Electronic Colloquium on Computational Complexity, Report No. 74 (2010)

Learning and Lower Bounds for AC^0 with Threshold Gates

Parikshit Gopalan
Microsoft Research – Silicon Valley
parik@microsoft.com

Rocco A. Servedio*
Columbia University
rocco@cs.columbia.edu

April 20, 2010

Abstract

In 2002 Jackson et al. [JKS02] asked whether AC^0 circuits augmented with a threshold gate at the output can be efficiently learned from uniform random examples. We answer this question affirmatively by showing that such circuits have fairly strong Fourier concentration; hence the low-degree algorithm of Linial, Mansour and Nisan [LMN93] learns such circuits in sub-exponential time. Under a conjecture of Gotsman and Linial [GL94] which upper bounds the total influence of low-degree polynomial threshold functions, the running time is quasi-polynomial. Our results extend to AC^0 circuits augmented with a small super-constant number of threshold gates at arbitrary locations in the circuit. We also establish some new structural properties of AC^0 circuits augmented with threshold gates, which allow us to prove a range of separation results and lower bounds.

Our techniques combine classical random restriction arguments with more recent results [DRST09, HKM09, She09] on polynomial threshold functions.

*Supported by NSF grants CCF-0347282, CCF-0523664 and CNS-0716245, and by DARPA award HR0011-08-1-0069.

ISSN 1433-8092

1 Introduction

The seminal result of Linial, Mansour and Nisan [LMN93] showed how to learn the class AC^0 of constant depth circuits in quasi-polynomial time under the uniform distribution with random examples. Their work introduced the *Low-Degree Algorithm* which can learn any class of functions where the Fourier spectrum is concentrated on low-degree coefficients; this algorithm and its extensions have since found numerous applications in learning, see e.g. [FJS91, BT96, JKS02, KOS04, MOS04, OS07, BOW08, KKMS08, KOS08].

In the two decades since their work, despite much effort, there has been limited progress in designing learning algorithms for more expressive circuit classes. Circuit classes like AC^0 with parity gates ($\mathsf{AC}^0[2]$) and depth-2 TC^0 remain beyond the reach of currently known algorithms. One obstacle is that there are no lower bounds known for some of these classes, such as depth-2 TC^0, and the existence of lower bounds seems to be a pre-requisite for any learning algorithm (see [FK09]). Devising learning algorithms and lower bound techniques that can handle more powerful classes of circuits is a central open problem at the intersection of computational learning theory and circuit lower bounds.

Jackson et al. made some progress on learning circuits more expressive than AC^0 in [JKS02]. They gave a quasipolynomial-time algorithm that can learn Majority-of-AC^0 circuits – polynomial-size, constant-depth circuits augmented with a single Majority gate at the output – under the uniform distribution. Using a result of [Bei94], this yields a quaspolynomial-time algorithm that can learn AC^0 circuits augmented with $\mathrm{polylog}(n)$ many Majority gates at arbitrary locations in the circuit. The algorithm of Jackson et al. uses the low-degree algorithm as a weak learner and combines it with boosting. [JKS02] posed as an open question whether any efficient algorithm can learn Threshold-of-AC^0 circuits, in which the the topmost gate is a threshold gate (i.e. a weighted majority in which the weights may be arbitrary). It is observed in [JKS02] via an explicit counterexample that the analysis of their boosting-based algorithm breaks down for Threshold-of-AC^0. In this work, we take a significant step towards answering the question of [JKS02].

AC^0 circuits augmented with a few threshold gates have been well studied in the complexity theory literature, see e.g. [ABFR94, Bei94, GHR92, Gol97]. This is a natural class of circuits lying between the classes AC^0 (which we understand well) and TC^0 (for which we do not know lower bounds). One focus of this work has been on understanding the difference in power between unweighted threshold gates (i.e. majorities) versus threshold gates with arbitrary weights. Aspnes et al. [ABFR94] prove that any AC^0 circuit with a single threshold gate at the top cannot compute (or even approximate) parity. However, we are not aware of prior lower bounds known even for AC^0 augmented with two threshold gates. In contrast, when we restrict ourselves to Majority gates, an elegant result of Beigel [Bei94] alluded to above shows that any polynomial-size AC^0 circuit with $\mathrm{polylog}(n)$ Majority gates is equivalent to a quasi-polynomial size AC^0 circuit with a single majority gate at the top, and lower bounds for such circuits follow from [ABFR94].

1.1 Our Results

We show that AC^0 circuits augmented with a few threshold gates with arbitrary weights can be learned in subexponential time under the uniform distribution. In doing this we establish some new structural properties of such circuits, which allow us to prove new lower bounds and separations for such circuits.

1.1.1 Learning AC^0 with threshold gates

Our first main result is a Fourier concentration bound for Threshold-of-AC^0 circuits: roughly speaking, this bound says that any size-M, constant-depth Threshold-of-AC^0 circuit C must satisfy

$$\sum_{|\alpha|>t} \widehat{C}(\alpha)^2 \leq \epsilon \qquad \text{for} \qquad t = \frac{(\log M)^{\Theta(d)} 2^{\Theta((\log M)^{2/3})}}{\epsilon^{(\log M)^{1/3}}}.$$

This can be viewed as a natural extension of the [ABFR94] result showing that Threshold-of-AC^0 cannot compute parity; we show that such circuits in fact exhibit strong Fourier concentration. (Thus, roughly speaking, our result is to [ABFR94] as the [LMN93] Fourier concentration bound for AC^0 is to the earlier AC^0 lower bounds of Håstad [Hås86].) We note that Fourier concentration bounds of the sort we establish were not known even for Majority-of-AC^0 prior to this work; the [JKS02] algorithm requires boosting and its analysis does not establish Fourier concentration.

With our Fourier concentration bound for Threshold-of-AC^0 in hand, applying the Low-Degree Algorithm of [LMN93] we get the first subexponential-time learning result for this class: any size-M, constant-depth Threshold-of-AC^0 can be learned to any constant accuracy ϵ in time $n^{2^{\Theta((\log M)^{2/3})}}$.

An important ingredient in our proof is a recent $2^{O(d)}n^{1-1/O(d)}$ upper bound on the total influence of degree-d polynomial threshold functions over n Boolean variables, proved recently by [HKM09] and [DRST09]. In 1994 Gotsman and Linial [GL94] conjectured a stronger bound, that every degree-d PTF has total influence $O(d\sqrt{n})$. We show that under the [GL94] conjecture our results become significantly stronger: every size-M depth-d Threshold-of-AC^0 circuit C has Fourier concentration

$$\sum_{|\alpha|>t} \widehat{C}(\alpha)^2 \leq \epsilon \qquad \text{for} \qquad t = \frac{2^{O(d)}(\log M)^d}{\epsilon^2}$$

and consequently such circuits can be learned to constant accuracy in time $n^{2^{O(d)}(\log M)^d}$.

We extend the above results by giving Fourier concentration and learning results for AC^0 circuits with r threshold gates in arbitrary locations in the circuit. We unconditionally learn such circuits with $r = O((\log M)^{1/3})$ many threshold gates, to any constant accuracy, in time $n^{2^{\Theta((\log M)^{2/3})}}$. Assuming the [GL94] conjecture, we learn such circuits with $r = O(\log \log M)$ to any constant accuracy in time $n^{2^{O(d)}(\log M)^{O(d)}}$. These results are achieved building on our results for Threshold-of-AC^0.

1.1.2 Lower bounds and separation results

To complement the positive (learning) results described above, in Section 6 we establish new lower bounds and separation results for AC^0 circuits augmented with threshold gates. These results separate the classes Majority-of-AC^0 and Threshold-of-AC^0 and highlight some interesting contrasts between them.

1. Since Majority-of-AC^0 is already known to be learnable in quasi-polynomial time, our learning results are only of interest if Threshold-of-AC^0 is actually a broader class than Majority-of-AC^0. We show that this is indeed the case, by exhibiting a single threshold gate for which any equivalent depth-d Majority-of-AC^0 circuit must have size $2^{\Omega(n^{1/(d-1)})}$. (See Section 6.1.)

2. Beigel [Bei94] showed that any size-s, depth-d circuit that contains m Majority gates is computed by a size-$2^{m(O(\log s))^{2d+1}}$, depth-$(d+2)$ circuit with a single Majority gate at the root. We show that this size bound cannot be improved to polynomial, by showing that a simple AND of two Majority gates requires any constant-depth circuit with a single Majority gate at the top (or even an arbitrary Threshold gate at the top) to have $n^{\Omega_d(\log n)}$ size. (See Section 6.2.)

3. A natural question is whether Beigel's result can be extended from Majority gates to arbitrary Threshold gates. Perhaps every AC^0 circuit which contains polylog(n) many Threshold gates is equivalent to a quasipoly(n)-size Threshold-of-AC^0? In fact the answer is no: we show that no analogue of Beigel's result is possible for Threshold gates, by showing that any Threshold-of-AC^0 circuit that computes the AND of two (high-weight) Threshold gates must have exponential size. (See Section 6.3.)

4. We also give lower bounds for AC^0 circuits with relatively many Threshold gates. We prove that any AC^0 circuit with $\epsilon \log n$ Threshold gates cannot compute parity, for a small constant $\epsilon > 0$. Previously, Aspnes et al. [ABFR94] proved this claim for AC^0 with a single threshold gate at the top. Beigel [Bei94] showed that any AC^0 circuit must be augmented with $n^{\Omega(1)}$ many Majority gates in order to compute parity. Our bound allows for a smaller number of gates augmenting the basic AC^0 circuit, but the gates (Threshold instead of Majority) are more powerful. (See Section 6.4.)

We note that the previous lower bounds on Threshold-of-AC^0 due to [ABFR94] apply to functions which have high PTF degree. This approach cannot be used for results (1) and (2) above, where we are proving lower bounds against functions which have low PTF degree.

2 Preliminaries

2.1 MAC^0 and TAC^0 and $\mathsf{TAC}^0[r]$

Recall that a *threshold function*, or halfspace, over n variables is a Boolean function $h : \{-1,1\}^n \to \{-1,1\}$, $h(x) = \operatorname{sign}\left(\sum_{i=1}^n w_i x_i - \theta\right)$, where $w_1, \ldots, w_n, \theta$ may be arbitrary real values. We will sometimes write Thr to denote a single threshold gate and Maj to denote a single Majority gate, where the Majority function is the threshold function for which each w_i equals 1 and the threshold θ equals 0.

A *Threshold-of-*AC^0 *circuit*, or TAC^0, is a circuit consisting of a threshold function (with arbitrary weights and fanin) as the output gate and AC^0 circuits feeding into it. A depth-d TAC^0 is one in which each of the AC^0 circuits feeding into the output threshold gate has depth at most $d-1$. The size of a TAC^0 is the total number of gates (so in particular, in a size-M TAC^0 each of the AC^0 circuits is of size at most M).

A *Majority-of-*AC^0 *circuit*, or MAC^0, is a TAC^0 in which the top threshold function is a majority gate.

Finally, we will also consider AC^0 circuits that have r arbitrary Thr gates buried at arbitrary locations in the circuit; we refer to such a circuit as a "Threshold-of-r-AC^0s", or $\mathsf{TAC}^0[r]$.

Our arguments involve polynomial threshold functions, influence of variables on Boolean functions, noise sensitivity, and the basics of Fourier analysis. We briefly define the relevant notions and then recall some facts we will need about how random restrictions affect AC^0 circuits. Then in Section 2.5 we explain the high-level idea of our Fourier concentration bound for TAC^0 circuits.

2.2 Polynomial Threshold Functions

Threshold functions are equivalent to degree-1 *polynomial threshold functions*; higher-degree polynomial threshold functions will play an important role in our proofs. A Boolean function $f : \{-1,1\}^n \to \{-1,1\}$ is said to be a degree-k polynomial threshold function (PTF) if $f(x) = \operatorname{sign}(p(x))$ for all $x \in \{-1,1\}^n$, where $p(x)$ is a real-valued polynomial of degree at most k. (Since we are dealing with Boolean inputs the polynomial p can always be taken without loss of generality to be multlinear.) If $p(x)$ is a polynomial with all integer coefficients, we say the *weight* of the PTF $\operatorname{sign}(p)$ is the sum of the absolute value of those integer coefficients.

2.3 Fourier background, Influence, and Noise Sensitivity

We briefly recall the rudiments of Fourier analysis over the Boolean hypercube. Every real-valued function $f : \{-1,1\}^n \to \{-1,1\}$ has a unique *Fourier representation* as a linear combination of parity basis functions χ_α:

$$f(x) = \sum_{\alpha \subseteq [n]} \widehat{f}(\alpha)\chi_\alpha(x).$$

Note that this is precisely the unique representation of f as a multilinear polynomial, since (over $\{-1,1\}^n$ inputs) the parity function $\chi_\alpha(x)$ is simply the monomial $\prod_{i\in\alpha} x_i$.

Plancherel's identity says that $\mathbf{E}[fg] = \sum_\alpha \widehat{f}(\alpha)\widehat{g}(\alpha)$ for all f, g; in particular this implies that for every Boolean function f with range $\{-1,1\}$, we have $\sum_\alpha \widehat{f}(\alpha)^2 = 1$.

The *Fourier degree*, or simply *degree*, of a Boolean function f is the size of the largest $\alpha \subseteq [n]$ such that $\widehat{f}(\alpha) \neq 0$. We denote this $\deg(f)$. We recall the easy fact that if f is computed by a decision tree of depth d, then $\deg(f) \leq d$.

The *influence* of variable i on a Boolean function $f : \{-1,1\}^n \to \{-1,1\}$ is defined to be $\Pr[f(x) \neq f(x \oplus e_i)]$, where x is uniform from $\{-1,1\}^n$ and $x + e_i$ denotes x with the i-th bit flipped. The *total influence* of f is $\mathrm{Inf}(f) = \sum_{i=1}^n \mathrm{Inf}_i(f)$.

For $f : \{-1,1\}^n \to \{-1,1\}$ and $\epsilon > 0$, the *noise sensitivity* of f at noise rate ϵ is defined to be

$$\mathrm{ns}_\epsilon(f) = \Pr_{x,y}[f(x) \neq f(y)]$$

where x is uniform from $\{-1,1\}^n$ and y is obtained from x by flipping each bit independently with probability ϵ.

2.4 Random Restrictions and AC^0

We write "$\rho \sim \mathcal{R}_p$" to indicate that ρ is a *random restriction with parameter* p. Such a restriction ρ is chosen by independently fixing each variable to $+1$ or -1 each with probability $\frac{1-p}{2}$, and leaving the variable unfixed with probability p. We write f_ρ to denote the function that results from applying ρ to f.

We will use several facts from [Hås86, LMN93] about the behavior of AC^0 circuits under random restrictions. The first of these facts is Håstad's Switching Lemma:

Lemma 1 ([Hås86]) *Let C be a depth-2 circuit (i.e. a DNF or a CNF) of bottom fan-in s. Then $\Pr_\rho[C_\rho$ cannot be written as a depth-t decision tree$] \leq (5ps)^t$, where ρ is a random restriction with parameter p.*

(The above statement is implicit in [Hås86] and is made explicit in e.g. [Hås01].) Repeated applications of the Switching Lemma can be used to prove the following in a rather straightforward way:

Lemma 2 ([LMN93], Lemma 2) *If C is a size-M depth-d AC^0 circuit, then for any $t \geq 0$ we have $\Pr_\rho[C_\rho$ cannot be written as a depth-t decision tree$] \leq M2^{-t}$, where ρ is a random restriction with parameter $p = \frac{1}{10^d t^{d-1}}$.*

([LMN93] actually state a slightly weaker form in which the LHS is replaced by "$\Pr_\rho[\deg(C_\rho) > t]$." It is easy to check that using Lemma 1, the [LMN93] proof directly yields Lemma 2 as stated above.)

2.5 Sketch of the Random Restriction Argument.

The high-level idea of our proof is quite simple, and is similar to the high-level idea of [LMN93]. We show that when a TAC^0 is hit with a random restriction, with high probability it collapses into a "much simpler function," specifically a low-degree PTF. Recent results on the Fourier concentration of low-degree PTFs due to [DRST09, HKM09] let us infer that the original TAC^0 must similarly have had good Fourier concentration. In the rest of this section we elaborate on this argument.

We begin by recalling the basic outline of [LMN93]'s Fourier concentration bound for AC^0 circuits. It will be useful for us to view the [LMN93] argument as proceeding in two stages:

1. The first stage analyzes what happens to a size-M, depth-d AC^0 circuit C when it is hit with a random restriction with parameter $p \approx \frac{1}{(\log M)^{d-1}}$ (recall that p is the probability that a variable "survives" the restriction, i.e. is left unfixed). [LMN93] show that with high probability such a restriction causes C_ρ to collapse down to a $(\log M)$-depth decision tree.

2. The second stage is the observation that a $(\log M)$-depth decision tree T, being a degree $\log M$ polynomial, has extremely strong Fourier concentration: $\sum_{|\alpha|>\log M} \widehat{T}(\alpha)^2 = 0$. Linial et al. then use the Fourier concentration of C_ρ to argue that the original AC^0 function computed by C must have had most of its Fourier weight at levels $\leq (\log M)^d$.

Our argument for TAC^0 has a similar high-level structure, but with some significant differences in both stages. Let C now denote a size-M, depth-d TAC^0 circuit.

1′. In the first stage, we consider hitting C with a "stronger" random restriction with a smaller value of p (so fewer variables survive the restriction). We show that with high probability such a restriction causes C_ρ to collapse down to a "low-degree" PTF of degree $k \ll \log M$. The stronger restriction is necessary since the results of [DRST09, HKM09] are non-trivial only when the degree of the PTF is $o(\sqrt{\log n})$.

2′. The results of [DRST09, HKM09] imply that C_ρ must have some nontrivial Fourier concentration. The Fourier concentration for C_ρ is much weaker than what one gets for decision trees, but one can adapt the original [LMN93] argument to show that the original circuit C itself must have had some Fourier concentration.

The conjecture of Gotsman & Linial significantly strengthens the bounds on total influence and noise sensitivity of low-degree PTFs that are currently known; it implies non-trivial bounds as long as the degree is $o(\sqrt{n})$. This in turn strengthens the Fourier concentration that we get for C_ρ in Stage 2′, and hence also for C. We present each of the stages of the above argument in as self-contained a way as possible in Section 3. Section 4 puts the pieces together to prove the main results.

3 Random Restrictions of TAC^0.

3.1 Stage 1: Collapsing TAC^0 to a low-degree PTF

In this section we prove the following:

Lemma 3 *Let C be a size-M, depth-d TAC^0. Let ρ be a random restriction with parameter p (specified below) and let $k \geq 1$. Then for any $0 < p' < 1$, with failure probability at most δ the function C_ρ is a degree-k PTF, where*

$$\delta = M^{-2} + M^5 \left(\frac{4e\log(M)p'}{k}\right)^k \quad \textit{and} \quad p = \frac{1}{10^{d-1}(4\log M)^{d-2}} \cdot p'.$$

Proof: The proof is conceptually quite simple. Let $C = \mathsf{Thr}(C_1, \ldots, C_\ell)$ where Thr is the topmost threshold gate, $\ell \leq M$ is its fan-in, and each C_i is an AC^0 circuit of depth at most $d-1$ and size M_i, where $M_1, \ldots, M_\ell \leq M$. We view the restriction ρ as being obtained in two steps. The first step collapses each C_i to a decision tree of depth $O(\log M)$. The second step significantly reduces the depth of each decision tree, down to k. After these two steps, with high probability each C_i has collapsed down to $(C_i)_\rho$ which is a degree-k polynomial. Thus C_ρ is a PTF of degree k.

In the first step we take a random restriction ρ_1 with parameter $p_1 = \frac{1}{10^{d-1}(4\log M)^{d-2}}$. For a given i, Lemma 2 gives that with failure probability at most $M_i \cdot M^{-4}$, the function $(C_i)_{\rho_1}$ is equivalent to a decision tree T_i of depth $4\log M$. Summing failure probabilities over all $i = 1, \ldots, \ell$, this occurs for every C_i with overall failure probability at most $(M_1 + \cdots + M_\ell)M^{-4} \leq M^{-2}$.

In the second step, we take a random restriction with parameter p' (thus the overall probability that a variable survives the combined restriction is $p = p_1 p'$ as desired). The following simple lemma analyzes the effect of a random restriction on a depth-t decision tree:

Lemma 4 *Let T be a depth-t decision tree and ρ be a random restriction with parameter p'. Then for $k \geq 1$, we have $\Pr[T_\rho$ cannot be written as a depth-k decision tree$] \leq 2^t \left((etp')/k\right)^k$.*

Proof: Suppose that under ρ at most k variables survive in each root-to-leaf path in T. Then it is clear that T_ρ can be written as a decision tree of depth at most k. So fix any given path of length at most t in T; wlog the variables appearing on this path are $x_1, \ldots, x_t$. The probability that at least k of these variables survive ρ is at most

$$\binom{t}{k}(p')^k \leq \left(\frac{et}{k}\right)^k (p')^k = \left(\frac{etp'}{k}\right)^k.$$

A union bound over all (at most 2^t) paths in T finishes the proof. ■

We apply this lemma to each of the $\ell \leq M$ decision trees T_i from step 1, taking $t = 4\log M$. A union bound gives that the probability that any T_i fails to have its depth reduced to k is at most $M \cdot 2^t \cdot \left(etp'/k\right)^k$. Any decision tree of depth k is exactly computed by a Fourier polynomial of degree at most k; the top-level Thr gate takes the sign of a weighted sum of these polynomials, and we obtain Lemma 3. ■

3.2 Stage 2: From Fourier concentration of C_ρ to Fourier concentration of C

We will use the following recent bound on the noise sensitivity of degree-k PTFs due to Diakonikolas et al. [DRST09] and Harsha et al. [HKM09] (see Appendix 2.3 for the definition of $\mathrm{ns}_\epsilon(f)$, the noise sensitivity of f at noise rate ϵ):

Theorem 5 *For any degree-k PTF f over $\{-1,1\}^n$ and any $0 \leq \epsilon \leq 1$, we have $\mathrm{ns}_\epsilon(f) \leq 2^{O(k)} \cdot \epsilon^{\frac{1}{O(k)}}$.*

The following simple result (Corollary 17 of [KOS04]) converts noise sensitivity upper bounds to Fourier concentration bounds:

Lemma 6 *Let $f : \{-1,1\}^n \to \{-1,1\}$ be any Boolean function and let $\kappa : [0,1/2] \to \mathbf{R}^+$ be an increasing function such that $\mathrm{ns}_\epsilon(f) \leq \kappa(\epsilon)$. Then*

$$\sum_{|\alpha| \geq m} \widehat{f}(\alpha)^2 \leq \epsilon \quad \textit{for} \quad m = \frac{1}{\kappa^{-1}(\epsilon/2.32)}.$$

Plugging in Theorem 5 gives the following Fourier concentration bound:

Corollary 7 *For any degree-k PTF f over $\{-1,1\}^n$ and any $0 \leq \epsilon \leq 1$, we have*

$$\sum_{|\alpha| \geq m(\epsilon)} \widehat{f}(\alpha)^2 \leq \epsilon \ \textit{ where } \ m(\epsilon) = \frac{2^{\Theta(k^2)}}{\epsilon^{\Theta(k)}}.$$

We now show that if f_ρ has good Fourier concentration (w.h.p. over the choice of random restriction ρ), then f itself has good Fourier concentration. This is done by the following lemma, adapting arguments from [LMN93].

Lemma 8 *Let $f : \{-1,1\}^n \to \{-1,1\}$ and let t, p be parameters such that $pt > 8$. Then*

$$\sum_{|\alpha|>t} \widehat{f}(\alpha)^2 \leq 2\mathbf{E}_\rho[\sum_{|\beta|>pt/2} \widehat{f_\rho}(\beta)^2],$$

where ρ is a random restriction with parameter p.

Proof: Lemma 6 of [LMN93] gives us

$$\sum_{|\alpha|>t} \widehat{f}(\alpha)^2 \leq 2\mathbf{E}_\gamma[\sum_{|\alpha\cap\gamma|>pt/2} \widehat{f}(\alpha)^2]$$

where $\gamma \subseteq [n]$ is a subset chosen at random by including each variable independently with probability p. [LMN93] also show that for any subset $\gamma \subseteq [n]$ and any k,

$$\sum_{\alpha:|\alpha\cap\gamma|>k} \widehat{f}(\alpha)^2 = \mathbf{E}_R[\sum_{|\beta|>k} \widehat{f_{\overline{\gamma}\leftarrow R}}(\beta)^2]$$

where R is a ± 1 assignment to the variables in $\overline{\gamma}$ chosen at random. Combining these, we have

$$\sum_{|\alpha|>t} \widehat{f}(\alpha)^2 \leq 2\mathbf{E}_\gamma[\mathbf{E}_R[\sum_{|\beta|>pt/2} \widehat{f_{\overline{\gamma}\leftarrow R}}(\beta)^2]] = 2\mathbf{E}_\rho[\sum_{|\beta|>pt/2} \widehat{f_\rho}(\beta)^2],$$

since the combination of randomly choosing γ and R as described is exactly equivalent to choosing a random restriction ρ with parameter p. ■

As an easy corollary of Lemma 8 we have the following:

Corollary 9 *Let $f : \{-1,1\}^n \to \{-1,1\}$ and let t, p be parameters such that $tp > 8$. Suppose that with probability at least $1-\delta$ (over the choice of a random restriction ρ with parameter p) the function f_ρ has Fourier concentration $\sum_{|\beta|>pt/2} \widehat{f_\rho}(\beta)^2 \leq \epsilon$. Then we have $\sum_{|\alpha|>t} \widehat{f}(\alpha)^2 \leq 2\epsilon + 2\delta$.*

(This follows from the lemma because $\widehat{f_\rho}$ is a Boolean function and consequently always has total Fourier weight at most 1.)

4 Proof of the Fourier concentration results for TAC^0

Throughout this section C is a size-M, depth-d TAC^0. The regime we are most interested in is when the circuit size M is poly(n) and the error parameter ϵ is something like a small constant; in particular, we are most interested in situations where $\epsilon > M^{-1}$. (We note that even the Majority function has $\widehat{f}([n])^2 = \Theta(1/n)$, so Fourier concentration bounds for TAC^0 must certainly be vacuous for $\epsilon < 1/n$.)

4.1 The unconditional result

Putting together all the pieces, we have established a Fourier concentration bound for TAC^0:

Theorem 10 *Let C be a size-M, depth-d TAC^0. Let $\epsilon \geq 2M^{-2}$. Then C has Fourier concentration*

$$\sum_{|\alpha|>t} \widehat{C}(\alpha)^2 \leq 4\epsilon \qquad \textit{for} \qquad t = \frac{(\log M)^{\Theta(d)} \cdot 2^{\Theta((\log M)^{2/3})}}{\epsilon^{\Theta((\log M)^{1/3})}}. \tag{1}$$

Proof: In Stage 1 we shall take (with foresight) $k = (\log M)^{1/3}$ and $p' = \frac{k}{4eM^{7/k}\log M}$, so consequently $p = \frac{k}{(40\log M)^{d-1}\cdot e\cdot M^{7/k}}$. This choice of parameters gives failure probability at most $\delta = 2M^{-2}$ in Lemma 3, so with this failure probability we have that C_ρ is a degree-k PTF which satisfies

$$\sum_{\alpha \geq m} \widehat{f}(\alpha)^2 \leq \epsilon \text{ where } m = \frac{2^{\Theta((\log M)^{2/3})}}{\epsilon^{\Theta((\log M)^{1/3})}}.$$

In Step 3, we take $t = 2m/p$ so $tp/2 = m$ which is at least 8. Corollary 9 thus gives us

$$\sum_{|\alpha|>t} \widehat{C}(\alpha)^2 \leq 2\epsilon + 2\delta \text{ where } t = \frac{(\log M)^{\Theta(d)} \cdot 2^{\Theta((\log M)^{2/3})}}{\epsilon^{\Theta((\log M)^{1/3})}}.$$

■

Applying the well-known [LMN93] machinery for uniform distribution learning of Boolean functions with good Fourier concentration, we get the following:

Corollary 11 *Size-M depth-d TAC^0 circuits can be learned to accuracy ϵ in time n^t where*

$$t = \frac{(\log M)^{\Theta(d)} \cdot 2^{\Theta((\log M)^{2/3})}}{\epsilon^{\Theta((\log M)^{1/3})}}.$$

Thus as long as $\epsilon \geq 1/2^{O((\log M)^{1/3}))}$ and $d \leq O((\log M)^{2/3}/(\log\log M))$ this gives an algorithm to learn size-M depth-d TAC^0 in time $n^{2^{\Theta((\log M)^{2/3})}}$, i.e. sub-exponential time ($2^{n^{o(1)}}$) for any $M = \mathrm{poly}(n)$.

4.2 The Gotsman-Linial Conjecture and its consequences

In 1994 Gotsman and Linial [GL94] asked the question of what is the maximum total influence of any degree-k PTF over n variables. They conjectured that the symmetric function which changes sign on the k middle layers of the Boolean hypercube has the highest total influence of any degree-k PTF (it is easy to see that this function is indeed a degree-k PTF). Since each layer of edges in the Boolean hypercube contains at most $\sqrt{n}2^{n-1}$ edges, a direct consequence of their conjecture (which is nearly equivalent to it for $k = o(\sqrt{n})$) is the following:

Conjecture 12 ([GL94]) *Every degree-k PTF f over n variables has $\mathrm{Inf}(f) \leq k\sqrt{n}$.*

We show that using our approach, Conjecture 12 yields significantly improved Fourier concentration (and significantly more efficient learnability) for TAC^0. The noise sensitivity bounds of [DRST09] and [HKM09] follow from a bound of $2^{O(k)}n^{1-1/O(k)}$ on the average sensitivity of degree-k PTFs. This bound becomes trivial for $k = \Omega(\sqrt{\log n})$, and hence we needed to use a very strong random restriction in order to reduce our initial TAC^0 to a PTF of degree $o(\sqrt{\log n})$. Conjecture 12 implies that a weaker random restriction will suffice. We use the following noise sensitivity and Fourier concentration consequences of the Gotsman-Linial conjecture:

Corollary 13 *If Conjecture 12 holds, then for any degree k PTF f over $\{-1,1\}^n$ and any $0 \leq \epsilon \leq 1$,*

$$\mathrm{ns}_\epsilon(f) \leq 2k\sqrt{\epsilon} \qquad \textit{and} \qquad \sum_{|\alpha| \geq m} \widehat{f}(\alpha)^2 \leq \epsilon \ \textit{where} \ m = \frac{24k^2}{\epsilon^2}.$$

The first inequality follows from the reduction from total influence to noise sensitivity for PTFs given in [DRST09] (see Section 7), and the second inequality then follows from Lemma 6. We thus obtain:

Theorem 14 *Let C be a size-M, depth-d TAC^0. Let $\epsilon \geq 2M^{-2}$. If Conjecture 12 is true, then we have*

$$\sum_{|\alpha| > t} \widehat{C}(\alpha)^2 \leq 4\epsilon \qquad \textit{for} \qquad t = \frac{2^{O(d)}(\log M)^d}{\epsilon^2}.$$

Proof: In Stage 1 we shall take $k = \log M$ and $p' = 10^{-4}$, so $p = \frac{1}{10^{d+3}(4 \log M)^{d-2}}$. Lemma 3 gives that with probability at least $1 - 2M^{-2}$, the function C_ρ is a degree-k PTF, in which case we have, for $m = 24k^2/\epsilon^2$, $\sum_{|\alpha|>m} \widehat{C_\rho}(\alpha)^2 \leq \epsilon$. For Stage 3, in Corollary 9 we take $t = \frac{2m}{p} = \frac{(c_1 \log M)^d}{\epsilon^2}$ for some absolute constant c_1. Corollary 9 thus gives us $\sum_{|\alpha|>t} \widehat{C}(\alpha)^2 \leq 2\epsilon + 2\delta$. ■

Similar to before, the [LMN93] low-degree algorithm gives us:

Corollary 15 *If Conjecture 12 is true, then size-M, depth-d TAC^0 can be learned to accuracy ϵ in time*

$$n^{\frac{2^{O(d)}(\log M)^d}{\epsilon^2}}.$$

This gives quasi-polynomial time learning for $M = \mathrm{poly}(n)$-size TAC^0 for any constant (or even $1/\mathrm{polylog}(n)$) accuracy ϵ.

5 Learning $\mathsf{TAC}^0[r]$

Our learning results can be extended from TAC^0 circuits to $\mathsf{TAC}^0[r]$ circuits for small (but superconstant) values of r. The high-level approach is as follows: We first prove a general result showing that if a class $\mathcal{C}$ has Fourier concentration, then any R-junta-of-functions-from-$\mathcal{C}$ must also have fairly good Fourier concentration provided that R is not too large. We then argue that any $\mathsf{TAC}^0[r]$ is equivalent to a R-junta-of-TAC^0 for $R = (r+1)2^r$. This lemma and the arguments used in its proof are similar to arguments found in [BRS95]. Combining the above two ingredients with the Fourier concentration bounds for TAC^0 which we obtained in Section 4, we get Fourier concentration bounds for $\mathsf{TAC}^0[r]$.

We show unconditionally that $\mathsf{TAC}^0[O((\log M)^{1/3})]$ circuits can be learned in essentially the same time bound that we achieved for unconditionally learning TAC^0 circuits:

Theorem 16 *The class of $\mathsf{TAC}^0[O((\log M)^{1/3})]$ circuits of size M and depth d can be learned to accuracy ϵ (for $\epsilon > 2M^{-2}$) in time n^t, where*

$$t = (\log M)^{\Theta(d)} \cdot 2^{\Theta((\log M)^{2/3})} \epsilon^{-\Theta((\log M)^{1/3})}.$$

Assuming the Gotsman-Linial conjecture, we obtain

Theorem 17 *If Conjecture 12 is true, then the class of $\mathsf{TAC}^0[r]$ circuits of size M and depth d can be learned to accuracy ϵ (for $\epsilon > 2M^{-2}$) in time n^t, where*

$$t = 2^{O(d+r)} \cdot (\log M)^{3d} \epsilon^{-3}.$$

For constant d this gives quasi-polynomial time learning for r as large as $O(\log \log M)$.

5.1 Fourier concentration of $\mathcal{C}$ implies Fourier concentration of junta-of-$\mathcal{C}$

Let $\mathcal{C}$ be a class of Boolean functions. Any function of the form $h = g(f_1, \ldots, f_R)$, where each f_i belongs to $\mathcal{C}$ and g is an arbitrary R-argument Boolean function, is said to be an *R-junta-of-$\mathcal{C}$*. Let us assume that every function $f \in \mathcal{C}$ satisfies the Fourier concentration bound:

$$\sum_{|\alpha| > \ell(\epsilon)} \widehat{f}(\alpha)^2 \leq \epsilon. \tag{2}$$

We first convert this to a statement about noise stability:

Lemma 18 *Set $\delta \leq \frac{\epsilon}{\ell(\epsilon)}$. For any function $f \in \mathcal{C}$, we have $\mathrm{ns}_\delta(f) \leq 2\epsilon$.*

Proof: A simple calculation (see e.g. [KOS04]) shows that

$$\mathrm{ns}_\delta(f) = \frac{1}{2} \sum_{\alpha} \widehat{f}(\alpha)^2 (1 - (1-2\delta)^{|\alpha|}).$$

For $|\alpha| \leq \ell(\epsilon)$, we have $(1 - (1-2\delta)^{|\alpha|}) \leq 2\delta\ell(\epsilon)$. Hence

$$\begin{aligned} 2\mathrm{ns}_\delta(f) &= \sum_{\alpha} \widehat{f}(\alpha)^2 (1 - (1-2\delta)^{|\alpha|}) \\ &\leq \sum_{|\alpha| \leq \ell(\epsilon)} \widehat{f}(\alpha)^2 (1 - (1-2\delta)^{|\alpha|}) + \sum_{|\alpha| > \ell(\epsilon)} \widehat{f}(\alpha)^2 \\ &\leq (2\delta\ell(\epsilon))(\sum_{|\alpha| \leq \ell(\epsilon)} \widehat{f}(\alpha)^2) + \epsilon \leq 3\epsilon. \end{aligned}$$

■

Recalling the definition of noise sensitivity, for $h = g(f_1, \ldots, f_R)$ any R-junta-of-$\mathcal{C}$ we have that $h(x) \neq h(y)$ only if $f_i(x) \neq f_i(y)$ for some i. Thus the union bound gives:

Corollary 19 *Let h be any R-junta-of-$\mathcal{C}$, and let $\delta \leq \frac{\epsilon}{\ell(\epsilon)}$. Then $\mathrm{ns}_\delta(h) \leq 2R\epsilon$.*

This noise stability bound for h can of course be converted into a Fourier concentration bound using Lemma 6; we do this in Section 5.3 below.

5.2 Every TAC^0 is a junta-of-TAC^0

Lemma 20 *Let C be a depth-d, size-M $\mathsf{TAC}^0[r]$ circuit. Then C is equivalent to a $((r+1)2^r)$-junta of TAC^0 circuits, each of which has depth at most d and size at most M.*

Proof: By increasing the number of Thr gates in C to $r+1$, we may assume that the output gate of C is itself a Thr gate. Let $\mathsf{Thr}_1, \ldots, \mathsf{Thr}_{r+1}$ denote the $r+1$ Thr gates in C, where Thr_1 is the root gate.

For each $i \in [r+1]$, let C_i denote the sub-circuit of C whose root is Thr_i (note that C_1 is equivalent to C). Let $n_i \in \{0, 1, \ldots, r\}$ denote the number of Thr gates that lie below the root Thr_i in C_i (so $n_1 = r$).

For each $i \in [r+1]$, for each n_i-bit string $b = (b_1, \ldots, b_{n_i})$, let $C_i^{(b)}$ denote the circuit obtained by replacing the j-th of the n_i Thr gates occurring below the root in C_i with the bit b_j, for all $j \in [n_i]$. Note that for each i and each b, the circuit $C_i^{(b)}$ is a TAC^0 of depth at most d and size at most M. Note also that there are at most $(r+1)2^r$ circuits $C_i^{(b)}$.

The lemma follows on observing that for every input string $x \in \{0,1\}^n$, the value of $C(x) = C_1(x)$ is completely determined by the values of all the $C_i^{(b)}(x)$'s in a bottom-up fashion. ■

As a simple example to illustrate how $C(x)$ is determined as a function of the $C_i^{(b)}(x)$'s, consider a circuit C containing $k = 3$ Thr gates: one root gate Thr_1 which has two Thr gates, Thr_2 and Thr_3, among its inputs (note that Thr_1 may have other inputs which are AC^0 circuits, and Thr_2 and Thr_3 may have AC^0 circuits as inputs). The value $C(x)$ is computed by the 6-junta-of-TAC^0 which works in the following way (we write ε to denote the empty string):

"For all $(b_1, b_2) \in \{0,1\}^2$, if $(C_2^{(\varepsilon)}(x), C_3^{(\varepsilon)}(x)) = (b_1, b_2)$ then output $C_1^{(b_1,b_2)}(x)$."

5.3 Learning $\mathsf{TAC}^0[r(n)]$

Let $h : \{-1,1\}^n \to \{-1,1\}$ be a depth-d, size-M $\mathsf{TAC}^0[r]$ circuit and let $R = (r+1)2^r$. Let

$$\ell(\epsilon) = \frac{(\log M)^{\Theta(d)} \cdot 2^{\Theta((\log M)^{2/3})}}{(\epsilon/R)^{\Theta((\log M)^{1/3})}}$$

so by Theorem 10 (assuming $\epsilon \geq 2M^{-2}$) we have that any size-M depth-d TAC^0 f satisfies $\sum_{|\alpha|>\ell(\epsilon)} \widehat{f}(\alpha)^2 \leq \epsilon/R$. Putting together Lemma 20 and Corollary 19, we see that for $\delta = \frac{\epsilon}{10\ell(\epsilon/10)}$ we have $\mathrm{ns}_\delta(h) \leq \epsilon/5$. We may rephrase this as $\mathrm{ns}_\epsilon(h) \leq \kappa(\epsilon)$, where

$$\kappa(\epsilon) = R\left(\epsilon \cdot (\log M)^{\Theta(d)} \cdot 2^{\Theta((\log M)^{2/3})}\right)^{1/\Theta((\log M)^{1/3})}.$$

Now appealing to Lemma 6, we get

$$\sum_{|\alpha| \geq m(\epsilon)} \widehat{f}(\alpha)^2 \leq \epsilon \quad \text{where} \quad m(\epsilon) = \frac{R^{\Theta((\log M)^{1/3})} \cdot (\log M)^{\Theta(d)} \cdot 2^{\Theta((\log M)^{2/3})}}{\epsilon^{\Theta((\log M)^{1/3})}}.$$

This is easily seen to give Theorem 16.

Assuming the Gotsman-Linial conjecture, we apply Theorem 14 instead of Theorem 10 and we take

$$\ell(\epsilon) = \frac{2^{O(d)}(\log M)^d}{(\epsilon/R)^2}.$$

This yields

$$\sum_{|\alpha| \geq m(\epsilon)} \widehat{f}(\alpha)^2 \leq \epsilon \quad \text{where} \quad m(\epsilon) = \frac{R^2 \cdot 2^{O(d)} \cdot (\log M)^{3d}}{\epsilon^3},$$

and we obtain Theorem 17.

6 Lower bounds

6.1 MAC^0 cannot compute TAC^0

In this section we prove that there are TAC^0 circuits that have no small equivalent MAC^0 circuit.

Theorem 21 *There is a threshold function over $N = O(n^2)$ variables such that any equivalent MAC^0 circuit of depth $d \geq 2, d = \Theta(1)$ must have size $2^{\Omega(n^{1/(d-1)})}$.*

The desired function is the function $U_{n,4n}(x)$ defined by Goldmann *et al.* in Section 4 of [GHR92]:

$$U_{n,4n}(x) = \mathrm{sign}(2r_{n,4n}(x) + 1), \qquad r_{n,4n}(x) = \sum_{i=0}^{n-1} \sum_{j=0}^{4n-1} 2^i x_{ij}, \tag{3}$$

where all variables take values ± 1.

It is clear that $U_{n,4n}$ is a TAC^0 circuit (of depth 1), consisting of a single threshold gate over $N = 4n^2$ input variables. It remains to show that any depth-d MAC^0 circuit for $U_{n,4n}(x)$ must be large. We do this in two steps as follows. Suppose that C is a depth-d, size-M MAC^0 circuit that computes $U_{n,4n}(x)$. If $M = 2^{\Omega(n^{1/(d-1)})}$ then there is nothing to show, so we assume $M = 2^{O(n^{1/(d-1)})}$. We shall consider the effect of applying a random restriction with parameter $r = \frac{1}{10^{d-1}s^{d-2}}$ to C, where we select $s = 3 \log M$. We will establish the following two lemmas:

Lemma 22 *With probability at least $1 - M^{-1}$ over the random choice of ρ, the function $(U_{n,4n})_\rho$ is a polynomial threshold function of total weight at most M^7.*

Lemma 23 *With probability at least $1 - 2n^{-2}$ over the random choice of ρ, the function $(U_{n,4n})_\rho$ has a sub-function (obtained by possibly fixing some additional variables in $(U_{n,4n})_\rho$) that is equivalent, up to renaming variables, to $U_{m,4m}$ where $m = \Omega(n/(\log M)^{d-2})$.*

Fix a restriction ρ that satisfies both Lemmas (such a ρ must exist since each of the two events has probability greater than $1/2$). The function $U_{m,4m}$ is a restriction of the function $(U_{n,4n})_\rho$ from Lemma 22, and thus $(U_{m,4m})_\rho$ must have a polynomial threshold function of weight at most M^7. However, the discussion following Corollary 8 of [GHR92] shows that the total weight of any PTF for $U_{m,4m}$ must be at least $\Omega(2^{m/2}/\sqrt{m})$. Since $m = \Omega(n/(\log M)^{d-2})$, straightforward manipulation yields the desired lower bound $M = 2^{\Omega(n^{1/(d-1)})}$ and proves Theorem 21.

Proof of Lemma 22: Fix a sub-circuit C' that is one of the inputs to the Majority gate (so the depth of C' is at most $d-1$ and the size is at most M). Lemma 2 implies that with probability at least $1 - M^{-2}$ we have that $(C')_\rho$ is a decision tree of depth at most $3 \log M$ and thus $\deg((C')_\rho) \leq 3 \log M$ (see Section 2.3). We now recall the easy fact (from [DLM$^+$07]) that if a function $g : \{-1,1\}^n \to \{-1,1\}$ has $\deg(g) \leq k$, then every Fourier coefficient of g is of the form (integer)$/2^{k-1}$. Hence we may rewrite $(C')_\rho$ as

$$(C')_\rho(x) = \frac{1}{2^{\lfloor 3 \log M \rfloor}} \sum_\alpha a_\alpha \chi_\alpha(x)$$

where the a_α's are integers whose squares sum to $2^{2 \cdot \lfloor 3 \log M \rfloor} \leq M^6$ and hence $\sum_\alpha |a_\alpha| \leq M^6$.

We apply this analysis to each of the (at most M) sub-circuits that feed into the Majority gate. Summing failure probabilities, we get that with overall probability at least $1 - M^{-1}$, the sum of $(C')_\rho$ (summed over all sub-circuits C' that are input to the Majority gate) can be expressed as $\frac{1}{2^{\lfloor 3 \log M \rfloor}} \times$ (some integer linear combination of parities), where the sum of the magnitudes of the integer coefficients is at most M^7. We may rescale by multiplying by $2^{\lfloor 3 \log M \rfloor}$, and thus obtain Lemma 22. ■

Proof of Lemma 23: Fix any $i \in \{0, 1, \ldots, n-1\}$. Let $live_i$ denote the number of variables x_{ij} that survive the random restriction ρ. We have $\mathbf{E}_\rho[live_i] = 4rn$ and a standard Chernoff bound gives that $\Pr_\rho[live_i \leq 2rn] \leq e^{-rn/2}$. A union bound over all n possibilities for i gives

$$\Pr_\rho[live_i \leq 2rn \text{ for any } i] \leq ne^{-rn/2} \ll n^{-2},$$

where the last inequality follows from our assumption that $M = 2^{O(n^{1/(d-1)})}$ (recall that $r = \frac{1}{10^{d-1}(3\log M)^{d-2}}$).

For a given i, let pos_i denote the number of variables x_{ij} that are set to $+1$ by ρ, and let neg_i denote the number of variables x_{ij} that are set to -1 by ρ. Another standard Chernoff bound gives that for each fixed i, we have $\Pr_\rho[|pos_i - neg_i| > 10\sqrt{n\log n}] \leq n^{-3}$ (with room to spare). So we further get

$$\Pr_\rho[|pos_i - neg_i| \geq 10\sqrt{n\log n} \text{ for any } i] \leq n^{-2}.$$

We henceforth assume that for every i we have $live_i > 2rn$ and $|pos_i - neg_i| < 10\sqrt{n\log n}$.

We now observe that for any fixed choice of i, by setting $|pos_i - neg_i| < 10\sqrt{n\log n}$ of the variables x_{ij} that survive ρ, it is possible to "undo" any nonzero contribution to a constant term in $(r_{n,4n})_\rho$ that came from summands of the form $2^i\rho_{ij}$ (i.e., that came from variables x_{ij} that were set by ρ). Let $\rho^\star$ denote the combined restriction obtained by extending ρ in this way for all i. For each i, the restriction $\rho^\star$ keeps at least $2rn - 10\sqrt{n\log n}$ variables of the form x_{ij} free. It is straightforward to check that $2rn - 10\sqrt{n\log n}$ is at least $m = \Omega(n/(\log M)^{d-2})$. Thus the restriction $(U_{n,4n})_{\rho^\star}$ can be restricted to yield a sub-function equivalent to $U_{n,4m}$, and it is clear that $U_{n,4m}$ can be restricted to yield $U_{m,4m}$. This gives Lemma 23. ■

6.2 Lower bounds on MAC^0

Beigel [Bei94] showed that any size-s, depth-d circuit that contains m Maj gates is computed by a size-$2^{m(O(\log s))^{2d+1}}$, depth-$(d+2)$ circuit with a single Maj gate at the root. It is natural to ask whether this simulation can be improved to a polynomial-size (rather than quasi-polynomial) Maj of AC^0. In this section we observe that no such strengthened version of Beigel's theorem can exist, by proving that there is no polynomial-size MAC^0 (or even TAC^0) for an AND of two Maj gates:

Theorem 24 *For any constant d, any TAC^0 circuit of depth d that computes $f(x,y) = \mathsf{Maj}(x_1,\ldots,x_n) \wedge \mathsf{Maj}(y_1,\ldots,y_n)$ must have size $n^{\Omega_d(\log n)}$.*

Proof: The proof is by contradiction. Let $M = n^{o(\log n)}$ and let C be a depth-d TAC^0 of size M that computes $f(x,y)$. We analyze the effect of hitting C with a very strong random restriction ρ, one which has parameter $p = n^{-0.1}$. It is easy to see that with extremely high probability – much more than $1/2$ – f_ρ turns into some function of the form

$$f_\rho(x,y) = \mathrm{sign}(\sum_{i\in S_1} x_i + C_1) \wedge \mathrm{sign}(\sum_{j\in S_2} y_j + C_2),$$

where $|S_1|, |S_2| \geq n^{0.8}$ and $|C_1|, |C_2| \leq n^{0.51}$. For any such ρ, by fixing at most $2n^{.51}$ additional variables, we get $\mathsf{Maj}(x') \wedge \mathsf{Maj}(y')$ where x', y' are $\Omega(n^{0.8})$-bit strings. By the recent result of Sherstov [She09], any PTF for this function must have degree at least $c_1 \log n$ for some absolute constant $c_1 > 0$.

On the other hand, let us consider what happens to the TAC^0 C under such a strong random restriction using Lemma 3. Since $p = n^{-0.1}$, we have $p' = n^{-0.1} \cdot 10^{d-1}(4\log M)^{d-2} < n^{-0.09}$ for n sufficiently large. Taking $k = (c_1/2)\log n$, Lemma 3 gives us that C_ρ has a PTF of degree at most $(c_1/2)\log n$ with failure probability at most

$$M^{-2} + M^5(4e\log(M)p'/k)^k = M^{-2} + M^5 n^{-\Omega(\log n)} < 1/2$$

since $M = n^{o(\log n)}$. Thus, there must be some restriction ρ such that f_ρ has PTF degree at least $c_1\log n$, but C_ρ has PTF degree at most $(c_1/2)\log n$. This contradiction proves the theorem. ■

Aspnes et al. [ABFR94] prove lower bounds on the size of TAC^0 circuits that compute various functions such as parity. The method of [ABFR94] is useful for functions that have high weak PTF degree (such as parity). In contrast, our argument above gives us a TAC^0 lower bound for the function $\mathsf{Maj}(x) \wedge \mathsf{Maj}(y)$, which is known [BRS95] to have PTF degree only $O(\log n)$.

6.3 Lower bounds on TAC^0

We prove that no analogue of Beigel's theorem [Bei94] is possible for Thr gates: even an AND of two Thr gates may require a TAC^0 of more than quasi-polynomial size. The proof is similar to that of Theorem 24, it uses a recent result of Sherstov [She09] showing that the function $f(x,y) = U_{n,4n}(x) \wedge U_{n,4n}(y)$ (see Section 6.1) has PTF degree $\Omega(n)$.

Theorem 25 *Fix any absolute constant d. Any TAC^0 circuit of depth d that computes $f(x,y) = U_{n,4n}(x) \wedge U_{n,4n}(y)$ must have size $2^{\Omega(n^{1/(d-1)})}$.*

Proof: Let C be a depth-d TAC^0 of size M that computes $f(x,y)$. As in Section 6.1, we consider the effect of hitting C with a random restriction with parameter $r = \frac{1}{10^{d-1}s^{d-2}}$, where $s = 3\log M$.

The proof of Lemma 22 shows that with failure probability at most M^{-2} over the choice of ρ, the restricted circuit C_ρ can be expressed as a PTF of degree at most $3\log M$. And Lemma 23 gives that with failure probability at most $4n^{-2}$, both $(U_{n,4n})_\rho(x)$ and $(U_{n,4n})_\rho(y)$ have sub-functions that are equivalent to $U_{m,4m}(x)$ and $U_{m,4m}(y)$ respectively, where $m = \Omega(n/(\log M)^{d-2})$. Applying Sherstov's lower bound, we get that for such a restriction ρ, the function $f_\rho(x,y)$ must have PTF degree $\Omega(n/(\log M)^{d-2})$. We thus have $M = 2^{\Omega(n^{1/(d-1)})}$, and the theorem is proved. ■

6.4 Lower bounds on $\mathsf{TAC}^0[t(n)]$

Inspection of the proof of Theorem 16 is easily seen to imply that the parity function cannot be computed by a $\mathsf{TAC}^0[(\log n)^{2/3}]$ circuit. We give an improved bound that allows up to $O(\log n)$ threshold gates.

Theorem 26 *Fix any absolute constant d. Any poly(n)-size, depth-d $\mathsf{TAC}^0[t(n)]$ circuit that computes the parity function must have $t(n) = \Omega(\log n)$.*

Proof: Fix any constant $c > 1$ and $d \geq 1$, and let C be a depth-d $\mathsf{TAC}^0[t(n)]$ circuit of size at most $M = n^c$ that computes the parity function. We write t as shorthand for $t(n)$, and we may assume that $(t+1)2^t$ is less than n.

By Lemma 20, we have that C is equivalent to a $(t+1)2^t$-junta-of-TAC^0. We write this function as $J(C_1(x), \ldots, C_{(t+1)2^t}(x))$, where each C_i is a TAC^0 of depth d and size at most $M = n^c$.

We consider the effect of applying a random restriction with parameter $p = n^{-0.5}$ to C. As in the proof of Theorem 24, let us consider what happens to C – or rather, to $J(C_1, \ldots, C_{(t+1)2^t})$ – under such a strong random restriction using Lemma 3. Since $p = n^{-0.5}$, we have

$$p' = n^{-0.5} \cdot 10^{d-1}(4c\log n)^{d-2} = n^{-0.5} \cdot \mathrm{polylog}(n).$$

Now we shall take k in Lemma 3 to be $k = 20c$. For any fixed $i \in [(t+1)2^t]$, the Lemma gives that with failure probability at most

$$M^{-2} + M^5(4ec\log(n)p'/(20c))^{20c} = n^{-2c} + n^{5c} \cdot (n^{-0.5} \cdot \mathrm{polylog}(n))^{20c} < 2n^{-2c} < n^{-2},$$

the restriction ρ causes the TAC^0 C_i to become a function $(C_i)_\rho$ which is a PTF of degree at most $k = 20c$. Since there are at most $(t+1)2^t < n$ many TAC^0 circuits C_i, with overall probability at least $1 - n^{-1}$ the function C_ρ is equivalent to a $(t+1)2^t$-junta of degree-$20c$ PTFs, $J((C_1)_\rho, \ldots, (C_{(t+1)2^t})_\rho)$.

We now observe that since C computes the parity function, under any restriction the function computed by C_ρ is simply the parity function (or its negation) on all variables that survive the restriction ρ. A simple Chernoff bound shows that with probability at least $99/100$, at least (say) $0.05 \cdot n^{0.5}$ many variables survive the restriction ρ.

Thus there is some restriction ρ such that (1) at least $0.05n^{0.5}$ many variables survive ρ, and (2) the function C_ρ is equivalent to a $(t+1)2^t$-junta of degree-$20c$ PTFs $(C_1)_\rho, \ldots, (C_{(t+1)2^t})_\rho$ over these variables. For simplicity we restrict further variables if necessary so that there are precisely $n' = 0.05n^{0.5}$ surviving variables. We now recall that by the [DRST09, HKM09] bound on the total influence of any degree-d PTF over n' variables, each function $(C_i)_\rho$ has total influence at most $O(1) \cdot (n')^{1-1/100d}$. The total influence of a T-junta $J(c_1, \ldots, c_T)$ is easily seen to be at most T times the maximum total influence of any c_i. Since $J((C_1)_\rho, \ldots, (C_{(t+1)2^t})_\rho)$ computes parity (or its negation) over n' variables, its total influence is exactly n'. Thus we must have $(t+1)2^t = \Omega(n'^{1/(100d)})$, which means that $t = \Omega(\log n)$. ■

We note that a careful inspection of our proof shows that it does not use the depth restriction on the threshold gates: it applies to any threshold circuit of size $\log n$ augmented with AC^0 inputs. It is known that there are circuits of $\log n$ threshold gates that can compute parity [SRK94], and hence any improvement of our bound must exploit the depth restriction on the threshold gates.

References

[ABFR94] J. Aspnes, R. Beigel, M. Furst, and S. Rudich. The expressive power of voting polynomials. *Combinatorica*, 14(2):1–14, 1994.

[Bei94] R. Beigel. When do extra majority gates help? polylog(n) majority gates are equivalent to one. *Computational Complexity*, 4:314–324, 1994.

[BOW08] E. Blais, R. O'Donnell, and K. Wimmer. Polynomial regression under arbitrary product distributions. In *Proc. 21st Annual Conference on Learning Theory (COLT)*, pages 193–204, 2008.

[BRS95] R. Beigel, N. Reingold, and D. Spielman. PP is closed under intersection. *Journal of Computer & System Sciences*, 50(2):191–202, 1995.

[BT96] N. Bshouty and C. Tamon. On the Fourier spectrum of monotone functions. *Journal of the ACM*, 43(4):747–770, 1996.

[DLM$^+$07] I. Diakonikolas, H. Lee, K. Matulef, K. Onak, R. Rubinfeld, R. Servedio, and A. Wan. Testing for concise representations. In *Proc. 48th Ann. Symposium on Computer Science (FOCS)*, pages 549–558, 2007.

[DRST09] I. Diakonikolas, P. Raghavendra, R. Servedio, and L.-Y. Tan. Average sensitivity and noise sensitivity of polynomial threshold functions, 2009. Manuscript.

[FJS91] M. Furst, J. Jackson, and S. Smith. Improved learning of AC^0 functions. In *Proc. 4th Annual Conference on Learning Theory (COLT)*, pages 317–325, 1991.

[FK09] Lance Fortnow and Adam Klivans. Efficient learning algorithms yield circuit lower bounds. *Journal of Computer & System Sciences*, 75(1):27–36, 2009.

[GHR92] M. Goldmann, J. Håstad, and A. Razborov. Majority gates vs. general weighted threshold gates. *Computational Complexity*, 2:277–300, 1992.

[GL94] C. Gotsman and N. Linial. Spectral properties of threshold functions. *Combinatorica*, 14(1):35–50, 1994.

[Gol97] M. Goldmann. On the power of a threshold gate at the top. *Information Processing Letters*, 63(6):287–293, 1997.

[Hås86] J. Håstad. *Computational Limitations for Small Depth Circuits*. MIT Press, Cambridge, MA, 1986.

[Hås01] J. Håstad. A slight sharpening of LMN. *Journal of Computer and System Sciences*, 63(3):498–508, 2001.

[HKM09] P. Harsha, A. Klivans, and R. Meka. Bounding the sensitivity of polynomial threshold functions. Available at http://arxiv.org/abs/0909.5175, 2009.

[JKS02] J. Jackson, A. Klivans, and R. Servedio. Learnability beyond AC^0. In *Proc. 34th Annual ACM Symposium on Theory of Computing (STOC)*, pages 776–784, 2002.

[KKMS08] A. Kalai, A. Klivans, Y. Mansour, and R. Servedio. Agnostically learning halfspaces. *SIAM Journal on Computing*, 37(6):1777–1805, 2008.

[KOS04] A. Klivans, R. O'Donnell, and R. Servedio. Learning intersections and thresholds of halfspaces. *Journal of Computer & System Sciences*, 68(4):808–840, 2004.

[KOS08] A. Klivans, R. O'Donnell, and R. Servedio. Learning geometric concepts via Gaussian surface area. In *Proc. 49th IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 541–550, 2008.

[LMN93] N. Linial, Y. Mansour, and N. Nisan. Constant depth circuits, Fourier transform and learnability. *Journal of the ACM*, 40(3):607–620, 1993.

[MOS04] E. Mossel, R. O'Donnell, and R. Servedio. Learning functions of k relevant variables. *Journal of Computer & System Sciences*, 69(3):421–434, 2004. Previously published as "Learning juntas".

[OS07] R. O'Donnell and R. Servedio. Learning monotone decision trees in polynomial time. *SIAM J. Comput.*, 37(3):827–844, 2007.

[She09] A. Sherstov. The intersection of two halfspaces has high threshold degree. In *Proc. 50th IEEE Symposium on Foundations of Computer Science (FOCS)*, 2009.

[SRK94] K.-Y. Siu, V.P. Roychowdhury, and T. Kailath. Rational approximation techniques for analysis of neural networks. *IEEE Transactions on Information Theory*, 40(2):445–474, 1994.

ECCC ISSN 1433-8092

http://eccc.hpi-web.de