New Degree Bounds for Polynomial Threshold Functions

[Extended Abstract]

Ryan O'Donnell *
MIT Department of Mathematics
Cambridge, MA
odonnell@theory.lcs.mit.edu

Rocco A. Servedio †
Computer Science Department, Columbia University
New York, NY
rocco@cs.columbia.edu

ABSTRACT

We give new upper and lower bounds on the degree of real multivariate polynomials which sign-represent Boolean functions. Our upper bounds for Boolean formulas yield the first known subexponential time learning algorithms for formulas of *superconstant* depth. Our lower bounds for constant-depth circuits and intersections of halfspaces are the first new degree lower bounds since 1968, improving results of Minsky and Papert. The lower bounds are proved *constructively*; we give explicit dual solutions to the necessary linear programs.

Categories and Subject Descriptors

G.0 [**Mathematics of Computing**]: General

General Terms

Theory

Keywords

polynomial threshold functions, lower bounds, ptfs, learning, degree, polynomials, duality, formulas

1. INTRODUCTION

Let f be a Boolean function $f : \{-1,1\}^n \to \{-1,1\}$ and let p be a degree d multilinear polynomial in n variables with real coefficients. If the sign of $p(x)$ equals $f(x)$ for every $x \in \{-1,1\}^n$, then we say that f is computed by a *polynomial threshold function* of degree d; equivalently we say that p *sign-represents* f.

*Supported by NSF grant 99-12342.

†Supported by an NSF Mathematical Sciences Postdoctoral Research Fellowship and by NSF grant CCR-98-77049. Most of this research was conducted when the third author was a postdoc at Harvard University

Polynomial threshold functions are an interesting and natural representation for Boolean functions which have many applications in complexity theory and learning theory, see, e.g., [2, 5, 6, 4, 22, 14, 13]. Positive results showing that functions have low degree polynomial threshold functions can be used to obtain efficient learning algorithms via linear programming; see, e.g., [14, 13]. Negative results showing that a function requires threshold polynomials of large degree and/or large coefficients can be used to obtain oracles separating PP from smaller classes; see, e.g., [5, 25].

In this paper we give new upper and lower bounds on polynomial threshold function degree for several interesting and natural classes of functions which have been previously considered (but not resolved) in the literature. It seems likely that both the upper and lower bound techniques we use will prove useful for broader classes of functions.

1.1 Previous work

The study of polynomial threshold functions began with Minsky and Papert in their 1968 book on perceptrons [18]. Minsky and Papert gave three lower bounds on the degree of polynomial threshold functions:

- Any polynomial threshold function which computes parity on n variables must have degree at least n. This result has since been reproved many times, see, e.g., [2, 7].
- Any polynomial threshold function which computes a particular linear-size CNF formula, the "one-in-a-box" function on n variables, must have degree $\Omega(n^{1/3})$. By Boolean duality this lower bound also holds for a corresponding DNF formula.
- Any polynomial threshold function which computes the AND of two majorities each on n variables must have degree $\omega(1)$.

Despite the fact that many researchers in learning theory and complexity theory have studied polynomial threshold functions, relatively little progress has been made on improving these lower bounds since 1968. In particular, Vereshchagin [25] has a lower bound for a promise-problem extension of one-in-a-box and Beigel [5] has a lower bound for a certain linear threshold function; however, both of these show degree lower bounds for polynomial threshold functions only under the added assumption that the polynomials have small integer coefficients. (Krause and Pudlak

[15] have given lower bounds on the number of nonzero coefficients which must be present in any polynomial threshold function for a particular depth-3 Boolean circuit, but their lower bounds are not strong enough to imply new lower bounds on polynomial threshold function degree.) More progress has been made on upper bounds; Beigel, Reingold, and Spielman [6] proved that there is a polynomial threshold function of degree $O(\log n)$ which computes the AND of two n-bit majorities. More recently, Klivans and Servedio [14] showed that any polynomial-size DNF formula (equivalently, CNF formula) has a polynomial threshold function of degree $O(n^{1/3}\log n)$, and Klivans *et al.* [13] showed that any Boolean function of a polylogarithmic number of halfspaces with quasipolynomially-bounded weights has a polynomial threshold function of polylogarithmic degree.

1.2 Our results

We give new upper and lower bounds on polynomial threshold functions for several interesting and natural classes of functions. Our main results are:

- We prove that any Boolean formula of depth d and size s is computed by a polynomial threshold function of degree $\sqrt{s}(\log s)^{O(d)}$. This gives us the first known upper bound for Boolean formulas of superconstant depth. In particular, any Boolean formula of size $o(n^2)$ and depth $o(\frac{\log n}{\log\log n})$ has a polynomial threshold of nontrivial (sublinear) degree. We use our upper bound to provide the first known subexponential learning algorithm for such formulas. Note that since parity on $\sqrt{s}$ variables can be computed by a formula of size s, the best possible degree upper bound which depends only on s is $\sqrt{s}$.
- We give an $\Omega(\frac{\log n}{\log\log n})$ lower bound on the degree of any polynomial threshold function which computes the AND of two n-bit majorities. Equivalently, this lower bound holds for the degree of any bivariate real polynomial $p(x, y)$ which is positive on the lattice points in the upper-right quadrant with coordinates bounded by n, and is negative on the lattice points in the other three quadrants with coordinates bounded in magnitude by n. This result (and our next) is the first new unconditional lower bound for polynomial threshold degree since 1968; it improves on Minsky and Papert's lower bound of $\omega(1)$ and nearly matches the $O(\log n)$ upper bound of Beigel, Reingold and Spielman.
- We prove an "XOR lemma" for polynomial threshold function degree and use this lemma to obtain an $\Omega(n^{1/3}\log^{2d/3} n)$ lower bound on the degree of an explicit Boolean circuit of polynomial size and depth $d+2$. This is the first improvement on Minsky and Papert's $\Omega(n^{1/3})$ lower bound for any constant-depth circuit.

1.3 Our techniques

Perhaps surprisingly, our lower bounds are achieved constructively. The question of whether a given function has a polynomial threshold function of degree d can be formulated as the feasibility question for a certain linear program. By duality, we can show the linear program is infeasible — and hence the function has polynomial threshold degree exceeding d — by showing that the dual linear program is feasible. We construct explicit dual solutions. (Interestingly, Vereschagin's lower bound [25] involves showing that a certain linear program *is* feasible by explicitly demonstrating the infeasibility of the dual.)

Our upper bounds build on ideas from [14, 13] and use tools from real approximation theory.

1.4 Organization

Section 2 gives preliminaries on polynomial threshold functions and describes the duality technique we use for our lower bounds. In Section 3 we give our upper bounds for Boolean formulas and the application to learning. In Section 4 we prove our XOR lemma for polynomial threshold functions using the duality technique, and use this lemma to obtain new lower bounds for constant depth circuits. In Section 5 we apply the lower bound technique to prove our $\Omega(\frac{\log n}{\log\log n})$ lower bound for the AND of two majorities. Finally, in Section 6 we make some conjectures and sketch possible future applications of our upper and lower bound techniques.

2. PRELIMINARIES

We make the following standard definitions of sign-representing polynomials (see [2]). Let $f : \{-1,1\}^n \to \{-1,1\}$ be a Boolean function. Let $p : \{-1,1\}^n \to \mathbf{R}$ be a multilinear polynomial of degree at most n which is not identically 0. Define the *support* of p to be the set of monomials $\mathcal{S} \subseteq 2^{[n]}$ on which p has nonzero coefficients.

Definition 1. *We say that p* weakly *(sign-)represents f if $f(x) = sgn(p(x))$ for all x such that $p(x) \neq 0$. If $p(x) \neq 0$ for every $x \in \{-1,1\}$ we say that p* strongly *(sign-) represents (or simply* (sign-)represents*) f. We let $thr(f)$ denote the minimum degree of a polynomial strongly representing f, and $thr^w(f)$ denote the minimum degree of a polynomial weakly representing f.*

On occasion we will view the domain of f as $\{0,1\}^n$ instead of $\{-1,1\}^n$; it is easy to see that this does not change the degree of any sign-representing polynomial.

There is a sense in which sign-representing polynomials are equivalent to distributions over $\{-1,1\}^n$.

Definition 2. *We call a map $w : \{-1,1\}^n \to \mathbf{R}^{\geq 0}$ which is not identically 0 a* distribution*. The set of points $\{x : w(x) \neq 0\}$ is called the* support *of w. If the support of w is all of $\{-1,1\}^n$ we call w a* total *distribution. If $\sum_{x\in\{-1,1\}^n} w(x) = 1$ we call w a* probability distribution*. If w is a map $w : \{-1,1\}^n \to \mathbf{R}$, not identically 0, which takes on at least one negative value, we call w an* improper *distribution. Given a monomial x_S, $S \subseteq [n]$, we say that the* correlation of x_S with f under w *is $\mathbf{E}_w[f(x)x_S] := \sum_{x\in\{-1,1\}^n} f(x)x_S w(x)$. (Here x_S denotes $\prod_{i\in S} x_i$.)*

Notice that multilinear polynomials of degree at most n are given by vectors of 2^n real coefficients. Improper distributions too are given by vectors of 2^n real weights. The connection between sign-representations and distributions is this:

Proposition 3. *For any Boolean function $f : \{-1,1\}^n \to \{-1,1\}$, there is an (orthogonal) linear bijection A_f between weak representations of f and distributions. If p and w are in correspondence then $p(x) = |w(x)|$ and hence strong representations are in bijective correspondence with total distributions. Further, the S coefficient of p is proportional to* *the correlation of x_S with f under w. Hence p is supported on $\mathcal{S}$ iff f has zero correlation with x_S under w for every monomial $S \notin \mathcal{S}$. (Finally, sign-representations which make mistakes correspond to improper distributions.)*

PROOF. The bijection maps column vectors of polynomial coefficients indexed by monomials $S \subseteq [n]$ to column vectors of distribution weights indexed by points $x \in \{-1,1\}^n$. The map is given by the matrix A_f with rows indexed by $x \in \{-1,1\}^n$ and columns indexed by monomials $S \subseteq [n]$; the entry $A_f[x, S]$ is equal to $f(x)x_S$. This matrix is orthogonal, being a Hadamard matrix. $\square$

Our main tool for proving polynomial threshold degree lower bounds is the following so-called "Theorem of the Alternative." It can be proved immediately using linear programming duality, as was essentially done by Aspnes *et al.* in [2]; a completely different proof based on the distribution perspective can be given by combining the "Discriminator Lemma" of [11] with the learning-theoretic technique of boosting, see [9, 10].

THEOREM 4. *Let $f : \{-1,1\}^n \to \{-1,1\}$ be a Boolean function. Let $\mathcal{S} \subseteq 2^{[n]}$ be any set of monomials. Then exactly one of the following holds:*

- *f has a strong representation with support in $\mathcal{S}$; or,*
- *f has a weak representation with support in $2^{[n]} \setminus \mathcal{S}$.*

Given the equivalence of sign-representations and distributions, there are three other ways of restating Theorem 4. We will need one more:

THEOREM 5. *Let $f : \{-1,1\}^n \to \{-1,1\}$ be a Boolean function. Let $\mathcal{S} \subseteq 2^{[n]}$ be any set of monomials. Then exactly one of the following holds:*

- *f has a strong representation with support in $\mathcal{S}$; or,*
- *there is a distribution on $\{-1,1\}^n$ under which f has zero correlation to every monomial in $\mathcal{S}$.*

3. UPPER BOUNDS FOR BOOLEAN FORMULAS

In this section we consider Boolean formulas composed of NOT gates and unbounded fan-in AND and OR gates. The *depth* of a formula is the length of the longest path from the root to any leaf, and the *size* is the number of occurrences of variables.

We will also consider variants of polynomial threshold functions in which the polynomial is subject to a stricter requirement than just sign-representing f. Following Nisan and Szegedy [20], we write $\widetilde{\deg}(f)$ to denote the minimum degree of any polynomial which approximates f to within $1/3$ on all inputs; i.e., such a polynomial $p(x)$ must satisfy:

$$\forall x \in \{0,1\}^n \quad |f(x) - p(x)| \leq \frac{1}{3}.$$

Clearly we have $\widetilde{\deg}(f) \geq \mathrm{thr}(f)$ for all f. We write $|p - f|_\infty$ to denote $\max_{x \in \{0,1\}^n} |p(x) - f(x)|$. Thus if $|p - f|_\infty \leq \frac{1}{3}$ we have $\deg(p) \geq \widetilde{\deg}(f) \geq \mathrm{thr}(f)$.

We prove two similar theorems bounding the polynomial threshold degree of Boolean formulas:

THEOREM 6. *Let f be computed by a Boolean formula of depth d and size s. Then there is a polynomial $p(x_1, \ldots, x_n)$ of degree at most $2^{O(d)}(\log s)^{5d/2}\sqrt{s}$ such that $|p - f|_\infty \leq \frac{1}{s}$.*

THEOREM 7. *Let f be computed by a Boolean formula of depth d and size s. Then there is a polynomial $p(x_1, \ldots, x_n)$ of degree at most $2^{O(d)}(\log s)^{5d} s^{\frac{1}{2} - \frac{1}{2^{d+1}-2}}$ such that $sgn(p(x)) = f(x)$.*

The proof technique in both cases is to first manipulate the formula to get a more structured form, and then to apply real approximating functions (Chebyshev polynomials, the rational functions of [6]) at each gate.

Some preliminary notes: Throughout this section we let 0 represent FALSE and 1 represent TRUE, and thus we view Boolean functions as mappings from $\{0,1\}^n$ to $\{0,1\}$. Without loss of generality we may assume that our formulas contain no NOT gates; i.e., they consist only of AND and OR gates. This is because any negations in a formula F can be pushed to the leaves using DeMorgan's laws with no increase in size or depth. Once all negations are at the leaves we can replace each negated variable $\neg x_i$ with a variable y_i to obtain a formula F' which has no negations. Given a polynomial which sign-represents or approximates F', we can obtain a corresponding polynomial for F by replacing each y_i with $1 - x_i$, and this will not increase the degree.

3.1 Proof of Theorem 6

Henceforth the variables $c_1, c_2, \ldots$ refer to fixed universal constants.

Theorem 6 *Let f be computed by a Boolean formula of depth d and size s. Then there is a polynomial $p(x_1, \ldots, x_n)$ of degree at most $c_1^d(\log s)^{5d/2}\sqrt{s}$ such that $|p - f|_\infty \leq \frac{1}{s}$.*

We will use the following lemma:

LEMMA 8. *Let $f = \bigwedge_{i=1}^{\ell} f_i$ be a Boolean formula where $\ell \geq 2$. For $1 \leq i \leq \ell$ let p_i be a polynomial with $\deg(p_i) \leq r$ such that $|p_i - f_i|_\infty \leq \epsilon$, where $0 < \epsilon < \frac{1}{8\ell}$. Then there is a polynomial p with $\deg(p) \leq (4\sqrt{\ell}\log\frac{1}{\epsilon})r$ such that $|p - f|_\infty \leq (c_2 \ell \log \frac{1}{\epsilon})\epsilon$.*

PROOF. The following convention will be useful: for P a polynomial we write "$P(x) \in_f ([a,b],[c,d])$" as shorthand for

"$\forall x \in \{0,1\}^n$: if $f(x) = 0$ then $P(x) \in [a,b]$
and if $f(x) = 1$ then $P(x) \in [c,d]$."

Thus by assumption we have $p_i(x) \in_{f_i} ([-\epsilon, \epsilon], [1-\epsilon, 1+\epsilon])$ for each i.

Let $P(x)$ denote $p_1(x) + \cdots + p_\ell(x) + \ell\epsilon$. It is easy to verify that we have

$$P(x) \in_f ([0, \ell - 1 + 2\ell\epsilon], [\ell, \ell + 2\ell\epsilon]).$$

Let $Q(x)$ denote $P(x)/(\ell - 1 + 2\ell\epsilon)$. We then have

$$Q(x) \in_f ([0,1], [1 + \frac{1 - 2\ell\epsilon}{\ell - 1 + 2\ell\epsilon}, 1 + \frac{1}{\ell - 1 + 2\ell\epsilon}]).$$

Let $k = \frac{1-2\ell\epsilon}{\ell-1+2\ell\epsilon}$. We can rewrite and say $Q(x) \in_f ([0,1], [1+k, 1+k+\frac{2\ell\epsilon}{\ell-1+2\ell\epsilon}])$. Since $\frac{2\ell\epsilon}{\ell-1+2\ell\epsilon} < \frac{2\ell\epsilon}{\ell-1} \leq 4\epsilon$ we have $Q(x) \in_f ([0,1], [1+k, 1+k+4\epsilon])$.

Recall that the Chebyshev polynomial of the first kind $C_d(t)$ is a univariate polynomial of degree d. The following fact is straightforward to prove; we omit the proof from this extended abstract.

FACT 9. *For all $d \geq 1$ we have:*

1. $C_d(t) \in [-1,1]$ *for* $t \in [0,1]$.
2. *Let t_d denote $C_{\lceil\sqrt{d}\rceil}(1+1/d)$. Then $t_d > 2$.*
3. *For all $0 < \tau < \frac{1}{d}$ we have $C_{\lceil\sqrt{d}\rceil}(1+1/d+\tau) \in [t_d, t_d + 26d\tau]$.*

Let $R(x)$ denote $C_{\lceil k^{-1/2}\rceil}(Q(x))$. Since $4\epsilon < \frac{1}{2\ell} < k$, by parts 1 and 3 of Fact 9 we have that $R(x) \in_f ([-1,1],[t_k, t_k + \frac{104\epsilon}{k}])$. Let $S(x)$ denote $(\frac{1}{t_k}R(x))^{\lceil \log \frac{1}{\epsilon}\rceil}$. Using part 2 of Fact 9 we find that $S(x) \in_f ([-\epsilon,\epsilon],[1,(1+\frac{104\epsilon}{t_k k})^{\lceil \log \frac{1}{\epsilon}\rceil}])$. We now use the fact that $\alpha^r \leq 1-(1-\alpha)r$ for all $0 \leq r \leq 1$ and $\alpha > 0$ (this can be proved using a simple convexity argument). We thus find that

$$\left(1+\frac{104\epsilon}{t_k k}\right)^{\lceil \log \frac{1}{\epsilon}\rceil} \leq 1 + \frac{104\epsilon\lceil \log \frac{1}{\epsilon}\rceil}{t_k k} \leq 1 + \frac{208 \log \frac{1}{\epsilon}}{t_k k}\epsilon.$$

Using our bounds on t_k and k, this is at most $1+(c_2\ell\log\frac{1}{\epsilon})\epsilon$ as desired.

It remains only to bound $\deg(S)$. From our construction it is clear that $\deg(S) \leq r \cdot \lceil k^{-1/2}\rceil \cdot \lceil \log \frac{1}{\epsilon}\rceil$. We have that $\lceil k^{-1/2}\rceil \leq \lceil\sqrt{2\ell}\rceil \leq 2\sqrt{\ell}$ and $\lceil \log\frac{1}{\epsilon}\rceil < 2\log\frac{1}{\epsilon}$. Thus $\deg(S) \leq 4r\sqrt{\ell}\log\frac{1}{\epsilon}$ and the lemma is proved. □

It is easy to see that an identical result holds if $f = \bigvee_{i=1}^{\ell} f_i$, i.e. f's top-level gate is an OR instead of an AND. The following lemma is now easy to establish:

LEMMA 10. *Let f be computed by a Boolean formula F of depth d and size s. Suppose that for any path from the root of F to a leaf, the product of the fanins of the gates on the path is at most t. Then there is a polynomial p with $\deg(p) \leq (c_3 \log s)^d\sqrt{t}$ such that $|p-f|_\infty \leq \frac{1}{s}$.*

PROOF. Note first that for any Boolean formula of size s, there is a multilinear interpolating polynomial which computes the formula exactly and is of degree at most s. Consequently if $(c_3\log s)^d\sqrt{t} \geq s$ the lemma is trivially true, so we assume that $(c_3 \log s)^d\sqrt{t} < s$.

Consider the formula F. Each leaf contains some variable x_i, so clearly there is a degree-1 polynomial which exactly computes the function at each leaf. Now apply Lemma 8 successively to every gate in F, going up from the leaves to the root. At each leaf we may take ϵ in Lemma 8 to be any positive value; we take $\epsilon = \frac{1}{s^3}$. Each time we go up through a gate of fanin ℓ the value of ϵ which we may use in Lemma 8 is multiplied by at most $c_2\ell\log(s^3) = c_3\ell\log s$. An easy induction on the depth of F shows that at the root we obtain a polynomial p such that

$$\deg(p) \leq (4\log(s^3))^d\sqrt{t} < (c_3 \log s)^d \sqrt{t}$$

and

$$|p-f|_\infty \leq \frac{1}{s^3}\cdot(c_3\log s)^d t < \frac{1}{s^3}\cdot s^2 = \frac{1}{s}$$

as desired. □

With Lemmas 8 and 10 in hand, in order to prove Theorem 6 it suffices to bound the product of the fanins on any path from the root to a leaf. In an arbitrary formula this product can be quite large; it is easy to construct a formula of size s and depth d in which there is a path composed of d gates each of fanin $\frac{s}{d}$. Thus in general this product can be as large as $(\frac{s}{d})^d$; however we can remedy this situation as described below.

LEMMA 11. *Let F be a formula of size s and depth d. There is a formula G of size s and depth $2d$ computing the same function as F such that the product of the fanins on any root-to-leaf path in G is at most $(4\log s)^d s$.*

PROOF. We prove the following slightly stronger statement: for any formula F of size s and depth d, there is a formula G of size s and depth $2d$ computing F such that the product of the fanins on any root-to-leaf path in G is at most $(2\lceil\log s\rceil)^d s$. The lemma follows since $2\log s \geq \lceil \log s\rceil$ for all s.

The proof is by induction on d. For $d = 0$ the formula must be a single variable so $s = 1$ and the claim is trivially true. Suppose without loss of generality that $F = \bigwedge_{i=1}^{\ell} F_i$ where $\ell \geq 2$, each F_i has depth at most $d-1$, and the sum of the sizes of $F_1,\ldots,F_\ell$ is s. Let $|F_i|$ denote the size of F_i. We partition the formulas $F_1,\ldots,F_\ell$ into disjoint classes $C_1,\ldots,C_{\lceil\log s\rceil}$ where the class C_j contains exactly those F_i such that $2^{j-1} \leq |F_i| < 2^j$. By the induction hypothesis each formula $F_i \in C_j$ has an equivalent formula G_i of size $|F_i|$ and depth at most $2d-2$ such that the product of the fanins along any root-to-leaf path in G_i is at most $(2\lceil\log s\rceil)^{d-1}|F_i| < 2^{d+j-1}\lceil\log s\rceil^{d-1}$. Let $G = \bigwedge_{j=1}^{\lceil\log s\rceil} H_j$ where the formula H_j is defined as $H_j = \bigwedge_{i:F_i\in C_j} G_i$.

To see that this works, first observe that each C_j contains at most $s/2^{j-1}$ formulas F_i. Thus the fanin at the root of H_j is at most $s/2^{j-1}$, and hence the product of the fanins along any path in H_j is at most $2^d s\lceil\log s\rceil^{d-1}$. Thus the product of the fanins along any path in G is at most $(2\lceil\log s\rceil)^d s$ as desired and the lemma is proved. □

Theorem 6 follows from combining Lemmas 10 and 11.

3.2 Proof of Theorem 7

Recall Theorem 7:

Theorem 7 *Let f be computed by a Boolean formula of depth d and size s. Then there is a polynomial $p(x_1,\ldots,x_n)$ of degree at most $c_4^d(\log s)^{5d}s^{\frac{1}{2}-\frac{1}{2^{d+1}-2}}$ such that $sgn(p(x)) = f(x)$.*

This bound is asymptotically superior to the one in Theorem 6, for any constant d. However, Theorem 7 only produces a polynomial which sign-represents the formula's values, not one that closely approximates them. The proof of Theorem 7 builds on the proof of Theorem 6 and uses the rational functions constructed by Beigel *et al.* [6] for approximating the sgn function. We omit the proof from this extended abstract.

3.3 Discussion

In earlier work Klivans and Servedio [14] showed that any Boolean formula of constant depth d and size s has a polynomial threshold function of degree $\tilde{O}(s^{1-\frac{1}{3\cdot 2^{d-3}}})$. For even moderately large constant values of d, this bound is not far from the trivial upper bound of s. In contrast, our new bounds are considerably stronger. Theorem 7 gives an

$o(s^{1/2})$ bound for some $d = \Omega(\log\log s)$, and Theorems 6 and 7 both give a bound of $O(s^{1/2+\epsilon})$ for any $d = o(\frac{\log s}{\log\log s})$. To our knowledge Theorems 6 and 7 are the first nontrivial upper bounds on polynomial threshold function degree for formulas of superconstant depth.

In other earlier work, Buhrman, Cleve and Wigderson [3] gave an $O(s^{1/2}\log^{d-1}(s))$ upper bound on the degree of polynomials that approximate (in the sense of Theorem 6) certain Boolean formulas of size s and depth d. Their bound applies only to "balanced formulas," namely to formulas in which all of the gates at any given depth have the same fanin (the fanin can be different for gates at different depths). Our Theorem 6 thus generalizes their bound on the degree of approximating polynomials to a substantially broader class of formulas. The motivation for the upper bounds of Buhrman *et al.* was to obtain upper bounds on the bounded-error quantum complexity of predicates corresponding to balanced formulas. Our Theorem 6 immediately implies corresponding upper bounds on the bounded-error quantum complexity of a broader class of predicates corresponding to general formulas.[1]

3.4 Learning Boolean formulas of superconstant depth in subexponential time

We close this section by describing some consequences of our results in computational learning theory. It is known (see [14, 13]) that if a class C of Boolean functions has $\text{thr}(f) \le r$ for all $f \in C$, then C can be learned in time $n^{O(r)}$ in either of two well-studied and demanding learning models, the Probably Approximately Correct (PAC) model of learning from random examples [12, 24] and the online model of learning from adversarially generated examples [1, 16]. Thus our polynomial threshold function upper bounds from Theorems 6 and 7 immediately give a range of new subexponential time learning results for various classes of Boolean formulas. For example, we immediately obtain:

THEOREM 12. *The class of linear-size Boolean formulas of depth $o(\frac{\log n}{\log\log n})$ can be learned in time $2^{n^{1/2+\epsilon}}$ for all $\epsilon > 0$.*

This is the first subexponential time learning algorithm for linear size formulas of superconstant depth.

We emphasize that the PAC learning results which follow from our upper bounds hold for the general PAC model of learning from random examples which are drawn from an arbitrary probability distribution over $\{0,1\}^n$. This is in contrast with many results in learning theory (such as the quasipolynomial time algorithm of Linial *et al.* [17] for learning constant-depth circuits) which require the random examples to be drawn from the uniform distribution on $\{0,1\}^n$.

4. AN XOR LEMMA FOR PTF DEGREE

Let f be any Boolean function $\{-1,1\}^n \to \{-1,1\}$ defined on variables $x_1,\dots,x_n$ and let g be any Boolean function $\{-1,1\}^n \to \{-1,1\}$ defined on variables $y_1,\dots,y_n$. Let $f \oplus g$ denote the XOR (parity) of f and g. We will prove the following "XOR lemma:"

THEOREM 13. *Let f and g be Boolean functions on disjoint sets of variables. Then $thr(f \oplus g) = thr(f) + thr(g)$.*

We note that Theorem 13 is similar in spirit (though incomparable) to a recent result of Sieling [23] which shows that $DT(f \oplus g) = DT(f)\cdot DT(g)$, where $DT(f)$ is the minimum decision tree size of f.

Proof of Theorem 13: The upper bound is easy; if $p_f(x)$ is a strong sign-representation of f of degree $\text{thr}(f)$ and $p_g(y)$ is a strong sign-representation of g with degree $\text{thr}(g)$ then $p_f(x)p_g(y)$ is easily seen to be a strong sign-representation of $f \oplus g$, and $\deg(p_f(x)p_g(y)) = \text{thr}(f) + \text{thr}(g)$.

For the lower bound, since f has no strong representation on the set of monomials of degree strictly less than $\text{thr}(f)$, Theorem 4 tells us that f has a weak representation $q_f(x)$ supported on the monomials x_S with $|S| \ge \text{thr}(f)$. Similarly, g has a weak representation $q_g(y)$ supported on the monomials y_T with $|T| \ge \text{thr}(g)$. Now $q_f(x)q_g(y)$ is a weak representation of $f \oplus g$; in particular, it is not identically zero because there is at least one x for which $q_f(x) \ne 0$ and at least one y for which $q_g(y) \ne 0$, so $q_f(x)q_g(y) \ne 0$ for these inputs. Note that $q_f(x)q_g(y)$ is supported on the set of monomials which have degree at least $\text{thr}(f)$ in x and at least $\text{thr}(g)$ in y. Applying Theorem 4 again we conclude that any strong representation for $f \oplus g$ must use some monomial with degree at least $\text{thr}(f)$ in x and at least $\text{thr}(g)$ in y; this is more than sufficient to prove that $\text{thr}(f \oplus g) \ge \text{thr}(f) + \text{thr}(g)$. □

For f a Boolean function let $\oplus_k f$ denote the XOR of k copies of f on disjoint sets of variables. From Theorem 13 we obtain:

COROLLARY 14. *$thr(\oplus_k f) = k \cdot thr(f)$.*

This corollary thus includes Minsky and Papert's lower bound of n for the parity function as a special case.

Corollary 14 also yields the following lower bound for constant depth circuits:

THEOREM 15. *For all $d \ge 1$ there is an AND/OR/NOT circuit C of depth $d+2$ and size $poly(n)$ which has polynomial threshold function degree $\Omega(n^{1/3}(\log n)^{2d/3})$.*

PROOF. The circuit C computes the parity of $(\log n)^d$ disjoint copies of Minsky and Papert's "one-in-a-box" function, where each one-in-a-box function is defined on $n/(\log n)^d$ variables. It is well known that for any constant d, parity on $(\log n)^d$ variables can be computed by an AND/OR/NOT circuit of depth $d+1$ and size $\text{poly}(n)$. Since the one-in-a-box function on $n/(\log n)^d$ variables is a depth-2 circuit of size $O(n/(\log n)^d)$, by substituting the appropriate one-in-a-box function for each input to the parity we see that C is a circuit of $\text{poly}(n)$ size and depth $d+2$ (we save one on depth by collapsing gates of the same kind on the next to bottom layer). By Minsky and Papert's lower bound, we know that any polynomial threshold function for one-in-a-box on $n/(\log n)^d$ variables must have degree $\Omega((n/(\log n)^d)^{1/3})$. Consequently Corollary 14 implies that $\text{thr}(C) = \Omega(n^{1/3}(\log n)^{2d/3})$ and the theorem is proved. □

[1] We note in passing that an easy argument shows that any balanced formula of size s has a polynomial threshold function approximator of degree at most $s^{1/2}$; the proof is based on the observation that either the product of the odd-depth fanins or the even-depth fanins in any balanced formula must be at most $s^{1/2}$.

In fact, we can actually give an alternate proof of Minsky and Papert's lower bound for one-in-a-box by using our lower bound of technique of applying the Theorem of the Alternative (Theorem 5) and constructing the necessary distribution explicitly. The proof will appear in the final version of this extended abstract.

Theorem 15 is of interest since it gives the first $\omega(n^{1/3})$ lower bound for any function in AC^0. We note that Theorem 15 also shows that the $n^{1/3}\log n$ upper bound of Klivans and Servedio for depth-2 AC^0 circuits does not hold for depth-4 AC^0.

5. A LOWER BOUND FOR THE AND OF TWO MAJORITIES

Let n be odd, and let $\mathsf{AND\text{-}MAJ}_n : \{-1,1\}^n \times \{-1,1\}^n \to \{-1,1\}$ be the function which on input (x,y), $x,y \in \{-1,1\}^n$, outputs 1 if both $\mathsf{MAJ}_n(x) = 1$ and $\mathsf{MAJ}_n(y) = 1$. Here MAJ_n is the majority function on n bits, $x \mapsto \mathrm{sgn}(\sum_{i=1}^n x_i)$. In this section we show that $\mathrm{thr}(\mathsf{AND\text{-}MAJ}_n) = \Omega(\frac{\log n}{\log\log n})$, improving on the $\omega(1)$ lower bound of Minsky and Papert. Note that $O(\log n)$ is an upper bound, by Beigel, Reingold, and Spielman [6].

We begin by applying a simple symmetrization due to Minsky and Papert. Suppose p is a polynomial threshold function for $\mathsf{AND\text{-}MAJ}_n$ where n is odd. Let $\mathbf{Z}_n^{\mathrm{odd}}$ denote the set $\{-n, -(n-2), \ldots, -1, 1, \ldots, n-2, n\} \subseteq \mathbf{Z}$. Let $\mathsf{AND\text{-}sgn}_n : \mathbf{Z}_n^{\mathrm{odd}} \times \mathbf{Z}_n^{\mathrm{odd}} \to \{-1,1\}$ be the function which on input (x,y) is 1 iff $x > 0$ and $y > 0$. Minsky and Papert show:

CLAIM 16. *There exists a polynomial threshold function for $\mathsf{AND\text{-}MAJ}_n$ of degree d if and only if there exists a bivariate polynomial of degree d which sign-represents AND-sgn_n.*

It follows that if we prove a lower bound on the degree of a bivariate polynomial which sign-represents $\mathsf{AND\text{-}sgn}_n$, we get a lower bound on $\mathrm{thr}(\mathsf{AND\text{-}MAJ}_n)$. Following Theorem 5, we shall show that there is a probability distribution over $\mathbf{Z}_n^{\mathrm{odd}} \times \mathbf{Z}_n^{\mathrm{odd}}$ under which every bivariate monomial of degree at most $d = \Omega(\frac{\log n}{\log\log n})$ has zero correlation with $\mathsf{AND\text{-}sgn}_n$. To see that this is enough, suppose that $\tilde{q}$ is a bivariate polynomial of degree d sign-representing $\mathsf{AND\text{-}sgn}_n$ and w is a probability distribution over $\mathbf{Z}_n^{\mathrm{odd}} \times \mathbf{Z}_n^{\mathrm{odd}}$ with the stated property. Then on one hand,

$$\mathbf{E}_w[\mathsf{AND\text{-}sgn}_n(x,y)\tilde{q}(x,y)] = 0,$$

by linearity of expectation, since each monomial in $\tilde{q}$ has zero correlation with $\mathsf{AND\text{-}sgn}_n$ under w. But on the other hand, since $\tilde{q}$ strongly sign-represents $\mathsf{AND\text{-}sgn}_n$, $\mathsf{AND\text{-}sgn}_n(x,y)\tilde{q}(x,y) > 0$ for all (x,y), hence,

$$\mathbf{E}_w[\mathsf{AND\text{-}sgn}_n(x,y)\tilde{q}(x,y)] > 0,$$

which gives a contradiction.

The problem is now set up to our satisfaction. Fix an integer d. We shall try to find a support (set of points) $\mathcal{Z} \subset \mathbf{Z}^{\mathrm{odd}} \times \mathbf{Z}^{\mathrm{odd}}$ and a probability distribution $\mathbf{w}$ over these points such that $\mathsf{AND\text{-}sgn}_n$ has zero correlation under $\mathbf{w}$ with every monomial $x^i y^j$ of total degree at most d. That is, we want $\mathbf{w} : \mathcal{Z} \to \mathbf{R}^{\geq 0}$ with $\sum_{z\in\mathcal{Z}} \mathbf{w}(z) = 1$ such that:

$$\forall\ 0 \leq i+j \leq d,\quad \mathbf{E}_{\mathbf{w}}[f(x,y)\ x^i y^j] = \sum_{(x,y)\in\mathcal{Z}} \mathbf{w}(x,y) f(x,y)\ x^i y^j = 0.$$

In addition we would like to find a solution in which $\mathrm{size}(\mathcal{Z})$ is as small as possible, where $\mathrm{size}(\mathcal{Z})$ is defined to be $\max_{(x,y)\in\mathcal{Z}}\{\max\{|x|,|y|\}\}$. Once we have such a $\mathcal{Z}$ and $\mathbf{w}$, we get a lower bound of $d+1$ for the degree of a polynomial threshold function computing $\mathsf{AND\text{-}MAJ}_{\mathrm{size}(\mathcal{Z})}$. In the remainder of this section we give a construction in which $\mathrm{size}(\mathcal{Z}) = d^{O(d)}$. This gives us the main result of this section:

THEOREM 17. *$thr(\mathsf{AND\text{-}MAJ}_n) = \Omega(\frac{\log n}{\log\log n})$.*

5.1 Proof of Theorem 17

Our *constraints* are all bivariate monomials $x^i y^j$ of total degree at most d. We will refer to $x^i y^j$ as the "(i,j) constraint monomial." There are a total of $D = \frac{(d+1)(d+2)}{2}$ constraint monomials, and for definiteness we will consider them to be ordered as follows: 1, x, y, x^2, xy, y^2, x^3, etc.

Our *support* will be:

$$\mathcal{Z} = \{((-1)^\ell\ h^k, (-1)^k\ h^\ell) : 0 \leq k+\ell \leq d\} \cup \{(-1,-1)\},$$

where here h is a large quantity to be chosen later (eventually we will take $h = \Theta(d^9)$). The support $\mathcal{Z}$ is symmetric about the line $y = x$ and contains exactly $D+1$ points. We will refer to $((-1)^\ell\ h^k, (-1)^k\ h^\ell)$ as the "(k,ℓ) support point" and consider the points to be ordered in the same order as the monomials (i.e., $(1,1)$, $(h,-1)$, $(-1,h)$, $(h^2,1)$, $(-h,-h)$, $(1,h^2)$, $(h^3,-1)$, etc.), with the special point $(-1,-1)$ coming last. Note that the value of f on the (k,ℓ) support point is $(-1)^{k\ell+k+\ell}$.

Let $\tilde{A}$ be a $D\times(D+1)$ matrix whose columns are indexed by the support points and whose rows are indexed by the constraint monomials. Define $\tilde{A}[(i,j),(k,\ell)]$ to be the value of the (i,j)th constraint monomial at the (k,ℓ)th support point, *times* the value of f at the (k,ℓ)th support point. This definition shall include the case of the special $(-1,-1)$ support point, to whose column we assign the index $(0',0')$ for reasons that will become clear soon. Let A be the $(D+1)\times(D+1)$ matrix given by adding a row of 1's to the bottom of $\tilde{A}$. For notational convenience we will also give this row the index $(0',0')$. So for $(i,j),(k,\ell) \neq (0',0')$ we have:

$$A[(i,j),(k,\ell)] = (-1)^{k(j+1)+\ell(i+1)+k\ell} h^{ik+j\ell}. \qquad (1)$$

Recall that we want to find values $\mathbf{w} : \mathcal{Z} \to \mathbf{R}$ such that $\sum_{(x,y)\in\mathcal{Z}} \mathbf{w}(x,y) f(x,y)\ x^i y^j = 0$ for all constraints and such that $\sum_{(x,y)\in\mathcal{Z}} \mathbf{w}(x,y) = 1$. By construction these values are uniquely given by the solution to the following system of linear equations:

$$A\begin{bmatrix} w_{(0,0)} \\ w_{(1,0)} \\ w_{(0,1)} \\ w_{(2,0)} \\ \vdots \\ w_{(0,d)} \\ w_{(-1,-1)} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}. \qquad (2)$$

In the remainder of the proof we show that by taking $h = \Theta(d^9)$, we can ensure that the solution to Equation (2) consists entirely of nonnegative numbers, and hence $\mathbf{w}$ corresponds to a true probability distribution as desired. Since $h = O(d^9)$ means that $\text{size}(\mathcal{Z}) = d^{O(d)}$, and we may take h to be odd, this proves Theorem 17.

We shall consider solving Equation (2) via Cramer's rule. Cramer's rule tells us that Equation (2) implies:

$$w_{(u,v)} = \frac{\det A_{(u,v)}}{\det A},$$

where $A_{(u,v)}$ denotes the matrix A with the (u,v) column replaced by the right hand side of Equation (2), namely $\begin{bmatrix} 0 & 0 & 0 & \cdots & 0 & 1 \end{bmatrix}^T$. To show that each $w_{(u,v)}$ is nonnegative we will show that $\det A_{(u,v)}$ and $\det A$ have the same sign.

Let $\sigma \in \{+1, -1\}$ be the sign of the product of the diagonal entries in A. We will prove the following two lemmas and thus prove Theorem 17:

LEMMA 18. *sign*$(\det A) = \sigma$.

LEMMA 19. *sign*$(\det A_{(u,v)}) = \sigma$ *for all* (u,v).

5.1.1 Proof of Lemma 18

To prove Lemma 18 we view $\det A$ as a polynomial in h. Let $T := \deg(\det A)$ be the degree of $\det A$. We show that the leading term of $\det A$ (corresponding to h^T) dominates all the other terms for h sufficiently large, and thus the sign of $\det A$ is the same as the sign of the leading term. More precisely, we establish the following two facts:

CLAIM 20. *The coefficient of* h^T *in* $\det A$ *is* 2σ.

CLAIM 21. *For all* $u \geq 1$ *the coefficient of* h^{T-u} *in* $\det A$ *is at most* $2(D+2)^{4u}$ *in magnitude.*

Claim 21 implies that the sum of the absolute values of the lower-order terms in $\det A$ is at most $\sum_{u=1}^{T} 2(D+2)^{4u} h^{T-u} \leq h^T \sum_{u=1}^{T} (2(D+2)^4/h)^u$. If we take h to be $\Theta(d^9)$ then this quantity will be strictly smaller than h^T. But by Claim 20 we have that the leading term of $\det A$ is $2\sigma h^T$. Thus $\text{sgn}(\det A) = \sigma$ and Lemma 18 holds.

We set the stage before proving Claims 20 and 21 with some notation and some observations. Let $\mathbf{S}$ denote the permutation group on the indices $(0,0)$, $(1,0)$, $(0,1)$, $(2,0)$, $\ldots$, $(0,d)$, $(0',0')$. Then:

$$\det A = \sum_{\pi \in \mathbf{S}} \text{sgn}(\pi) \prod_{(i,j)} A[(i,j), \pi(i,j)]. \tag{3}$$

Recall that for $(i,j),(k,\ell) \neq (0',0')$, the entry $A[(i,j),(k,\ell)]$ is $\pm h^{ik+j\ell}$, which we will write as $\pm \exp_h((i,j)\cdot(k,\ell))$, with $\exp_h(t)$ denoting h^t and $\cdot$ being the usual dot product. In the case that $(i,j) = (0',0')$ or $(k,\ell) = (0',0')$, the entry $A[(i,j),(k,\ell)]$ is $\pm 1 = \pm h^0$. If we define $(0',0')\cdot(a,b)$ to be 0, then we have that for any permutation $\pi \in \mathbf{S}$,

$$\prod_{(i,j)} A[(i,j), \pi(i,j)] = \pm \exp_h\left(\sum_{(i,j)} (i,j)\cdot\pi(i,j)\right).$$

Given a permutation $\pi \in \mathbf{S}$, write $t(\pi) = \sum_{(i,j)} (i,j) \cdot \pi(i,j)$, so the permutation π contributes ± 1 to the coefficient of $h^{t(\pi)}$ in $\det A$. Then the absolute value of the coefficient of h^u in $\det A$ is at most $|\{\pi \in \mathbf{S} : t(\pi) = u\}|$. We will use this fact to bound all the lower-order terms in $\det A$; for the leading term we will pay more attention to the signs.

To calculate $t(\pi)$ from π, we decompose the permutation π as a product of cycles. For each cycle $\pi_0 = ((i_1,j_1)\,(i_2,j_2)\cdots(i_m,j_m))$ we have by simple arithmetic:

$$\sum_{r=1}^{m} (i_r,j_r)\cdot(i_r,j_r) - \sum_{r=1}^{m} (i_r,j_r)\cdot\pi_0(i_r,j_r) = \frac{1}{2}\sum_{r=1}^{m} (i_r - i_{r-1})^2 + (j_r - j_{r-1})^2, \tag{4}$$

where we use the notation $(i_0,j_0) = (i_m,j_m)$. (Note that a geometric interpretation of this quantity is that it is half the sum of the squares of the lengths of the line segments which make up the cycle in the two-dimensional plane from (i_1,j_1) to (i_2,j_2) to (i_3,j_3) to $\ldots$ to (i_m,j_m) to (i_1,j_1).) In particular, this quantity is at least 1 for every nontrivial cycle, where a trivial cycle for us is either a cycle of length 1 or the transposition exchanging $(0,0)$ and $(0',0')$. The quantity in Equation (4) is 0 for trivial cycles. Thus we have that the identity permutation and the transposition $((0,0),(0',0'))$ are the only two permutations which achieve the maximum value $t(\pi) = T$. It is easy to see that this maximum value T is $\sum_{(i,j)} i^2 + j^2$, which one easily calculates to be $T := d(d+1)^2(d+2)/6$. We further see that every other permutation "pays a penalty" in its t value for each nontrivial cycle it contains, and this penalty is given by the right-hand side of Equation (4). Hence to calculate $t(\pi)$ from π we simply sum up the penalties for each cycle in its cycle decomposition and subtract the total from T.

Proof of Claim 20: As described above, we have that there are exactly two permutations which lead to the maximum power h^T in Equation (3): the identity permutation which takes all the diagonal elements, and the $((0,0),(0',0'))$ transposition which takes the top-right entry of A, the bottom-left entry of A, and the diagonal elements otherwise. The product of the top-left and bottom-right entries of A is 1. The product of the top-right and bottom-left entries is -1; however this gets flipped to $+1$ by the sign of the permutation (it is a transposition so its sign is -1). We conclude that leading term of $\det A$ is $2\sigma h^T$ where $\sigma \in \{-1,1\}$ is the sign of the product of the diagonal entries in A. $\square$

Proof of Claim 21: To bound the coefficient on the lower-order term h^{T-u} in $\det A$ we simply count the number of permutations π which have $t(\pi) = T - u$. This count gives an upper bound on the magnitude of the coefficient. If $t(\pi) = T - u$ then the penalty accounting scheme from Equation (4) tells us that π has at most u nontrivial cycles. In fact we can say more: any nontrivial cycle of length m must incur a penalty of at least $\lfloor m/2 \rfloor$. (This can be verified using the geometric interpretation described earlier, together with the fact that any nontrivial cycle of length $m \geq 3$ can include at most one segment of length 0 between $(0,0)$ and $(0',0')$.) Consequently, if $t(\pi) = T - u$ then the lengths of the nontrivial cycles in π's cycle decomposition must sum to at most $3u$ (in the worst case all its cycles may be 3-cycles each of which incurs a penalty of 1). Now observe that there are at most $(D+2)^{4u}$ permutations on $D+1$ elements which decompose into at most u cycles whose total length is at most $3u$. (Any such sequence of cycles can be written as a string of length $4u$ over a $D+2$ element alphabet, where the extra symbol is used to mark the end of each cycle.) Doubling this upper bound covers the optional addition of the trivial $((0,0),(0',0'))$ transposition. We thus may conclude that there are at most $2(D+2)^{4u}$ permutations $\pi \in \mathbf{S}$ which have $t(\pi) = T - u$. $\square$

5.1.2 Proof of Lemma 19

It now remains to show that $\mathrm{sgn}(\det A_{(u,v)}) = \sigma$ for each (u,v). By the nature of cofactor expansion, $\det A_{(u,v)}$ is equal to a certain sign ρ, times the determinant of A with the bottom row and the (u,v) column deleted. In the case $(u,v) = (0',0')$ we have $\rho = 1$ and we shall write $A'_{(0',0')}$ for the matrix A with its last row and column deleted. For all $(u,v) \neq (0',0')$, let us write $A'_{(u,v)}$ for the matrix gotten by first deleting the bottom row and (u,v) column from A, and then moving the $(0',0')$ column leftward until it is in the place where the old (u,v) used to be. Shifting the $(0',0')$ column like this incurs a sign change equal to $-\rho$; we conclude that $\det A_{(u,v)} = -\det A'_{(u,v)}$. Hence it is sufficient for us to show that $\mathrm{sgn}(\det A'_{(0',0')}) = \sigma$ and that $\mathrm{sgn}(\det A'_{(u,v)}) = -\sigma$ for all $(u,v) \neq (0',0')$.

Let us begin by dispensing with the cases $(u,v) = (0',0')$ or $(0,0)$. In both of these cases $A'_{(u,v)}$ is very similar to A with the last row and column deleted; when $(u,v) = (0',0')$ this is exactly what $A'_{(u,v)}$ is, and when $(u,v) = (0,0)$ some of the signs in the first column are changed. Hence the analysis of $\det A'_{(u,v)}$ is virtually identical to the above analysis of $\det A$, except that $(0',0')$ is no longer present. The leading term will therefore be equal to the top-left entry of $A'_{(u,v)}$ times σh^T; this entry is 1 when $(u,v) = (0',0')$ and is -1 when $(u,v) = (0,0)$, as desired. The analysis bounding the lower-order terms goes through in essentially the same way as before (again without $(0',0')$) and we conclude that $\mathrm{sgn}(\det A'_{(0',0')}) = \sigma$ and $\mathrm{sgn}(\det A'_{(0,0)}) = -\sigma$ as desired.

Throughout the rest of this section we assume that $(u,v) \neq (0',0'), (0,0)$. Let $T_{(u,v)}$ denote the degree of $\det(A'_{(u,v)})$. We will prove the following two claims:

CLAIM 22. *The coefficient of $h^{T_{(u,v)}}$ in $\det(A'_{(u,v)})$ is $-2\sigma C$.*

CLAIM 23. *For all $s \geq 1$ the coefficient of $h^{T_{(u,v)}-s}$ in $\det(A'_{(u,v)})$ is at most $4C(D+2)^{4s}$ in magnitude.*

As in the previous subsection, these two claims show that we may take $h = \Theta(d^9)$ to obtain $\mathrm{sgn}(\det(A'_{(u,v)})) = -\sigma$, so they suffice to prove the lemma.

Studying $\det A'_{(u,v)}$ is slightly more complex than studying $\det A$ because its rows and columns no longer have the same names; the rows of $A'_{(u,v)}$ are named $(0,0)$, $(1,0)$, $(0,1)$, $(2,0), \ldots, (u,v), \ldots, (0,d)$, whereas the columns are named $(0,0)$, $(1,0)$, $(0,1)$, $(2,0)$, $\ldots$, $(0',0')$, $\ldots$, $(0,d)$. To deal with this, we will let $\mathbf{S}'$ denote the permutation group on the D *row* indices of $A'_{(u,v)}$, and we will view (u,v) as $(0',0')$ whenever it is the "output" of a permutation. To be precise, let ι be a mapping which maps (i,j) to (i,j) for each $(i,j) \neq (u,v)$, and maps (u,v) to $(0',0')$. Then our determinant equation becomes:

$$\det A'_{(u,v)} = \sum_{\pi \in \mathbf{S}} \mathrm{sgn}(\pi) \prod_{(i,j)} A[(i,j), \iota(\pi(i,j))]. \qquad (5)$$

We may write $t(\pi) = \sum_{(i,j)} (i,j) \cdot \iota(\pi(i,j))$, so we have $\prod_{(i,j)} A[(i,j), \iota(\pi(i,j))] = \pm h^{t(\pi)}$.

As before we will calculate $t(\pi)$ by considering the cycle decomposition of π and computing the penalty difference from $T = d(d+1)^2(d+2)/6$ for each cycle. Since now the "identity" permutation does not exist, the permutations maximizing $t(\pi)$ may not achieve T; indeed, since $(u,v) \neq (0',0')$ it is the case that maximizing permutations will not achieve $t(\pi) = T$. Let us now find the new highest value for $t(\pi)$. The cycle decomposition of π contains a unique cycle (which may be a 1-cycle) containing (u,v), and perhaps other cycles which do not contain (u,v). For the cycles not containing (u,v), ι does not enter into the picture in calculating $t(\pi_0)$; hence Equation (4) still holds and we conclude that any π with maximal $t(\pi)$ has no nontrivial cycles involving (u,v). Thus, in order to find all maximizing π's, it is sufficient to determine which cycles containing (u,v) give the smallest penalty.

Let π^* be a cycle containing (u,v); say $\pi^* = ((u,v)\ (i_1,j_1)\ (i_2,j_2)\ \cdots\ (i_m,j_m))$, so according to our conventions π^* maps (u,v) to (i_1,j_1), maps (i_r,j_r) to (i_{r+1},j_{r+1}) for $1 \leq r \leq m-1$, and maps (i_m,j_m) to $\iota(u,v) = (0',0')$. Write $(i_0,j_0) = (u,v)$. Then akin to Equation (4) we have:

$$\begin{aligned}
& \sum_{r=0}^{m} (i_r,j_r)\cdot(i_r,j_r) - \sum_{r=0}^{m} (i_r,j_r)\cdot \iota(\pi^*(i_r,j_r)) \\
= \ & \sum_{r=0}^{m} (i_r,j_r)\cdot(i_r,j_r) - \\
& \quad \sum_{r=0}^{m} (i_r,j_r)\cdot(i_{r+1 \bmod m+1}, j_{r+1 \bmod m+1}) + i_r u + j_r v \\
= \ & \frac{1}{2}\left(\left(\sum_{r=1}^{m} (i_r - i_{r-1})^2 + (j_r - j_{r-1})^2\right)\right. \\
& \quad \left. + (u - i_r)^2 + (v - j_r)^2\right) + i_r u + j_r v \quad \text{(as in (4))} \\
= \ & \frac{1}{2}\left(\left(\sum_{r=1}^{m} (i_r - i_{r-1})^2 + (j_r - j_{r-1})^2\right)\right. \\
& \quad \left. + i_m^2 + j_m^2 + u^2 + v^2\right). \qquad (6)
\end{aligned}$$

The geometric interpretation of the quantity on the right-hand side of Equation (6) is that it is half the sum of the squares of the path segments on the closed path from (u,v) to (i_1,j_1) to (i_2,j_2) to $\cdots$ to (i_m,j_m) to $(0,0)$ to (u,v). It is immediate that in a cycle minimizing this quantity, there should be no path step which has either x or y displacement greater than 1 in magnitude (aside from the step from $(0,0)$ to (u,v) which is forced). Consequently, the permutations π which maximize $t(\pi)$ are precisely those cycles π^* such that (1) $i_{r+1} - i_r \in \{-1,0\}$ and $j_{r+1} - j_r \in \{-1,0\}$ for $0 \leq r < m$, and (2) $i_m, j_m \in \{0,1\}$. It is easy to see that each such maximizing permutation has $t(\pi) = T_{(u,v)} = T - \frac{1}{2}(u + v + u^2 + v^2)$.

Proof of Claim 22: Now we can compute the coefficient of $h^{T_{(u,v)}}$ in $\det A'_{(u,v)}$. Given a permutation π maximizing $t(\pi)$, let $\epsilon(\pi)$ denote the sign of π's contribution to the determinant computation of Equation (5), i.e. $\epsilon(\pi) = \mathrm{sgn}(\pi)\prod_{(i,j)} \mathrm{sgn}(A[(i,j),\iota(\pi(i,j))])$. Then the leading coefficient of $\det A'_{(u,v)}$ is just the sum of $\epsilon(\pi)$ over all maximizing π.

Let $\pi = ((u,v)\ (i_1,j_1)\ (i_2,j_2)\ \cdots\ (i_m,j_m))$ be a maximizing permutation; as before we write $(i_0,j_0) = (u,v)$. By the definition of σ as the product of the signs of A's diagonal elements, we get that $\sigma\epsilon(\pi)$ is equal to $\mathrm{sgn}(\pi)$ times:

$$\left(\prod_{r=0}^{m-1} \mathrm{sgn}(A[(i_r,j_r),(i_r,j_r)])\mathrm{sgn}(A[(i_r,j_r),(i_{r+1},j_{r+1})])\right) \cdot \mathrm{sgn}(A[(i_m,j_m),(i_m,j_m)])\mathrm{sgn}(A[(i_m,j_m),(0',0')]).$$

We claim that for each $0 \le r \le m-1$ we have:

$$\mathrm{sgn}(A[(i_r,j_r),(i_r,j_r)])\mathrm{sgn}(A[(i_r,j_r),(i_{r+1},j_{r+1})]) = -1,$$

independent of (i_r,j_r). For from Equation (1) we know that:

$$\begin{aligned}
&\mathrm{sgn}(A[(i_r,j_r),(i_r,j_r)])\mathrm{sgn}(A[(i_r,j_r),(i_{r+1},j_{r+1})]) \\
=\ & \exp_{-1}(i_r(j_r+1) + j_r(i_r+1) + i_rj_r) \\
&\cdot \exp_{-1}(i_{r+1}(j_r+1) + j_{r+1}(i_r+1) + i_{r+1}j_{r+1}) \\
=\ & \exp_{-1}(i_rj_r + i_{r+1}j_r + i_rj_{r+1} + i_{r+1}j_{r+1} \\
&\quad + i_r + i_{r+1} + j_r + j_{r+1}) \\
=\ & \exp_{-1}((i_r+i_{r+1}+1)(j_r+j_{r+1}+1) - 1),
\end{aligned}$$

which is always -1 as claimed, because $(i_r,j_r)-(i_{r+1},j_{r+1}) \in \{(1,0),(0,1),(1,1)\}$.

Thus we have:

$$\begin{aligned}
\sigma\epsilon(\pi) &= \mathrm{sgn}(\pi)(-1)^m\mathrm{sgn}(A[(i_m,j_m),(i_m,j_m)]) \\
&\quad \cdot \mathrm{sgn}(A[(i_m,j_m),(0',0')]) \\
&= +\mathrm{sgn}(A[(i_m,j_m),(i_m,j_m)])\mathrm{sgn}(A[(i_m,j_m),(0',0')]) \quad (*),
\end{aligned}$$

because π is a cycle of length $m+1$. If $(i_m,j_m) = (1,1)$ then $(*) = -1$; otherwise, $(*) = +1$. Hence we conclude that $\epsilon(\pi) = \sigma$ if $(i_m,j_m) = (1,1)$ and $\epsilon(\pi) = -\sigma$ if $(i_m,j_m) \in \{(0,0),(1,0),(0,1)\}$. For each maximizing cycle π of length $m+1$ with $(i_m,j_m) \neq (0,0)$, there is a corresponding maximizing cycle π' of length $m+2$ obtained by appending $(i_{m+1},j_{m+1}) = (0,0)$ to π. Thus we have $\epsilon(\pi)+\epsilon(\pi') = 0$ when $(i_m,j_m) = (1,1)$ and $\epsilon(\pi)+\epsilon(\pi') = -2\sigma$ when $(i_m,j_m) = (1,0)$ or $(0,1)$. In conclusion, the leading term in $\det A'_{(u,v)}$ is exactly $-2\sigma Ch^{T_{(u,v)}}$, where C is the number of paths from (u,v) to $(1,0)$ plus the number of paths from (u,v) to $(0,1)$, where each path uses steps $(-1,0)$, $(0,-1)$, and $(-1,-1)$. (Such paths are known as *Delannoy paths*, and the number of such paths between a pair of points is a *Delannoy number*; hence C is a sum of two Delannoy numbers.) Since $(u,v) \neq (0,0)$ we have $C \ge 1$, and the claim is proved. $\square$

Proof of Claim 23: We must upper-bound the magnitude of the lower-order terms in $\det A'_{(u,v)}$. We do this as in the analysis of $\det A$ by upper-bounding the number of permutations π with $t(\pi) = T_{(u,v)} - s$. To each $\pi \in \mathbf{S}'$ we will associate a *maximizing* permutation π^* (i.e., one for which $t(\pi^*) = T_{(u,v)}$), and a "deviation description." We will show that the longer the deviation description, the smaller $t(\pi)$ is compared to $t(\pi^*)$. Thus the number of permutations π with $t(\pi)$ close to $T_{(u,v)}$ will be upper-bounded by the number of optimal permutations times the number of short deviation descriptions.

Let π be an arbitrary permutation in $\mathbf{S}'$ and write π as the product of a cycle π_0 involving (u,v), and some other cycles $\pi_1,\ldots,\pi_s$. The maximizing permutation π^* we associate with π will depend only on π_0. View π_0 geometrically as a path from (u,v) to $\pi_0^{-1}(u,v)$. Call a path "optimal" if it only uses steps $(-1,0)$, $(0,-1)$, and $(-1,-1)$, so in particular every maximizing permutation contains one nontrivial cycle containing (u,v) whose corresponding path is optimal. We will split π_0 up into its optimal and nonoptimal segments. Specifically, $a_i,b_i,c_i,d_i,\ldots,a_r,b_r,c_r,d_r$ are defined as follows: π_0 proceeds optimally from (u,v) to (a_1,b_1), at which point it takes a nonoptimal step. Let (c_1,d_1) be the first point it proceeds to subsequently with the property that $c_1 \le a_1$, $d_1 \le b_1$. Then π_0 proceeds optimally from (c_1,d_1) to (a_2,b_2), at which point it makes a nonoptimal step. Let (c_2,d_2) be the first point it proceeds to subsequently with $c_2 \le a_2$, $d_2 \le b_2$. Continuing in this fashion, let (a_r,b_r) be the last point reached in the last optimal segment of π_0; π_0 may optionally go on and reach $\pi_0^{-1}(u,v)$ We will let the maximizing permutation π^* associated with π be any optimal path that agrees with π_0 on all steps from (u,v) to (a_1,b_1), all steps from (c_1,d_1) to (a_2,b_2), $\ldots$, all steps from (c_{r-1},d_{r-1}) to (a_r,b_r), and then ends by proceeding optimally to $(0,0)$.

The deviation description of π will simply be a list of all of the cycles $\pi_1,\ldots,\pi_s$ not containing (u,v), along with a description of π_0's deviation from π^*. This deviation consists of the path from (a_1,b_1) to (c_1,d_1), from (a_2,b_2) to (c_2,d_2), etc., possibly ending with some path from (a_r,b_r) to a point not in $\{0,1\}^2$. Note that π can be recovered from π^* and the deviation description.

Now let us compute $t(\pi^*) - t(\pi)$. This difference is equal to $(T - t(\pi)) - (T - t(\pi^*))$, and Equations (4) and (6) tell us how to compute these quantities. By Equation (4), $t(\pi)$ pays an extra penalty over $t(\pi^*)$ for each of its cycles not involving (u,v), $\pi_1,\ldots,\pi_s$. As in the analysis of $\det A$ we know that such a cycle of length m incurs a penalty of at least $\lfloor m/2 \rfloor$. Equation (6) allows us to compare the penalties against T that each of $t(\pi^*)$ and $t(\pi)$ pays. Every time π_0 deviates from π^* it pays an extra penalty of at least 1. Indeed, just as in the analysis of extraneous cycles, a deviation path from (a_i,b_i) to (c_i,d_i) which touches m nodes must incur an extra penalty of at least $\lfloor m/2 \rfloor$. This holds also for a final deviation path which does not end up in $\{0,1\}^2$, since it must pay for half the squared distance from the origin of its endpoint. Both π^* and π_0 pay equally for the final $\frac{1}{2}(k^2+\ell^2)$ term.

In conclusion, if the total length of the cycles and deviation paths in π's deviation description is m then $(T - t(\pi)) - (T - t(\pi^*))$ is at least $\lfloor m/2 \rfloor$; i.e., $t(\pi) \le T_{(u,v)} - \lfloor m/2 \rfloor$. Hence as in the analysis of $\det A$ we can get an upper bound of $(D+2)^{4s} \cdot \#\{\text{number of maximizing } \pi_0\}$ for the number of permutations π with $t(\pi) = T_{(u,v)} - s$. But note that the leading coefficient in $\det A'_{(u,v)}$ has magnitude $2C$, and $2C$ is at least half the number of maximizing permutations π_0. To see this, recall that C counts the number of optimal paths from (u,v) to either $(1,0)$ or $(0,1)$, and each maximizing permutation corresponds to an optimal path to one of $(0,0),(0,1),(1,0),(1,1)$. The number of optimal paths to $(1,1)$ is at most C (each such path can be extended to a path ending in $(1,0)$ or $(0,1)$), and hence the number of optimal paths to $(0,0)$ is at most $2C$ (since the next to last point on any such path is either $(1,0),(0,1)$ or $(1,1)$). It follows that the magnitude of the sum of all lower-order terms in $\det A'_{(u,v)}$ is at most $\sum_{s=1}^{T(u,v)} 4C(D+2)^{4s} h^{T(u,v)-s}$, and the claim is proved. $\square$

6. CONJECTURES AND FUTURE WORK

Many questions remain for further research on the polynomial threshold degree of Boolean functions. We believe the new techniques introduced in this paper will lead to the solution of some of them. Below we give some open problems and conjectures which we hope will spur further research.

- Can lower bounds of $\Omega(n^{1/3+\epsilon})$ for some $\epsilon > 0$ be proved for constant depth circuits of depth 3 or greater? In particular, let f be the function computed by the following depth-3 read-once formula: the top gate is an AND of fan-in $n^{1/5}$, the middle gates are ORs of fan-in $n^{2/5}$, and the bottom gates are ANDs of fan-in $n^{2/5}$. We conjecture that f requires PTF degree $\Omega(n^{2/5})$, and believe that this may be provable via our lower bound techniques. (Krause and Pudlak [15] have given lower bounds on the number of nonzero coefficients in any polynomial threshold function for this circuit, but as mentioned earlier their results do not imply new degree lower bounds.)

- Does every Boolean formula of size s have a polynomial threshold function of degree $O(\sqrt{s})$ independent of its depth? This is the best possible upper bound since parity on $\sqrt{s}$ variables is computed by a formula of size s and depth $O(\log s)$.

  One particular function that seems difficult is the following: Let B be an integer and consider the function $g(x_1, \ldots, x_B, y_1, \ldots, y_B) = (\mathrm{OR}(x_1, \ldots, x_B))$ OR $(\mathrm{AND}(y_1, \ldots, y_B))$. Let f be the Boolean formula given by a tree of copies of g. Let f be on n total variables and let $B = \log n$, so that f has depth $\Theta(\log n / \log \log n)$. We do not know how to show that this function has PTF degree $O(\sqrt{n})$.

- Can our $\Omega(\frac{\log n}{\log \log n})$ lower bound for the AND of two majorities be strengthened to $\Omega(\log n)$? We conjecture that $\Omega(\log n)$ is the true lower bound and that hence the Beigel *et al.* construction is optimal.

7. REFERENCES

[1] D. Angluin. Queries and concept learning. *Machine Learning*, 2:319–342, 1988.

[2] J. Aspnes, R. Beigel, M. Furst, and S. Rudich. The expressive power of voting polynomials. *Combinatorica*, 14(2):1–14, 1994.

[3] H. Buhrman, R. Cleve, and A. Wigderson. Quantum vs. classical communication and computation. In *Proceedings of the 30th Annual ACM Symposium on Theory of Computing*, pages 63–68, 1998.

[4] R. Beigel. The polynomial method in circuit complexity. In *Proceedings of the Eigth Conference on Structure in Complexity Theory*, pages 82–95, 1993.

[5] R. Beigel. Perceptrons, PP, and the Polynomial Hierarchy. *Computational Complexity*, 4:339–349, 1994.

[6] R. Beigel, N. Reingold, and D. Spielman. PP is closed under intersection. *Journal of Computer and System Sciences*, 50(2):191–202, 1995.

[7] J. Bruck. Harmonic analysis of polynomial threshold functions. *SIAM Journal on Discrete Mathematics*, 3(2):168–177, 1990.

[8] E. Cheney. *Introduction to approximation theory.* McGraw-Hill, New York, New York, 1966.

[9] Y. Freund. Boosting a weak learning algorithm by majority. *Information and Computation*, 121(2):256–285, 1995.

[10] M. Goldmann. On the power of a threshold gate at the top. *Information Processing Letters*, 63(6):287–293, 1997.

[11] A. Hajnal, W. Maass, P. Pudlak, M. Szegedy, and G. Turan. Threshold circuits of bounded depth. *Journal of Computer and System Sciences*, 46:129–154, 1993.

[12] M. Kearns and U. Vazirani. *An introduction to computational learning theory.* MIT Press, Cambridge, MA, 1994.

[13] A. Klivans, R. O'Donnell, and R. Servedio. Learning intersections and thresholds of halfspaces. In *Proceedings of the Forty-Third Annual Symposium on Foundations of Computer Science*, 2002.

[14] A. Klivans and R. Servedio. Learning DNF in time $2^{\tilde{O}(n^{1/3})}$. In *Proceedings of the Thirty-Third Annual Symposium on Theory of Computing*, pages 258–265, 2001.

[15] M. Krause and P. Pudlak. Computing Boolean functions by polynomials and threshold circuits. *Computational Complexity*, 7:346–370, 1998.

[16] N. Littlestone. Learning quickly when irrelevant attributes abound: a new linear-threshold algorithm. *Machine Learning*, 2:285–318, 1988.

[17] N. Linial, Y. Mansour, and N. Nisan Constant depth circuits, Fourier transform and learnability. *Journal of the ACM*, 40(3):607–620, 1993.

[18] M. Minsky and S. Papert. *Perceptrons: an introduction to computational geometry (expanded edition).* MIT Press, Cambridge, MA, 1988.

[19] D. J. Newman. Rational approximation to $|x|$. *Michigan Mathematical Journal*, 11:11–14, 1964.

[20] N. Nisan and M. Szegedy. On the degree of boolean functions as real polynomials. In *Proceedings of the Twenty-Fourth Annual Symposium on Theory of Computing*, pages 462–467, 1992.

[21] R. Paturi and M. Saks. Approximating threshold circuits by rational functions. *Information and Computation*, 112(2):257–272, 1994.

[22] M. Saks. *Slicing the hypercube.* Cambridge University Press, 1993.

[23] D. Sieling. Minimization of decision trees is hard to approximate. Technical Report ECCC Report TR02-054, Electronic colloquium on computational complexity, 2002.

[24] L. Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, 1984.

[25] N. Vereshchagin. Lower bounds for perceptrons solving some separation problems and oracle separation of AM from PP. In *Proceedings of the Third Annual Israel Symposium on Theory of Computing and Systems*, 1995.