The Approximate Degree of DNF and CNF Formulas

Alexander A. Sherstov
University of California, Los Angeles
Los Angeles, California, USA
sherstov@cs.ucla.edu

ABSTRACT

The *approximate degree* of a Boolean function $f\colon \{0,1\}^n \to \{0,1\}$ is the minimum degree of a real polynomial p that approximates f pointwise: $|f(x) - p(x)| \leqslant 1/3$ for all $x \in \{0,1\}^n$. For any $\delta > 0$, we construct DNF and CNF formulas of polynomial size with approximate degree $\Omega(n^{1-\delta})$, essentially matching the trivial upper bound of n. This fully resolves the approximate degree of constant-depth circuits (AC^0), a question that has seen extensive research over the past 10 years. Prior to our work, an $\Omega(n^{1-\delta})$ lower bound was known only for AC^0 circuits of depth that grows with $1/\delta$ (Bun and Thaler, FOCS 2017). Furthermore, the DNF and CNF formulas that we construct are the simplest possible in that they have *constant* width.

Our result gives the first near-linear lower bounds on the bounded-error communication complexity of polynomial-size DNF and CNF formulas in the challenging k-party number-on-the-forehead model and two-party quantum model: $\Omega(n/4^k k^2)^{1-\delta}$ and $\Omega(n^{1-\delta})$, respectively, where $\delta > 0$ is any constant. Our lower bounds are essentially optimal. Analogous to above, such lower bounds were previously known only for AC^0 circuits of depth that grows with $1/\delta$.

CCS CONCEPTS

• **Theory of computation → Algebraic complexity theory**; **Communication complexity**; **Circuit complexity**; **Quantum complexity theory**.

KEYWORDS

DNF formulas, CNF formulas, AC^0, constant-depth circuits, approximate degree, polynomial approximation, communication complexity, number-on-the-forehead model, quantum communication

ACM Reference Format:
Alexander A. Sherstov. 2022. The Approximate Degree of DNF and CNF Formulas. In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing (STOC '22), June 20–24, 2022, Rome, Italy.* ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3519935.3520000

1 INTRODUCTION

Representations of Boolean functions by real polynomials play a central role in theoretical computer science. Our focus in this paper is on *approximate degree*, a particularly natural and useful complexity measure. Formally, the ε-approximate degree of a Boolean function $f\colon \{0,1\}^n \to \{0,1\}$ is denoted $\deg_\varepsilon(f)$ and defined as the minimum degree of a real polynomial p that approximates f within ε pointwise: $|f(x) - p(x)| \leqslant \varepsilon$ for all $x \in \{0,1\}^n$. The standard choice of the error parameter is $\varepsilon = 1/3$, which is a largely arbitrary setting that can be replaced by any other constant in $(0, 1/2)$ without affecting the approximate degree by more than a multiplicative constant. Since every function $\{0,1\}^n \to \{0,1\}$ can be computed with zero error by a polynomial of degree at most n, we see that the ε-approximate degree is always an integer between 0 and n.

The notion of approximate degree originated three decades ago in the pioneering work of Nisan and Szegedy [38] and has since proved to be a powerful tool in theoretical computer science. Upper bounds on approximate degree have algorithmic applications, whereas lower bounds are a staple in complexity theory. On the algorithmic side, approximate degree underlies many of the strongest results obtained to date in computational learning [5, 26, 29, 30, 39, 59], differentially private data release [23, 60], and algorithm design in general [25, 35, 45]. In complexity theory, the notion of approximate degree has produced breakthroughs in quantum query complexity [1, 2, 4, 10, 14, 17, 28], communication complexity [11, 15, 16, 24, 34, 42–44, 46, 47, 51, 52], and circuit complexity [6, 11, 13, 31, 32, 40, 46, 58].

Approximate degree has been particularly prominent in the study of AC^0, the class of polynomial-size constant-depth circuits with gates $\vee, \wedge, \neg$ of unbounded fan-in. The simplest functions in AC^0 are conjunctions and disjunctions, which have depth 1, followed by polynomial-size CNF and DNF formulas, which have depth 2, followed in turn by higher-depth circuits. Lower bounds on the approximate degree of AC^0 functions have been used to settle the quantum query complexity of Grover search [10], element distinctness [2], and a host of other problems [17]; resolve the communication complexity of set disjointness in the two-party quantum model [42, 47] and number-on-the-forehead multiparty model [11, 24, 34, 44, 46, 47, 51, 52]; separate the communication complexity classes PP and UPP [15, 46]; and separate the polynomial hierarchy in communication complexity from the communication class UPP [43]. Despite this array of applications and decades of study, our understanding of the approximate degree of AC^0 has remained surprisingly fragmented and incomplete. In this paper, we set out to resolve this question in full.

In more detail, previous work on the approximate degree of AC^0 started with the seminal 1994 paper of Nisan and Szegedy [38], who proved that the OR function on n bits has approximate degree $\Theta(\sqrt{n})$. This was the best lower bound for an AC^0 function until Aaronson and Shi's celebrated lower bound of $\Omega(n^{2/3})$ for the element distinctness problem [2]. In a beautiful result from 2017, Bun and Thaler [20] showed that AC^0 contains functions in n variables with approximate degree $\Omega(n^{1-\delta})$, where the constant $\delta > 0$ can be made arbitrarily small at the expense of increasing the

STOC '22, June 20–24, 2022, Rome, Italy

ACM ISBN 978-1-4503-9264-8/22/06.
https://doi.org/10.1145/3519935.3520000

depth of the circuit. In follow-up work, Bun and Thaler [21] proved that the $\Omega(n^{1-\delta})$ lower bound for AC^0 holds even for approximation to error exponentially close to $1/2$. A stronger yet result was obtained by Sherstov and Wu [57], who showed that AC^0 has essentially the maximum possible *threshold degree* (defined as the limit of ε-approximate degree as $\varepsilon \nearrow 1/2$) and *sign-rank* (a generalization of threshold degree to arbitrary bases rather than just the basis of monomials). Quantitatively, the authors of [57] proved a lower bound of $\Omega(n^{1-\delta})$ for threshold degree and $\exp(\Omega(n^{1-\delta}))$ for sign-rank, essentially matching the trivial upper bounds. As before, $\delta > 0$ can be made arbitrarily small at the expense of increasing the circuit depth. In particular, AC^0 requires a polynomial of degree $\Omega(n^{1-\delta})$ even for approximation to error doubly (triply, quadruply, quintuply…) exponentially close to $1/2$.

The lower bounds of [20, 21, 57] show that AC^0 functions have essentially the maximum possible complexity—but only if one is willing to look at circuits of *arbitrarily large* constant depth. What happens at *small* depths has been a wide open problem, with no techniques to address it. Bun and Thaler [20] observe that their AC^0 circuit with approximate degree $\Omega(n^{1-\delta})$ can be flattened to produce a DNF formula of size $\exp(\log^{O(\log(1/\delta))} n)$, but this is superpolynomial and thus no longer in AC^0. The only progress of which we are aware is an $\Omega(n^{3/4-\delta})$ lower bound obtained for polynomial-size DNF formulas in [17, 36]. This leaves a polynomial gap in the approximate degree for small depth versus arbitrary constant depth. Our main contribution is to definitively resolve the approximate degree of AC^0 by constructing, for any constant $\delta > 0$, a polynomial-size DNF formula with approximate degree $\Omega(n^{1-\delta})$. We now describe our main result and applications in technical detail.

1.1 Approximate Degree of DNF and CNF Formulas

Recall that a *literal* is a Boolean variable $x_1, x_2, \ldots, x_n$ or its negation $\overline{x_1}, \overline{x_2}, \ldots, \overline{x_n}$. A conjunction of literals is called a *term*, and a disjunction of literals is called a *clause*. The *width* of a term or clause is the number of literals that it contains. A *DNF formula* is a disjunction of terms, and analogously a *CNF formula* is a conjunction of clauses. The *width* of a DNF or CNF formula is the maximum width of a term or clause in it. One often refers to DNF and CNF formulas of width k as k-DNF and k-CNF formulas, respectively. The *size* of a DNF or CNF formula is the total number of terms or clauses that it contains. Thus, AC^0 circuits of depth 1 correspond precisely to clauses and terms, whereas AC^0 circuits of depth 2 correspond precisely to polynomial-size DNF and CNF formulas. Our main result on approximate degree is as follows.

Theorem 1.1 (Main result). *Let $\delta > 0$ be any constant. Then for each $n \geqslant 1$, there is an (explicitly given) function $f\colon \{0,1\}^n \to \{0,1\}$ that has approximate degree*

$$\deg_{1/3}(f) = \Omega(n^{1-\delta})$$

and is computable by a DNF formula of size $n^{O(1)}$ and width $O(1)$.

Theorem 1.1 almost matches the trivial upper bound of n on the approximate degree of any function. Thus, the theorem shows that AC^0 circuits of depth 2 already achieve essentially the maximum possible approximate degree. This depth cannot be reduced further because AC^0 circuits of depth 1 have approximate degree $O(\sqrt{n})$. Finally, the DNF formulas constructed in Theorem 1.1 are the simplest possible in that they have *constant* width.

Recall that previously, a lower bound of $\Omega(n^{1-\delta})$ for AC^0 was known only for circuits of large constant depth that grows with $1/\delta$. The lack of progress on small-depth AC^0 prior to this paper had experts seriously entertaining [21] the possibility that AC^0 circuits of any given depth d have approximate degree $O(n^{1-\delta_d})$, for some constant $\delta_d = \delta_d(d) > 0$. Such an upper bound would have far-reaching consequences in computational learning and circuit complexity. Theorem 1.1 rules it out.

We obtain the following strengthening of our main result, in which the allowed approximation error is relaxed from $1/3$ to an optimal $1/2 - 1/n^{\Theta(1)}$.

Theorem 1.2 (Main result for large error). *Let $\delta > 0$ and $C \geqslant 1$ be any constants. Then for each $n \geqslant 1$, there is an (explicitly given) function $f\colon \{0,1\}^n \to \{0,1\}$ that has approximate degree*

$$\deg_{\frac{1}{2}-\frac{1}{n^C}}(f) = \Omega(n^{1-\delta})$$

and is computable by a DNF formula of size $n^{O(1)}$ and width $O(1)$.

To rephrase Theorem 1.2, polynomial-size DNF formulas require degree $\Omega(n^{1-\delta})$ for approximation not only to constant error but even to error $\frac{1}{2} - \frac{1}{n^C}$, where $C \geqslant 1$ is an arbitrarily large constant. Thus, Theorem 1.2 assumes a weaker hypothesis but produces the same conclusion as Theorem 1.1. The error parameter in Theorem 1.2 cannot be relaxed further to $\frac{1}{2} - \frac{1}{n^{\omega(1)}}$ because any DNF formula with m terms can be approximated to error $\frac{1}{2} - \Omega(\frac{1}{m})$ by a polynomial of degree $O(\sqrt{n \log m})$.

Negating a function has no effect on the approximate degree. Indeed, if f is approximated to error ε by a polynomial p, then the negated function $\neg f = 1 - f$ is approximated to the same error ε by the polynomial $1 - p$. With this observation, Theorems 1.1 and 1.2 carry over to CNF formulas:

Corollary 1.3. *Let $\delta > 0$ and $C \geqslant 1$ be any constants. Then for each $n \geqslant 1$, there is an (explicitly given) function $g\colon \{0,1\}^n \to \{0,1\}$ that has approximate degree*

$$\deg_{\frac{1}{2}-\frac{1}{n^C}}(g) = \Omega(n^{1-\delta})$$

and is computable by a CNF formula of size $n^{O(1)}$ and width $O(1)$.

We now turn to applications of our work to basic questions in communication complexity.

1.2 Multiparty Communication Complexity

We adopt the *number-on-the-forehead* model of Chandra, Furst, and Lipton [22], which is the most powerful formalism of multiparty communication. The model features k communicating players and a Boolean function $F\colon X_1 \times X_2 \times \cdots \times X_k \to \{0,1\}$ with k arguments. An input $(x_1, x_2, \ldots, x_k)$ is distributed among the k players by giving the i-th player the arguments $x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_k$ but not x_i. This arrangement can be visualized as having the k players seated in a circle with x_i written on the i-th player's forehead, whence the name of the model. Number-on-the-forehead is the

canonical model in the area because any other way of assigning arguments to players results in a less powerful model—provided of course that one does not assign all the arguments to some player, in which case there is never a need to communicate.

The players communicate according to a protocol agreed upon in advance. The communication occurs in the form of broadcasts, with a message sent by any given player instantly reaching everyone else. The players' objective is to compute F on any given input with minimal communication. To this end, the players have access to an unbounded supply of shared random bits which they can use in deciding what message to send at any given point in the protocol. The *cost* of a protocol is the total bit length of all the messages broadcast in a worst-case execution. The *ε-error randomized communication complexity* $R_\varepsilon(F)$ of a given function F is the least cost of a protocol that computes F with probability of error at most ε on every input. As with approximate degree, the standard setting of the error parameter is $\varepsilon = 1/3$.

The number-on-the-forehead communication complexity of constant depth circuits is a challenging question that has been the focus of extensive research, e.g., [11, 12, 20, 24, 34, 44, 51, 52]. In contrast to the two-party model, where a lower bound of $\Omega(\sqrt{n})$ for AC^0 circuits is straightforward to prove from first principles [7], the first $n^{\Omega(1)}$ multiparty lower bound [52] for AC^0 was obtained only in 2012. The strongest known multiparty lower bounds for AC^0 are obtained using the *pattern matrix method* of [51], which transforms approximate degree lower bounds in a black-box manner into communication lower bounds. In the most recent application of this method, Bun and Thaler [20] gave a k-party communication problem $F\colon (\{0,1\}^n)^k \to \{0,1\}$ in AC^0 with communication complexity $\Omega(n/4^k k^2)^{1-\delta}$, where the constant $\delta > 0$ can be taken arbitrarily small at the expense of increasing the depth of the AC^0 circuit. This shows that AC^0 has essentially the maximum possible multiparty communication complexity—as long as one is willing to use circuits of arbitrarily large constant depth. For circuits of small depth, on the other hand, the lower bounds are polynomially weaker: the best lower bound of which we are aware is $\Omega(n/4^k k^2)^{3/4-\delta}$ for the k-party communication complexity of polynomial-size DNF formulas, which results from applying the pattern matrix method to the approximate degree lower bounds in [17, 36]. What is more, if one changes the question slightly by allowing communication protocols with error $\frac{1}{2} - \frac{1}{n^C}$ for a large constant $C \geqslant 1$, then no lower bounds at all were known on the multiparty communication complexity of polynomial-size DNF formulas. This fragmented state of the art closely parallels that for approximate degree prior to our work.

We resolve the multiparty communication complexity of AC^0 in detail in the following theorem.

Theorem 1.4. *Fix any constants $\delta \in (0,1]$ and $C \geqslant 1$. Then for all integers $n, k \geqslant 2$, there is an (explicitly given) k-party communication problem $F_{n,k}\colon (\{0,1\}^n)^k \to \{0,1\}$ with*

$$R_{1/3}(F_{n,k}) \geqslant \left(\frac{n}{c' 4^k k^2}\right)^{1-\delta},$$

$$R_{\frac{1}{2}-\frac{1}{n^C}}(F_{n,k}) \geqslant \frac{n^{1-\delta}}{c' 4^k},$$

where $c' \geqslant 1$ is a constant independent of n and k. Moreover, each $F_{n,k}$ is computable by a DNF formula of size $n^{c'}$ and width $c'k$.

Theorem 1.4 essentially represents the state of the art for multiparty communication lower bounds. Indeed, the best communication lower bound to date for any explicit function $F\colon (\{0,1\}^n)^k \to \{0,1\}$, whether or not F is computable by an AC^0 circuit, is $\Omega(n/2^k)$ [8]. Theorem 1.4 comes close to matching the trivial upper bound of $n+1$ for any communication problem, thereby showing that AC^0 circuits of depth 2 achieve nearly maximum possible communication complexity. Moreover, our result holds not only for bounded-error communication but also for communication with error $\frac{1}{2} - \frac{1}{n^C}$ for any $C \geqslant 1$. The error parameter in Theorem 1.4 is optimal and cannot be further increased to $\frac{1}{2} - \frac{1}{n^{\omega(1)}}$; indeed, it is straightforward to see that any DNF formula with m terms has a communication protocol with error $\frac{1}{2} - \Omega(\frac{1}{m})$ and cost 2 bits. Theorem 1.4 is also optimal with respect to circuit depth because the multiparty communication complexity of AC^0 circuits of depth 1 is at most 2 bits.

Since randomized communication complexity is invariant under function negation, Theorem 1.4 remains valid with the word "DNF" replaced with "CNF."

1.3 Quantum Communication Complexity

We adopt the standard model of quantum communication, where two parties exchange quantum messages according to an agreed-upon protocol in order to solve a two-party communication problem $F\colon X \times Y \to \{0,1\}$. As usual, an input $(x,y) \in X \times Y$ is split between the parties, with one party knowing only x and the other party knowing only y. We allow arbitrary prior entanglement at the start of the communication. A measurement at the end of the protocol produces a single-bit answer, which is interpreted as the protocol output. An *ε-error protocol for F* is required to output, on every input $(x,y) \in X \times Y$, the correct value $F(x,y)$ with probability at least $1-\varepsilon$. The *cost* of a quantum protocol is the total number of quantum bits exchanged in the worst case on any input. The *ε-error quantum communication complexity of F*, denoted $Q^*_\varepsilon(F)$, is the least cost of an ε-error quantum protocol for F. The asterisk in $Q^*_\varepsilon(F)$ indicates that the parties share arbitrary prior entanglement. The standard setting of the error parameter is $\varepsilon = 1/3$, which is as usual without loss of generality. For a detailed formal description of the quantum model, we refer the reader to [42, 47].

Proving lower bounds for bounded-error quantum communication is significantly more challenging than for randomized communication. An illustrative example is the set disjointness problem on n bits. Babai, Frankl, and Simon [7] obtained an $\Omega(\sqrt{n})$ randomized communication lower bound using a short and elementary proof, which was later improved to a tight $\Omega(n)$ in [9, 27, 41]. This is in stark contrast with the quantum model, where the best lower bound for set disjointness was for a long time a trivial $\Omega(\log n)$ until a tight $\Omega(\sqrt{n})$ was proved by Razborov [42].

A completely different proof of the $\Omega(\sqrt{n})$ lower bound for set disjointness was given in [47] by introducing the pattern matrix method. Since then, the pattern matrix method has produced the strongest known quantum lower bounds for AC^0. Of these, the best lower bound prior to our work was $\Omega(n^{1-\delta})$ due to Bun and Thaler [20], where the constant $\delta > 0$ can be taken arbitrarily small at the expense of circuit depth. In the following theorem, we resolve the quantum communication complexity of AC^0 essentially

in full by proving that polynomial-size DNF formulas achieve near-maximum communication complexity.

Theorem 1.5. *Let $\delta > 0$ and $C \geqslant 1$ be any constants. Then for each $n \geqslant 1$, there is an (explicitly given) two-party communication problem $F\colon \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ that has quantum communication complexity*

$$Q^*_{\frac{1}{2}-\frac{1}{n^C}}(F) \geqslant \Omega(n^{1-\delta})$$

and is representable by a DNF formula of size $n^{O(1)}$ and width $O(1)$.

This theorem remains valid for CNF formulas since quantum communication complexity is invariant under function negation. As in all of our results, Theorem 1.5 essentially matches the trivial upper bound, showing that AC^0 circuits of depth 2 achieve nearly maximum possible complexity. Again analogous to our other results, Theorem 1.5 holds not only for bounded-error communication but also for communication with error $\frac{1}{2} - \frac{1}{n^C}$ for any $C \geqslant 1$. The error parameter in Theorem 1.5 is optimal and cannot be further increased to $\frac{1}{2} - \frac{1}{n^{\omega(1)}}$: as remarked above, any DNF formula with m terms has a classical communication protocol with error $\frac{1}{2} - \Omega(\frac{1}{m})$ and cost 2 bits. Lastly, Theorem 1.5 is optimal with respect to circuit depth because AC^0 circuits of depth 1 have communication complexity at most 2 bits even in the classical deterministic model.

In our overview so far, we have separately considered the classical multiparty model and the quantum two-party model. By combining the features of these models, one arrives at the k-party number-on-the-forehead model with quantum players. Our results readily generalize to this setting. Specifically, for any constants $\delta > 0$ and $C \geqslant 1$, we give an explicit DNF formula $F_{n,k}\colon (\{0,1\}^n)^k \to \{0,1\}$ of size $n^{O(1)}$ and width $O(k)$ such that computing $F_{n,k}$ in the k-party quantum number-on-the-forehead model with error $\frac{1}{2} - \frac{1}{n^C}$ requires $\Omega(n^{1-\delta}/4^k k)$ quantum bits. For more details, see the full version [56].

1.4 Previous Approaches

In the remainder of the introduction, we sketch our proof of Theorems 1.1 and 1.2. To properly set the stage for our work, we start by reviewing relevant background and presenting previous approaches and their limitations. The notation that we adopt below is standard, and we defer its formal review to Section 2.

Dual view of approximation. Let $f\colon X \to \{0,1\}$ be a Boolean function of interest, where X is an arbitrary finite subset of Euclidean space. The approximate degree of f is defined analogously to functions on the Boolean hypercube: $\deg_\varepsilon(f)$ is the minimum degree of a real polynomial p such that $|f(x) - p(x)| \leqslant \varepsilon$ for every $x \in X$. A valuable tool in the analysis of approximate degree is linear programming duality, which gives a powerful *dual* view of approximation [47]. This dual characterization states that $\deg_\varepsilon(f) \geqslant d$ if and only if there is a function $\phi\colon X \to \mathbb{R}$ with the following two properties: $\langle \phi, f\rangle > \varepsilon \|\phi\|_1$; and $\langle \phi, p\rangle = 0$ for every polynomial p of degree less than d. Rephrasing, ϕ must have large inner product with f but zero inner product with every low-degree polynomial. Equivalently, one may think of ϕ as being heavily correlated with f but completely uncorrelated with any polynomial of degree less than d. The function ϕ is variously referred to in the literature as a "dual object," "dual polynomial," or "witness" for f. The dual characterization makes it possible—in principle if not in practice—to prove any approximate degree lower bound by constructing the corresponding witness ϕ. This good news comes with a sobering caveat: for all but the simplest functions, the construction of ϕ is very demanding, and linear programming duality gives no guidance whatsoever in this regard.

Componentwise composition. The construction of a dual object is more approachable for composed functions since one can hope to break them up into constituent parts, construct a dual object for each, and recombine these results. Formally, define the *componentwise composition* of functions $f\colon \{0,1\}^n \to \{0,1\}$ and $g\colon X \to \{0,1\}$ as the Boolean function $f \circ g\colon X^n \to \{0,1\}$ given by $(f \circ g)(x_1, \ldots, x_n) = f(g(x_1), \ldots, g(x_n))$. To construct a dual object for $f \circ g$, one starts by obtaining dual objects ϕ and ψ for the constituent functions f and g, respectively, either by direct construction or by appeal to linear programming duality. They are then combined to yield a dual object Φ for the composed function, using *dual componentwise composition* [33, 49]:

$$\Phi(x_1, x_2, \ldots, x_n) = \phi(\mathbf{I}[\psi(x_1) > 0], \ldots, \mathbf{I}[\psi(x_n) > 0]) \prod_{i=1}^{n} |\psi(x_i)|. \quad (1)$$

This composed dual object typically requires additional work to ensure strong enough correlation with the composed function $f \circ g$. Among the generic tools available to assist in this process is a "corrector" object ζ due to Razborov and Sherstov [43], with the following four properties: (i) ζ is orthogonal to low-degree polynomials; (ii) ζ takes on 1 at a prescribed point of the hypercube; (iii) ζ is bounded on inputs of low Hamming weight; and (iv) ζ vanishes on all other points of the hypercube. Using ζ, suitably shifted and scaled, one can surgically correct the behavior of a given dual object Φ on a substantial fraction of inputs, thus modifying the metric properties of Φ without affecting its orthogonality to low-degree polynomials. This technique has played an important role in recent work, e.g., [17, 20, 21, 57].

Componentwise composition by itself does not allow one to construct hard-to-approximate functions from easy ones. To see why, consider arbitrary functions $f\colon \{0,1\}^{n_1} \to \{0,1\}$ and $g\colon \{0,1\}^{n_2} \to \{0,1\}$ with approximate degree at most n_1^α and n_2^α, respectively, for some $0 < \alpha < 1$. It is well-known [50] that the composed function $f \circ g$ on $n_1 n_2$ variables has approximate degree $O(n_1^\alpha n_2^\alpha) = O(n_1 n_2)^\alpha$. This means that relative to the new number of variables, the composed function $f \circ g$ is asymptotically no harder to approximate than the constituent functions f and g. In particular, one cannot use componentwise composition to transform functions on n bits with 1/3-approximate degree at most n^α into functions on $N \geqslant n$ bits with 1/3-approximate degree $\omega(N^\alpha)$.

Previous best bound for AC^0. In the previous best result on the 1/3-approximate degree of AC^0, Bun and Thaler [20] approached the componentwise composition $f \circ g$ in an ingenious way to amplify the approximate degree for a careful choice of g. Let $f\colon \{0,1\}^n \to \{0,1\}$ be given, with 1/3-approximate degree n^α for some $0 \leqslant \alpha < 1$. Bun and Thaler consider the componentwise composition $F = f \circ (\mathrm{AND}_{\Theta(\log m)} \circ \mathrm{OR}_m)$, for an appropriate

parameter $m = \mathrm{poly}(n)$. It was shown in earlier work [19, 49] that dual componentwise composition witnesses the lower bound $\deg_{1/3}(F) = \Omega(\deg_{1/3}(\mathrm{OR}_m)\deg_{1/3}(f)) = \Omega(\sqrt{m}\deg_{1/3}(f))$. Bun and Thaler make the crucial observation that the dual object for OR_m has most of its ℓ_1 mass on inputs of Hamming weight $O(1)$, which in view of (1) implies that the dual object for F places most of its ℓ_1 mass on inputs of Hamming weight $\tilde{O}(n)$. The authors of [20] then use the Razborov–Sherstov corrector object to transfer the small amount of ℓ_1 mass that the dual object for F places on inputs of high Hamming weight, to inputs of low Hamming weight. The resulting dual object is supported entirely on inputs of low Hamming weight and therefore witnesses a lower bound on the approximate degree of the *restriction* F' of F to inputs of low Hamming weight.

The restriction F' takes as input $N := \Theta(nm\log m)$ variables but is defined only when its input string has Hamming weight $\tilde{O}(n)$. This makes it possible to represent the input to F' more economically, by specifying the locations of the $\tilde{O}(n)$ nonzero bits inside the array of N variables. Since each such location can be specified using $\lceil\log N\rceil$ bits, the entire input to F' can be specified using $\lceil\log N\rceil\cdot\tilde{O}(n) = \tilde{O}(n)$ bits. This yields a function F'' on $\tilde{O}(n)$ variables. A careful calculation shows that this "input compression" does not hurt the approximate degree. Thus, the approximate degree of F'' is at least the approximate degree of F', which as discussed above is $\Omega(\sqrt{m}\deg_{1/3}(f))$. With m set appropriately, the approximate degree of F'' is polynomially larger than that of f. The passage from f to F'' is the desired hardness amplification for approximate degree. In summary, Bun and Thaler's hardness amplification involves three steps: (i) start with standard componentwise composition; (ii) restrict the input to strings of low Hamming weight; (iii) compress the input to a near-linear number of variables. To obtain an $\Omega(n^{1-\delta})$ lower bound on the approximate degree of AC^0, the authors of [20] start with a trivial circuit and iteratively apply the hardness amplification step a constant number of times, until approximate degree $\Omega(n^{1-\delta})$ is reached.

Limitations of previous approaches to AC^0. Bun and Thaler's hardness amplification for approximate degree rests on two pillars. The first is componentwise composition, whereby the given function $f\colon\{0,1\}^n\to\{0,1\}$ is composed componentwise with n independent copies of the gadget $\mathrm{AND}_{\Theta(\log m)}\circ\mathrm{OR}_m$. In this gadget, the $\mathrm{AND}_{\Theta(\log m)}$ gate is necessary to prevent accumulation of error and to ensure the correlation property of the dual polynomial. The resulting composed function $F = f\circ(\mathrm{AND}_{\Theta(\log m)}\circ\mathrm{OR}_m)$ is defined on $N = \Theta(nm\log m)$ variables. The standard dual object for $F = f\circ(\mathrm{AND}_{\Theta(\log m)}\circ\mathrm{OR}_m)$ places nearly all of its ℓ_1 mass on inputs of Hamming weight $\tilde{O}(n)\ll N$ to start with, and the ℓ_1 mass can further be redistributed to make sure the dual object is supported entirely on inputs of Hamming weight $\tilde{O}(n)$. This brings us to the second pillar of [20], input compression. Here, the length-N input to F is represented compactly as an array of $\tilde{O}(n)$ strings of length $\lceil\log N\rceil$ each, to indicate the locations of the $\tilde{O}(n)$ ones among the N input bits. The circuitry to implement these two pillars is expensive, requiring in both cases a polynomial-size DNF formula of width $\Theta(\log n+\log m)$. As a result, even a *single* iteration of Bun–Thaler hardness amplification cannot be implemented as a polynomial-size DNF or CNF formula.

To prove an $\Omega(n^{1-\delta})$ approximate degree lower bound for small $\delta>0$ in the framework of [20], one needs a number of iterations that grows with $1/\delta$. Thus, the overall circuit produced in [20] has a large constant number of alternating layers of AND and OR gates of logarithmic and polynomial fan-in, respectively, and in particular cannot be flattened into a polynomial-size DNF or CNF formula. Proving Theorem 1.1 within this framework would require reducing the fan-in of the AND gates from $\Theta(\log n+\log m)$ to $O(1)$, which would completely destroy the componentwise composition and input compression pillars of [20]. These pillars are present in all follow-up papers [17, 20, 21, 57] and seem impossible to get around, prompting the authors of [21, p. 14] to entertain the possibility that the approximate degree of AC^0 is much smaller than once conjectured. We show that this is not the case.

1.5 Our Proof

In this paper, we design hardness amplification from first principles, without using componentwise composition or input compression. Our approach efficiently amplifies the approximate degree even for functions with sparse input, while ensuring that each hardness amplification stage is implementable by a monotone circuit of constant depth with AND gates of constant fan-in and OR gates of polynomial fan-in. As a result, repeating our process any constant number of times produces a polynomial-size DNF formula of constant width.

Our approach at a high level. Let $f\colon\{0,1\}^N\to\{0,1\}$ be a given function. Let $f|_{\leqslant\theta}$ denote the restriction of f to inputs of Hamming weight at most θ, and let $d = \deg_{1/3}(f|_{\leqslant\theta})$ be the approximate degree of this restriction. The use of uppercase N is meant to emphasize that the total number of variables can be vastly larger than θ, making $f|_{\leqslant\theta}$ a function with sparse input. In actual usage, we have $N=\theta^C$ for any desired constant $C\geqslant 1$. Since an input $y\in\{0,1\}^N$ to $f|_{\leqslant\theta}$ is guaranteed to have Hamming weight at most θ, we can think of y as the disjunction of θ vectors of Hamming weight at most 1 each:

$$y = y_1\vee y_2\vee\cdots\vee y_\theta,$$

where each y_i is either the zero vector 0^N or a basis vector $e_1,e_2,\ldots,e_N$, and the disjunction on the right-hand side is applied coordinate-wise. Our approach centers around encoding each y_i as a string of $n\ll N$ bits so as to make the decoding difficult for polynomials but easy for circuits. Specifically, we seek a decoding function $h\colon\{0,1\}^n\to\{0,1\}^N$ with the following properties:

(i) the sets $h^{-1}(v)$ for $v\in\{e_1,e_2,\ldots,e_N,0^N\}$ are indistinguishable by polynomials of degree up to D, for some parameter D;
(ii) the sets $h^{-1}(v)$ for $v\in\{e_1,e_2,\ldots,e_N,0^N\}$ contain strings only of Hamming weight $O(1)$;
(iii) h is computable by a constant-depth monotone circuit with AND gates of constant fan-in and OR gates of polynomial fan-in.

With such h in hand, we define $F\colon(\{0,1\}^n)^\theta\to\{0,1\}$ by

$$F(x_1,x_2,\ldots,x_\theta) = f\left(\bigvee_{i=1}^{\theta}h(x_i)\right).$$

Then, one can reasonably expect that approximating F is harder than approximating $f|_{\leqslant\theta}$. Indeed, an approximating polynomial has access only to the encoded input $(x_1, x_2, \ldots, x_\theta)$. Decoding this input presumably involves computing $(x_1, x_2, \ldots, x_\theta) \mapsto (h(x_1), h(x_2), \ldots, h(x_\theta))$ one way or another, which by property (i) requires a polynomial of degree greater than D. Once the decoded string $h(x_1) \vee h(x_2) \vee \cdots \vee h(x_\theta)$ is available, the polynomial further needs to compute f on that input, which in and of itself requires degree d. Altogether, we expect F to have approximate degree on the order of Dd. Moreover, property (ii) ensures that F is hard to approximate even on inputs of Hamming weight $O(\theta)$, putting us in a strong position for another round of hardness amplification. Finally, property (iii) guarantees that the result of constantly many rounds of hardness amplification is computable by a DNF formula of polynomial size.

Actual implementation. As one might suspect, the above program is wildly optimistic and cannot be implemented literally. In the actual proof, we are able to ensure properties (i) and (ii) only approximately. In our construction, (i) holds only with respect to suitably chosen distributions on the sets $h^{-1}(v)$. Furthermore, for the decoding function h to be hard for polynomials, property (ii) needs to be relaxed by adding inputs of high Hamming weight to each $h^{-1}(v)$. We are still able to ensure that with respect to our distribution, nearly all inputs in $h^{-1}(v)$ have constant Hamming weight. Property (iii) is implemented as stated, allowing us to obtain a polynomial-size constant-width DNF formula in the end.

The design of h is the most demanding part of the proof. At its core, our construction of h contributes the following result of independent interest. Let k be a parameter, which we take to be a sufficiently large constant. For each $v \in \{e_1, e_2, \ldots, e_N, 0^N\}$, we construct a probability distribution λ_v on $\{0,1\}^n$ that has all but a vanishing fraction of its mass on inputs of Hamming weight exactly k, and moreover any two such distributions λ_v and $\lambda_{v'}$ are indistinguishable by polynomials of low degree. We are further able to ensure that an input of Hamming weight k belongs to the support of at most one of the distributions λ_v. Thus, the λ_v are in essence supported on pairwise disjoint sets of strings of Hamming weight k, and are pairwise indistinguishable by polynomials of low degree. The decoding function h works by taking an input $x \in \{0,1\}^n$ of Hamming weight k and determining which of the distributions has x in its support—a highly efficient computation realizable as a monotone k-DNF formula. With small probability, h will receive as input a string of Hamming weight larger than k, in which case the decoding may fail.

Construction of the λ_v. Our starting point is the number-theoretic notion of *m-discrepancy*, which is a measure of pseudorandomness or aperiodicity of a given set of integers modulo m. Formally, the m-discrepancy of a nonempty finite set $S \subseteq \mathbb{Z}$ is defined as

$$\operatorname{disc}_m(S) = \max_{k=1,2,\ldots,m-1} \left| \frac{1}{|S|} \sum_{s \in S} \xi^{ks} \right|,$$

where ξ is a primitive m-th root of unity. The construction of sparse sets with low discrepancy is a well-studied problem in combinatorics and theoretical computer science. By building on previous work [3, 55], we construct a sparse set of integers with small discrepancy in our regime of interest. For our application, we set the modulus $m = N + 1$.

Continuing, let $\binom{[n]}{k}$ denote the family of cardinality-k subsets of $[n] = \{1, 2, \ldots, n\}$. To design the distributions λ_v, we need an explicit coloring $\gamma\colon \binom{[n]}{k} \to [N+1]$ that is *balanced*, in the sense that for nearly all large enough subsets $A \subseteq \{1, 2, \ldots, n\}$ and all $i \in [N+1]$, the family $\gamma^{-1}(i)$ accounts for almost exactly a $1/(N+1)$ fraction of all cardinality-k subsets of A. The existence of a highly balanced coloring follows by the probabilistic method, and we construct one explicitly by leveraging the sparse set of integers with small $(N+1)$-discrepancy constructed earlier in the proof.

Our next ingredient is a dual polynomial ω for the OR function, a staple in approximate degree lower bounds. An important property of ω is that it places a constant fraction of its ℓ_1 mass on the point 0^n. Translating ω from 0^n to a point z of slightly higher Hamming weight results in a new dual polynomial, call it ω_z. Analogous to ω, the new dual polynomial has a constant fraction of its ℓ_1 mass on z and the rest on inputs greater than z in lexicographic order.

For notational convenience, let us now rename γ's range elements $1, 2, \ldots, N+1$ to $e_1, e_2, \ldots, e_N, 0^N$, respectively. For $v \in \{e_1, e_2, \ldots, e_N, 0^N\}$, define Φ_v to be the average of the dual polynomials ω_z where z ranges over all characteristic vectors of the sets in $\gamma^{-1}(v)$. Being a convex combination of dual polynomials, each Φ_v is a dual object orthogonal to polynomials of low degree. Observe further that each Φ_v is supported on inputs of Hamming weight at least k, and any input of Hamming weight exactly k belongs to the support of exactly one Φ_v. For inputs x of Hamming weight greater than k, a remarkable thing happens: $\Phi_v(x)$ is almost the same for all v. We prove this by exploiting the fact that γ is highly balanced. As a result, the "common part" of the Φ_v for inputs of Hamming weight greater than k can be subtracted out to obtain a function $\widetilde{\Phi_v}$ for each $v \in \{e_1, e_2, \ldots, e_N, 0^N\}$. While these new functions are not dual polynomials, the *difference* of any two of them is since $\widetilde{\Phi_v} - \widetilde{\Phi_{v'}} = \Phi_v - \Phi_{v'}$. Put another way, the $\widetilde{\Phi_v}$ are pairwise indistinguishable by low-degree polynomials. By defining the $\widetilde{\Phi_v}$ in a somewhat more subtle way, we further ensure that each $\widetilde{\Phi_v}$ is nonnegative. The distribution λ_v is then defined to be the normalized function $\widetilde{\Phi_v}/\|\widetilde{\Phi_v}\|_1$. This construction ensures all the properties that we need: λ_v has nearly all of its mass on inputs of Hamming weight k; an input of Hamming weight k belongs to the support of at most one distribution λ_v; and any pair of distributions $\lambda_v, \lambda_{v'}$ are indistinguishable by a low-degree polynomial. Observe that in our construction, λ_v is close to the uniform probability distribution on the characteristic vectors of the sets in $\gamma^{-1}(v)$.

This completes the proof sketch of our main results on approximate degree (Theorems 1.1 and 1.2 and Corollary 1.3). To obtain the communication lower bounds (Theorems 1.4 and 1.5), we invoke the pattern matrix method for the corresponding models.

2 PRELIMINARIES

2.1 General Notation

For a string $x \in \{0,1\}^n$ and a set $S \subseteq \{1, 2, \ldots, n\}$, we let $x|_S$ denote the restriction of x to the indices in S. In other words, $x|_S = x_{i_1} x_{i_2} \ldots x_{i_{|S|}}$, where $i_1 < i_2 < \cdots < i_{|S|}$ are the elements of S. The

characteristic vector $\mathbf{1}_S$ of a set $S \subseteq \{1, 2, \ldots, n\}$ is given by

$$(\mathbf{1}_S)_i = \begin{cases} 1 & \text{if } i \in S, \\ 0 & \text{otherwise.} \end{cases}$$

Given an arbitrary set X and elements $x, y \in X$, the Kronecker delta $\delta_{x,y}$ is defined by

$$\delta_{x,y} = \begin{cases} 1 & \text{if } x = y, \\ 0 & \text{otherwise.} \end{cases}$$

For a logical condition C, we use the Iverson bracket

$$\mathbf{I}[C] = \begin{cases} 1 & \text{if } C \text{ holds}, \\ 0 & \text{otherwise.} \end{cases}$$

We let $\mathbb{N} = \{0, 1, 2, 3, \ldots\}$ denote the set of natural numbers. We use the comparison operators in a unary capacity to denote one-sided intervals of the real line. Thus, $<a$, $\leqslant a$, $>a$, $\geqslant a$ stand for $(-\infty, a)$, $(-\infty, a]$, (a, ∞), $[a, \infty)$, respectively. We let $\ln x$ and $\log x$ stand for the natural logarithm of x and the logarithm of x to base 2, respectively. The term *Euclidean space* refers to $\mathbb{R}^n$ for some positive integer n. We let e_i denote the vector whose i-th component is 1 and the others are 0. Thus, the vectors $e_1, e_2, \ldots, e_n$ form the standard basis for $\mathbb{R}^n$. For a complex number x, we denote the real part, imaginary part, and complex conjugate of x as usual by $\mathrm{Re}(x)$, $\mathrm{Im}(x)$, and $\overline{x}$, respectively. We typeset the imaginary unit $\mathbf{i}$ in boldface to distinguish it from the index variable i. For an arbitrary integer a and a positive integer m, recall that $a \bmod m$ denotes the unique element of $\{0, 1, 2, \ldots, m-1\}$ that is congruent to a modulo m.

For a set X, we let $\mathbb{R}^X$ denote the linear space of real-valued functions on X. The *support* of a function $f \in \mathbb{R}^X$ is denoted $\operatorname{supp} f = \{x \in X : f(x) \neq 0\}$. For real-valued functions with finite support, we adopt the usual norms and inner product:

$$\begin{aligned} \|f\|_\infty &= \max_{x \in \operatorname{supp} f} |f(x)|, \\ \|f\|_1 &= \sum_{x \in \operatorname{supp} f} |f(x)|, \\ \langle f, g \rangle &= \sum_{x \in \operatorname{supp} f \cap \operatorname{supp} g} f(x)g(x). \end{aligned}$$

This covers as a special case functions on finite sets. Analogous to functions, we adopt the familiar norms for vectors $x \in \mathbb{R}^n$ in Euclidean space: $\|x\|_\infty = \max_{i=1,\ldots,n} |x_i|$ and $\|x\|_1 = \sum_{i=1}^n |x_i|$. The *tensor product* of $f \in \mathbb{R}^X$ and $g \in \mathbb{R}^Y$ is denoted $f \otimes g \in \mathbb{R}^{X \times Y}$ and given by $(f \otimes g)(x, y) = f(x)g(y)$. The tensor product $f \otimes f \otimes \cdots \otimes f$ (n times) is abbreviated $f^{\otimes n}$. We frequently omit the argument in equations and inequalities involving functions, as in $\operatorname{sgn} p = (-1)^f$. Such statements are to be interpreted pointwise. For example, the statement "$f \geqslant 2|g|$ on X" means that $f(x) \geqslant 2|g(x)|$ for every $x \in X$.

We adopt the standard notation for function composition, with $f \circ g$ defined by $(f \circ g)(x) = f(g(x))$. In addition, we use the $\circ$ operator to denote the *componentwise composition* of Boolean functions. Formally, the componentwise composition of $f \colon \{0,1\}^n \to \{0,1\}$ and $g \colon X \to \{0,1\}$ is the function $f \circ g \colon X^n \to \{0,1\}$ given by $(f \circ g)(x_1, x_2, \ldots, x_n) = f(g(x_1), g(x_2), \ldots, g(x_n))$. Componentwise composition is consistent with standard composition, which in the context of Boolean functions is only defined for $n = 1$. Thus, the meaning of $f \circ g$ is determined by the range of g and is never in doubt.

For a natural number n, we abbreviate $[n] = \{1, 2, \ldots, n\}$. For a set S and an integer k, we let $\binom{S}{k}$ stand for the family of cardinality-k subsets of S:

$$\binom{S}{k} = \{A \subseteq S : |A| = k\}.$$

Analogously, for any set I, we define

$$\binom{S}{I} = \{A \subseteq S : |A| \in I\}.$$

To illustrate, $\binom{S}{\leqslant k}$ denotes the family of subsets of S that have cardinality at most k. Analogously, we have the symbols $\binom{S}{<k}$, $\binom{S}{\geqslant k}$, $\binom{S}{>k}$. Throughout this manuscript, we use brace notation as in $\{z_1, z_2, \ldots, z_n\}$ to specify *multisets* rather than *sets*, the distinction being that the number of times an element occurs is taken into account. The *cardinality* $|Z|$ of a finite multiset Z is defined to be the total number of element occurrences in Z, with each element counted as many times as it occurs. The equality and subset relations on multisets are defined analogously, with the number of element occurrences taken into account. For example, $\{1, 1, 2\} = \{1, 2, 1\}$ but $\{1, 1, 2\} \neq \{1, 2\}$. Similarly, $\{1, 2\} \subseteq \{1, 1, 2\}$ but $\{1, 1, 2\} \not\subseteq \{1, 2\}$.

2.2 Boolean Strings and Functions

We identify the Boolean values "true" and "false" with 1 and 0, respectively, and view Boolean functions as mappings $X \to \{0, 1\}$ for a finite set X. The familiar functions $\mathrm{OR}_n \colon \{0,1\}^n \to \{0,1\}$ and $\mathrm{AND}_n \colon \{0,1\}^n \to \{0,1\}$ are given by $\mathrm{OR}_n(x) = \bigvee_{i=1}^n x_i$ and $\mathrm{AND}_n(x) = \bigwedge_{i=1}^n x_i$. We abbreviate $\mathrm{NOR}_n = \neg\mathrm{OR}_n$. For Boolean strings $x, y \in \{0,1\}^n$, we let $x \oplus y$ denote their bitwise XOR. The strings $x \wedge y$ and $x \vee y$ are defined analogously, with the binary operator applied bitwise.

For a vector $v \in \mathbb{N}^n$, we define its *weight* $|v|$ to be $|v| = v_1 + v_2 + \cdots + v_n$. If $x \in \{0,1\}^n$ is a Boolean string, then $|x|$ is precisely the Hamming weight of x. For any sets $X \subseteq \mathbb{N}^n$ and $W \subseteq \mathbb{R}$, we define $X|_W$ to be the subset of strings in X whose weight belongs to W :

$$X|_W = \{x \in X : |x| \in W\}.$$

In the case of a one-element set $W = \{w\}$, we further shorten $X|_{\{w\}}$ to $X|_w$. For example, $\mathbb{N}^n|_{\leqslant w}$ denotes the set of vectors whose n components are natural numbers and sum to at most w, whereas $\{0,1\}^n|_w$ denotes the set of Boolean strings of length n and Hamming weight exactly w. For a function $f \colon X \to \mathbb{R}$ on a subset $X \subseteq \{0,1\}^n$, we let $f|_W$ denote the restriction of f to $X|_W$. Thus, $f|_W$ is a function with domain $X|_W$ given by $f|_W(x) = f(x)$. A typical instance of this notation would be $f|_{\leqslant w}$ for some real number w, corresponding to the restriction of f to Boolean strings of Hamming weight at most w.

2.3 Concentration of Measure

Throughout this manuscript, we view probability distributions as real functions. This convention makes available the shorthand notation introduced above. In particular, for probability distributions μ and λ, the symbol $\operatorname{supp} \mu$ denotes the support of μ, and $\mu \otimes \lambda$ denotes

the probability distribution given by $(\mu \otimes \lambda)(x, y) = \mu(x)\lambda(y)$. We use the notation $\mu \times \lambda$ interchangeably with $\mu \otimes \lambda$, the former being more standard for probability distributions. If μ is a probability distribution on X, we consider μ to be defined also on any superset of X with the understanding that $\mu = 0$ outside X.

We will need a concentration-of-measure result due to Bun and Thaler [20, Lemma 4.7] for product distributions on $\mathbb{N}^n$.

Lemma 2.1 (cf. Bun and Thaler). *Let $\lambda_1, \lambda_2, \ldots, \lambda_n$ be distributions on $\mathbb{N}$ with finite support such that*

$$\lambda_i(t) \leqslant \frac{C\alpha^t}{(t+1)^2}, \qquad t \in \mathbb{N}, \tag{2}$$

where $C \geqslant 0$ and $0 \leqslant \alpha \leqslant 1$. Then for all $T \geqslant 8Ce n(1 + \ln n)$,

$$\mathop{\mathbf{P}}_{v \sim \lambda_1 \times \lambda_2 \times \cdots \times \lambda_n} [\|v\|_1 \geqslant T] \leqslant \alpha^{T/2}.$$

Bun and Thaler's result in [20, Lemma 4.7] differs slightly from the statement above. The proof of Lemma 2.1 as stated can be found in [57, Lemma 3.6]. By leveraging Lemma 2.1, we obtain the following concentration result for probability distributions that are supported on the Boolean hypercube, rather than $\mathbb{N}$, and are shifted from the origin.

Lemma 2.2. *Fix integers $B \geqslant k \geqslant 0$. Let $\lambda_1, \lambda_2, \ldots, \lambda_\ell$ be probability distributions on $\{0,1\}^B$ with support contained in $\{0,1\}^B|_{\geqslant k}$. Suppose further that*

$$\lambda_i(\{0,1\}^B|_t) \leqslant \frac{C\alpha^{t-k}}{(t-k+1)^2}, \quad i \in [\ell], \quad t \in \{k, k+1, \ldots, B\},$$

where $C \geqslant 0$ and $0 \leqslant \alpha \leqslant 1$. Then for all $T \geqslant 8Ce\ell(1 + \ln \ell) + \ell k$,

$$\mathop{\mathbf{P}}_{(x_1,\ldots,x_\ell) \sim \lambda_1 \times \cdots \times \lambda_\ell} \left[\sum_{i=1}^{\ell} |x_i| \geqslant T \right] \leqslant \alpha^{(T - \ell k)/2}.$$

Proof. For $i = 1, 2, \ldots, \ell$, consider the distribution μ_i on $\{0, 1, \ldots, B-k\}$ given by $\mu_i(t) = \lambda_i(\{0,1\}^B|_{t+k})$. Then

$$\mu_i(t) \leqslant \frac{C\alpha^t}{(t+1)^2}, \qquad i \in [\ell], \quad t \geqslant 0. \tag{3}$$

Moreover, the random variable $|x_i|$ with $x_i \sim \lambda_i$ has the same distribution as the random variable $u_i + k$ for $u_i \sim \mu_i$. As a result,

$$\begin{aligned}
\mathop{\mathbf{P}}_{(x_1,\ldots,x_\ell) \sim \lambda_1 \times \cdots \times \lambda_\ell} \left[\sum_{i=1}^{\ell} |x_i| \geqslant T \right] &= \mathop{\mathbf{P}}_{u \sim \mu_1 \times \cdots \times \mu_\ell} \left[\sum_{i=1}^{\ell} (u_i + k) \geqslant T \right] \\
&= \mathop{\mathbf{P}}_{u \sim \mu_1 \times \cdots \times \mu_\ell} [\|u\|_1 \geqslant T - k\ell] \\
&\leqslant \alpha^{(T-\ell k)/2},
\end{aligned}$$

where the last step uses Lemma 2.1 along with (3) and the hypothesis that $T \geqslant 8Ce\ell(1 + \ln \ell) + \ell k$. □

2.4 Orthogonal Content

For a multivariate polynomial $p\colon \mathbb{R}^n \to \mathbb{R}$, we let $\deg p$ denote the total degree of p, i.e., the largest degree of any monomial of p. We use the terms *degree* and *total degree* interchangeably in this paper. It will be convenient to define the degree of the zero polynomial by $\deg 0 = -\infty$. For a real-valued function ϕ supported on a finite subset of $\mathbb{R}^n$, the *orthogonal content of ϕ*, denoted $\operatorname{orth} \phi$, is the minimum degree of a real polynomial p for which $\langle \phi, p \rangle \neq 0$. We adopt the convention that $\operatorname{orth} \phi = \infty$ if no such polynomial exists. It is clear that $\operatorname{orth} \phi \in \mathbb{N} \cup \{\infty\}$, with the extremal cases $\operatorname{orth} \phi = 0 \iff \langle \phi, 1 \rangle \neq 0$ and $\operatorname{orth} \phi = \infty \iff \phi = 0$. Additional facts about orthogonal content are given by the following two propositions.

Proposition 2.3. *Let X and Y be nonempty finite subsets of Euclidean space. Then:*

(i) $\operatorname{orth}(\phi + \psi) \geqslant \min\{\operatorname{orth} \phi, \operatorname{orth} \psi\}$ *for all* $\phi, \psi\colon X \to \mathbb{R}$;
(ii) $\operatorname{orth}(\phi \otimes \psi) = \operatorname{orth}(\phi) + \operatorname{orth}(\psi)$ *for all* $\phi\colon X \to \mathbb{R}$ *and* $\psi\colon Y \to \mathbb{R}$.

A proof of Proposition 2.3 can be found in [57, Proposition 2.1].

Proposition 2.4. *Define $V = \{0^N, e_1, e_2, \ldots, e_N\} \subseteq \mathbb{R}^N$. Fix functions $\phi_v\colon X \to \mathbb{R}$ $(v \in V)$, where X is a finite subset of Euclidean space. Suppose that*

$$\operatorname{orth}(\phi_u - \phi_v) \geqslant D, \qquad u, v \in V, \tag{4}$$

where D is a positive integer. Then for every polynomial $p\colon X^\ell \to \mathbb{R}$, the mapping $z \mapsto \langle \bigotimes_{i=1}^{\ell} \phi_{z_i}, p \rangle$ is a polynomial on V^ℓ of degree at most $(\deg p)/D$.

Proof. By linearity, it suffices to consider factored polynomials $p(x_1, \ldots, x_\ell) = \prod_{i=1}^{\ell} p_i(x_i)$, where each p_i is a nonzero polynomial on X. In this setting,

$$\left\langle \bigotimes_{i=1}^{\ell} \phi_{z_i}, p \right\rangle = \prod_{i=1}^{\ell} \langle \phi_{z_i}, p_i \rangle. \tag{5}$$

By (4), we have $\langle \phi_{0^N}, p_i \rangle = \langle \phi_{e_1}, p_i \rangle = \langle \phi_{e_2}, p_i \rangle = \cdots = \langle \phi_{e_N}, p_i \rangle$ for any index i with $\deg p_i < D$. As a result, polynomials p_i with $\deg p_i < D$ do not contribute to the degree of the right-hand side of (5) as a function of z. For the other polynomials p_i, the inner product $\langle \phi_{z_i}, p_i \rangle$ is a linear polynomial in z_i, namely,

$$\begin{aligned}
\langle \phi_{z_i}, p_i \rangle = z_{i,1} \langle \phi_{e_1}, p_i \rangle + z_{i,2} \langle \phi_{e_2}, p_i \rangle + \cdots + z_{i,N} \langle \phi_{e_N}, p_i \rangle \\
+ \left(1 - \sum_{j=1}^{N} z_{i,j} \right) \langle \phi_{0^N}, p_i \rangle.
\end{aligned}$$

Thus, polynomials p_i with $\deg p_i \geqslant D$ contribute at most 1 each to the degree. Summarizing, the right-hand side of (5) is a real polynomial in $z_1, z_2, \ldots, z_\ell$ of degree at most $|\{i : \deg p_i \geqslant D\}| \leqslant \frac{\deg p}{D}$. □

Proposition 2.4 generalizes an analogous result in [57, Proposition 2.2], where the special case $N = 1$ was treated.

2.5 Approximation by Polynomials

For a real number $\varepsilon \geqslant 0$ and a function $f\colon X \to \mathbb{R}$ on a finite subset X of Euclidean space, the *ε-approximate degree of f* is denoted $\deg_\varepsilon(f)$ and is defined to be the minimum degree of a polynomial p such that $\|f - p\|_\infty \leqslant \varepsilon$. For $\varepsilon < 0$, it will be convenient to define $\deg_\varepsilon(f) = +\infty$ since no polynomial satisfies $\|f - p\|_\infty \leqslant \varepsilon$ in this case. We focus on the approximate degree of Boolean functions $f\colon X \to \{0,1\}$. In this setting, the standard choice of the error parameter is $\varepsilon = 1/3$. This choice is without loss of generality since $\deg_\varepsilon(f) = \Theta(\deg_{1/3}(f))$ for every Boolean function f and every constant $0 < \varepsilon < 1/2$. In what follows, we refer to 1/3-approximate

degree simply as "approximate degree." The notion of approximate degree has the following dual characterization [47, 48].

Fact 2.5. *Let $f\colon X \to \mathbb{R}$ be given, for a finite set $X \subset \mathbb{R}^n$. Let $d \geqslant 0$ be an integer. Then $\deg_\varepsilon(f) \geqslant d$ if and only if there exists a function $\psi\colon X \to \mathbb{R}$ such that*

$$\begin{aligned} &\langle f, \psi\rangle > \varepsilon\|\psi\|_1, \\ &\operatorname{orth}\psi \geqslant d. \end{aligned}$$

This characterization of approximate degree can be verified using linear programming duality, cf. [47, 48]. Fact 2.5 makes it possible to prove lower bounds on approximate degree in a constructive manner, by exhibiting a dual object ψ that serves a witness. This object as referred to as a *dual polynomial*. Often, a dual polynomial for a composed function f can be constructed by combining dual objects for various components of f. Of particular importance in the study of AC^0 is the dual object for the OR function. The first dual polynomial for OR was constructed by Špalek [61], with many refinements and generalizations obtained in follow-up work [17, 18, 20, 53, 54, 57]. We will use the following construction from [57, Lemma B.2].

Lemma 2.6. *Let ε be given, $0 < \varepsilon < 1$. Then for some constant $c = c(\varepsilon) \in (0,1)$ and every integer $n \geqslant 1$, there is an (explicitly given) function $\omega\colon \{0,1,2,\dots,n\} \to \mathbb{R}$ such that*

$$\begin{aligned} &\omega(0) > \frac{1-\varepsilon}{2} \cdot \|\omega\|_1, \\ &|\omega(t)| \leqslant \frac{1}{ct^2\, 2^{ct/\sqrt{n}}} \cdot \|\omega\|_1 && (t = 1,2,\dots,n), \\ &(-1)^t\omega(t) \geqslant 0 && (t = 0,1,2,\dots,n), \\ &\operatorname{orth}\omega \geqslant c\sqrt{n}. \end{aligned}$$

The following lemma is useful when one needs to adjust a dual object's metric properties while preserving its orthogonality to low-degree polynomials. The lemma plays a basic role in several recent papers [17, 20, 21, 43, 57] as well as our work. Its proof is available in the full version [56].

Lemma 2.7. *Let $\Phi\colon \{0,1\}^B \to \mathbb{R}$ be given. Fix integers $T \geqslant D \geqslant 0$. Then there is an (explicitly given) function $\tilde{\Phi}\colon \{0,1\}^B \to \mathbb{R}$ such that*

$$\operatorname{supp}\tilde{\Phi} \subseteq \{0,1\}^B|_{\leqslant T}, \tag{6}$$

$$\operatorname{orth}(\Phi - \tilde{\Phi}) > D, \tag{7}$$

$$\|\Phi - \tilde{\Phi}\|_1 \leqslant \left(1 + 2^D\binom{B}{D}\right) \sum_{x\colon |x| > T} |\Phi(x)|. \tag{8}$$

2.6 Symmetrization

Let S_n denote the symmetric group on n elements. For a permutation $\sigma \in S_n$ and an arbitrary sequence $x = (x_1, x_2, \dots, x_n)$, we adopt the shorthand $\sigma x = (x_{\sigma(1)}, x_{\sigma(2)}, \dots, x_{\sigma(n)})$. A function $f(x_1, x_2, \dots, x_n)$ is called *symmetric* if it is invariant under permutation of the input variables: $f(x_1, x_2, \dots, x_n) = f(x_{\sigma(1)}, x_{\sigma(2)}, \dots, x_{\sigma(n)})$ for all x and σ. Symmetric functions on $\{0,1\}^n$ are intimately related to univariate polynomials, as was first observed by Minsky and Papert in their *symmetrization argument* [37].

Proposition 2.8 (Minsky and Papert). *Let $p\colon \mathbb{R}^n \to \mathbb{R}$ be a given polynomial. Then the mapping*

$$t \mapsto \mathop{\mathbf{E}}_{x \in \{0,1\}^n|_t} p(x)$$

is a univariate polynomial on $\{0,1,2,\dots,n\}$ of degree at most $\deg p$.

The next result, proved in [57, Corollary 2.13], generalizes Minsky and Papert's symmetrization to the setting when $x_1, x_2, \dots, x_n$ are vectors rather than bits.

Fact 2.9 (Sherstov and Wu). *Let $p\colon (\mathbb{R}^N)^\theta \to \mathbb{R}$ be a given polynomial. Then the mapping*

$$v \mapsto \mathop{\mathbf{E}}_{\substack{x \in \{0^N, e_1, e_2, \dots, e_N\}^\theta\colon \\ x_1 + x_2 + \cdots + x_\theta = v}} p(x) \tag{9}$$

is a polynomial on $\mathbb{N}^N|_{\leqslant\theta}$ of degree at most $\deg p$.

Minsky and Papert's symmetrization corresponds to $N = 1$ in Fact 2.9.

3 BALANCED COLORINGS

For integers $n \geqslant k \geqslant 1$ and $r \geqslant 1$, consider a mapping $\gamma\colon \binom{[n]}{k} \to [r]$. We refer to any such γ as a *coloring of $\binom{[n]}{k}$ with r colors*. An important ingredient in our main result is the construction of a *balanced* coloring, in the following technical sense.

Definition 3.1. Let $\gamma\colon \binom{[n]}{k} \to [r]$ be a given coloring. For a subset $A \subseteq [n]$, we say that γ *is ε-balanced on A* iff for each $i \in [r]$,

$$\frac{1-\varepsilon}{r}\binom{|A|}{k} \leqslant \left|\gamma^{-1}(i) \cap \binom{A}{k}\right| \leqslant \frac{1+\varepsilon}{r}\binom{|A|}{k}.$$

We define γ to be (ε, δ, m)-*balanced* iff

$$\mathop{\mathbf{P}}_{A \in \binom{[n]}{\ell}}[\gamma \text{ is } \varepsilon\text{-balanced on } A] \geqslant 1 - \delta$$

for all $\ell \in \{m, m+1, \dots, n\}$.

The next lemma, proved in the full version [56], uses the probabilistic method to establish the existence of balanced colorings with excellent parameters.

Lemma 3.2. *Let $\varepsilon, \delta \in (0,1]$ be given. Let n, m, k, r be positive integers with $n \geqslant m \geqslant k$ and*

$$\binom{m}{k} \geqslant \frac{3r}{\varepsilon^2} \cdot \ln\frac{2rn}{\delta}. \tag{10}$$

Then there exists an (ε, δ, m)-balanced coloring $\gamma\colon \binom{[n]}{k} \to [r]$.

Corollary 3.3. *Let n, m, k, r be positive integers with $n \geqslant m \geqslant k^2$. Then there is an $(\varepsilon, \varepsilon, m)$-balanced coloring $\gamma\colon \binom{[n]}{k} \to [r]$, where*

$$\varepsilon = \frac{3r\sqrt{k\ln(n+1)}}{m^{k/4}}.$$

In the full version of our paper [56], we give the following construction of balanced colorings.

Theorem 3.4 (Explicit balanced coloring). *Let n, m, k, r be integers with $n/2 \geqslant m \geqslant k \geqslant 1$ and $r \geqslant 2$. Then there is an*

(explicitly given) integer $n' \in (n/2, n]$ *and an (explicitly given)* (ε, δ, m)*-balanced coloring* $\gamma\colon \binom{[n']}{k} \to [r]$*, where*

$$\varepsilon = 4r^2 k \exp\left(-\frac{\sqrt{m}}{16k}\right) + r\left(\frac{3C^* \log^2(n+r)}{m^{1/4}}\right)^k, \tag{11}$$

$$\delta = 4r \exp\left(-\frac{\sqrt{m}}{8}\right), \tag{12}$$

and $C^* \geqslant 1$ *is an absolute constant.*

4 PSEUDODISTRIBUTIONS FROM BALANCED COLORINGS

Recall from the introduction that our approach centers around encoding the vectors $e_1, e_2, \ldots, e_N, 0^N$ as n-bit strings with $n \ll N$ so as to make the decoding easy for circuits but hard for low-degree polynomials. The construction of this code requires several steps. As a first step, we show how to convert any balanced coloring of $\binom{[n]}{k}$ with r colors into an explicit sequence of functions $\phi_1, \phi_2, \ldots, \phi_r\colon \{0,1\}^n \to \mathbb{R}$ that are almost everywhere nonnegative, are supported almost entirely on pairwise disjoint sets of strings of Hamming weight k, and are pairwise indistinguishable by low-degree polynomials. We call them *pseudodistributions* to highlight the fact that each ϕ_i has ℓ_1 norm approximately 1, nearly all of it coming from the points where ϕ_i is nonnegative.

Theorem 4.1. *Let* $\varepsilon, \delta \in [0,1)$ *be given. Let* n, m, k, r *be positive integers with* $n \geqslant m > k$*. Let* $\gamma\colon \binom{[n]}{k} \to [r]$ *be a given* (ε, δ, m)*-balanced coloring. Then there are (explicitly given) functions* $\phi_1, \phi_2, \ldots, \phi_r\colon \{0,1\}^n \to \mathbb{R}$ *with the following properties.*

(i) ***Support:*** $\operatorname{supp}\phi_i \subseteq \{x \in \{0,1\}^n : |x| = k \text{ or } |x| \geqslant m\}$;

(ii) ***Essential support:*** $\{0,1\}^n|_k \cap \operatorname{supp}\phi_i = \{\mathbf{1}_S : S \in \gamma^{-1}(i)\}$;

(iii) ***Nonnegativity:*** $\phi_i \geqslant 0$ *on* $\{0,1\}^n|_k$;

(iv) ***Normalization:*** $\sum_{x:|x|=k} \phi_i(x) = 1$;

(v) ***Tail bound:*** $\sum_{x:|x|\neq k} |\phi_i(x)| \leqslant (8\varepsilon + 4r\delta)/(1-\varepsilon)$;

(vi) ***Graded bound:*** *for some absolute constant* $c' \in (0,1)$,

$$\sum_{x:|x|=\ell} |\phi_i(x)| \leqslant \frac{\varepsilon + r\delta}{1-\varepsilon} \cdot \frac{m^2}{c'\ell^2} \cdot \exp\left(-\frac{c'(\ell-k)}{\sqrt{nm}}\right), \quad \ell > k;$$

(vii) ***Orthogonality:*** *for some absolute constant* $c'' \in (0,1)$,

$$\operatorname{orth}(\phi_i - \phi_j) \geqslant c''\sqrt{\frac{n}{m}}, \qquad i, j \in [r].$$

The proof of this result is available in the full version [56].

5 ENCODING VIA INDISTINGUISHABLE DISTRIBUTIONS

As our next step, we will show that the pseudodistributions $\phi_1, \phi_2, \ldots, \phi_r$ in Theorem 4.1 can be turned into actual probability distributions $\lambda_1, \lambda_2, \ldots, \lambda_r$ provided that the underlying coloring of $\binom{[n]}{k}$ is sufficiently balanced. The resulting distributions λ_i inherit all the desirable analytic properties established for the ϕ_i in Theorem 4.1. Specifically, the λ_i are supported almost entirely on pairwise disjoint sets of inputs of Hamming weight k and are pairwise indistinguishable by low-degree polynomials.

Theorem 5.1. *Let* $0 < \beta < 1$ *be given. Let* n, n', m, k, r *be positive integers with* $n \geqslant n' \geqslant m > k$*. Let* $\gamma\colon \binom{[n']}{k} \to [r]$ *be a given* $(\frac{\beta}{16rm^2}, \frac{\beta}{16r^2m^2}, m)$*-balanced coloring. Then there are (explicitly given) probability distributions* $\lambda_1, \lambda_2, \ldots, \lambda_r$ *on* $\{0,1\}^n$ *such that*

$$\operatorname{supp}\lambda_i \subseteq \{x \in \{0,1\}^n : |x| = k \text{ or } |x| \geqslant m\}, \quad i \in [r], \tag{13}$$

$$\{0,1\}^n|_k \cap \operatorname{supp}\lambda_i = \{\mathbf{1}_S : S \in \gamma^{-1}(i)\}, \quad i \in [r], \tag{14}$$

$$\lambda_i(\{0,1\}^n|_k) \geqslant 1 - \beta, \quad i \in [r], \tag{15}$$

$$\lambda_i(\{0,1\}^n|_\ell) \leqslant \frac{\exp(-c(\ell-k)/\sqrt{n'm})}{c(\ell-k+1)^2}, \quad i \in [r],\ \ell \geqslant k, \tag{16}$$

$$\operatorname{orth}(\lambda_i - \lambda_j) \geqslant c\sqrt{\frac{n'}{m}}, \quad i, j \in [r], \tag{17}$$

where $c \in (0,1)$ *is an absolute constant, independent of* n, n', m, k, r, β.

The proof of this result is available in the full version [56].

6 HARDNESS AMPLIFICATION FOR APPROXIMATE DEGREE

We have reached the crux of our proof, a hardness amplification theorem for approximate degree. Unlike previous work, our hardness amplification is directly applicable to Boolean functions with sparse input and does not use componentwise composition or input compression.

Theorem 6.1. *Let* $C^* \geqslant 1$ *and* $c \in (0,1)$ *be the absolute constants from Theorems 3.4 and 5.1, respectively. Fix a real number* $0 < \beta < 1$ *and positive integers* n, m, k, N, θ, D, T *such that*

$$n/2 \geqslant m > k, \tag{18}$$

$$4(N+1)^2 k \exp\left(-\frac{\sqrt{m}}{16k}\right) + (N+1)\left(\frac{3C^* \log^2(n+N+1)}{m^{1/4}}\right)^k \leqslant \frac{\beta}{16(N+1)^2 m^2}, \tag{19}$$

$$T \geqslant \frac{8\mathrm{e}}{c} \cdot \theta(1 + \ln\theta) + \theta k, \tag{20}$$

$$T \geqslant D. \tag{21}$$

Define

$$\Delta = \left(1 + 2^D \binom{n\theta}{D}\right) \exp\left(-\frac{c(T-\theta k)}{2\sqrt{nm}}\right). \tag{22}$$

Then there is an (explicitly given) mapping $H\colon (\{0,1\}^n)^\theta \to \{0,1\}^N$ *such that each output bit of* H *is computable by a monotone* k*-DNF formula and*

$$\deg_{\varepsilon - 2\beta\theta - 2\Delta}((f \circ H)|_{\leqslant T}) \geqslant \min\left\{c \deg_\varepsilon(f|_{\leqslant\theta})\sqrt{\frac{n}{2m}}, D\right\} \tag{23}$$

for every function $f\colon \{0,1\}^N \to \{0,1\}$ *and every* $\varepsilon \in [0,1]$.

Proof. We may assume that

$$\varepsilon - 2\beta\theta - 2\Delta > 0 \tag{24}$$

since otherwise the left-hand side of (23) is by definition $+\infty$. Define $V \subseteq \mathbb{R}^N$ by $V = \{0^N, e_1, e_2, \ldots, e_N\}$, and set $r = N + 1$. In view of (18) and (19), Theorem 3.4 gives an explicit integer $n' \in (n/2, n]$ and an explicit $(\frac{\beta}{16r^2m^2}, \frac{\beta}{16r^2m^2}, m)$-balanced coloring $\gamma\colon \binom{[n']}{k} \to [r]$. Alternately, if one is not concerned about explicitness, the existence of γ can be deduced from the much simpler Corollary 3.3. Specifically, (19) forces $\sqrt{m} \geqslant k$ and in particular $n \geqslant m \geqslant k^2 \geqslant 1$. Moreover, (19) implies that $3r\sqrt{k\ln(n+1)}/m^{k/4} \leqslant \frac{\beta}{16r^2m^2}$. Now Corollary 3.3 guarantees the existence of a $(\frac{\beta}{16r^2m^2}, \frac{\beta}{16r^2m^2}, m)$-balanced coloring $\gamma\colon \binom{[n]}{k} \to [r]$.

Since $n' \geqslant m > k$, Theorem 5.1 gives explicit distributions $\lambda_{0^N}, \lambda_{e_1}, \lambda_{e_2}, \ldots, \lambda_{e_N}$ on $\{0,1\}^n$ such that

$$\operatorname{supp}\lambda_v \subseteq \{x \in \{0,1\}^n : |x| = k \text{ or } |x| \geqslant m\}, \qquad v \in V, \tag{25}$$

$$\{0,1\}^n|_k \cap \operatorname{supp}\lambda_{e_i} = \{\mathbf{1}_S : S \in \gamma^{-1}(i)\}, \qquad i \in [N], \tag{26}$$

$$\{0,1\}^n|_k \cap \operatorname{supp}\lambda_{0^N} = \{\mathbf{1}_S : S \in \gamma^{-1}(N+1)\}, \tag{27}$$

$$\lambda_v(\{0,1\}^n|_k) \geqslant 1 - \beta, \qquad v \in V, \tag{28}$$

$$\lambda_v(\{0,1\}^n|_t) \leqslant \frac{\exp(-c(t-k)/\sqrt{nm})}{c(t-k+1)^2}, \qquad v \in V,\ t \geqslant k, \tag{29}$$

$$\operatorname{orth}(\lambda_v - \lambda_u) \geqslant c\sqrt{\frac{n}{2m}}, \qquad v, u \in V. \tag{30}$$

Properties (26) and (27) imply that

$$\{0,1\}^n|_k \cap \operatorname{supp}\lambda_u \cap \operatorname{supp}\lambda_v = \varnothing, \qquad u, v \in V,\ u \neq v. \tag{31}$$

For $\mathbf{v} = (\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_\theta) \in V^\theta$, define

$$\Lambda_{\mathbf{v}} = \bigotimes_{i=1}^{\theta} \lambda_{\mathbf{v}_i}.$$

Equations (20), (25), and (29) ensure that Lemma 2.2 is applicable to the distributions $\lambda_{\mathbf{v}_1}, \lambda_{\mathbf{v}_2}, \ldots, \lambda_{\mathbf{v}_\theta}$ with parameters $\ell = \theta$, $B = n$, $C = 1/c$, and $\alpha = \exp(-c/\sqrt{nm})$, whence

$$\Lambda_{\mathbf{v}}((\{0,1\}^n)^\theta|_{>T}) \leqslant \exp\left(-\frac{c(T-\theta k)}{2\sqrt{nm}}\right), \qquad \mathbf{v} \in V^\theta.$$

In view of (21), we can now invoke Lemma 2.7 with parameter $B = n\theta$ to obtain a function $\widetilde{\Lambda_{\mathbf{v}}}\colon (\{0,1\}^n)^\theta \to \mathbb{R}$ such that

$$\operatorname{supp}\widetilde{\Lambda_{\mathbf{v}}} \subseteq (\{0,1\}^n)^\theta|_{\leqslant T}, \qquad \mathbf{v} \in V^\theta, \tag{32}$$

$$\operatorname{orth}(\Lambda_{\mathbf{v}} - \widetilde{\Lambda_{\mathbf{v}}}) > D, \qquad \mathbf{v} \in V^\theta, \tag{33}$$

$$\|\Lambda_{\mathbf{v}} - \widetilde{\Lambda_{\mathbf{v}}}\|_1 \leqslant \Delta, \qquad \mathbf{v} \in V^\theta. \tag{34}$$

We now turn to the construction of the monotone mapping H in the theorem statement. Define $h\colon \{0,1\}^n \to \{0,1\}^N$ by

$$(h(z))_j = \bigvee_{S \in \binom{[n]}{k} : \mathbf{1}_S \in \operatorname{supp}\lambda_{e_j}} \bigwedge_{s \in S} z_s, \qquad j = 1, 2, \ldots, N. \tag{35}$$

Define $H\colon (\{0,1\}^n)^\theta \to \{0,1\}^N$ by

$$H(x_1, x_2, \ldots, x_\theta) = \bigvee_{i=1}^{\theta} h(x_i), \qquad x_1, x_2, \ldots, x_\theta \in \{0,1\}^n, \tag{36}$$

where the right-hand side is the componentwise disjunction of the Boolean vectors $h(x_1), h(x_2), \ldots, h(x_\theta)$. Observe that both h and H are monotone and are given explicitly in closed form in terms of the probability distributions λ_v constructed at the beginning of the proof.

Claim 6.2. *Let $v \in V$ be given. Then for all $z \in \{0,1\}^n|_k \cap \operatorname{supp}\lambda_v$, one has $h(z) = v$.*

Proof. Consider an arbitrary string $z \in \{0,1\}^n|_k$. In this case, (35) simplifies to $(h(z))_j = \mathbf{I}[z \in \operatorname{supp}\lambda_{e_j}]$. Thus, $h(z)$ can be written out explicitly as

$$h(z) = (\mathbf{I}[z \in \operatorname{supp}\lambda_{e_1}], \mathbf{I}[z \in \operatorname{supp}\lambda_{e_2}], \ldots, \mathbf{I}[z \in \operatorname{supp}\lambda_{e_N}]). \tag{37}$$

Now recall from (31) that a string z of Hamming weight k can belong to at most one of the sets $\operatorname{supp}\lambda_{0^N}, \operatorname{supp}\lambda_{e_1}, \operatorname{supp}\lambda_{e_2}, \ldots, \operatorname{supp}\lambda_{e_N}$. As a result, if $z \in \operatorname{supp}\lambda_{e_i}$ then $z \notin \operatorname{supp}\lambda_{e_j}$ for all $j \neq i$ and consequently $h(z) = e_i$ by (37). Analogously, if $z \in \operatorname{supp}\lambda_{0^N}$ then $z \notin \operatorname{supp}\lambda_{e_j}$ for all j and consequently $h(z) = 0^N$ by (37). This settles the claim for all $v \in V$. □

Now, fix an arbitrary function $f\colon \{0,1\}^N \to \{0,1\}$ and an error parameter $\varepsilon \in [0,1]$. Our objective is to prove (23). To begin with, abbreviate

$$d = \deg_\varepsilon(f|_{\leqslant\theta}). \tag{38}$$

By the dual characterization of approximate degree (Fact 2.5), there is a function $\psi\colon \{0,1\}^N|_{\leqslant\theta} \to \mathbb{R}$ such that

$$\|\psi\|_1 = 1, \tag{39}$$

$$\langle f, \psi\rangle > \varepsilon, \tag{40}$$

$$\operatorname{orth}\psi \geqslant d. \tag{41}$$

Define $\Psi\colon (\{0,1\}^n)^\theta \to \mathbb{R}$ by

$$\Psi = \sum_{u \in \{0,1\}^N|_{\leqslant\theta}} \psi(u) \mathop{\mathbf{E}}_{\substack{\mathbf{v} \in V^\theta : \\ \mathbf{v}_1 + \mathbf{v}_2 + \cdots + \mathbf{v}_\theta = u}} \widetilde{\Lambda_{\mathbf{v}}}.$$

Observe from (32) that Ψ is a linear combination of functions supported on inputs of Hamming weight at most T. Therefore,

$$\operatorname{supp}\Psi \subseteq (\{0,1\}^n)^\theta|_{\leqslant T}. \tag{42}$$

Furthermore,

$$\begin{aligned}
\|\Psi\|_1 &\leqslant \sum_{u \in \{0,1\}^N|_{\leqslant\theta}} |\psi(u)| \mathop{\mathbf{E}}_{\substack{\mathbf{v} \in V^\theta : \\ \mathbf{v}_1 + \mathbf{v}_2 + \cdots + \mathbf{v}_\theta = u}} \|\widetilde{\Lambda_{\mathbf{v}}}\|_1 \\
&\leqslant \left(\sum_{u \in \{0,1\}^N|_{\leqslant\theta}} |\psi(u)|\right) \max_{\mathbf{v} \in V^\theta} \|\widetilde{\Lambda_{\mathbf{v}}}\|_1 \\
&= \|\psi\|_1 \max_{\mathbf{v} \in V^\theta} \|\widetilde{\Lambda_{\mathbf{v}}}\|_1 \\
&\leqslant \|\psi\|_1 \max_{\mathbf{v} \in V^\theta} \{\|\Lambda_{\mathbf{v}}\|_1 + \|\widetilde{\Lambda_{\mathbf{v}}} - \Lambda_{\mathbf{v}}\|_1\} \\
&\leqslant 1 + \Delta,
\end{aligned} \tag{43}$$

where the first and fourth steps apply the triangle inequality, and the last step uses (34) and (39).

Next, we claim that

$$\operatorname{orth}\Psi \geqslant \min\left\{cd\sqrt{\frac{n}{2m}}, D\right\}. \tag{44}$$

Indeed, consider an arbitrary polynomial $P\colon (\{0,1\}^n)^\theta \to \mathbb{R}$ of degree less than $\min\{cd\sqrt{n/(2m)}, D\}$. Then

$$\begin{aligned}
\langle \Psi, P\rangle &= \sum_{u\in\{0,1\}^N|_{\leqslant\theta}} \psi(u) \mathop{\mathbf{E}}_{\substack{\mathbf{v}\in V^\theta:\\ \mathbf{v}_1+\mathbf{v}_2+\cdots+\mathbf{v}_\theta=u}} \langle \widetilde{\Lambda_{\mathbf{v}}}, P\rangle \\
&= \sum_{u\in\{0,1\}^N|_{\leqslant\theta}} \psi(u) \mathop{\mathbf{E}}_{\substack{\mathbf{v}\in V^\theta:\\ \mathbf{v}_1+\mathbf{v}_2+\cdots+\mathbf{v}_\theta=u}} [\langle \Lambda_{\mathbf{v}}, P\rangle + \langle \widetilde{\Lambda_{\mathbf{v}}} - \Lambda_{\mathbf{v}}, P\rangle] \\
&= \sum_{u\in\{0,1\}^N|_{\leqslant\theta}} \psi(u) \mathop{\mathbf{E}}_{\substack{\mathbf{v}\in V^\theta:\\ \mathbf{v}_1+\mathbf{v}_2+\cdots+\mathbf{v}_\theta=u}} \langle \Lambda_{\mathbf{v}}, P\rangle,
\end{aligned} \tag{45}$$

where the first and second steps use the linearity of inner product, and the third step is valid by (33). Equation (30) allows us to invoke Proposition 2.4 with $\ell = \theta$ and $\phi_v = \lambda_v$ to infer that the inner product $\langle \Lambda_{\mathbf{v}}, P\rangle$ is a polynomial in $\mathbf{v}$ of degree less than d. As a result, Fact 2.9 implies that the expected value in (45) is a polynomial in u of degree less than d. In summary, (45) is the inner product of ψ with a polynomial of degree less than d and is therefore zero by (41). The proof of (44) is complete.

To analyze the approximate degree of $(f\circ H)|_{\leqslant T}$, we need to study the inner product $\langle f\circ H, \Psi\rangle$. To this end, we prove the following claim.

Claim 6.3. *Let $u \in \{0,1\}^N|_{\leqslant\theta}$ and $\mathbf{v} = (\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_\theta) \in V^\theta$ be given such that $\mathbf{v}_1 + \mathbf{v}_2 + \cdots + \mathbf{v}_\theta = u$. Then*

$$|f(u) - \langle \widetilde{\Lambda_{\mathbf{v}}}, f\circ H\rangle| \leqslant 2\beta\theta + \Delta.$$

Proof. Since u is a Boolean vector, the equality $\mathbf{v}_1 + \mathbf{v}_2 + \cdots + \mathbf{v}_\theta = u$ forces

$$\mathbf{v}_1 \vee \mathbf{v}_2 \vee \cdots \vee \mathbf{v}_\theta = u, \tag{46}$$

where the disjunction is applied componentwise. For any input $(x_1, x_2, \ldots, x_\theta)$ where $x_i \in \{0,1\}^n|_k \cap \operatorname{supp}\lambda_{\mathbf{v}_i}$, we have

$$(f\circ H)(x_1, x_2, \ldots, x_\theta) = f\left(\bigvee_{i=1}^\theta h(x_i)\right) = f\left(\bigvee_{i=1}^\theta \mathbf{v}_i\right) = f(u),$$

where the second and third steps use Claim 6.2 and(46), respectively. Since $\operatorname{supp}\Lambda_{\mathbf{v}} = \prod_{i=1}^\theta \operatorname{supp}\lambda_{\mathbf{v}_i}$, we have shown that

$$f\circ H \equiv f(u) \qquad \text{on} \quad (\{0,1\}^n|_k)^\theta \cap \operatorname{supp}\Lambda_{\mathbf{v}}. \tag{47}$$

Furthermore,

$$\Lambda_{\mathbf{v}}((\{0,1\}^n|_k)^\theta) = \prod_{i=1}^\theta \lambda_{\mathbf{v}_i}(\{0,1\}^n|_k) \geqslant (1-\beta)^\theta \geqslant 1 - \beta\theta, \tag{48}$$

where the second step uses (28). Now

$$\begin{aligned}
&|f(u) - \langle \widetilde{\Lambda_{\mathbf{v}}}, f\circ H\rangle| \\
&\quad\leqslant |f(u) - \langle \Lambda_{\mathbf{v}}, f\circ H\rangle| + |\langle \Lambda_{\mathbf{v}} - \widetilde{\Lambda_{\mathbf{v}}}, f\circ H\rangle| \\
&\quad\leqslant |f(u) - \langle \Lambda_{\mathbf{v}}, f\circ H\rangle| + \|\Lambda_{\mathbf{v}} - \widetilde{\Lambda_{\mathbf{v}}}\|_1 \\
&\quad= \left|f(u) - \mathop{\mathbf{E}}_{\Lambda_{\mathbf{v}}} f\circ H\right| + \|\Lambda_{\mathbf{v}} - \widetilde{\Lambda_{\mathbf{v}}}\|_1 \\
&\quad\leqslant \mathop{\mathbf{E}}_{\Lambda_{\mathbf{v}}} |f(u) - f\circ H| + \|\Lambda_{\mathbf{v}} - \widetilde{\Lambda_{\mathbf{v}}}\|_1 \\
&\quad\leqslant 0\cdot\Lambda_{\mathbf{v}}((\{0,1\}^n|_k)^\theta) + 2\cdot\Lambda_{\mathbf{v}}(\overline{(\{0,1\}^n|_k)^\theta}) + \|\Lambda_{\mathbf{v}} - \widetilde{\Lambda_{\mathbf{v}}}\|_1 \\
&\quad\leqslant 2\beta\theta + \|\Lambda_{\mathbf{v}} - \widetilde{\Lambda_{\mathbf{v}}}\|_1 \\
&\quad\leqslant 2\beta\theta + \Delta,
\end{aligned}$$

where the last three steps use (47), (48), and (34), respectively. □

We are now in a position to finish the proof of Theorem 6.1. We have

$$\begin{aligned}
&\langle f, \psi\rangle - \langle f\circ H, \Psi\rangle \\
&\quad= \sum_{u\in\{0,1\}^N|_{\leqslant\theta}} \psi(u) f(u) \\
&\qquad\qquad - \sum_{u\in\{0,1\}^N|_{\leqslant\theta}} \psi(u) \mathop{\mathbf{E}}_{\substack{\mathbf{v}\in V^\theta:\\ \mathbf{v}_1+\mathbf{v}_2+\cdots+\mathbf{v}_\theta=u}} \langle \widetilde{\Lambda_{\mathbf{v}}}, f\circ H\rangle \\
&\quad= \sum_{u\in\{0,1\}^N|_{\leqslant\theta}} \psi(u) \mathop{\mathbf{E}}_{\substack{\mathbf{v}\in V^\theta:\\ \mathbf{v}_1+\mathbf{v}_2+\cdots+\mathbf{v}_\theta=u}} [f(u) - \langle \widetilde{\Lambda_{\mathbf{v}}}, f\circ H\rangle] \\
&\quad\leqslant \sum_{u\in\{0,1\}^N|_{\leqslant\theta}} |\psi(u)| \mathop{\mathbf{E}}_{\substack{\mathbf{v}\in V^\theta:\\ \mathbf{v}_1+\mathbf{v}_2+\cdots+\mathbf{v}_\theta=u}} |f(u) - \langle \widetilde{\Lambda_{\mathbf{v}}}, f\circ H\rangle| \\
&\quad\leqslant \|\psi\|_1 \max_{u\in\{0,1\}^N|_{\leqslant\theta}} \max_{\substack{\mathbf{v}\in V^\theta:\\ \mathbf{v}_1+\mathbf{v}_2+\cdots+\mathbf{v}_\theta=u}} |f(u) - \langle \widetilde{\Lambda_{\mathbf{v}}}, f\circ H\rangle| \\
&\quad\leqslant \|\psi\|_1(2\beta\theta + \Delta) \\
&\quad\leqslant 2\beta\theta + \Delta,
\end{aligned} \tag{49}$$

where the last two steps use Claim 6.3 and (39), respectively. Then

$$\begin{aligned}
\langle f\circ H, \Psi\rangle &> \varepsilon - 2\beta\theta - \Delta \\
&\geqslant \frac{\varepsilon - 2\beta\theta - \Delta}{1+\Delta}\cdot\|\Psi\|_1 \\
&\geqslant (\varepsilon - 2\beta\theta - 2\Delta)\|\Psi\|_1,
\end{aligned} \tag{50}$$

where the first step uses (40) and (49), the second step is justified by (24) and (43), and the third step is legitimate since $a/(1+b) \geqslant a-b$ for all $a \in [0,1]$ and $b \geqslant 0$. Recall from (42) that Ψ is supported on inputs of Hamming weight at most T and can therefore be regarded as a function on $(\{0,1\}^n)^\theta|_{\leqslant T}$. Now the claimed bound (23) is immediate from (44), (50), and Fact 2.5. □

Corollary 6.4. *Fix reals $\alpha \in (0,1]$, $A \geqslant 1$, and $C \geqslant 1$ arbitrarily. Then for all large enough integers θ, there is an (explicitly given) mapping $H\colon \{0,1\}^{\lfloor T^{1+\alpha}\rfloor} \to \{0,1\}^{\lfloor\theta^C\rfloor}$ with $T = \lfloor\theta\log^2\theta\rfloor$ such that the output bits of H are computable by monotone $\lceil 50(A+C)/\alpha\rceil$-DNF formulas and*

$$\deg_{\varepsilon - \frac{1}{T^A}}((f\circ H)|_{\leqslant T}) \geqslant T^{1-\frac{2}{3}\alpha} \tag{51}$$

for every $\varepsilon \in [0,1]$ *and every function* $f\colon \{0,1\}^{\lfloor \theta^C \rfloor} \to \{0,1\}$ *with* $\deg_\varepsilon(f|_{\leqslant\theta}) \geqslant \theta^{1-\alpha}$.

The proof of this corollary is available in the full version [56].

7 MAIN RESULT ON APPROXIMATE DEGREE

We will now establish our main results on the approximate degree of DNF formulas, stated in the introduction as Theorems 1.1 and 1.2 and Corollary 1.3. Our proof amounts to starting with the trivial one-variable formula x_1 and iteratively applying the hardness amplification of Corollary 6.4.

Theorem 7.1. *For every* $\delta \in (0,1]$ *and* $\Delta \geqslant 1$, *there is a constant* $c \geqslant 1$ *and an (explicitly given) family* $\{f_n\}_{n=1}^\infty$ *of functions* $f_n\colon \{0,1\}^n \to \{0,1\}$ *such that each* f_n *is computable by a monotone c-DNF formula and satisfies*

$$\deg_{\frac{1}{2}-\frac{1}{n^\Delta}}(f_n) \geqslant \frac{1}{c}\cdot n^{1-\delta}, \qquad n = 1,2,3,\ldots. \tag{52}$$

Proof. Let K be the smallest integer such that

$$\frac{1-(2/3)^K}{1+(2/3)^{K-1}} > 1-\delta. \tag{53}$$

Define

$$A = 2\Delta + 3. \tag{54}$$

Now, let $n \geqslant 1$ be any large enough integer. Define $T_0, T_1, T_2, \ldots, T_K$ recursively by $T_0 = \lfloor n/\log^{2K} n \rfloor$ and $T_i = \lfloor T_{i-1}\log^2 T_{i-1} \rfloor$ for $i \geqslant 1$. Thus,

$$T_i \leqslant \frac{n}{\log^{2(K-i)} n}, \qquad i = 0,1,2,\ldots,K, \tag{55}$$

$$T_i \sim \frac{n}{\log^{2(K-i)} n}, \qquad i = 0,1,2,\ldots,K, \tag{56}$$

where $\sim$ denotes equality up to lower-order terms in n. Provided that n is larger than a certain constant, inductive application of Theorem 6.4 gives functions

$$g_{n,i}\colon \{0,1\}^{\lfloor T_i^{1+(2/3)^{i-1}} \rfloor} \to \{0,1\}, \qquad i = 0,1,2,\ldots,K, \tag{57}$$

such that

$$\deg_{\frac{1}{2}-\frac{1}{T_0^A}-\frac{1}{T_1^A}-\cdots-\frac{1}{T_i^A}}(g_{n,i}|_{\leqslant T_i}) \geqslant T_i^{1-(2/3)^i}, \quad i = 0,1,2,\ldots,K, \tag{58}$$

and each $g_{n,i}$ is an explicitly constructed monotone c_i-DNF formula for some constant c_i independent of n. In more detail, the requirement (58) for $i = 0$ amounts to $\deg_{\frac{1}{2}-\frac{1}{T_0^A}}(g_{n,0}|_{\leqslant T_0}) \geqslant 1$ and is trivially satisfied by the "dictator" function $g_{n,0}(x) = x_1$, whereas for $i \geqslant 1$ the function $g_{n,i}$ is obtained constructively from $g_{n,i-1}$ by invoking Theorem 6.4 with

$$\begin{aligned}
\alpha &= \left(\frac{2}{3}\right)^{i-1},\\
C &= 1 + \left(\frac{2}{3}\right)^{i-2},\\
\theta &= T_{i-1},\\
f &= g_{n,i-1},\\
\varepsilon &= \frac{1}{2} - \frac{1}{T_0^A} - \frac{1}{T_1^A} - \cdots - \frac{1}{T_{i-1}^A}.
\end{aligned}$$

Specializing (55)–(58) to $i = K$, the function $g_{n,K}$ is a monotone c_K-DNF formula for some constant c_K independent of n, takes at most $N := n^{1+(2/3)^{K-1}}$ input variables, and has approximate degree

$$\begin{aligned}
\deg_{\frac{1}{2}-\frac{1}{N^{\Delta+1}}}(g_{n,K}) &\geqslant \deg_{\frac{1}{2}-\frac{1}{T_0^A}-\frac{1}{T_1^A}-\cdots-\frac{1}{T_K^A}}(g_{n,K})\\
&\geqslant \deg_{\frac{1}{2}-\frac{1}{T_0^A}-\frac{1}{T_1^A}-\cdots-\frac{1}{T_K^A}}(g_{n,K}|_{\leqslant T_K})\\
&= \Omega(n^{1-(2/3)^K})\\
&= \omega(N^{1-\delta}),
\end{aligned}$$

where the first and last steps hold for all large enough n due to (54) and (53), respectively. The desired function family $\{f_n\}_{n=1}^\infty$ can then be defined by setting $f_n = g_{\lfloor n^{1/(1+(2/3)^{K-1})} \rfloor, K}$ for all n larger than a certain constant n_0, and taking the remaining functions $f_1, f_2, \ldots, f_{n_0}$ to be the dictator function $x \mapsto x_1$. □

The remainder of this paper is available in the full version [56].

ACKNOWLEDGMENTS

This work was supported in part by NSF grant CCF-1814947. The author is thankful to Justin Thaler and Mark Bun for useful comments on an earlier version of this paper.

REFERENCES

[1] Scott Aaronson. 2005. Limitations of Quantum Advice and One-Way Communication. *Theory of Computing* 1, 1 (2005), 1–28. https://doi.org/10.4086/toc.2005.v001a001

[2] Scott Aaronson and Yaoyun Shi. 2004. Quantum lower bounds for the collision and the element distinctness problems. *J. ACM* 51, 4 (2004), 595–605. https://doi.org/10.1145/1008731.1008735

[3] Miklós Ajtai, Henryk Iwaniec, János Komlós, János Pintz, and Endre Szemerédi. 1990. Construction of a Thin Set with small Fourier Coefficients. *Bulletin of the London Mathematical Society* 22, 6 (1990), 583–590. https://doi.org/10.1112/blms/22.6.583

[4] Andris Ambainis. 2005. Polynomial Degree and Lower Bounds in Quantum Complexity: Collision and Element Distinctness with Small Range. *Theory of Computing* 1, 1 (2005), 37–46. https://doi.org/10.4086/toc.2005.v001a003

[5] Andris Ambainis, Andrew M. Childs, Ben Reichardt, Robert Špalek, and Shengyu Zhang. 2010. Any AND-OR Formula of Size N can be Evaluated in time $N^{1/2+o(1)}$ on a Quantum Computer. *SIAM J. Comput.* 39, 6 (2010), 2513–2530. https://doi.org/10.1137/080712167

[6] James Aspnes, Richard Beigel, Merrick L. Furst, and Steven Rudich. 1994. The Expressive Power of Voting Polynomials. *Combinatorica* 14, 2 (1994), 135–148. https://doi.org/10.1007/BF01215346

[7] László Babai, Peter Frankl, and Janos Simon. 1986. Complexity classes in communication complexity theory. In *Proceedings of the Twenty-Seventh Annual IEEE Symposium on Foundations of Computer Science* (FOCS). 337–347. https://doi.org/10.1109/SFCS.1986.15

[8] László Babai, Noam Nisan, and Mario Szegedy. 1992. Multiparty Protocols, Pseudorandom Generators for Logspace, and Time-Space Trade-Offs. *J. Comput. Syst. Sci.* 45, 2 (1992), 204–232. https://doi.org/10.1016/0022-0000(92)90047-M

[9] Ziv Bar-Yossef, T. S. Jayram, Ravi Kumar, and D. Sivakumar. 2004. An information statistics approach to data stream and communication complexity. *J. Comput. Syst. Sci.* 68, 4 (2004), 702–732. https://doi.org/10.1016/j.jcss.2003.11.006

[10] Robert Beals, Harry Buhrman, Richard Cleve, Michele Mosca, and R. de Wolf. 2001. Quantum lower bounds by polynomials. *J. ACM* 48, 4 (2001), 778–797. https://doi.org/10.1145/502090.502097

[11] Paul Beame and Trinh Huynh. 2012. Multiparty Communication Complexity and Threshold Circuit Size of AC^0. *SIAM J. Comput.* 41, 3 (2012), 484–518. https://doi.org/10.1137/100792779

[12] Paul Beame, Toniann Pitassi, Nathan Segerlind, and Avi Wigderson. 2006. A Strong Direct Product Theorem for Corruption and the Multiparty Communication Complexity of Disjointness. *Computational Complexity* 15, 4 (2006), 391–432. https://doi.org/10.1007/s00037-007-0220-2

[13] Richard Beigel, Nick Reingold, and Daniel A. Spielman. 1995. PP Is Closed under Intersection. *J. Comput. Syst. Sci.* 50, 2 (1995), 191–202. https://doi.org/10.1006/jcss.1995.1017

[14] Harry Buhrman, Richard Cleve, R. de Wolf, and Christof Zalka. 1999. Bounds for Small-Error and Zero-Error Quantum Algorithms. In *Proceedings of the Fortieth Annual IEEE Symposium on Foundations of Computer Science* (FOCS). 358–368. https://doi.org/10.1109/SFFCS.1999.814607

[15] Harry Buhrman and R. de Wolf. 2001. Communication complexity lower bounds by polynomials. In *Proceedings of the Sixteenth Annual IEEE Conference on Computational Complexity* (CCC). 120–130. https://doi.org/10.1109/CCC.2001.933879

[16] Harry Buhrman, Nikolai K. Vereshchagin, and R. de Wolf. 2007. On computation and communication with small bias. In *Proceedings of the Twenty-Second Annual IEEE Conference on Computational Complexity* (CCC). 24–32. https://doi.org/10.1109/CCC.2007.18

[17] Mark Bun, Robin Kothari, and Justin Thaler. 2020. The Polynomial Method Strikes Back: Tight Quantum Query Bounds via Dual Polynomials. *Theory Comput.* 16 (2020), 1–71. https://doi.org/10.4086/toc.2020.v016a010

[18] Mark Bun and Justin Thaler. 2015. Dual lower bounds for approximate degree and Markov–Bernstein inequalities. *Inf. Comput.* 243 (2015), 2–25. https://doi.org/10.1016/j.ic.2014.12.003

[19] Mark Bun and Justin Thaler. 2015. Hardness Amplification and the Approximate Degree of Constant-Depth Circuits. In *Proceedings of the Forty-Second International Colloquium on Automata, Languages and Programming* (ICALP). 268–280. https://doi.org/10.1007/978-3-662-47672-7_22

[20] Mark Bun and Justin Thaler. 2020. A Nearly Optimal Lower Bound on the Approximate Degree of AC^0. *SIAM J. Comput.* 49, 4 (2020). https://doi.org/10.1137/17M1161737

[21] Mark Bun and Justin Thaler. 2021. The Large-Error Approximate Degree of AC^0. *Theory of Computing* 17, 7 (2021), 1–46. https://doi.org/10.4086/toc.2021.v017a007

[22] Ashok K. Chandra, Merrick L. Furst, and Richard J. Lipton. 1983. Multi-Party Protocols. In *Proceedings of the Fifteenth Annual ACM Symposium on Theory of Computing* (STOC). 94–99. https://doi.org/10.1145/800061.808737

[23] Karthekeyan Chandrasekaran, Justin Thaler, Jonathan Ullman, and Andrew Wan. 2014. Faster private release of marginals on small databases. In *Proceedings of the Fifth Conference on Innovations in Theoretical Computer Science* (ITCS). 387–402. https://doi.org/10.1145/2554797.2554833

[24] Arkadev Chattopadhyay and Anil Ada. 2008. Multiparty Communication Complexity of Disjointness. In *Electronic Colloquium on Computational Complexity (ECCC)*. Report TR08-002.

[25] Jeff Kahn, Nathan Linial, and Alex Samorodnitsky. 1996. Inclusion-Exclusion: Exact and Approximate. *Combinatorica* 16, 4 (1996), 465–477. https://doi.org/10.1007/BF01271266

[26] Adam Tauman Kalai, Adam R. Klivans, Yishay Mansour, and Rocco A. Servedio. 2008. Agnostically learning halfspaces. *SIAM J. Comput.* 37, 6 (2008), 1777–1805. https://doi.org/10.1137/060649057

[27] Bala Kalyanasundaram and Georg Schnitger. 1992. The Probabilistic Communication Complexity of Set Intersection. *SIAM J. Discrete Math.* 5, 4 (1992), 545–557. https://doi.org/10.1137/0405044

[28] Hartmut Klauck, Robert Špalek, and R. de Wolf. 2007. Quantum and Classical Strong Direct Product Theorems and Optimal Time-Space Tradeoffs. *SIAM J. Comput.* 36, 5 (2007), 1472–1493. https://doi.org/10.1137/05063235X

[29] Adam R. Klivans, Ryan O'Donnell, and Rocco A. Servedio. 2004. Learning intersections and thresholds of halfspaces. *J. Comput. Syst. Sci.* 68, 4 (2004), 808–840. https://doi.org/10.1016/j.jcss.2003.11.002

[30] Adam R. Klivans and Rocco A. Servedio. 2004. Learning DNF in time $2^{\tilde{O}(n^{1/3})}$. *J. Comput. Syst. Sci.* 68, 2 (2004), 303–318. https://doi.org/10.1016/j.jcss.2003.07.007

[31] Matthias Krause and Pavel Pudlák. 1997. On the Computational Power of Depth-2 Circuits with Threshold and Modulo Gates. *Theor. Comput. Sci.* 174, 1–2 (1997), 137–156. https://doi.org/10.1016/S0304-3975(96)00019-9

[32] Matthias Krause and Pavel Pudlák. 1998. Computing Boolean functions by polynomials and threshold circuits. *Comput. Complex.* 7, 4 (1998), 346–370. https://doi.org/10.1007/s000370050015

[33] Troy Lee. 2009. A note on the sign degree of formulas. Available at http://arxiv.org/abs/0909.4607.

[34] Troy Lee and Adi Shraibman. 2009. Disjointness is Hard in the Multiparty Number-on-the-Forehead Model. *Computational Complexity* 18, 2 (2009), 309–336. https://doi.org/10.1007/s00037-009-0276-2

[35] Nathan Linial and Noam Nisan. 1990. Approximate Inclusion-Exclusion. *Combinatorica* 10, 4 (1990), 349–365. https://doi.org/10.1007/BF02128670

[36] Nikhil S. Mande, Justin Thaler, and Shuchen Zhu. 2020. Improved Approximate Degree Bounds for k-Distinctness. In *Proceedings of the 15th Conference on the Theory of Quantum Computation, Communication and Cryptography* (TQC), Vol. 158. 2:1–2:22. https://doi.org/10.4230/LIPIcs.TQC.2020.2

[37] Marvin L. Minsky and Seymour A. Papert. 1969. *Perceptrons: An Introduction to Computational Geometry.* MIT Press, Cambridge, Mass.

[38] Noam Nisan and Mario Szegedy. 1994. On the degree of Boolean functions as real polynomials. *Computational Complexity* 4 (1994), 301–313. https://doi.org/10.1007/BF01263419

[39] Ryan O'Donnell and Rocco A. Servedio. 2010. New degree bounds for polynomial threshold functions. *Combinatorica* 30, 3 (2010), 327–358. https://doi.org/10.1007/s00493-010-2173-3

[40] Ramamohan Paturi and Michael E. Saks. 1994. Approximating Threshold Circuits by Rational Functions. *Inf. Comput.* 112, 2 (1994), 257–272. https://doi.org/10.1006/inco.1994.1059

[41] Alexander A. Razborov. 1992. On the distributional complexity of disjointness. *Theor. Comput. Sci.* 106, 2 (1992), 385–390. https://doi.org/10.1016/0304-3975(92)90260-M

[42] Alexander A. Razborov. 2002. Quantum communication complexity of symmetric predicates. *Izvestiya of the Russian Academy of Sciences, Mathematics* 67 (2002), 145–159.

[43] Alexander A. Razborov and Alexander A. Sherstov. 2010. The sign-rank of AC^0. *SIAM J. Comput.* 39, 5 (2010), 1833–1855. https://doi.org/10.1137/080744037

[44] Alexander A. Sherstov. 2008. Communication Lower Bounds Using Dual Polynomials. *Bulletin of the EATCS* 95 (2008), 59–93.

[45] Alexander A. Sherstov. 2009. Approximate Inclusion-Exclusion for Arbitrary Symmetric Functions. *Computational Complexity* 18, 2 (2009), 219–247. https://doi.org/10.1007/s00037-009-0274-4

[46] Alexander A. Sherstov. 2009. Separating AC^0 from depth-2 majority circuits. *SIAM J. Comput.* 38, 6 (2009), 2113–2129. https://doi.org/10.1137/08071421X

[47] Alexander A. Sherstov. 2011. The pattern matrix method. *SIAM J. Comput.* 40, 6 (2011), 1969–2000. https://doi.org/10.1137/080733644

[48] Alexander A. Sherstov. 2012. Strong Direct Product Theorems for Quantum Communication and Query Complexity. *SIAM J. Comput.* 41, 5 (2012), 1122–1165. https://doi.org/10.1137/110842661

[49] Alexander A. Sherstov. 2013. The intersection of two halfspaces has high threshold degree. *SIAM J. Comput.* 42, 6 (2013), 2329–2374. https://doi.org/10.1137/100785260

[50] Alexander A. Sherstov. 2013. Making polynomials robust to noise. *Theory of Computing* 9 (2013), 593–615. https://doi.org/10.4086/toc.2013.v009a018

[51] Alexander A. Sherstov. 2014. Communication lower bounds using directional derivatives. *J. ACM* 61, 6 (2014), 1–71. https://doi.org/10.1145/2629334

[52] Alexander A. Sherstov. 2016. The multiparty communication complexity of set disjointness. *SIAM J. Comput.* 45, 4 (2016), 1450–1489. https://doi.org/10.1137/120891587

[53] Alexander A. Sherstov. 2018. Breaking the Minsky–Papert Barrier for Constant-Depth Circuits. *SIAM J. Comput.* 47, 5 (2018), 1809–1857. https://doi.org/10.1137/15M1015704

[54] Alexander A. Sherstov. 2018. The Power of Asymmetry in Constant-Depth Circuits. *SIAM J. Comput.* 47, 6 (2018), 2362–2434. https://doi.org/10.1137/16M1064477

[55] Alexander A. Sherstov. 2021. The hardest halfspace. *Comput. Complex.* 30, 11 (2021), 1–85. https://doi.org/10.1007/s00037-021-00211-4

[56] Alexander A. Sherstov. 2022. The Approximate Degree of DNF and CNF Formulas. In *Electronic Colloquium on Computational Complexity (ECCC)*.

[57] Alexander A. Sherstov and Pei Wu. 2019. Near-optimal lower bounds on the threshold degree and sign-rank of AC^0. In *Proceedings of the Fifty-First Annual ACM Symposium on Theory of Computing* (STOC). 401–412. https://doi.org/10.1145/3313276.3316408

[58] Kai-Yeung Siu, Vwani P. Roychowdhury, and Thomas Kailath. 1994. Rational approximation techniques for analysis of neural networks. *IEEE Transactions on Information Theory* 40, 2 (1994), 455–466. https://doi.org/10.1109/18.312168

[59] Jun Tarui and Tatsuie Tsukiji. 1999. Learning DNF by Approximating Inclusion-Exclusion Formulae. In *Proceedings of the Fourteenth Annual IEEE Conference on Computational Complexity* (CCC). 215–221. https://doi.org/10.1109/CCC.1999.766279

[60] Justin Thaler, Jonathan Ullman, and Salil P. Vadhan. 2012. Faster Algorithms for Privately Releasing Marginals. In *Proceedings of the Thirty-Ninth International Colloquium on Automata, Languages and Programming* (ICALP). 810–821. https://doi.org/10.1007/978-3-642-31594-7_68

[61] Robert Špalek. 2008. A Dual Polynomial for OR. Available at http://arxiv.org/abs/0803.4516.