3-approximates $f|_\rho$. By property (2) of $\mathcal{D}$, we have:

$$\deg(f|_\rho) = |\rho|_* \geqslant \ell = c \cdot k^2 > c \cdot (\deg(Q|_\rho))^2 \geqslant c \cdot \widetilde{\deg}(f|_\rho)^2.$$

This contradicts the known relationship between degree and approximate degree for Boolean functions, namely that for all g, $\deg(g) \leqslant c \cdot \widetilde{\deg}(g)^2$ for some universal constant c (see Theorem 2.7).

Hence, our assumption was false, and the claim follows. □

Claim 3.9. *Let $f : \{0,1\}^n \to \{0,1\}$, and suppose there exists an ℓ-variable max-degree distribution $\mathcal{D}$ for f. Then,*

$$\log \widetilde{wt}(f) = \Omega(\sqrt{\ell}).$$

Proof. Let $\mathcal{D}$ be an ℓ-variable max-degree distribution for f. Suppose, for the sake of contradiction, that the claim does not hold. Let $k = (1/c_1) \cdot \sqrt{\ell/c}$ for appropriate positive constants c and c_1 to be determined later. Assume there exists a real polynomial $Q = \sum_{S \subseteq [n]} q_S \prod_{i \in S} x_i$ that 1/3-approximates f and has ℓ_1-norm

$$\mathrm{wt}(Q) \leqslant \frac{1}{100} \cdot 2^k.$$

We will argue that such a polynomial cannot exist, thereby proving the claim.

Sample a restriction ρ from $\mathcal{D}$, and consider the restricted polynomial $Q|_\rho$. We analyze the expected ℓ_1-mass of high-degree monomials in $Q|_\rho$. For any polynomial $P = \sum_{S \subseteq [n]} a_S \prod_{i \in S} x_i$, define the degree-$d$ tail of its ℓ_1-norm as

$$\mathrm{wt}(P)^{\geqslant d} := \sum_{\substack{S \subseteq [n] \\ |S| \geqslant d}} |a_S|.$$

Using property (3) of the distribution $\mathcal{D}$, we get:

$$\mathbb{E}_\rho \left[\mathrm{wt}(Q|_\rho)^{\geqslant k}\right] \leqslant \sum_{\substack{S \subseteq [n] \\ |S| \geqslant k}} |q_S| \cdot \Pr_\rho \left(\deg \left(\prod_{i \in S} x_i|_\rho \right) \geqslant k \right) \leqslant \mathrm{wt}(Q) \cdot 2^{-k} \leqslant \frac{1}{100}.$$

By Markov's inequality, with probability at least 0.9, we have $\mathrm{wt}(Q|_\rho)^{\geqslant k} < 0.1$. Combining this with property (1) of $\mathcal{D}$, which ensures $|\rho|_* \geqslant \ell$ with probability at least 0.9, we conclude that with probability at least 0.8, a random restriction $\rho \sim \mathcal{D}$ satisfies both:

$$|\rho|_* \geqslant \ell \quad \text{and} \quad \mathrm{wt}(Q|_\rho)^{\geqslant k} < 0.1.$$

Fix such a restriction ρ. Let $\bar{Q}$ be the polynomial obtained from $Q|_\rho$ by discarding all monomials of degree at least k. Since $Q|_\rho$ 1/3-approximates $f|_\rho$ and the total weight of the discarded tail is at most 0.1, it follows that $\bar{Q}$ 0.44-approximates $f|_\rho$, with $\deg(\bar{Q}) < k$.

By standard error reduction (see Theorem 2.6), we can boost the success probability of $\bar{Q}$ to obtain a polynomial that 1/3-approximates $f|_\rho$ with degree at most $c_1 k$. Thus, $\widetilde{\deg}(f|_\rho) < c_1 k = \sqrt{\ell/c}$. On the other hand, by property (2) of $\mathcal{D}$, we have $\deg(f|_\rho) = |\rho|_* \geqslant \ell$. But this contradicts the known relationship between degree and approximate degree for Boolean functions, which asserts that for any Boolean function g, $\deg(g) \leqslant c \cdot \widetilde{\deg}(g)^2$ for some universal constant c (see Theorem 2.7).

Hence, our assumption was false, and the claim follows. □

Theorem 3.10. *For every total Boolean function $f : \{0,1\}^n \to \{0,1\}$, we have*

$$\log(wt(f)) = O\big(\log(\widetilde{spar}(f))^2 \cdot \log n\big).$$

Proof. Assume $\mathrm{wt}(f) \geqslant 10(4n)^{40}$, as the claim is trivial otherwise. By Theorem 3.7, there exists an ℓ-variable max-degree distribution $\mathcal{D}$ for f, where $\ell = \Omega\left(\frac{\log \mathrm{wt}(f)}{\log n}\right)$. Applying Claim 3.8 to $\mathcal{D}$, we obtain

$$\log \widetilde{\mathrm{spar}}(f) = \Omega(\sqrt{\ell}) = \Omega\left(\sqrt{\frac{\log \mathrm{wt}(f)}{\log n}}\right).$$ □

Since $\mathrm{spar}(f) \leqslant \mathrm{wt}(f)$ for any Boolean function f (see Remark 2.4), the above theorem implies that for all total Boolean functions f,

$$\log(\mathrm{spar}(f)) = O\big(\log^2(\widetilde{\mathrm{spar}}(f)) \cdot \log n\big),$$

thereby proving Theorem 1.1.

Combining Theorem 3.7 with Claim 3.9, we also obtain:

Theorem 1.5 (Restated)**.** *For every total Boolean function $f : \{0,1\}^n \to \{0,1\}$, we have*

$$\log wt(f) = O\left((\log \widetilde{wt}(f))^2 \cdot \log n\right).$$

Proof. Assume $\mathrm{wt}(f) \geqslant 10(4n)^{40}$, as the claim is trivial otherwise. By Theorem 3.7, there exists an ℓ-variable max-degree distribution $\mathcal{D}$ for f, where $\ell = \Omega\left(\frac{\log \mathrm{wt}(f)}{\log n}\right)$. Applying Claim 3.9 to $\mathcal{D}$ yields the desired bound. □

Remark 3.11. *It is known that $\log \widetilde{spar}(f) = O(\log \widetilde{wt}(f) + \log n)$, a result referred to as Grolmusz's theorem [29, 52], which has its roots in a paper by Bruck and Smolensky [13]. The works [29, 52] show this bound for the Fourier basis, but the underlying proof technique is more general and, in fact, provides a method for converting a weighting measure to a counting measure. In particular, it applies to the De Morgan basis as well. Therefore, by combining this relationship with Theorem 3.10, we could have directly obtained a bound relating the exact and approximate ℓ_1-norms, without relying on Claim 3.9. However, this approach yields a slightly weaker result, incurring an additive $(\log n)^{O(1)}$ loss. Specifically, we would get $\log wt(f) = O\left((\log \widetilde{wt}(f))^2 \cdot \log n + (\log n)^3\right)$.*

Finally, combining Theorems 1.1, 1.5 and 3.10, we conclude that, on a logarithmic scale, the exact sparsity, approximate sparsity, exact ℓ_1-norm, and approximate ℓ_1-norm of any Boolean function are all polynomially related, up to polylogarithmic factors in n.

Remark 3.12. *Although we state our result for Boolean functions, the proof relies on minimal properties specific to Boolean-valuedness. In fact, the argument extends to any class of functions over the Boolean domain that is closed under variable fixing and satisfies a universal relationship between degree and approximate degree. In such settings, this relationship can be lifted to one between the logarithm of sparsity and the logarithm of approximate sparsity. In particular, since Boolean functions are closed under variable fixing and their degree and approximate degree are polynomially related, it follows that their sparsity and approximate sparsity are polynomially related on a logarithmic scale.*

3.3 Discussion on the Optimality of Our Results

Optimality of the bound in Theorem 1.1. Our result in Theorem 1.1 is optimal up to polynomial factors in $\log n$, as witnessed by the $\textsc{Or}_n$ function. The exact sparsity of $\textsc{Or}_n$ is $2^n - 1$, while its approximate degree is $\Theta(\sqrt{n})$ [44], implying that its approximate sparsity is at most $n^{O(\sqrt{n})}$. Thus, $\log \mathrm{spar}(\textsc{Or}_n) = \Theta(n)$ and $\log \widetilde{\mathrm{spar}}(\textsc{Or}_n) = O(\sqrt{n} \log n)$, showing that the upper bound in Theorem 1.1 is essentially tight up to polynomial factors in $\log n$.

The dependence on n is also unavoidable. Consider the function $\textsc{Thr}^n_{n-1} : \{0,1\}^n \to \{0,1\}$, defined as

$$\textsc{Thr}^n_{n-1}(x) = 1 \quad \text{iff} \quad |x| \geqslant n-1,$$

namely, the function evaluates to 1 if the input has at most one zero. Its exact sparsity is $n+1$, via

$$\textsc{Thr}^n_{n-1}(x) = \sum_{\substack{S \subseteq [n] \\ |S| = n-1}} \prod_{i \in S} x_i - (n-1) \prod_{i \in [n]} x_i,$$

while we show its approximate sparsity is only $O(\log n)$, implying that an additive $O(\log n)$ or multiplicative $O\left(\frac{\log n}{\log \log n}\right)$ factor is necessary in Theorem 1.1.

Claim 3.13. $\widetilde{spar}(\textsc{Thr}^n_{n-1}) = O(\log n)$.

Proof. We begin by introducing a combinatorial structure that underlies our construction.

Separating collections. Let $\{i, j\} \in \binom{[n]}{2}$ be an unordered pair of distinct indices. A set $S \subseteq [n]$ is said to *separate* $\{i, j\}$ if exactly one of i or j belongs to S. A pair $(S_1, S_2) \in 2^{[n]} \times 2^{[n]}$ is said to separate $\{i, j\}$ if at least one of S_1 or S_2 separates it.

We say that a collection $F \subseteq 2^{[n]} \times 2^{[n]}$ is *δ-separating* if, for every pair $\{i, j\} \in \binom{[n]}{2}$, at least a δ-fraction of the elements in F separate it. Formally,

$$\forall \{i, j\} \in \binom{[n]}{2}, \quad |\{(S_1, S_2) \in F : (S_1, S_2) \text{ separates } \{i, j\}\}| \geqslant \delta |F|.$$

We will show that there exists a 2/3-separating collection F of size $O(\log n)$. Assuming such a collection exists, we describe a low-sparsity approximator for $\textsc{Thr}^n_{n-1}$.

Approximator construction. Let $F \subseteq 2^{[n]} \times 2^{[n]}$ be a 2/3-separating collection of size $O(\log n)$. For each pair $(S_1, S_2) \in F$, define

$$f_{(S_1,S_2)}(x) = \left(1 - (1 - \prod_{i \in S_1} x_i)(1 - \prod_{i \notin S_1} x_i)\right) \cdot \left(1 - (1 - \prod_{i \in S_2} x_i)(1 - \prod_{i \notin S_2} x_i)\right).$$

Each function $f_{(S_1,S_2)}$ evaluates to 1 if the input $x \in \{0, 1\}^n$ contains at most one zero, and evaluates to 0 if, for some $S \in \{S_1, S_2\}$, the input x contains zeros in both S and its complement. Define

$$g(x) := \frac{1}{|F|} \sum_{(S_1,S_2) \in F} f_{(S_1,S_2)}(x).$$

We claim that g is a 1/3-approximator for $\textsc{Thr}^n_{n-1}$.

- If x is a 1-input, i.e., x has at most one zero, then for every $S \subseteq [n]$, either $\prod_{i \in S} x_i = 1$ or $\prod_{i \notin S} x_i = 1$. Thus, each term $f_{(S_1,S_2)}(x) = 1$, so $g(x) = 1$.
- If x is a 0-input, i.e., it contains at least two zeros, let $i, j \in [n]$ be distinct positions where $x_i = x_j = 0$. For any $(S_1, S_2) \in F$ that separates $\{i, j\}$, one of S_1 or S_2 contains exactly one of i, j, so one of the products in the corresponding $f_{(S_1,S_2)}(x)$ vanishes, and hence $f_{(S_1,S_2)}(x) = 0$. Since F is 2/3-separating, at least 2/3 of the terms in the sum are 0, so $g(x) \leqslant 1/3$.

Hence, g is a 1/3-approximator for $\textsc{Thr}^n_{n-1}$. Each function $f_{(S_1,S_2)}$ has constant sparsity, and there are $O(\log n)$ such terms in the sum, so the total sparsity of g is $O(\log n)$.

Existence of separating collections. It remains to show that a 2/3-separating collection of size $O(\log n)$ exists. We do this via the probabilistic method.

Let $t = 216 \ln(n^2) = O(\log n)$, and sample $F = \{(S_1^{(k)}, S_2^{(k)})\}_{k=1}^t$, where each set $S_u^{(k)} \subseteq [n]$ (for $u \in \{1, 2\}$) is formed by including each element independently with probability 1/2. We show that with positive probability, F is 2/3-separating.

Fix a pair $\{i, j\} \in \binom{[n]}{2}$, and let X_i be the indicator that $(S_1^{(i)}, S_2^{(i)})$ separates $\{i, j\}$. Each X_i has expectation $\mathbb{E}[X_i] = 3/4$, so the sum $\sum_{i=1}^t X_i$ has expectation $\frac{3t}{4}$. By a Chernoff bound,

$$\Pr\left[\sum_{i=1}^t X_i \leqslant \frac{2t}{3}\right] \leqslant e^{-t/216} \leqslant \frac{1}{n^2}.$$

Taking a union bound over all $\binom{n}{2} < n^2$ pairs, the probability that F fails to be 2/3-separating for some pair is less than 1/2. Hence, with positive probability, a 2/3-separating set F of size $t = O(\log n)$ exists. $\square$

Optimality of the bound in Theorem 1.5. The bound in Theorem 1.5 is also tight up to polynomial factors in $\log n$, again witnessed by $\textsc{Or}_n$. Its exact ℓ_1-norm is $2^n - 1$, while its approximate ℓ_1-norm is at most $n^{O(\sqrt{n})}$, as shown via a standard Chebyshev polynomial approximator.

Observation 3.14 ($\widetilde{\mathrm{wt}}(\textsc{Or}_n) \leqslant n^{O(\sqrt{n})}$)**.** *Let T_d denote the degree-d Chebyshev polynomial defined recursively by $T_0(z) = 1$, $T_1(z) = z$, and $T_d(z) = 2zT_{d-1}(z) - T_{d-2}(z)$ for $d \geqslant 2$. For $d = 2\sqrt{n}$, define*

$$p(z) = 1 - \frac{T_d\left(\frac{n-z}{n-1}\right)}{T_d\left(\frac{n}{n-1}\right)}, \quad \text{and} \quad q(x_1, \ldots, x_n) = p\left(\sum_{i=1}^{n} x_i\right).$$

Then q 1/3-approximates $\textsc{Or}_n$ (see [44, Example 2]). Furthermore, using the recursive definition, it is easy to verify that the coefficients of T_d are bounded in absolute value by 3^d. Therefore, for $d = 2\sqrt{n}$, the ℓ_1-norm of q is at most $n^{O(\sqrt{n})}$.

As with sparsity, the dependence on n in Theorem 1.5 cannot be avoided as well. For $\textsc{Thr}^n_{n-1}$, the exact ℓ_1-norm is $2n - 1$, while the approximator from Claim 3.13 has constant ℓ_1-norm. Thus, a $\log n$ factor—either additive or multiplicative—is necessary.

3.4 Implications for the And Query Model

The measure $\log \mathrm{spar}(f)$ naturally connects to the And-query model—a variant of the standard decision tree model where each query computes the And of an arbitrary subset of input bits. Just as polynomial degree characterizes ordinary deterministic query complexity up to polynomial loss, Knop et al. [40] showed that $\log \mathrm{spar}(f)$ characterizes deterministic query complexity in the And-query model, up to polynomial loss and polylogarithmic factors in n.

In the randomized setting, it is easy to see that $\log \widetilde{\mathrm{spar}}(f)$ lower bounds randomized And-query complexity. Let $\mathrm{R}^{\wedge dt}(f)$ denote the randomized And-query complexity of f. The following is easy to verify:

Claim 3.15 ([39, Claim 3.20])**.** *For every total Boolean function $f : \{0,1\}^n \to \{0,1\}$, we have*

$$\log \widetilde{spar}(f) = O(R^{\wedge dt}(f) + \log n).$$

However, it was unknown whether it also characterizes the randomized query complexity up to polynomial loss. Our results, combined with those of [40], establish that this is indeed the case.

Knop et al. showed that for any Boolean function f,

$$\mathrm{D}^{\wedge dt}(f) = O\left((\log \mathrm{spar}(f))^5 \cdot \log n\right),$$

which, when combined with Theorem 1.1, implies

$$\mathrm{R}^{\wedge dt}(f) \leqslant \mathrm{D}^{\wedge dt}(f) = O\left((\log \widetilde{\mathrm{spar}}(f))^{10} \cdot (\log n)^6\right).$$

A tighter bound can be obtained using a structural result of Knop et al., which relates deterministic And-query complexity to sparsity and a combinatorial measure called *monotone block sensitivity*:

Definition 3.16 (Monotone Block Sensitivity)**.** *The* monotone block sensitivity *of a Boolean function $f : \{0,1\}^n \to \{0,1\}$, denoted $MBS(f)$, is a variant of block sensitivity that only considers flipping 0's to 1's. A subset $B \subseteq [n]$ is called a* sensitive 0-block of f at input x *if $x_i = 0$ for all $i \in B$, and $f(x) \neq f(x \oplus 1_B)$, where $x \oplus 1_B$ denotes the input obtained by flipping all bits in B from 0 to 1. For an input $x \in \{0,1\}^n$, let $MBS(f,x)$ denote the maximum number of pairwise disjoint sensitive 0-blocks of f at x. Then, $MBS(f) = \max_{x \in \{0,1\}^n} MBS(f,x)$.*

Claim 3.17 ([40, Lemma 3.2, Claim 4.4, Lemma 4.6])**.** *For any Boolean function f,*

$$D^{\wedge dt}(f) = O\left((\log MBS(f))^2 \cdot \log spar(f) \cdot \log n\right).$$

Intuitively, a large value of $\mathrm{MBS}(f)$ indicates that a large-arity PROMISE-OR function can be embedded into f via suitable restrictions and identifications of variables.

To tighten our upper bound on $\mathrm{R}^{\wedge dt}(f)$, we now upper bound $\mathrm{MBS}(f)$ in terms of $\log \widetilde{\mathrm{spar}}(f)$. While Knop et al. showed $\mathrm{MBS}(f) = O((\log \mathrm{spar}(f))^2)$, the same proof idea gives a similar bound in terms of approximate sparsity:

Claim 3.18. *For any Boolean function f,*

$$MBS(f) = O\left((\log \widetilde{spar}(f))^2\right).$$

Proof. Assume $\mathrm{MBS}(f) = k \geqslant 40$; otherwise, the claim is trivial. Let this be witnessed by an input $z \in \{0,1\}^n$ and disjoint 0-blocks $B_1, \ldots, B_k \subseteq [n]$, such that $f(z) \neq f(z \oplus 1_{B_i})$ for all $i \in [k]$.

Define $g : \{0,1\}^k \to \{0,1\}$ by identifying variables within each B_i, fixing all others according to z, and letting g be the resulting function. Then $g(0^k) = f(z)$ and $g(x) \neq f(z)$ for all x with Hamming weight 1. Thus, g has sensitivity k at 0^k. Since restrictions and identifications do not increase approximate sparsity, we have $\widetilde{\mathrm{spar}}(g) \leqslant \widetilde{\mathrm{spar}}(f)$, so it suffices to show $\widetilde{\mathrm{spar}}(g)$ is large.

Suppose, for contradiction, that g is 1/3-approximated by a polynomial Q of sparsity

$$\mathrm{spar}(Q) \leqslant \frac{1}{10} \cdot 2^{\ell},$$

for $\ell = c \cdot \sqrt{k/4}$, where $c > 0$ is a constant to be fixed later. We will argue that such a polynomial cannot exist, thereby proving the claim.

Define a distribution $\mathcal{D}$ over restrictions $\rho : \{x_1, \ldots, x_k\} \to \{0, 1, *\}$, where each variable is independently set to 0 with probability 1/2 and left free with probability 1/2. This distribution satisfies the following:

1. By a standard Chernoff bound,

   $$\Pr_{\rho}[|\rho|_* \leqslant k/4] \leqslant e^{-k/16} \leqslant 0.1,$$

   where the last inequality uses $k \geqslant 40$. Thus, with probability at least 0.9, $|\rho|_* \geqslant k/4$.

2. For every ρ in the support of $\mathcal{D}$, the restricted function $g|_\rho$ has sensitivity $|\rho|_*$ at the all-zero input. Hence, by Theorem 4.19, $\widetilde{\deg}(g|_\rho) \geqslant c \cdot \sqrt{|\rho|_*}$.

3. For any monomial M over $\{x_1, \ldots, x_k\}$, we have $\Pr_\rho[\deg(M|_\rho) > 0] = 2^{-\deg(M)}$.

Now consider the restricted polynomial $Q|_\rho$. By property (3) of $\mathcal{D}$, the probability that any fixed monomial in Q of degree at least ℓ survives is at most $2^{-\ell}$, so by a union bound:

$$\Pr_{\rho}[\deg(Q|_\rho) \geqslant \ell] \leqslant \mathrm{spar}(Q) \cdot 2^{-\ell} \leqslant \frac{1}{10}.$$

By property (1) of $\mathcal{D}$, with probability at least 0.9, ρ leaves at least $k/4$ variables free. Thus, with probability at least 0.8, both of the following hold:

$$|\rho|_* \geqslant k/4 \quad \text{and} \quad \deg(Q|_\rho) < \ell.$$

Fix such a restriction ρ. Then $Q|_\rho$ is a polynomial of degree less than ℓ that 1/3-approximates $f|_\rho$. Hence, $\widetilde{\deg}(f|_\rho) < \ell = c \cdot \sqrt{k/4} \leqslant c \cdot \sqrt{|\rho|_*}$, contradicting property (2) of $\mathcal{D}$. Hence, our assumption was false, and the claim follows. □

Theorem 1.4 (Restated). *For every total Boolean function $f : \{0,1\}^n \to \{0,1\}$, the following holds:*

$$\Omega(\log(\widetilde{spar}(f)) - \log n) \overset{(1)}{=} R^{\wedge dt}(f) \leqslant D^{\wedge dt}(f) \overset{(2)}{=} O\left((\log \widetilde{spar}(f))^6 \cdot \log n\right).$$

Proof. The bound in (1) follows from Claim 3.15. For (2), combining Claim 3.17, Claim 3.18, and Theorem 1.1, we get:

$$\mathrm{D}^{\wedge dt}(f) = O\left((\log \mathrm{MBS}(f))^2 \cdot \log \mathrm{spar}(f) \cdot \log n\right) = O\left((\log \widetilde{\mathrm{spar}}(f))^6 \cdot \log n\right).$$ □

This parallels the classical setting, where deterministic and randomized query complexity, degree, and approximate degree are all polynomially related. In the AND-query model, $\log \mathrm{spar}(f)$ plays the role of degree, while $\log \widetilde{\mathrm{spar}}(f)$ plays the role of approximate degree. Combined with the results of [40], our work shows that deterministic and randomized AND-query complexities, log sparsity, and log approximate sparsity are all polynomially related—up to polylogarithmic loss factors.

4 Exact vs Approximate Generalized Representations

In this section, we present another application of our adaptive restriction technique. In the previous section, we showed that approximate polynomial representations of Boolean functions do not offer substantially more succinct representations than exact ones. Here, we study an analogous question in the setting of *generalized polynomials*:

> *Does allowing approximation lead to significantly sparser representations when using generalized polynomials?*

We show that for *monotone functions*, the answer is negative: their exact and approximate generalized sparsity and ℓ_1-norm are polynomially related on the logarithmic scale.

Unlike our result for standard polynomials, where we worked directly with polynomials exhibiting large exact sparsity or ℓ_1-norm, it is unclear how to apply similar reasoning to generalized polynomials. A key obstacle is the non-uniqueness of generalized representations. For example, $\mathrm{OR}_n(x_1, \ldots, x_n)$ can be expressed as $\sum_{\varnothing \neq S \subseteq [n]} (-1)^{|S|} \prod_{i \in S} x_i$, which contains $2^n - 1$ generalized monomials, or equivalently as $1 - \prod_{i=1}^{n}(1 - x_i)$, which involves only 2 generalized monomials. This non-uniqueness makes it difficult to reason directly via generalized complexity measures. Therefore, our approach here proceeds differently: we identify a combinatorial structure that bridges the gap between exact and approximate generalized measures. Our argument proceeds in two steps:

- First, we show that if a monotone function f has large exact generalized sparsity or ℓ_1-norm, then it must have either a large set of maxterms or a large set of minterms.
- Second, we show that the existence of such a large set—either maxterms or minterms—implies large approximate generalized sparsity and large approximate generalized ℓ_1-norm.

Together, these implications show that for monotone functions, the exact and approximate generalized sparsity and ℓ_1-norm cannot be too far apart.

The first implication is straightforward and follows from existing results. The second implication is our main technical contribution and is where we apply our adaptive restriction technique. In fact, we prove a more general result: we introduce a combinatorial notion called *separating set*—a structural property of a set of inputs with respect to a function f—and show that any sufficiently large separating set necessitates high approximate generalized sparsity and ℓ_1-norm. The case of a large set of minterms (or maxterms) is captured as a special instance of this framework.

Organization of this section. In Section 4.1, we present the necessary preliminaries: we define generalized polynomials and their associated complexity measures, and recall some basic properties of monotone functions. In Section 4.2, we define the notion of separating set. In Section 4.3, we establish that the existence of a separating set implies large approximate generalized sparsity and ℓ_1-norm. In Section 4.4, we put things together to conclude that the exact and approximate generalized measures are polynomially related on the logarithmic scale for monotone functions. Finally, in Section 4.5, we discuss implications for the decision tree size in the ordinary query model.

4.1 Preliminaries

Generalized Polynomials. In standard polynomial representations, even simple functions like $\mathrm{OR}_n = 1 - \prod_{i=1}^{n}(1 - x_i)$ can have high sparsity: the standard expansion of OR_n contains $2^n - 1$ monomials. To address this and allow for more compact representations, we consider *generalized polynomials*, which extend standard polynomials by introducing formal complements $\bar{x}_i$ for each variable x_i. For example, $\textsc{Or}_n$ can be written more succinctly as

$$\textsc{Or}_n(x_1, \ldots, x_n) = 1 - \prod_{i=1}^{n} \bar{x}_i,$$

where each $\bar{x}_i$ acts as a stand-in for $1 - x_i$. This representation uses only two monomials, offering exponential savings in sparsity.

We now define generalized polynomials formally.

Definition 4.1 (Generalized Polynomial)**.** *A* generalized polynomial *is a polynomial over the ring*

$$\mathbb{R}[x_1, \ldots, x_n, \bar{x}_1, \ldots, \bar{x}_n]/I,$$

where $\bar{x}_i$ denotes the formal complement of x_i, and I is the ideal generated by the relations:

$$x_i^2 - x_i = 0 \quad \text{and} \quad x_i + \bar{x}_i - 1 = 0 \quad \text{for all } i \in [n].$$

Definition 4.2 (Generalized Representation of Boolean Functions)**.** *A generalized polynomial $Q \in \mathbb{R}[x_1, \ldots, x_n, \bar{x}_1, \ldots, \bar{x}_n]/I$* represents *a function $f : \{0,1\}^n \to \mathbb{R}$ if $Q(x, \bar{x}) = f(x)$ for all $x \in \{0,1\}^n$, where $\bar{x}_i = 1 - x_i$.*

Definition 4.3 (Generalized Complexity Measures)**.** *As in the standard case, one can define the degree, sparsity, and ℓ_1-norm of a generalized polynomial. For a function $f : \{0,1\}^n \to \mathbb{R}$, we define $gdeg(f)$, $gspar(f)$, and $gwt(f)$ as the minimum degree, sparsity, and ℓ_1-norm, respectively, over all generalized polynomials that represent f exactly.*

Analogously, the approximate measures $\widetilde{gdeg}(f)$, $\widetilde{gspar}(f)$, and $\widetilde{gwt}(f)$ denote the minimum degree, sparsity, and ℓ_1-norm among all generalized polynomials that approximate f pointwise within error $1/3$.

Remark 4.4. *Using generalized polynomials offers no advantage in terms of degree. Indeed, each dual variable $\bar{x}_i$ can be replaced by $1 - x_i$, yielding a standard polynomial of the same degree. Therefore,* $\deg(f) = gdeg(f)$ *and* $\widetilde{deg}(f) = \widetilde{gdeg}(f)$*. Since the degree measures coincide, we will simply write* $\deg(f)$ *and* $\widetilde{deg}(f)$ *and avoid using the generalized notation $gdeg(f)$ and $\widetilde{gdeg}(f)$.*

On the other hand, generalized representations are not unique and can be exponentially more succinct. As discussed above, $\textsc{Or}_n$ *has a generalized representation with just two monomials, while its standard representation requires $2^n - 1$.*

Monotone Functions. For $x, y \in \{0,1\}^n$, we write $x \leqslant y$ if $x_i \leqslant y_i$ for all $i \in [n]$. A Boolean function $f : \{0,1\}^n \to \{0,1\}$ is *monotone* if $x \leqslant y$ implies $f(x) \leqslant f(y)$.

Definition 4.5 (Maxterms, Minterms, and Critical Inputs)**.** *Let f be a monotone Boolean function.*

- *A* maxterm *of f is a minimal set $S \subseteq [n]$ such that setting all variables in S to 0 forces f to output 0. The associated* critical 0-input *is the input $x_S \in \{0,1\}^n$ with $x_i = 0$ for $i \in S$ and $x_i = 1$ otherwise. Let $M_0(f) = \{x_S \mid S \text{ is a maxterm of } f\}$ denote the set of all such critical 0-inputs.*
- *A* minterm *of f is a minimal set $S \subseteq [n]$ such that setting all variables in S to 1 forces f to output 1. The associated* critical 1-input *is the input x_S with $x_i = 1$ for $i \in S$ and $x_i = 0$ otherwise. Let $M_1(f) = \{x_S \mid S \text{ is a minterm of } f\}$ denote the set of all such critical 1-inputs.*

Observation 4.6. *For any monotone function f, every $x \in M_1(f)$ is sensitive on all 1-bits: flipping any 1 to 0 changes $f(x)$ from 1 to 0. Similarly, every $x \in M_0(f)$ is sensitive on all 0-bits.*

The following result of Ehrenfeucht and Haussler [26, Lemmas 1 and 6] implies that if a monotone Boolean function has large exact generalized sparsity or ℓ_1-norm, then the number of minterms (i.e., critical 1-inputs) or maxterms (i.e., critical 0-inputs) must be large.

Theorem 4.7 (Ehrenfeucht and Haussler [26])**.** *For every monotone Boolean function $f : \{0,1\}^n \to \{0,1\}$, the following hold:*

(a) $\log gspar(f) = O\left(\log^2 M(f) \cdot \log n\right)$,

(b) $\log gwt(f) = O\left(\log^2 M(f) \cdot \log n\right)$,

where $M(f) = |M_0(f)| + |M_1(f)|$.

Remark 4.8. *Ehrenfeucht and Haussler [26] proved a more general result: for any Boolean function* f*, they showed*

$$\log DSize^{dt}(f) = O(\log^2 \mathrm{Cover}(f) \cdot \log n),$$

where $DSize^{dt}(f)$ *is the size (number of leaves) of the smallest ordinary decision tree computing* f*, and* $\mathrm{Cover}(f)$ *is the minimum number of monochromatic subcubes under* f *that cover* $\{0,1\}^n$*.*

For monotone functions, the minimal subcube cover corresponds exactly to subcubes defined by maxterms and minterms, so $\mathrm{Cover}(f) = M(f)$*. Hence, the above bound specializes to:*

$$\log DSize^{dt}(f) = O(\log^2 M(f) \cdot \log n).$$

Moreover, any decision tree of size s *and depth* d *computing a function* f *can be converted into a generalized polynomial of degree at most* d*, and sparsity and* ℓ_1*-norm at most* s*, that also computes* f*. Hence,*

$$gspar(f) \leqslant DSize^{dt}(f) \quad \text{and} \quad gwt(f) \leqslant DSize^{dt}(f).$$

and the bound in Theorem 4.7 follows.

4.2 Separating Sets of Inputs

For any $B \subseteq [n]$, let $1_B \in \{0,1\}^n$ denote the string with 1s in coordinates indexed by B and 0s elsewhere. For $x \in \{0,1\}^n$ and $i \in [n]$, we say that i is a *sensitive coordinate* of f at x if $f(x \oplus 1_{\{i\}}) \neq f(x)$. Let $S(f,x) \subseteq [n]$ denote the set of all such coordinates. The *sensitivity* of f is defined as $\mathrm{s}(f) = \max_{x \in \{0,1\}^n} |S(f,x)|$.

We now define the notion of a *separating* set of inputs for a Boolean function f. Informally, its a collections of inputs that differ on their sensitive coordinates. Formally,

Definition 4.9 (Separating Set of Inputs)**.** *Let* $f : \{0,1\}^n \to \{0,1\}$ *be a Boolean function. A set* $F \subseteq \{0,1\}^n$ *is said to be* separating *(with respect to* f*) if for every distinct pair* $x, y \in F$*, the projections of* x *and* y *to the union of their sensitive coordinates differ; that is, letting* $B = S(f,x) \cup S(f,y)$*, we require* $x|_B \neq y|_B$*. We refer to this condition as the* separation property*.*

We record some basic properties of separating sets that will be used later. First, the separation property is preserved under restrictions.

Claim 4.10 (Closure under restriction)**.** *Let* $F \subseteq \{0,1\}^n$ *be a separating set with respect to* f*, and let* ρ *be any restriction. Then the restricted set* $F|_\rho$ *is separating with respect to the restricted function* $f|_\rho$*.*

Proof. Suppose not. Then there exist $x, y \in F|_\rho$ such that

$$x|_B = y|_B \quad \text{for } B := S(f|_\rho, x) \cup S(f|_\rho, y).$$

Let $x', y' \in F$ be extensions of x, y consistent with ρ. Then

$$x'|_{B'} = y'|_{B'} \quad \text{for } B' := S(f, x') \cup S(f, y') \subseteq B \cup \mathsf{SetVars}(\rho),$$

contradicting the the separation property of F. □

The separation property also ensures that a nontrivial fraction of inputs ($\approx 1/n$) in any separating set share a common sensitive coordinate:

Claim 4.11. *Let* $F \subseteq \{0,1\}^n$ *be a separating set for* f *with* $|F| \geqslant 2$*. Then there exists an index* $i \in [n]$ *and a subset* $F' \subseteq F$ *such that* $i \in S(f,x)$ *for every* $x \in F'$*, and* $|F'| \geqslant |F|/(2n)$*.*

Proof. If $S(f,x) \neq \emptyset$ for all $x \in F$, then by averaging, some $i \in [n]$ appears in $S(f,x)$ for at least $|F|/n$ inputs; let $F' := \{x \in F : i \in S(f,x)\}$.

Otherwise, let $x \in F$ have $S(f,x) = \emptyset$. Separation property then forces every $y \in F\backslash\{x\}$ to satisfy $S(f,y) \neq \emptyset$. Averaging over $F\backslash\{x\}$, some $i \in [n]$ appears in $S(f,y)$ for at least $(|F|-1)/n \geqslant |F|/(2n)$ inputs. Let $F' := \{y \in F\backslash\{x\} : i \in S(f,y)\}$. ☐

We next observe that set of critical 1-inputs $M_1(f)$ and the set of critical 0-inputs $M_0(f)$ arising from monotone functions are separating:

Claim 4.12. *Let $f : \{0,1\}^n \to \{0,1\}$ be monotone. Then both the sets of critical 1-inputs $M_1(f)$ and critical 0-inputs $M_0(f)$ are separating.*

Proof. Consider distinct inputs $x, y \in M_1(f)$. By the definition of critical 1-inputs, we have $S(f,x) = \{i \in [n] : x_i = 1\}$, and $S(f,y) = \{i \in [n] : y_i = 1\}$. Since $x \neq y$, their sets of 1s differ, so there exists $i \in S(f,x) \cup S(f,y)$ such that $x_i \neq y_i$. Hence, $x|_B \neq y|_B$ for $B = S(f,x) \cup S(f,y)$, as required. A similar argument shows that $M_0(f)$ is also separating. ☐

Combining this with Theorem 4.7, we conclude that any monotone function with large exact generalized sparsity or exact ℓ_1-norm must have a large separating set. Therefore, to relate the exact and approximate generalized sparsity and ℓ_1-norm of monotone functions, it suffices to show that large separating sets lead to large approximate sparsity and ℓ_1-norm. We establish this in the next section.

4.3 Large Separating Set of Inputs Implies Large Approximate Generalized Sparsity and ℓ_1-norm

In this subsection, we show that the existence of a large separating set for a function f implies that f must have large approximate generalized sparsity and ℓ_1-norm. We begin with an outline of the proof for the sparsity case; a similar argument also yields a lower bound on the approximate generalized ℓ_1-norm.

Proof Outline. The proof closely follows the template used earlier to relate sparsity and approximate sparsity for ordinary polynomials, with a key variation. This may be viewed as a second application of that general proof strategy.

Given a large separating set F with respect to a function f, we sample a carefully designed random restriction ρ, as described in Algorithm 2, which satisfies the following key properties:

1. With high probability, ρ leaves a significant number of variables free—specifically, $\ell = \Omega(\log|F|/\log n)$.
2. The restricted function $f|_\rho$ has full sensitivity; that is, $\mathrm{s}(f|_\rho) = \ell$.
3. For any generalized monomial, the probability that its degree under the restriction ρ exceeds t is at most 2^{-t}.

These properties lead us to conclude that any generalized polynomial approximating f must have large sparsity. Suppose, for contradiction, that there exists a polynomial Q that 1/3-approximates f and has generalized sparsity at most $2^{c\sqrt{\ell}}$, for some constant c. Then, using property (3) and a standard probabilistic argument, we can find a restriction ρ such that all generalized monomials of degree greater than $c\sqrt{\ell}$ are eliminated. Consequently, the restricted polynomial $Q|_\rho$ has degree at most $c\sqrt{\ell}$ and approximate $f|_\rho$. However, property (2) tells us that $f|_\rho$ has sensitivity ℓ, and by the known relationship between sensitivity and approximate degree [44], this implies that any approximating polynomial for $f|_\rho$ must have degree at least $\Omega(\sqrt{\ell})$. This contradicts the assumption that $Q|_\rho$ has degree $c\sqrt{\ell}$. Thus, any approximating generalized polynomial for f must have sparsity at least $2^{\Omega(\sqrt{\ell})}$, where $\ell = \Omega(\log|F|/\log n)$.

Let us compare this restriction-based argument with the one from the previous section (Algorithm 1). Property (1) remains the same in spirit—the number of free variables is again $\ell = \log|F|/\log n$, as opposed to $\log \mathrm{spar}(f)/\log n$ in the earlier case. Property (2) differs: previously, the restricted function had full degree, while here it has full sensitivity. However, since we used large degree previously only to infer large approximate degree, we can make a similar inference here using the known connection between sensitivity and approximate degree. Hence, the role of property (2) in the argument remains essentially unchanged.

Property (3), on the other hand, is stronger than before. In the earlier argument, we could only reduce the degree of ordinary monomials, whereas here we are able to reduce the degree of even generalized monomials. This strengthens the conclusion of the argument.

As is evident, the overall structure of the argument closely mirrors the earlier one, with suitable adjustments to handle generalized monomials.

The following definition abstracts the essential properties of the random restriction process.

Definition 4.13 (ℓ-Variable Max-Sensitivity Distribution)**.** *Let $f : \{0,1\}^n \to \{0,1\}$, and let $\mathcal{D}$ be a distribution over restrictions $\rho : \{x_1, \dots, x_n\} \to \{0,1,*\}$. We say that $\mathcal{D}$ is an ℓ-variable* max-sensitivity distribution *for f if:*

1. *With probability at least 0.9, ρ leaves at least ℓ variables free.*
2. *For every ρ in the support of $\mathcal{D}$, we have $s(f|_\rho) = |\rho|_*$.*
3. *For any generalized monomial M and any $t \in \mathbb{N}$, $\Pr_{\rho\sim\mathcal{D}}[\deg(M|_\rho) \geqslant t] \leqslant 2^{-t}$.*

We will show that a large separating set F implies the existence of such a distribution, which in turn implies that any generalized polynomial approximating f must have large sparsity and ℓ_1-norm.

Algorithm 2 MaxSensitivityRestriction

1: **Input:** A set $V \subseteq \{x_1, \dots, x_n\}$; $f : \{0,1\}^V \to \{0,1\}$; a nonempty set $F \subseteq \{0,1\}^V$ separating w.r.t f
2: **Output:** A restriction $\rho : V \to \{0,1,*\}$.
3: **if** $|F| \leqslant 2$ **then**
4:     Let w be an input in F
5:     For each $x_i \in V$, set $\rho(x_i) \leftarrow w_i$
6: **else**
7:     **if** there exists $x_i \in V$, $u \in \{0,1\}$ such that $|F|_{x_i=u}| \geqslant \left(1-\frac{1}{n}\right)\cdot|F|$ **then**
8:         $\rho' \leftarrow$ MaxSensitivityRestriction$(V\backslash\{x_i\}, f|_{x_i=u}, F|_{x_i=u})$
9:         Set $\rho(x_i) \leftarrow u$, and for all $x_j \in V\backslash\{x_i\}$, set $\rho(x_j) \leftarrow \rho'(x_j)$
10:     **else**
11:         Choose $x_i \in V$ and $u \in \{0,1\}$ such that $|\{w \in F \mid i \in S(f,w) \text{ and } w_i = u\}| \geqslant \frac{|F|}{4n}$
12:         With probability 1/3:
13:             $\rho_0 \leftarrow$ MaxSensitivityRestriction$(V\backslash\{x_i\}, f|_{x_i=0}, F|_{x_i=0})$
14:             Set $\rho(x_i) \leftarrow 0$, and for all $x_j \in V\backslash\{x_i\}$, set $\rho(x_j) \leftarrow \rho_0(x_j)$
15:         With probability 1/3:
16:             $\rho_1 \leftarrow$ MaxSensitivityRestriction$(V\backslash\{x_i\}, f|_{x_i=1}, F|_{x_i=1})$
17:             Set $\rho(x_i) \leftarrow 1$, and for all $x_j \in V\backslash\{x_i\}$, set $\rho(x_j) \leftarrow \rho_1(x_j)$
18:         Otherwise:
19:             $F' \leftarrow \{w \in F \mid i \in S(f,w)\}$
20:             $\rho_* \leftarrow$ MaxSensitivityRestriction$(V\backslash\{x_i\}, f|_{x_i=u}, F'|_{x_i=u})$
21:             Set $\rho(x_i) \leftarrow *$, and for all $x_j \in V\backslash\{x_i\}$, set $\rho(x_j) \leftarrow \rho_*(x_j)$
22:     **end if**
23: **end if**
24: **return** ρ

Properties of the Restriction Process. Algorithm 2 describes how to sample random restrictions for a given separating set F w.r.t f. We will show that the resulting distribution over restrictions is ℓ-variable max-sesitivity for f, where $\ell = \Omega\left(\frac{\log|F|}{\log n}\right)$.

We begin with some observations concerning the validity of Algorithm 2. The step at line 11, which selects an $x_i \in V$, is justified by the following observation.

Observation 4.14. *By Claim 4.11, at line 11, Algorithm 2 is guaranteed to find $x_i \in V$ such that*

$$|\{w \in F \mid i \in S(f,w)\}| \geqslant \frac{|F|}{2n}.$$

Moreover, choosing $u \in \{0,1\}$ to be the more frequent value of the i-th coordinate among these inputs ensures that

$$|\{w \in F \mid i \in S(f,w) \text{ and } w_i = u\}| \geqslant \frac{|F|}{4n}.$$

We next address the correctness of the recursive structure. Since the separation property is preserved under restrictions, the recursive calls made on lines 8, 13, 16, and 20 satisfy the preconditions of Algorithm 2.

We now observe a key structural property of the restriction ρ returned by the algorithm, which shows property (2) of the max-sensitivity distribution.

Claim 4.15. *Let ρ be the final restriction returned by Algorithm 2 on input (V, f, F). Then there exists an input $w \in F$ such that:*

$$\forall x_i \in \mathsf{SetVars}(\rho),\ \rho(x_i) = w_i, \quad \textit{and} \quad \forall x_i \in \mathsf{FreeVars}(\rho),\ i \in S(f,w).$$

Proof. We prove the claim by induction on the size of the V set, which reduces with each recursive call.

Base case ($|V| = 1$). In this case, the input set F must have size at most 2. The algorithm simply fixes the lone variable in V to either 0 or 1 based on some $w \in F$, and returns the corresponding restriction. The claim then holds trivially for this choice of w.

Inductive step ($|V| > 1$). We consider the possible return paths in the algorithm depending on the branching conditions and the randomness involved (lines 3, 7, 12, 15, and 18):

1. **Case $|F| \leqslant 2$:** This is similar to the base case. The algorithm returns a restriction that sets all variables in V to match some $w \in F$, and the claim follows directly.
2. **Case: branch taken via line 7, 12, or 15.** In these branches, the algorithm chooses some $x_i \in V$ and $u \in \{0,1\}$, and returns a restriction of the form $\rho = \rho' \cup \{x_i \leftarrow u\}$, where

   $$\rho' \leftarrow \textsc{MaxSensitivityRestriction}(V \backslash \{x_i\}, f|_{x_i=u}, F|_{x_i=u}).$$

   By the inductive hypothesis applied to ρ', there exists $w' \in F|_{x_i=u}$ such that

   $$\forall x_j \in \mathsf{SetVars}(\rho'),\ \rho'(x_j) = w'_j, \quad \text{and} \quad \forall x_j \in \mathsf{FreeVars}(\rho'),\ j \in S(f|_{x_i=u}, w').$$

   Let $w \in F$ be an extension of w' with $x_i = u$. Then $\rho(x_j) = w_j$ for all set variables x_j, and for all free variables x_j, we have $j \in S(f,w)$ since $j \in S(f|_{x_i=u}, w')$. Thus, the claim holds for this w.
3. **Case: branch taken via line 18.** In this case, for $x_i \in V$ and $u \in \{0,1\}$ selected in line 10, the ρ returned is $\rho_* \cup \{x_i \leftarrow *\}$ where

   $$\rho_* \leftarrow \textsc{MaxSensitivityRestriction}(V \backslash \{x_i\}, f|_{x_i=u}, F'|_{x_i=u}).$$

   for $F' \leftarrow \{w \in F \mid i \in S(f,w)\}$. By the inductive hypothesis applied to ρ_*, there exists $w' \in F'|_{x_i=u}$ such that

   $$\forall x_j \in \mathsf{SetVars}(\rho_*),\ \rho_*(x_j) = w'_j, \quad \text{and} \quad \forall x_j \in \mathsf{FreeVars}(\rho_*),\ j \in S(f|_{x_i=u}, w').$$

   Let $w \in F'$ be an extension of w' with $x_i = u$. Since $i \in S(f,w)$, by definition of F', and since x_i is free in ρ, the required conditions are satisfied by w for the final restriction $\rho = \rho_* \cup \{x_i \leftarrow *\}$.

This completes the inductive proof. ☐

As a consequence, for any restriction ρ in the support of the distribution induced by Algorithm 2, the restricted function $f|_\rho$ has sensitivity exactly $|\rho|_*$, thereby satisfying property (2) of the max-sensitivity distribution.

We next show that the restriction ρ produced by Algorithm 2 leaves a significant number of variables free, establishing property (1) of the max-sensitivity distribution. The argument closely mirrors that of the earlier restriction algorithm.

Algorithm 2 proceeds recursively and follows one of the following three branches:

- If $|F| \leqslant 2$, the algorithm terminates and returns ρ immediately (line 3).
- If the condition on line 7 is satisfied, the algorithm makes a single recursive call (line 8).
- Otherwise (line 10), the algorithm selects one of three recursive calls (lines 13, 16, 20) uniformly at random, each with probability 1/3.

The recursion continues as long as $|F| > 2$, and halts when $|F| \leqslant 2$, at which point the algorithm backtracks to construct the final restriction.

As in earlier analyses, we classify recursive calls as either *active* (when the condition on line 10 holds) or *passive* (when the condition on line 7 holds). We argue that a substantial fraction of the calls must be active, and in each such call, a variable is left free with probability 1/3. Moreover, these choices are made independently across calls. Therefore, by standard concentration bounds, the number of free variables in the final restriction is close to its expectation, which is 1/3 the number of active calls. This leads to the following claim:

Claim 4.16. *Let (V, f, F) satisfy the input requirements of Algorithm 2, and suppose that $|F| \geqslant 20(4n)^{60}$. Then, with probability at least 0.9, the restriction $\rho = \textsc{MaxSensitivityRestriction}(V, f, F)$ produced by Algorithm 2 leaves at least $\Omega\left(\frac{\log|F|}{\log n}\right)$ variables free.*

Proof. We begin by showing that every execution of the algorithm encounters a substantial number of *active* recursive calls. Suppose the algorithm makes t recursive calls in total, of which ℓ are active. Since the size of V decreases by 1 with each recursive call, and the algorithm halts when $|V| = 1$ (which corresponds to $|F| \leqslant 2$), we have $t \leqslant n$.

We observe how the size of the separating set F evolves during recursion:

- In a *passive* call (i.e., when line 7 is satisfied), the size of F decreases by at most a factor of $1 - \frac{1}{n}$.
- In an *active* call (i.e., when line 10 is satisfied), the reduction in the size of F depends on the specific recursive branch taken:
  - For recursive calls on lines 13 and 16, the size of F decreases by at most a factor of $1/n$, owing to the balancedness condition enforced by line 10.
  - For the recursive call on line 20, the size of F decreases by at most a factor of $1/(4n)$, due to the choice of index i in line 11 (see Observation 4.14).

  Therefore, in any *active* call, the size of F decreases by at most a factor of $1/(4n)$, regardless of which of the three recursive branches is chosen.

Since the algorithm halts and backtracks when the $|F| \leqslant 2$, we obtain the following inequality:

$$\begin{aligned}
2 &\geqslant (1-1/n)^{t-\ell} \cdot (1/4n)^{\ell} \cdot |F| \\
&\geqslant (1-1/n)^{n} \cdot (1/4n)^{\ell} \cdot |F| && \text{(since } t-\ell \leqslant n\text{)} \\
&\geqslant \frac{1}{10} \cdot \left(\frac{1}{4n}\right)^{\ell} \cdot |F| && \text{(using } (1-1/n)^n \geqslant 1/10 \text{ for } n \geqslant 2\text{).}
\end{aligned}$$

Taking logarithms and rearranging, we obtain:

$$\ell \geqslant \frac{\log(|F|/20)}{\log(4n)}.$$

Define $\ell^* := \frac{\log(|F|/20)}{\log(4n)}$. Thus, every run of the algorithm contains at least ℓ^* active recursive calls.

Let $X_1, X_2, \ldots, X_{\ell^*}$ be indicator random variables, where $X_i = 1$ if the variable chosen in the i-th active call is left free (which occurs with probability 1/3), and 0 otherwise. These variables are independent by construction, and $\mathbb{E}\left[\sum_{i=1}^{\ell^*} X_i\right] = \frac{\ell^*}{3}$. By a standard Chernoff bound, we get:

$$\Pr\left(\sum_{i=1}^{\ell^*} X_i \leqslant \frac{\ell^*}{6}\right) \leqslant e^{-\ell^*/24} \leqslant 0.1,$$

where the last inequality follows from the assumption $|F| \geqslant 20(4n)^{60}$.

Since the number of variables left free in the final restriction is at least $\sum_{i=1}^{\ell^*} X_i$, we conclude that with probability at least 0.9, the algorithm leaves at least $\ell^*/6 = \Omega\left(\frac{\log|F|}{\log n}\right)$ variables free. ☐

Finally, we establish property (3) of the max-sensitivity distribution, showing that for any generalized monomial M, the degree of $M|_\rho$ under the sampled restriction ρ exhibits exponential tail decay.

Claim 4.17. *Let $\rho = \textsc{MaxSensitivityRestriction}(V, f, F)$ be the restriction returned by Algorithm 1. Then, for any generalized monomial M with $\mathsf{Vars}(M) \subseteq \{x_i \mid x_i \in V\}$, and any $t \in \mathbb{N}$, we have:*

$$\Pr_\rho\left(\deg(M|_\rho) \geqslant t\right) \leqslant 2^{-t}.$$

Proof. We prove the claim by induction on $|V|$.

Base case ($|V| = 1$). Here, we must have $|F| \leqslant 2$, the restriction ρ returned by the algorithm sets all variables in V to either 0 or 1, based on some $w \in F$. Thus, any monomial M becomes a constant under ρ. The claim follows.

Inductive step ($|V| > 1$). We consider the behavior of the algorithm based on the three possible branches (lines 3, 7, and 10):

1. **Case $|F| \leqslant 2$:** Similar to the base case, ρ returned by the algorithm sets all variables in V to either 0 or 1, based on some $w \in F$, and hence $\deg(M|_\rho) = 0$.
2. **Case where the "if" condition (line 7) is satisfied:** Suppose the condition is satisfied for some $x_i \in V$ and some value $u \in \{0, 1\}$. The algorithm then returns the restriction $\rho = \rho' \cup \{x_i \leftarrow u\}$, where ρ' is obtained from a recursive call with a strictly smaller support set. By the inductive hypothesis applied to ρ', for any generalized monomial M, we have:
$$\Pr_\rho\left(\deg(M|_\rho) \geqslant t\right) \leqslant \Pr_{\rho'}\left(\deg(M|_{\rho'}) \geqslant t\right) \leqslant 2^{-t}.$$
3. **Case: the `else` clause at line 9 is executed.** Let $x_i \in V$ be the variable selected in line 10, and let ρ_0, ρ_1, ρ_* be restrictions sampled from the recursive calls at lines 13, 16, and 20, respectively. For a generalized monomial M, we consider two cases:
   - ***M* does not contain x_i or $\bar{x}_i$:** In this case, the monomial is unaffected by the assignment to x_i. By applying induction hypothesis, we have:
   $$\Pr_\rho(\deg(M|_\rho) \geqslant t) = \tfrac{1}{3}\Pr_{\rho_0}(\deg(M|_{\rho_0}) \geqslant t) + \tfrac{1}{3}\Pr_{\rho_1}(\deg(M|_{\rho_1}) \geqslant t) + \tfrac{1}{3}\Pr_{\rho_*}(\deg(M|_{\rho_*}) \geqslant t) \leqslant 2^{-t}.$$
   - ***M* contains x_i or $\bar{x}_i$:** Without loss of generality, suppose $x_i \in M$ (the case $\bar{x}_i \in M$ is analogous). Let $M' = M/x_i$. Then:
   $$\Pr_\rho(\deg(M|_\rho) \geqslant t) = \frac{1}{3}\cdot 0 + \frac{1}{3}\Pr_{\rho_1}(\deg(M'|_{\rho_1}) \geqslant t) + \frac{1}{3}\Pr_{\rho_*}(\deg(M'|_{\rho_*}) \geqslant t-1),$$
   where the first term is zero because setting $x_i \leftarrow 0$ kills the monomial. Using the inductive hypothesis:
   $$\Pr_\rho(\deg(M|_\rho) \geqslant t) \leqslant \frac{1}{3}\cdot 2^{-t} + \frac{1}{3}\cdot 2^{-(t-1)} = 2^{-t}.$$

This completes the inductive proof. ☐

Applying Algorithm 2 to a separating set F for f, and combining Claim 4.15, Claim 4.16, and Claim 4.17, we conclude that the existence of a large separating set implies the existence of a max-sensitivity distribution. This yields the following theorem.

Theorem 4.18. *Let $f : \{0,1\}^n \to \{0,1\}$ be a Boolean function, and let $F \subseteq \{0,1\}^n$ be a separating set for f with $|F| \geqslant 20(4n)^{60}$. Then there exists an ℓ-variable max-sensitivity distribution for f, where $\ell = \Omega\left(\frac{\log|F|}{\log n}\right)$.*

Putting Everything Together. We first show that the existence of a max-sensitivity distribution forces any generalized polynomial approximating f to have large sparsity and ℓ_1-norm. Combined with Theorem 4.18, this implies that a large separating set yields large approximate generalized sparsity and large approximate generalized ℓ_1-norm.

We will use the following classical result of Nisan and Szegedy [44], which relates the approximate degree of a Boolean function to its sensitivity.

Theorem 4.19 (Nisan and Szegedy [44]). *Let $f : \{0,1\}^n \to \{0,1\}$ be a Boolean function. Then,*

$$\widetilde{deg}(f) \geqslant \sqrt{\frac{s(f)}{6}}.$$

Claim 4.20. *Let $f : \{0,1\}^n \to \{0,1\}$, and suppose there exists an ℓ-variable max-sensitivity distribution $\mathcal{D}$ for f. Then,*

$$\log \widetilde{gspar}(f) = \Omega(\sqrt{\ell}).$$

Proof. Let $\mathcal{D}$ be an ℓ-variable max-sensitvity distribution for f. Suppose, for contradiction, that the claimed bound does not hold. Let $k = \sqrt{\ell/6}$ and assume there exists a real generalized polynomial Q that 1/3-approximates f and has sparsity

$$\mathrm{spar}(Q) \leqslant \frac{1}{10} \cdot 2^k.$$

We will argue that such a polynomial cannot exist, thereby proving the claim.

Sample a restriction $\rho \sim \mathcal{D}$, and consider the restricted polynomial $Q|_\rho$. By property (3) of $\mathcal{D}$, the probability that any fixed generalized monomial in Q has degree at least k under ρ is at most 2^{-k}. Applying a union bound over all monomials in Q, we have

$$\Pr_\rho\left(\deg(Q|_\rho) \geqslant k\right) \leqslant \mathrm{spar}(Q) \cdot 2^{-k} \leqslant \frac{1}{10}.$$

By property (1) of $\mathcal{D}$, with probability at least 0.9, ρ leaves at least ℓ variables free. Thus, with probability at least 0.8, both of the following hold:

$$|\rho|_* \geqslant \ell \quad \text{and} \quad \deg(Q|_\rho) < k.$$

Fix such a restriction ρ. Then $Q|_\rho$ is a polynomial of degree less than k that 1/3-approximates $f|_\rho$. By property (2) of $\mathcal{D}$, we have:

$$\mathrm{s}(f|_\rho) = |\rho|_* \geqslant \ell = 6 \cdot k^2 > 6 \cdot (\deg(Q|_\rho))^2 \geqslant 6 \cdot \widetilde{\deg}(f|_\rho)^2.$$

This contradicts the relationship between approximate degree and sensitivity Theorem 4.19. Hence, our assumption was false, and the claim follows. ☐

Combining the above with Grolmusz's theorem [29, 52] (see Remark 3.11), which gives

$$\log \widetilde{\mathrm{gspar}}(f) = O\left(\log \widetilde{\mathrm{gwt}}(f) + \log n\right),$$

we also obtain a lower bound on the approximate generalized ℓ_1-norm from the existence of a max-sensitivity distribution. While this approach incurs an extra additive $\log n$ loss compared to the bound above, it can be avoided by directly arguing as in Claim 3.9. Since the proof involves no new ideas, we omit the details for brevity and state the resulting optimal bound:

Claim 4.21. *Let $f : \{0,1\}^n \to \{0,1\}$, and suppose there exists an ℓ-variable max-sensitivity distribution $\mathcal{D}$ for f. Then,*

$$\log \widetilde{gwt}(f) = \Omega(\sqrt{\ell}).$$

Combining Theorem 4.18 with Claim 4.20 and Claim 4.21, we obtain the following consequences of the existence of a large separating set:

Theorem 4.22. *Let $f : \{0,1\}^n \to \{0,1\}$ be a Boolean function, and let $F \subseteq \{0,1\}^n$ be a separating set with respect to f. Then,*

$$\log \widetilde{gspar}(f) = \Omega\left(\left(\frac{\log |F|}{\log n}\right)^{1/2}\right).$$

Theorem 4.23. *Let $f : \{0,1\}^n \to \{0,1\}$ be a Boolean function, and let $F \subseteq \{0,1\}^n$ be a separating set with respect to f. Then,*

$$\log \widetilde{gwt}(f) = \Omega\left(\left(\frac{\log |F|}{\log n}\right)^{1/2}\right).$$

4.4 Exact vs Approximate Generalized Measures for Monotone Functions

We now relate the generalized sparsity and generalized ℓ_1-norm of a monotone function to their approximate counterparts. This is done by combining Theorem 4.7 with the results from the previous section.

Theorem 1.8 (Restated). *For every monotone Boolean function $f : \{0,1\}^n \to \{0,1\}$, the following hold:*

(a) $\log gspar(f) = O\left((\log \widetilde{gspar}(f))^4 \cdot (\log n)^3\right)$,

(b) $\log gwt(f) = O\left((\log \widetilde{gwt}(f))^4 \cdot (\log n)^3\right)$.

Proof. From Theorem 4.7, we have:

$$\log M(f) = \Omega\left(\sqrt{\frac{\log \mathrm{gspar}(f)}{\log n}}\right) \quad \text{and} \quad \log M(f) = \Omega\left(\sqrt{\frac{\log \mathrm{gwt}(f)}{\log n}}\right).$$

This implies that either the number of critical 1-inputs or the number of critical 0-inputs is large. Without loss of generality, assume $|M_1(f)| \geqslant M(f)/2$.

Since both the sets of critical 1-inputs and critical 0-inputs are separating with respect to f, we can apply Theorem 4.22 and Theorem 4.23 to the set $M_1(f)$, yielding:

$$\log \mathrm{gspar}(f) = O\left(\log^2 |M_1(f)| \cdot \log n\right) = O\left((\log \widetilde{\mathrm{gspar}}(f))^4 \cdot (\log n)^3\right),$$

$$\log \mathrm{gwt}(f) = O\left(\log^2 |M_1(f)| \cdot \log n\right) = O\left((\log \widetilde{\mathrm{gwt}}(f))^4 \cdot (\log n)^3\right),$$

as claimed. ∎

4.5 Implications for Decision Tree Size in the Ordinary Query Model

The measures $\log \mathrm{gspar}(f)$ and $\log \mathrm{gwt}(f)$ are related to the decision tree size $\log \mathrm{DSize}^{dt}(f)$ in the ordinary query model, as noted in Remark 4.8. Specifically, any decision tree of size s computing f can be converted into a generalized polynomial for f with sparsity and ℓ_1-norm at most s.

For monotone functions, applying our result Theorem 1.8 together with Remark 4.8, we obtain:

Corollary 1.9 (Restated). *For every monotone Boolean function $f : \{0,1\}^n \to \{0,1\}$, the following hold:*

(a) $\Omega(\log(\widetilde{gspar}(f)) - \log n) \overset{(1)}{=} \log RSize^{dt}(f) \leqslant \log DSize^{dt}(f) \overset{(2)}{=} O\left((\log \widetilde{gspar}(f))^4 \cdot (\log n)^3\right)$.

(b) $\Omega(\log(\widetilde{gwt}(f))) \overset{(1)}{=} \log RSize^{dt}(f) \leqslant \log DSize^{dt}(f) \overset{(2)}{=} O\left((\log \widetilde{gwt}(f))^4 \cdot (\log n)^3\right)$.

Proof. For (1) of both (a) and (b), assume $\mathrm{RSize}^{dt}(f) = s$, i.e., there exists a distribution $\mathcal{D}$ over decision trees, each of size at most s, such that for every input $x \in \{0,1\}^n$, sampling a decision tree T from $\mathcal{D}$ and evaluating it on x yields:

$$\Pr_{T \sim \mathcal{D}}[T(x) = f(x)] \geqslant 2/3.$$

By a standard Chernoff bound argument, we can assume $\mathcal{D}$ is supported on $O(n)$ trees, as such a distribution always exists. Each decision tree in the support of $\mathcal{D}$ can be converted into a generalized polynomial of sparsity and ℓ_1-norm at most s that agrees with the tree. Taking a convex combination of these polynomials (weighted by $\mathcal{D}$) gives a generalized polynomial that 1/3-approximates f with sparsity $O(sn)$ and ℓ_1-norm $\leqslant s$, implying $\widetilde{\mathrm{gspar}}(f) = O(sn)$ and $\widetilde{\mathrm{gwt}}(f) \leqslant s$. Taking logarithms gives the desired bound (1) of (a) and (b).

For (2), we apply Theorem 1.8 together with Remark 4.8, obtaining:

$$\log \mathrm{DSize}^{dt}(f) = O(\log^2 M(f) \cdot \log n) = O\left((\log \widetilde{\mathrm{gspar}}(f))^4 \cdot (\log n)^3\right),$$

$$\log \mathrm{DSize}^{dt}(f) = O(\log^2 M(f) \cdot \log n) = O\left((\log \widetilde{\mathrm{gwt}}(f))^4 \cdot (\log n)^3\right). \qquad \square$$

Thus, for monotone functions, the complexity measures $\mathrm{gspar}(f)$, $\widetilde{\mathrm{gspar}}(f)$, $\mathrm{gwt}(f)$, $\widetilde{\mathrm{gwt}}(f)$, $\mathrm{DSize}^{dt}(f)$, and $\mathrm{RSize}^{dt}(f)$ are all polynomially related on the logarithmic scale, up to polylogarithmic factors in n. In contrast, such a relationship fails for general functions; see Remark 1.10. Specifically, there exists a function f on n bits with $\mathrm{gspar}(f) = O(\sqrt{n})$ but $\mathrm{RSize}^{dt}(f) = 2^{\Omega(\sqrt{n})}$.

References

[1] Scott Aaronson, Shalev Ben-David, Robin Kothari, Shravas Rao, and Avishay Tal. Degree vs. approximate degree and quantum implications of huang's sensitivity theorem. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 1330–1342, 2021.

[2] Scott Aaronson and Yaoyun Shi. Quantum lower bounds for the collision and the element distinctness problems. *J. ACM*, 51(4):595–605, 2004.

[3] Anil Ada, Omar Fawzi, and Hamed Hatami. Spectral norm of symmetric functions. In Anupam Gupta, Klaus Jansen, José D. P. Rolim, and Rocco A. Servedio, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques - 15th International Workshop, APPROX 2012, and 16th International Workshop, RANDOM 2012, Cambridge, MA, USA, August 15-17, 2012. Proceedings*, volume 7408 of *Lecture Notes in Computer Science*, pages 338–349. Springer, 2012.

[4] Josh Alman, Arkadev Chattopadhyay, and Ryan Williams. Sparsity lower bounds for probabilistic polynomials. In Raghu Meka, editor, *16th Innovations in Theoretical Computer Science Conference, ITCS 2025, January 7-10, 2025, Columbia University, New York, NY, USA*, volume 325 of *LIPIcs*, pages 3:1–3:25. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2025.

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

[6] Paul Beame and Dang-Trinh Huynh-Ngoc. Multiparty communication complexity and threshold circuit size of acˆ0. In *50th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2009, October 25-27, 2009, Atlanta, Georgia, USA*, pages 53–62. IEEE Computer Society, 2009.

[7] Stephen J. Bellantoni, Toniann Pitassi, and Alasdair Urquhart. Approximation and small-depth frege proofs. *SIAM J. Comput.*, 21(6):1161–1179, 1992.

[8] Eli Ben-Sasson and Avi Wigderson. Short proofs are narrow - resolution made simple. *J. ACM*, 48(2):149–169, 2001.

[9] Gal Beniamini and Noam Nisan. Bipartite perfect matching as a real polynomial. In *53rd ACM Symposium on Theory of Computing (STOC)*, pages 1118–1131. ACM, 2021.

[10] Andrej Bogdanov, Yuval Ishai, Emanuele Viola, and Christopher Williamson. Bounded indistinguishability and the complexity of recovering secrets. In *International Cryptology Conference*, volume LNCS 9816, pages 593–618. Springer, 2016.

[11] Andrej Bogdanov, Nikhil S. Mande, Justin Thaler, and Christopher Williamson. Approximate degree, secret sharing and concentration phenomena. In *Approximation, Randomization and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM*, volume LIPIcs, 145, pages 71:1–71:21. Schloss-Dagstuhl, 2019.

[12] Jehoshua Bruck and Roman Smolensky. Polyomial threshold functions, AC^0 functions, and spectral norms (extended abstract). In *31st Annual Symposium on Foundations of Computer Science (FOCS)*, volume II, pages 632–641. IEEE, 1990.

[13] Jehoshua Bruck and Roman Smolensky. Polynomial threshold functions, acˆ0 functions, and spectral norms. *SIAM Journal on Computing*, 21(1):33–42, 1992.

[14] Harry Buhrman and Ronald de Wolf. Communication complexity lower bounds by polynomials. In *Proceedings of the 16th Annual IEEE Conference on Computational Complexity, Chicago, Illinois, USA, June 18-21, 2001*, pages 120–130. IEEE Computer Society, 2001.

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

[16] Mark Bun, Robin Kothari, and Justin Thaler. The polynomial method strikes back: Tight quantum query bounds via dual polynomials. *Theory Comput.*, 16:1–71, 2020.

[17] Mark Bun and Justin Thaler. Approximate degree in classical and quantum computing. *Found. Trends Theor. Comput. Sci.*, 15(3-4):229–423, 2022.

[18] Karthekeyan Chandrasekaran, Justin Thaler, Jonathan Ullman, and Andrew Wan. Faster private release of marginals on small databases. In *Proceedings of the 5th conference on Innovations in theoretical computer science*, pages 387–402, 2014.

[19] Arkadev Chattopadhyay and Anil Ada. Multiparty communication complexity of disjointness. *Electron. Colloquium Comput. Complex.*, TR08-002, 2008.

[20] Arkadev Chattopadhyay, Yogesh Dahiya, Nikhil S Mande, Jaikumar Radhakrishnan, and Swagato Sanyal. Randomized versus deterministic decision tree size. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, pages 867–880, 2023.

[21] Arkadev Chattopadhyay and Nikhil S. Mande. Dual polynomials and communication complexity of XOR functions. *Electron. Colloquium Comput. Complex.*, TR17-062, 2017.

[22] Arkadev Chattopadhyay and Nikhil S. Mande. A lifting theorem with applications to symmetric functions. In Satya V. Lokam and R. Ramanujam, editors, *37th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2017, December 11-15, 2017, Kanpur, India*, volume 93 of *LIPIcs*, pages 23:1–23:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2017.

[23] Arkadev Chattopadhyay, Nikhil S. Mande, and Suhail Sherif. The log-approximate-rank conjecture is false. *J.ACM*, 67(4):1–28, 2020.

[24] Tsun-Ming Cheung, Hamed Hatami, Kaave Hosseini, Aleksandar Nikolov, Toniann Pitassi, and Morgan Shirley. A lower bound on the trace norm of boolean matrices and its applications. In *16th Innovations in Theoretical Computer Science Conference (ITCS 2025)*, pages 37–1. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2025.

[25] Ilias Diakonikolas, Parikshit Gopalan, Ragesh Jaiswal, Rocco A Servedio, and Emanuele Viola. Bounded independence fools halfspaces. *SIAM Journal on Computing*, 39(8):3441–3462, 2010.

[26] Andrzej Ehrenfeucht and David Haussler. Learning decision trees from random examples. *Information and Computation*, 82(3):231 – 246, 1989. Earlier version in COLT'88.

[27] Merrick L. Furst, James B. Saxe, and Michael Sipser. Parity, circuits, and the polynomial-time hierarchy. *Math. Syst. Theory*, 17(1):13–27, 1984.

[28] Ben Green and Tom Sanders. Boolean functions with small spectral norm. *Geometric and Functional Analysis*, 18:144 – 162, 2008.

[29] Vince Grolmusz. On the power of circuits with gates of low l1 norms. *Theoretical computer science*, 188(1-2):117–128, 1997.

[30] Johan Håstad. Almost optimal lower bounds for small depth circuits. In Juris Hartmanis, editor, *Proceedings of the 18th Annual ACM Symposium on Theory of Computing, May 28-30, 1986, Berkeley, California, USA*, pages 6–20. ACM, 1986.

[31] Johan Håstad. On small-depth frege proofs for PHP. In *64th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2023, Santa Cruz, CA, USA, November 6-9, 2023*, pages 37–49. IEEE, 2023.

[32] Johan Håstad, Benjamin Rossman, Rocco A. Servedio, and Li-Yang Tan. An average-case depth hierarchy theorem for boolean circuits. *J. ACM*, 64(5):35:1–35:27, 2017.

[33] Hao Huang. Induced subgraphs of hypercubes and a proof of the sensitivity conjecture. *CoRR*, abs/1907.00847, 2019.

[34] Russell Impagliazzo, Pavel Pudlák, and Jirí Sgall. Lower bounds for the polynomial calculus and the gröbner basis algorithm. *Comput. Complex.*, 8(2):127–144, 1999.

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

[36] Adam Tauman Kalai, Adam R Klivans, Yishay Mansour, and Rocco A Servedio. Agnostically learning halfspaces. *SIAM Journal on Computing*, 37(6):1777–1805, 2008.

[37] Hartmut Klauck. Lower bounds for quantum communication complexity. *SIAM J. Comput.*, 37(1):20–46, 2007.

[38] Adam R Klivans and Rocco A Servedio. Learning dnf in time 2õ (n1/3). *Journal of Computer and System Sciences*, 68(2):303–318, 2004.

[39] Alexander Knop, Shachar Lovett, Sam McGuire, and Weiqiang Yuan. Guest column: Models of computation between decision trees and communication. *ACM SIGACT News*, 52(2):46–70, 2021.

[40] Alexander Knop, Shachar Lovett, Sam McGuire, and Weiqiang Yuan. Log-rank and lifting for and-functions. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 197–208, 2021.

[41] Jan Krajícek, Pavel Pudlák, and Alan R. Woods. An exponenetioal lower bound to the size of bounded depth frege proofs of the pigeonhole principle. *Random Struct. Algorithms*, 7(1):15–40, 1995.

[42] Troy Lee and Adi Shraibman. Disjointness is hard in the multiparty number-on-the-forehead model. *Comput. Complex.*, 18(2):309–336, 2009.

[43] Noam Nisan. The demand query model for bipartite matching. In *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 592–599. ACM, 2021.

[44] Noam Nisan and Mario Szegedy. On the degree of boolean functions as real polynomials. *Computational complexity*, 4:301–313, 1994.

[45] Toniann Pitassi, Paul Beame, and Russell Impagliazzo. Exponential lower bounds for the pigeonhole principle. *Comput. Complex.*, 3:97–140, 1993.

[46] Alexander Razborov. Quantum communication complexity of symmetric predicates. *Izvestiya:Mathematics*, 67(1):145–159, 2003.

[47] Alexander A. Sherstov. On quantum-classical equivalence for composed communication problems. *Quantum Inf. Comput.*, 10(5&6):435–455, 2010.

[48] Alexander A. Sherstov. The pattern matrix method. *SIAM J. Comput.*, 40(6):1969–2000, 2011.

[49] Alexander A. Sherstov. Communication lower bounds using directional derivatives. *J. ACM*, 61(6):34:1–34:71, 2014.

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

[51] Justin Thaler, Jonathan Ullman, and Salil Vadhan. Faster algorithms for privately releasing marginals. In *International Colloquium on Automata, Languages, and Programming*, pages 810–821. Springer, 2012.

[52] Shengyu Zhang. Efficient quantum protocols for xor functions. In *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*, pages 1878–1885. SIAM, 2014.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il