On the average sensitivity and density of k-CNF formulas ⋆

Dominik Scheder[1] and Li-Yang Tan[2]⋆⋆

[1] Aarhus University
[2] Columbia University

Abstract. We study the relationship between the average sensitivity and density of k-CNF formulas via the isoperimetric function $\varphi : [0, 1] \to$ R,

$$\varphi(\mu) = \max \left\{ \frac{\mathbf{AS}(F)}{\textbf{CNF-width}(F)} : \mathbf{E}[F(\boldsymbol{x})] = \mu \right\},$$

where the maximum is taken over all Boolean functions $F : \{0,1\}^* \to \{0,1\}$ over a finite number of variables and $\mathbf{AS}(F)$ is the average sensitivity of F. Building on the work of Boppana [Bop97] and Traxler [Tra09], and answering an open problem of O'Donnell, Amano [Ama11] recently proved that $\varphi(\mu) \leq 1$ for all $\mu \in [0, 1]$. In this paper we determine φ exactly, giving matching upper and lower bounds. The heart of our upper bound is the Paturi-Pudlák-Zane (PPZ) algorithm for k-SAT [PPZ97], which we use in a unified proof that sharpens the three incomparable bounds of Boppana, Traxler, and Amano.

We extend our techniques to determine φ when the maximum is taken over monotone Boolean functions F, further demonstrating the utility of the PPZ algorithm in isoperimetric problems of this nature. As an application we show that this yields the largest known separation between the average and maximum sensitivity of monotone Boolean functions, making progress on a conjecture of Servedio.

Finally, we give an elementary proof that $\mathbf{AS}(F) \leq \log(s)(1 + o(1))$ for functions F computed by an s-clause CNF, which is tight up to lower order terms. This sharpens and simplifies Boppana's bound of $O(\log s)$ obtained using Håstad's switching lemma.

⋆ The authors acknowledge support from the Danish National Research Foundation and The National Science Foundation of China (under the grant 61061130540) for the Sino-Danish Center for the Theory of Interactive Computation, within which this work was performed.

⋆⋆ Part of this research was completed while visiting KTH Royal Institute of Technology, partially supported by ERC Advanced Investigator Grant 226203.

1 Introduction

The average sensitivity of a Boolean function $F : \{0,1\}^n \to \{0,1\}$ is a fundamental and well-studied complexity measure. The sensitivity of F at an input $x \in \{0,1\}^n$, denoted $S(F,x)$, is the number of coordinates $i \in [n]$ of x such that $F(x) \neq F(x \oplus e_i)$, where $x \oplus e_i$ denotes x with its i-th coordinate flipped. The average sensitivity of F, denoted $\mathbf{AS}(F)$, is the expected number of sensitive coordinates of F at an input $\boldsymbol{x}$ drawn uniformly at random from $\{0,1\}^n$. Viewing F as the indicator of a subset $A_F \subseteq \{0,1\}^n$, the average sensitivity of F is proportional to the number of edges going from A_F to its complement, and so $\mathbf{AS}(F)$ may be equivalently viewed as a measure of the normalized edge boundary of A_F.

The average sensitivity of Boolean functions was first studied in the computer science literature by Ben-Or and Linial [BL90] in the context of distributed computing. Owing in part to connections with the Fourier spectrum of F established in the celebrated work of Kahn, Kalai, and Linial [KKL88], this complexity measure has seen utility throughout theoretical computer science, receiving significant attention in a number of areas spanning circuit complexity [LMN93,OW07,BT13][3], learning theory [BT96,OS08,DHK+10], random graphs [Fri98,Fri99,BKS99], social choice theory, hardness of approximation [DS05], quantum query complexity [Shi00], property testing [RRS+12], *etc.* We remark that the study of average sensitivity in combinatorics predates its introduction in computer science. For example, the well-known edge-isoperimetric inequality for the Hamming cube [Har64,Ber67,Lin64,Har76] yields tight extremal bounds on the average sensitivity of Boolean functions in terms of the number of its satisfying assignments.

The focus of this paper is on the average sensitivity of k-CNF formulas, the AND of ORs of k or fewer variables; by Boolean duality our results apply to k-DNF formulas as well. Upper bounds on the average sensitivity of small-depth AC^0 circuits are by now classical results, having been the subject of study in several early papers in circuit complexity [LMN93,Man95,Bop97,Hås01]. Despite its apparent simplicity, though, gaps remain even in our understanding of the average sensitivity of depth-2 AC^0 circuits. The starting point of this research was the following basic question:

Question 1. What is the maximum average sensitivity of a k-CNF formula $F : \{0,1\}^n \to \{0,1\}$ that is satisfied by a μ fraction of assignments?

An easy folkloric argument (first appearing explicitly in [Bop97]) gives an upper bound of $2(1-\mu)k$. The maximum of $2k$ attained by this bound is a multiplicative factor of 2 away from the lower bound of k witnessed by the parity function over k variables, leading O'Donnell to ask if there is indeed a matching upper bound of

[3] Though couched in different terminology, Khrapchenko's classical lower bound [Khr71] on the formula size of Boolean functions also relies implicitly on average sensitivity.

k [O'D07]. O'Donnell's question was answered in a sequence of works by Traxler [Tra09] and Amano [Ama11], with Traxler proving a bound of $2\mu \log_2(1/\mu)k$ (attaining a maximum of $\sim 1.062k$ at $\mu = 1/e$), followed by Amano's bound of k independent of μ. These three incomparable bounds are shown in Figure 1 where they are normalized by k.

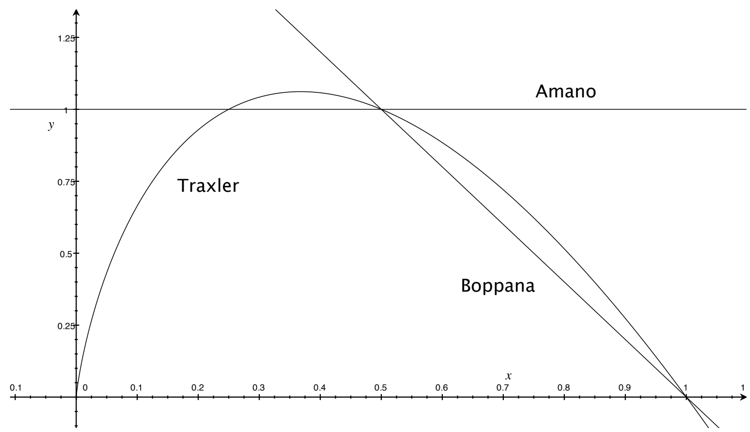


Fig. 1. The upper bounds of Boppana, Traxler, and Amano, normalized by k.

The natural question at this point is: what is the true dependence on μ? In this work we answer this question by giving matching upper and lower bounds. Traxler's upper bound of $2\mu \log_2(1/\mu)k$ is easily seen to be tight at the points $\mu = 2^{-I}$ for all positive integers $I \in \mathbb{N}$, since the AND of I variables is a 1-CNF with average sensitivity $2\mu \log_2(1/\mu)$, but we are not aware of any other matching lower bounds prior to this work. Like Traxler and Amano, the main technical tool for our upper bound is the Paturi-Pudlák-Zane (PPZ) randomized algorithm for k-SAT. We remark that this is not the first time the PPZ algorithm has seen utility beyond the satisfiability problem; in their original paper the authors use the algorithm and its analysis to obtain sharp lower bounds on the size of depth-3 AC^0 circuits computing the parity function.

We extend our techniques to determine φ when the maximum is taken over monotone Boolean functions F, further demonstrating the utility of the PPZ algorithm in isoperimetric problems of this nature. As an application we show that this yields the largest known separation between the average and maximum sensitivity of monotone functions, making progress on a conjecture of Servedio. Finally, we give an elementary proof that $\mathbf{AS}(F) \leq \log(s)(1+o(1))$ for functions F computed by an s-clause CNF; such a bound that is tight up to lower order terms does not appear to have been known prior to our work.

1.1 Our results

Our main object of study is the following isoperimetric function:

Definition 1. *Let $\varphi : [0,1] \to \mathrm{R}$ be the function:*

$$\varphi(\mu) = \max\left\{\frac{\mathbf{AS}(F)}{\mathbf{CNF\text{-}width}(F)} : \mathbf{E}[F(\boldsymbol{x})] = \mu\right\},$$

where the maximum is taken over all Boolean functions $F : \{0,1\}^* \to \{0,1\}$ over a finite number of variables.

Note that $\mathbf{E}[F(\boldsymbol{x})] = a2^{-b}$ for $a, b \in \mathrm{N}$, and thus $\varphi(\mu)$ is well-defined only at those points. However, these points are dense within the interval $[0,1]$ and thus one can continuously extend φ to all of $[0,1]$. As depicted in Figure 1 the upper bounds of Boppana, Traxler, and Amano imply that $\varphi(\mu) \le \min\{2(1-\mu), 2\mu\log(1/\mu), 1\}$. In this paper we determine φ exactly, giving matching upper and lower bounds.

Theorem 1. *$\varphi(\mu) : [0,1] \to \mathrm{R}$ is the piecewise linear continuous function that evaluates to $2\mu\log_2(1/\mu)$ when $\mu = 2^{-I}$ for some $I \in \mathrm{N} = \{0,1,2,\ldots\}$, and is linear between these points*[4]*. That is, if $\mu = t \cdot 2^{-(I+1)} + (1-t) \cdot 2^{I}$ for some $I \in \mathrm{N}$ and $t \in [0,1]$, then*

$$\varphi(\mu) = t \cdot \frac{(I+1)}{2^I} + (1-t) \cdot \frac{I}{2^{I-1}}.$$

We extend our techniques to also determine the variant of φ where the maximum is taken only over monotone Boolean functions. The reader familiar with the PPZ algorithm will perhaps recall the importance of Jensen's inequality in its analysis. Jensen's inequality is very helpful for dealing with random variables whose correlations one does not understand. It turns out that in case of monotone CNF formulas, certain events are *positively correlated* and we can replace Jensen's inequality by the FKG inequality [FKG71], leading to a substantial improvement in the analysis.

Theorem 2 (Upper bound for monotone k-CNFs). *Let F be a monotone k-CNF formula and $\mu = \mathbf{E}[f(\boldsymbol{x})]$. Then $\mathbf{AS}(F) \le 2k\mu\ln(1/\mu)(1+\varepsilon_k)$ for some ε_k that goes to 0 as k grows.*[5]

Theorem 3 (Lower bound for monotone k-CNFs). *Let $\mu \in [0,1]$ and $k \in \mathbb{N}$. There exists a monotone k-CNF formula F with $\mathbf{E}[F(\boldsymbol{x})] = \mu \pm \varepsilon_k$ and $\mathbf{AS}(F) \ge 2k\mu\ln(1/\mu)(1-\varepsilon_k)$ for some ε_k that goes to 0 as k grows.*

We apply Theorem 2 to obtain a separation between the average and maximum sensitivity of monotone Boolean functions, making progress on a conjecture of Servedio [O'D12]. Our result improves on the current best gap of $\mathbf{AS}(f) \le \sqrt{2/\pi} \cdot \mathbf{S}(f)(1+o(1)) \approx 0.797 \cdot \mathbf{S}(f)(1+o(1))$, which follows as a corollary of an isoperimetric inequality of Blais [Bla11].

Corollary 1. *Let f be a monotone Boolean function. Then $\mathbf{AS}(F) \le \ln(2) \cdot \mathbf{S}(F)(1+o(1)) \le 0.694 \cdot \mathbf{S}(f)(1+o(1))$, where $o(1)$ is a term that goes to 0 as $\mathbf{S}(F)$ grows.*

[4] We use the fact that $0\log_2(1/0) = 0$ here.

[5] Note that the additive term of ε_k is necessary since $\mathbf{AS}(F) = 1$ when $F = x_1$.

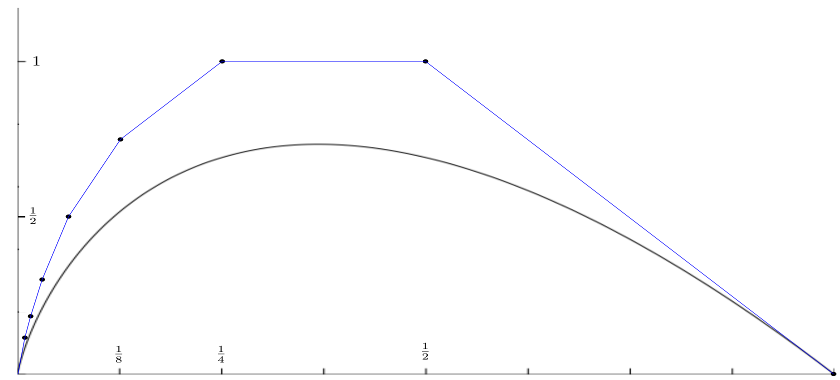


Fig. 2. Our matching bounds for all functions (top), and for monotone functions (bottom).

Finally, we give an elementary proof that $\mathbf{AS}(F) \le \log(s)(1 + o(1))$ for functions F computed by an s-clause CNF, which is tight up to lower order terms by considering the parity of $\log s$ variables. This sharpens and simplifies Boppana's bound of $O(\log s)$ obtained using Håstad's switching lemma.

Theorem 4. *Let F be an s-clause CNF. Then* $\mathbf{AS}(F) \le \log s + \log\log s + O(1)$.

1.2 Preliminaries

Throughout this paper all probabilities and expectations are with respect to the uniform distribution, and logarithms are in base 2 unless otherwise stated. We adopt the convention that the natural numbers N include 0. We use boldface letters (*e.g.* $\boldsymbol{x}$, $\boldsymbol{\pi}$) to denote random variables.

For any Boolean function $F : \{0,1\}^n \to \{0,1\}$, we write $\mu(F) \in [0,1]$ to denote the density $\mathbf{E}_{x\in\{0,1\}^n}[F(\boldsymbol{x})]$ of F, and $\mathrm{sat}(F) \subseteq \{0,1\}^n$ to denote the set of satisfying assignments of F (and so $|\mathrm{sat}(F)| = \mu \cdot 2^n$). The CNF width of F, which we will denote **CNF-width**(F), is defined to be the smallest $k \in [n]$ such that F is computed by a k-CNF formula; similarly, **DNF-width**(F) is the smallest k such that F is computed by a k-DNF formula. Note that by Boolean duality, we have the relation **CNF-width**(F) = **DNF-width**$(\neg F)$.

Definition 2. *Let $F : \{0,1\}^n \to \{0,1\}$ and $x \in \{0,1\}^n$. For any $i \in [n]$, we say that F is sensitive at coordinate i on x if $F(x) \ne F(x \oplus e_i)$, where $x \oplus e_i$ denotes x with its i-th coordinate flipped, and write $S(F,x,i)$ as the indicator for this event. The sensitivity of F at x, denoted $S(F,x)$, is $\#\{i \in [n] : F(x) \ne F(x \oplus e_i)\} = \sum_{i=1}^n S(F,x,i)$. The average sensitivity and maximum sensitivity of F, denoted $\mathbf{AS}(F)$ and $\mathbf{S}(F)$ respectively, are defined as follows:*

$$\mathbf{AS}(F) = \mathop{\mathbf{E}}_{x\in\{0,1\}^n}[S(F,\boldsymbol{x})], \qquad \mathbf{S}(F) = \max_{x\in\{0,1\}^n}[S(F,x)].$$

We will need the following basic fact:

Fact 11 *Let $F : \{0,1\}^n \to \{0,1\}$ and $\mu = \mathbf{E}[F(\boldsymbol{x})]$. Then $\mathbf{E}_{x \in \mathrm{sat}(F)}[S(F,\boldsymbol{x})] = \mathbf{AS}(F)/2\mu$.*

Proof. This follows by noting that

$$\mathbf{AS}(f) = \mathop{\mathbf{E}}_{x\in\{0,1\}^n}[S(F,\boldsymbol{x})] = \mathop{\mathbf{E}}_{x\in\{0,1\}^n}[2\cdot S(F,\boldsymbol{x})\cdot \mathbf{1}_{[F(\boldsymbol{x})=1]}] = 2\mu \mathop{\mathbf{E}}_{x\in \mathrm{sat}(F)}[S(F,\boldsymbol{x})].$$

Here the second identity holds by observing that for any $x \in \mathrm{sat}(F)$ and coordinate $i \in [n]$ on which F is sensitive at x, we have that $x \oplus e_i \notin \mathrm{sat}(F)$ and F is sensitive on i at $x \oplus e_i$. □

We remark that Boppana's bound follows easily from 11 and Boolean duality. For any k-CNF F with density $\mathbf{E}[f(\boldsymbol{x})] = \mu$, its negation $\neg F$ is a k-DNF with density $(1-\mu)$ and $\mathbf{AS}(\neg F) = \mathbf{AS}(F)$. Applying Fact 11 to $\neg F$ and noting that every satisfying assignment of a k-DNF has sensitivity at most k, we conclude that $\mathbf{AS}(F) = \mathbf{AS}(\neg F) \le 2(1-\mu)k$.

2 The PPZ Algorithm

The main technical tool for our upper bounds in both Theorems 1 and 2 is the PPZ algorithm (Figure 3), a remarkably simple and elegant randomized algorithm for k-SAT discovered by and named after Paturi, Pudlák, and Zane [PPZ97]. Perhaps somewhat surprisingly, the utility of the PPZ algorithm extends beyond its central role in the satisfiability problem. Suppose the PPZ algorithm is run on a k-CNF F, for which it is searching for an satisfying assignment $x \in \mathrm{sat}(F)$. Since the algorithm is randomized and may not return a satisfying assignment, it defines a probability distribution on $\mathrm{sat}(F) \cup \{\texttt{failure}\}$.

The key observation underlying the analysis of PPZ is that a satisfying assignment x for which $S(F,x)$ is large receives a higher probability under this distribution than its less sensitive brethren; the exact relationship depends on **CNF-width**(F), and is made precise by the Satisfiability Coding Lemma (Lemma 2). Since the probabilities of the assignments sum to at most 1, it follows that there cannot be too many high-sensitivity assignments. This intuition is the crux of the sharp lower bounds of Paturi, Pudlák, and Zane on the size of depth-3 AC^0 circuits computing parity; it is also the heart of Traxler's, Amano's, and our upper bounds on the average sensitivity of k-CNF formulas.

Let us add some bookkeeping to this algorithm. For every satisfying assignment $x \in \mathrm{sat}(F)$, permutation $\pi : [n] \to [n]$, and coordinate $i \in [n]$, we introduce an indicator variable $T_i(x,\pi,F)$ that takes value 1 iff the assignment x_i to the i-th coordinate was decided by a coin *T*oss, conditioned on PPZ returning x on inputs F and π (which we denote as $\texttt{ppz}(F,\pi) = x$). We also introduce the dual indicator variable $I_i(x,\pi,F) = 1 - T_i(x,\pi,F)$, which takes value 1 if the the assignment x_i was *I*nferred. We define $T(x,\pi,F) = T_1(x,\pi,F) + \cdots + T_n(x,\pi,F)$

The ppz algorithm takes as input a k-CNF formula F and a permutation $\pi : [n] \to [n]$.

1. for $i = 1$ to n:
2. if $x_{\pi(i)}$ occurs in a unit clause in F then set $x_{\pi(i)} \leftarrow 1$ in F.
3. else if $\overline{x}_{\pi(i)}$ occurs in a unit clause in F then set $x_{\pi(i)} \leftarrow 0$ in F.
4. else toss a fair coin and set $x_{\pi(i)}$ to 0 or 1 uniformly at random.

If $F \equiv 1$, the algorithm has found a satisfying assignment and returns it. Otherwise the algorithm reports `failure`.

Fig. 3. The PPZ k-SAT algorithm

to be the total number of coin tosses, and similarly $I(x, \pi, F) = I_1(x, \pi, F) + \ldots + I_n(x, \pi, F) = n - T(x, \pi, F)$ to be the number of inference steps. Note that if $x \in \text{sat}(F)$ and we condition on the event $\texttt{ppz}(F, \pi) = x$, then all coin tosses of the algorithm are determined and $T = T(x, \pi, F)$ becomes some constant in $\{0, 1, \ldots, n\}$; likewise for $I = I(x, \pi, F)$. The next lemma follows immediately from these definitions:

Lemma 1 (Probability of a solution under PPZ [PPZ97]). *Let F be a CNF formula over n variables and $x \in \text{sat}(F)$. Let π be a permutation over the variables. Then*

$$\mathbf{Pr}[\texttt{ppz}(F, \pi) = x] = 2^{-T(x,\pi,F)} = 2^{-n+I(x,\pi,F)} \ , \tag{1}$$

where $T(x, \pi, F)$ is the number of coin tosses used by the algorithm when finding x.

For completeness, we include a proof of the simple but crucial Satisfiability Coding Lemma:

Lemma 2 (Satisfiability Coding Lemma [PPZ97]). *Let F be a k-CNF formula and let $x \in \text{sat}(F)$. If F is sensitive at coordinate i on x then $\mathbf{E}_\pi[I_i(x, \boldsymbol{\pi}, F)] \geq 1/k$, and otherwise $I_i(x, \pi, F) = 0$ for all permutations π. Consquently, by linearity of expectation $\mathbf{E}_\pi[I(x, \boldsymbol{\pi}, F)] \geq S(F, x)/k$.*

Proof. Without loss of generality we assume that $x = (1, \ldots, 1)$, and since we condition on $\texttt{ppz}(F, \pi) = x$, all coin tosses made by the algorithm yield a 1. If F is sensitive to i at x, then certainly there must exist a clause C in which x_i is the only satisfied literal. That is, $C = x_i \vee \overline{x}_{i_2} \vee \cdots \vee \overline{x}_{i_k}$. With probabilitiy $1/k$, variable i comes after $i_2, \ldots, i_k$ in the permutation π and in this case, the PPZ algorithm has already set the variables $x_{i_2}, \ldots, x_{i_k}$ to 1 when it processes x_i. Thus, F will contain the unit clause $\{x_i\}$ at this point, and PPZ will not toss a coin for x_i (*i.e.* the value of x_i is forced), which means that $I_i(x, \pi, F) = 1$. Thus, $\mathbf{E}_\pi[I_i(x, \boldsymbol{\pi}, F)] \geq 1/k$. On the other hand if F is not sensitive to i at x, then every clause containing x_i also contains a second satisfied literal. Thus, PPZ will never encounter a unit clause containing only x_i and therefore $I_i(x, \pi, F) = 0$ for all permutations π. □

3 Average sensitivity of k-CNFs: Proof of Theorem 1

3.1 The upper bound

Let $g : \mathbb{R} \to \mathbb{R}$ be any monotone increasing convex function such that $g(I) \le 2^I$ for $I \in \mathbb{N}$. Choose a uniformly random permutation π of the variables of F and run PPZ. Summing over all $x \in \mathrm{sat}(F)$ and applying Lemma 1, we first note that

$$\begin{aligned} 1 \ \ge \sum_{x \in \mathrm{sat}(F)} \mathbf{Pr}[\mathrm{ppz}(F, \boldsymbol{\pi}) = x] &= \sum_{x \in \mathrm{sat}(F)} \mathop{\mathbf{E}}_{\pi} \left[2^{-T(x, \boldsymbol{\pi}, F)} \right] \\ &= 2^{-n} \sum_{x \in \mathrm{sat}(F)} \mathbf{E}_{\pi} \left[2^{I(x, \boldsymbol{\pi}, F)} \right] \\ &= \mu \mathop{\mathbf{E}}_{x \in \mathrm{sat}(F), \pi} \left[2^{I(\boldsymbol{x}, \boldsymbol{\pi}, F)} \right]. \end{aligned} \tag{2}$$

Next by our assumptions on g, we have

$$\mu \mathop{\mathbf{E}}_{x \in \mathrm{sat}(F), \pi} \left[2^{I(\boldsymbol{x}, \boldsymbol{\pi}, F)} \right] \ge \mu \mathop{\mathbf{E}}_{x \in \mathrm{sat}(F), \pi} [g(I(\boldsymbol{x}, \boldsymbol{\pi}, F))] \ge \mu \cdot g \left(\mathop{\mathbf{E}}_{x \in \mathrm{sat}(F), \pi} [I(\boldsymbol{x}, \boldsymbol{\pi}, F)] \right),$$

where the first inequality holds since g satisfies $g(I) \le 2^I$ for all $I \in \mathrm{N}$, and the second follows from Jensen's inequality and convexity of g. Combining these inequalities and applying the Satisfiability Coding Lemma (Lemma 2), we get

$$1 \ge \mu \cdot g \left(\mathop{\mathbf{E}}_{x \in \mathrm{sat}(F), \pi} [I(\boldsymbol{x}, \boldsymbol{\pi}, F)] \right) \ge \mu \cdot g \left(\mathop{\mathbf{E}}_{x \in \mathrm{sat}(F)} \left[\frac{S(F, \boldsymbol{x})}{k} \right] \right) = \mu \cdot g \left(\frac{\mathbf{AS}(F)}{2\mu k} \right).$$

Here we have used the assumption that g is monotone increasing in the second inequality, and Fact 11 for the final equality. Solving for $\mathbf{AS}(F)$, we obtain the following upper bound:

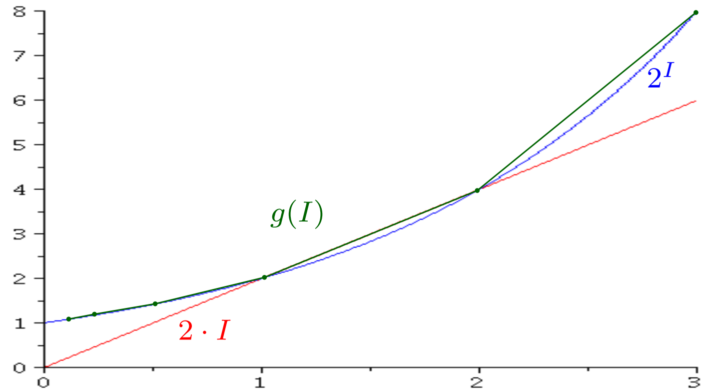
$$\mathbf{AS}(F) \le 2\mu g^{-1}(1/\mu) \cdot k. \tag{3}$$

At this point we note that we can easily recover Traxler, Amano, and Boppana's bounds from Equation (3) simply by choosing the appropriate function $g : \mathrm{R} \to \mathrm{R}$ that satisfies the necessary conditions (*i.e.* monotone increasing, convex, and $g(I) \le 2^I$ for all $I \in \mathrm{N}$).

- If $g(I) = 2^I$, then (3) becomes $\mathbf{AS}(F) \le 2\mu \log(1/\mu) \cdot k$, which is Traxler's bound.
- If $g(I) = 2I$, we obtain $\mathbf{AS}(F) \le k$, which is Amano's bound.
- If $g(I) = 1 + I$, we obtain Boppana's bound of $\mathbf{AS}(F) \le 2(1 - \mu) \cdot k$.[6]

We pick g to be the largest function that is monotone increasing, convex, and at most 2^I for all integers I. This is the convex envelope of the integer points $(I, 2^I)$:

[6] This observation was communicated to us by Lee [Lee12].



That is, g is the continuous function such that $g(I) = 2^I$ whenever $I \in \mathrm{N}$, and is linear between these integer points. Thus, the function $g^{-1}(1/\mu)$ is piecewise linear in $1/\mu$, and $2\mu g^{-1}(1/\mu) \cdot k$ is piecewise linear in μ. Therefore we obtain an upper bound on $\varphi(\mu)$ that is $2\mu \log(1/\mu)$ if $\mu = 2^{-I}$ for $I \in \mathbb{N}$ and piecewise linear between these points. This proves the upper bound in Theorem 1.

3.2 The lower bound

We will need a small observation:

Lemma 3. *Let $k, \ell \in \mathrm{N}_0$, and set $\mu = 2^{-\ell}$. There exists a Boolean function $F : \{0,1\}^n \to \{0,1\}$ with* $\textbf{CNF-width}(F) = k$ *and* $\mathbf{AS}(F) = 2\mu \log(1/\mu) \cdot k$.

Proof. We introduce $k \cdot \ell$ variables $x_j^{(i)}$, $1 \le i \le \ell$, $1 \le j \le k$ and let F be (k,ℓ)-*block parity*, defined as

$$F := \bigwedge_{i=1}^{\ell} \bigoplus_{j=1}^{k} x_j^{(i)} \ .$$

Note that F has density $\mathbf{E}[F(\boldsymbol{x})] = 2^{-\ell} = \mu$, and every satisfying assignment has sensitivity exactly $k\ell$. Thus by Fact 11, $\mathbf{AS}(F) = 2\mu\, \mathbf{E}_{x \in \mathrm{sat}}[S(F, \boldsymbol{x})] = 2k\ell 2^{-\ell} = 2k\mu \log(1/\mu) \cdot k$. □

By Lemma 3, for every $k \in \mathrm{N}$ there is a k-CNF which is a tight example for our upper bound whenever $\mu = 2^{-\ell}$ and $\ell \in \mathrm{N}$. The main idea is to interpolate φ linearly between $\mu = 2^{-\ell-1}$ and $2^{-\ell}$ for all integers ℓ. If μ is not an integer power of $1/2$, we choose ℓ such that $\mu \in \left(2^{-\ell-1}, 2^{-\ell}\right)$, and recall that we may assume that $\mu = a2^{-b}$ for some $a, b \in \mathrm{N}$ (since these points are dense within $[0,1]$). Choose $k \ge b$ and let F be a $(k, \ell+1)$-block parity. We consider the last block $x_1^{\ell+1}, \ldots, x_k^{\ell+1}$ of variables, and note that F contains 2^{k-1} clauses over this last block. Removing $0 \le t \le 2^{k-1}$ clauses over the last block of variables linearly changes μ from $2^{-\ell-1}$ at $t = 0$ to $2^{-\ell}$ at $t = 2^{k-1}$. Every time we remove a clause from the last block, $2^{(k-1)\ell}$ formerly unsatisfying x become satisfying. Before removal, F at x was sensitive to the k variables in the last block (flipping them used to make x satisfying) whereas after removal, F at x is not sensitive to them anymore (change them and x will still satisfy F). However, F at x is now sensitive to the ℓk variables in the first ℓ blocks: changing any of them makes x unsatisfying. Thus, each time we remove a clasue, the number of edges from $\mathrm{sat}(F)$ to its complement in $\{0,1\}^n$ changes by the same amount. Therefore, $\mathbf{AS}(F)$ moves linearly from $2(\ell+1)k2^{-\ell-1}$ at $t=0$ to $2\ell k 2^{-\ell}$ at $t=2^{k-1}$. At every step t, the point $(\mu(F), \mathbf{AS}(F)/k)$ lies on the line from $\left(2^{-\ell-1}, 2(\ell+1)2^{-\ell-1}\right)$ to $\left(2^{-\ell}, 2\ell 2^{-\ell}\right)$, *i.e.*, exactly on our upper bound curve. Choosing $t = a2^{k+\ell-b} - 2^{k+1}$ ensures F has density exactly $\mu = a2^{-b}$.

4 Average sensitivity of monotone k-CNFs

Revisiting Equation (2) in the proof of our upper bound in Section 3.1, recall that we used Jensen's inequality to handle the expression $\mathbf{E}_\pi[2^{I(x,\boldsymbol{\pi},F)}]$, where $I(x,\pi,F) = \sum_{i=1}^n I_i(x,\pi,F)$ is the number of inference steps made by the PPZ algorithm. The crux of our improvement for monotone k-CNFs is the observation that when F is monotone the indicator variables $I_1(x,\pi,F),\ldots,I_n(x,\pi,F)$ are *positively correlated*, *i.e.* $\mathbf{E}\left[2^I\right] \geq \prod_{i=1}^n \mathbf{E}\left[2^{I_i}\right]$, leading to a much better bound.

Lemma 4 (Positive correlation). *If F is monotone then the indicator variables $I_i(x,\pi,F)$ are positively correlated. That is, for every $x \in \mathrm{sat}(F)$,*

$$\mathop{\mathbf{E}}_{\pi}\left[2^{I(x,\boldsymbol{\pi},F)}\right] \geq \prod_{i=1}^{n} \mathop{\mathbf{E}}_{\pi}\left[2^{I_i(x,\boldsymbol{\pi},F)}\right]. \tag{4}$$

Proof (Proof of Theorem 2 assuming Lemma 4). We begin by analyzing each term in the product in the right-hand side of Equation (4). Let $x \in \mathrm{sat}(F)$ and $i \in [n]$. If F is sensitive to coordinate i at x then $\mathbf{Pr}_\pi[I_i(x,\boldsymbol{\pi},F)=1] \geq 1/k$ by Lemma 2, and so

$$\begin{aligned}
\mathop{\mathbf{E}}_{\pi}\left[2^{I_i(x,\boldsymbol{\pi},F)}\right] &= \mathop{\mathbf{Pr}}_{\pi}[I_i(x,\boldsymbol{\pi},F)=0]\cdot 1 + \mathop{\mathbf{Pr}}_{\pi}[I_i(x,\boldsymbol{\pi},F)=1]\cdot 2 \\
&= (1-\mathbf{Pr}[I_i(x,\boldsymbol{\pi},F)=1]) + \mathbf{Pr}[I_i(x,\boldsymbol{\pi},F)=1]\cdot 2 \\
&= 1 + \mathbf{Pr}[I_i(x,\boldsymbol{\pi},F)=1] \;\geq\; 1+\frac{1}{k}.
\end{aligned}$$

On the other hand if F is not sensitive to coordinate i at x, then $I_i(x,\pi,F)$ is always 0, and so $\mathbf{E}_\pi\left[2^{I_i(x,\boldsymbol{\pi},F)}\right] = 1$. Combining this with Lemma 4 shows that

$$\mathop{\mathbf{E}}_{\pi}\left[2^{I(x,\boldsymbol{\pi},F)}\right] \geq \prod_{i=1}^{n} \mathop{\mathbf{E}}_{\pi}\left[2^{I_i(x,\boldsymbol{\pi},F)}\right] \geq \left(1+\frac{1}{k}\right)^{S(F,x)}. \tag{5}$$

With this identity in hand Theorem 2 follows quite easily. Starting with Equation (2), we have

$$\begin{aligned}
1 &\geq \mu \mathop{\mathbf{E}}_{x\in \mathrm{sat}(F),\pi}\left[2^{I(\boldsymbol{x},\boldsymbol{\pi},F)}\right] && \text{(by Equation (2))}\\
&\geq \mu \mathop{\mathbf{E}}_{x\in \mathrm{sat}(F)}\left[\left(1+\frac{1}{k}\right)^{S(F,\boldsymbol{x})}\right] && \text{(by (5))}\\
&\geq \mu \left(1+\frac{1}{k}\right)^{\mathbf{E}_{x\in \mathrm{sat}(F)}[S(F,\boldsymbol{x})]} && \text{(by Jensen's inequality)}\\
&= \mu \left(1+\frac{1}{k}\right)^{\mathbf{AS}(F)/2\mu}. && \text{(by Fact 11)}
\end{aligned}$$

Solving for $\mathbf{AS}(F)$, we get

$$\mathbf{AS}(F) \leq \frac{2\mu \ln(1/\mu)}{\ln\left(1+\frac{1}{k}\right)} = \frac{2\mu \ln(1/\mu)\cdot k}{\ln\left(1+\frac{1}{k}\right)^k} = 2k\mu\ln(1/\mu)(1+\epsilon_k)\ ,$$

for some ϵ_k that goes to 0 as k grows. This proves Theorem 2. □

4.1 Proof of Lemma 4: Positive Correlation

Fix a satisfying assignment x of F. If F is insensitive to coordinate j on x (*i.e.* $S(F,x,j)=0$) then $I_j(x,\pi,F)=0$ for all permutations π, and so we first note that

$$\mathop{\mathbf{E}}_{\pi}\left[2^{I(x,\boldsymbol{\pi},F)}\right] = \mathop{\mathbf{E}}_{\pi}\left[\prod_{i:\ S(F,x,i)=1} 2^{I_i(x,\boldsymbol{\pi},F)}\right]. \tag{6}$$

Fix an i such that $S(F,x,i)=1$. At this point, it would be convenient to adopt the equivalent view of a random permutation π as a function $\pi:[n]\to[0,1]$ where we choose the value of each $\pi(k)$ independently and uniformly at random from $[0,1]$ (ordering $[n]$ according to π defines a uniformly random permutation). From this point of view $2^{I_i(x,\boldsymbol{\pi},F)}$ is a function from $[0,1]^n\to\{1,2\}$. The key observation that we make now is that the n functions $2^{I_i(x,\pi,F)}$ for $1\leq i\leq n$ are monotonically increasing in the coordinates at which x is 1, and decreasing in the coordinates at which x is 0.

By monotonicity, we know that $I_i(x,\pi,F)=1$ if only if there is a clause $C=x_i\vee x_{i_2}\vee\ldots\vee x_{i_{k'}}$ in F, where $x_{i_2}=\ldots=x_{i_{k'}}=0$ (note that $x_i=1$ by monotonicity) and $\pi(i_2),\pi(i_3),\ldots,\pi(i_{k'})<\pi(i)$. By this characterization, we see that

- Increasing $\pi(i)$ can only increase $I_i(x,\pi,F)$, and decreasing $\pi(i)$ can only decrease $I_i(x,\pi,F)$.
- Increasing $\pi(j)$ for some j where $x_j=0$ can only decrease $I_i(x,\pi,F)$, and decreasing $\pi(j)$ can only increase $I_i(x,\pi,F)$.

– Finally, $I_i(x, \pi, F)$ is not affected by changes to $\pi(j)$ when $j \neq i$ and $x_j = 1$.

Therefore, the functions $2^{I_i(x,\pi,F)}$ where $S(F, x, i) = 1$ are all unate with the same orientation[7] and so by the FKG correlation inequality [FKG71], they are positively correlated. We conclude that

$$\mathop{\mathbf{E}}_{\pi}\left[\prod_{i:\ S(F,x,i)=1} 2^{I_i(x,\boldsymbol{\pi},F)}\right] \geq \prod_{i:\ S(F,x,i)=1} \mathop{\mathbf{E}}_{\pi}\left[2^{I_i(x,\boldsymbol{\pi},F)}\right] = \prod_{i=1}^{n} \mathop{\mathbf{E}}_{\pi}\left[2^{I_i(x,\boldsymbol{\pi},F)}\right],$$

where in the final inequality we again use the fact that $I_j(x, \boldsymbol{\pi}, F) = 0$ for all π if $S(F, x, j) = 0$. This proves Lemma 4.

4.2 Proof of Theorem 3: The Lower Bound

In this section we construct a monotone k-CNF formula with large average sensitivity. We will need a combinatorial identity.

Lemma 5. *Let $k, \ell \geq 0$. Then $\sum_{s=0}^{m} \binom{m}{s} k^s \ell^{m-s} s = mk(k+\ell)^{m-1}$.*

Proof. First, if $k + \ell = 1$, then both sides equal the expected number of heads in m coin tosses with head probability k. Otherwise, we divide both sides by $(k+\ell)^m$ and apply that argument to $\frac{k}{k+\ell}$ and $\frac{\ell}{k+\ell}$. □

Proof (Proof of Theorem 3). This function will be the tribes function $F := \mathrm{Tribes}_m^k$ over $n = km$ variables:

$$(x_1^{(1)} \vee x_2^{(1)} \cdots \vee x_k^{(1)}) \wedge (x_1^{(2)} \vee x_2^{(2)} \cdots \vee x_k^{(2)}) \wedge \cdots \wedge (x_1^{(m)} \vee x_2^{(m)} \cdots \vee x_k^{(m)})$$

This is a k-CNF formula with $\mathbf{E}[F] = \left(1 - 2^{-k}\right)^m$. We set $m := \left\lceil \ln(\mu)/\ln\left(1 - 2^{-k}\right)\right\rceil$, which yields $\mu\left(1 - 2^{-k}\right) \leq \mathbf{E}[F] \leq \mu$. Let us compute $\mathbf{AS}(F)$. For a satisfying assignment x, $S(F, x)$ is the number of clauses in Tribes_m^k containing exactly one satisfied literal. The number of satisfying assignments x of sensitivity s is exactly $\binom{m}{s} k^s \left(2^k - k - 1\right)^{m-s}$, as there are $2^k - k - 1$ ways to satisfy more than one literal in a k-clause. Thus,

$$\mathbf{AS}(F) = 2^{-n+1} \sum_{x \in \mathrm{sat}(F)} S(F, x) = 2^{-mk+1} \sum_{s=0}^{m} \binom{m}{s} k^s \left(2^k - k - 1\right)^{m-s} s$$

Applying Lemma 5 with $\ell = 2^k - k - 1$, we get

$$\mathbf{AS}(F) = 2^{-mk+1} mk(2^k - 1)^{m-1} = \frac{\left(2^k - 1\right)^m}{2^{km}} \frac{2mk}{2^k - 1} = \frac{2\,\mathbf{E}[F]mk}{2^k - 1}$$

Recall that $m \geq \frac{\ln(\mu)}{\ln(1-2^{-k})}$ and $\mathbf{E}[F] \geq \left(1 - 2^{-k}\right)\mu$. Thus,

$$\mathbf{AS}(F) = \frac{2\,\mathbf{E}[F]mk}{2^k - 1} \geq \frac{2k\mu\left(1 - 2^{-k}\right)\ln(\mu)}{(2^k - 1)\ln\left(1 - 2^{-k}\right)} = \frac{2k\mu\ln(\mu)}{2^k \ln(1 - 2^{-k})} = 2k\mu \ln\left(\frac{1}{\mu}\right)(1 - \varepsilon_k),$$

for some ε_k that quickly converges to 0 as k grows. This proves Theorem 3. □

[7] This means for each $1 \leq i \leq n$, they are either all monotonically increasing in $\pi(i)$ or all decreasing in $\pi(i)$.

4.3 A gap between average and maximum sensitivity

Recall that the maximum sensitivity $\mathbf{S}(F)$ of a Boolean function F is the quantity $\max_{x\in\{0,1\}^n}[S(F,x)]$. Clearly we have that $\mathbf{AS}(F) \le \mathbf{S}(F)$, and this inequality is tight when $F = \mathsf{PAR}_n$, the parity function over n variables. Servedio conjectured that unlike the case for PAR_n, the average sensitivity of a *monotone* Boolean function F is always asymptotically smaller than its maximum sensitivity [O'D12]:

Conjecture 1 (Servedio). There exists universal constants $K > 0$ and $\delta < 1$ such that the following holds. Let $F : \{0,1\}^n \to \{0,1\}$ be any monotone Boolean function. Then $\mathbf{AS}(F) \le K \cdot \mathbf{S}(F)^\delta$.

In addition to being an interesting and natural question, Servedio's conjecture also has implications for Mansour's conjecture [Man94] on the Fourier spectrum of depth-2 AC^0, a longstanding open problem in analysis of Boolean functions and computational learning theory [O'D12,GKK08]. The conjecture can be checked to be true for the canonical examples of monotone Boolean functions such as majority ($\mathbf{AS}(\mathsf{MAJ}_n) = \Theta(\sqrt{n})$ whereas $\mathbf{S}(\mathsf{MAJ}_n) = \lceil n/2 \rceil$), and the Ben-Or-Linial Tribes function ($\mathbf{AS}(\mathsf{Tribes}_{k,2^k}) = \Theta(k)$ whereas $\mathbf{S}(\mathsf{Tribes}_{k,2^k}) = 2^k$). O'Donnell and Servedio have shown the existence of a monotone function F with $\mathbf{AS}(F) = \Omega(\mathbf{S}(F)^{0.61})$ [OS08], and this is the best known lower bound on the value of δ in Conjecture 1.

The current best separation between the two quantities is $\mathbf{AS}(F) \le \sqrt{2/\pi} \cdot \mathbf{S}(F)(1 + o(1)) \approx 0.797 \cdot \mathbf{S}(F)(1 + o(1))$ where $o(1)$ is a term that tends to 0 as $\mathbf{S}(F)$ grows[8], which follows as a corollary of Blais's [Bla11] sharpening of an isoperimetric inequality of O'Donnell and Servedio [OS08]. We now show that our upper bound in Theorem 2 yields an improved separation. We recall a basic fact from [Nis91] characterizing the maximum sensitivity of a monotone Boolean function by its CNF and DNF widths:

Fact 41 *Let $F : \{0,1\}^n \to \{0,1\}$ be a monotone Boolean function. Then*

$$\mathbf{S}(F) = \max\{\textbf{DNF-width}(F), \textbf{CNF-width}(F)\}.$$

Corollary 2. *monotone-sepration Let F be a monotone Boolean function. Then $\mathbf{AS}(F) \le \ln(2) \cdot \mathbf{S}(F)(1 + o(1)) \le 0.694 \cdot \mathbf{S}(f)(1 + o(1))$, where $o(1)$ is a term that goes to 0 as $\mathbf{S}(F)$ grows.*

Proof. By Fact 41, we have $\textbf{CNF-width}(F) \le \mathbf{S}(F)$ and $\textbf{CNF-width}(\neg F) = \textbf{DNF-width}(F) \le \mathbf{S}(F)$. Applying the upper bound of Theorem 2 to both F and $\neg F$, we get

$$\mathbf{AS}(F) \le \min\{2\mu\ln(1/\mu), 2(1-\mu)\ln(1/(1-\mu))\} \cdot \mathbf{S}(F)(1 + o(1)),$$

where $\mu = \mu(F)$. The proof is complete by noting that $\min\{2\mu\ln(1/\mu), 2(1-\mu)\ln(1/(1-\mu))\} \le \ln(2)$ for all $\mu \in [0,1]$. □

[8] Note that this additive $o(1)$ term is necessary as $\mathbf{AS}(F) = \mathbf{S}(F) = 1$ for the monotone function $F(x) = x_1$.

5 Average sensitivity of s-clause CNFs

Let F be computed by an s-clause CNF. It is straightforward to check that $\mathbf{Pr}[F(\boldsymbol{x}) \neq G(\boldsymbol{x})] \leq \varepsilon$ and $\mathbf{AS}(F) \leq \mathbf{AS}(G) + \varepsilon \cdot n$, if G is the CNF obtained from F by removing all clauses of width greater than $\log(s/\varepsilon)$. When $s = \Omega(n)$ we may apply Amano's theorem to G and take $\varepsilon = O(1/n)$ to conclude that $\mathbf{AS}(F) = O(\log s)$. Building on the work of Linial, Mansour and Nisan [LMN93], Boppana employed Håstad's switching lemma to prove that in fact $\mathbf{AS}(F) = O(\log s)$ continues to hold for all values of $s = o(n)$. Here we give an elementary proof of Theorem 4 that sharpens and simplifies Boppana's result. A bound of $\mathbf{AS}(F) \leq \log(s)(1 + o(1))$, which is tight up to lower order terms by considering the parity function over $\log s$ variables, does not appear to have been known prior to this work.[9]

Proof (Proof of Theorem 4). We write $F = G \wedge H$, where G consists of all clauses of width at most τ and the threshold $\tau \in [s]$ will be chosen later. By the subadditivity of average sensitivity, we see that

$$\mathbf{AS}(F) \leq \mathbf{AS}(G) + \mathbf{AS}(H) \leq \tau + \sum_{C \,\in\, H} \mathbf{AS}(C) = \tau + \sum_{C \in H} \frac{|C|}{2^{|C|-1}} \leq \tau + s \cdot \frac{\tau}{2^{\tau-1}}.$$

Here the second inequality is by Amano's theorem applied to G and the subadditivity of average sensitivity applied to H, and the last inequality holds because $z \mapsto z/2^{z-1}$ is a decreasing function. Choosing $\tau := \log s + \log\log s$ yields $\mathbf{AS}(F) \leq \log s + \log\log s + 2 + o(1)$ and completes the proof. □

6 Acknowledgements

We thank Eric Blais and Homin Lee for sharing [Bla11] and [Lee12] with us. We also thank Rocco Servedio and Navid Talebanfard for helpful discussions.

References

Ama11. Kazuyuki Amano. Tight bounds on the average sensitivity of k-CNF. *Theory of Computing*, 7(4):45–48, 2011. 1, 1

Ber67. Arthur Bernstein. Maximally connected arrays on the n-cube. *SIAM Journal on Applied Mathematics*, pages 1485–1489, 1967. 1

BKS99. Itai Benjamini, Gil Kalai, and Oded Schramm. Noise sensitivity of Boolean functions and applications to percolation. *Publications Mathématiques de l'IHÉS*, 90(1):5–43, 1999. 1

BL90. Michael Ben-Or and Nathan Linial. Collective coin flipping. In Silvio Micali and Franco Preparata, editors, *Randomness and Computation*, volume 5 of *Advances in Computing Research: A research annual*, pages 91–115. JAI Press, 1990. 1

[9] Working through the calculations in Boppana's proof one gets a bound of $\mathbf{AS}(F) \leq 3\log s$.

Bla11. Eric Blais. Personal communication, 2011. 1.1, 4.3, 6

Bop97. Ravi B. Boppana. The average sensitivity of bounded-depth circuits. *Inf. Process. Lett.*, 63(5):257–261, 1997. 1, 1, 1

BT96. Nader Bshouty and Christino Tamon. On the Fourier spectrum of monotone functions. *Journal of the ACM*, 43(4):747–770, 1996. 1

BT13. Eric Blais and Li-Yang Tan. Approximating Boolean functions with depth-2 circuits. In *Conference on Computational Complexity*, 2013. 1

DHK+10. Ilias Diakonikolas, Prahladh Harsha, Adam Klivans, Raghu Meka, Prasad Raghavendra, Rocco Servedio, and Li-Yang Tan. Bounding the average sensitivity and noise sensitivity of polynomial threshold functions. In *Proceedings of the 42nd Annual ACM Symposium on Theory of Computing*, pages 533–542, 2010. 1

DS05. Irit Dinur and Samuel Safra. On the hardness of approximating minimum vertex cover. *Annals of Mathematics*, 162(1):439–485, 2005. 1

FKG71. C. M. Fortuin, P. W. Kasteleyn, and J. Ginibre. Correlation inequalities on some partially ordered sets. *Comm. Math. Phys.*, 22:89–103, 1971. 1.1, 4.1

Fri98. Ehud Friedgut. Boolean functions with low average sensitivity depend on few coordinates. *Combinatorica*, 18(1):27–36, 1998. 1

Fri99. Ehud Friedgut. Sharp thresholds of graph properties, and the k-SAT problem. *Journal of the American Mathematical Society*, 12(4):1017–1054, 1999. 1

GKK08. Parikshit Gopalan, Adam Kalai, and Adam Klivans. A query algorithm for agnostically learning DNF? In *Proceedings of the 21st Annual Conference on Learning Theory*, pages 515–516, 2008. 4.3

Har64. Lawrence Harper. Optimal assignments of numbers to vertices. *Journal of the Society for Industrial and Applied Mathematics*, 12(1):131–135, 1964. 1

Har76. Sergiu Hart. A note on the edges of the n-cube. *Discrete Mathamatics*, 14(2):157–163, 1976. 1

Hås01. Johan Håstad. A slight sharpening of LMN. *Journal of Computer and System Sciences*, 63(3):498–508, 2001. 1

Khr71. V. M. Khrapchenko. A method of determining lower bounds for the complexity of π-schemes. *Math. Notes Acad. Sci. USSR*, 10(1):474–479, 1971. 3

KKL88. Jeff Kahn, Gil Kalai, and Nathan Linial. The influence of variables on Boolean functions. In *Proceedings of the 29th Annual Symposium on Foundations of Computer Science*, pages 68–80, 1988. 1

Lee12. Homin Lee. Personal communication, 2012. 6, 6

Lin64. J. H. Lindsey. Assignment of numbers to vertices. *Amer. Math. Monthly*, 71:508–516, 1964. 1

LMN93. Nathan Linial, Yishay Mansour, and Noam Nisan. Constant depth circuits, Fourier transform and learnability. *Journal of the ACM*, 40(3):607–620, 1993. 1, 5

Man94. Yishay Mansour. Learning Boolean functions via the Fourier Transform. In Vwani Roychowdhury, Kai-Yeung Siu, and Alon Orlitsky, editors, *Theoretical Advances in Neural Computation and Learning*, chapter 11, pages 391–424. Kluwer Academic Publishers, 1994. 4.3

Man95. Yishay Mansour. An $O(n^{\log\log n})$ learning algorithm for DNF under the uniform distribution. *Journal of Computer and System Sciences*, 50(3):543–550, 1995. 1

Nis91. Noam Nisan. CREW PRAMs and decision trees. *SIAM Journal on Computing*, 20(6):999–1007, 1991. 4.3

O'D07. Ryan O'Donnell. Lecture 29: Open problems. Scribe notes for a course on Analysis of Boolean Functions at Carnegie Mellon University, 2007. 1

O'D12. Ryan O'Donnell. Open problems in analysis of boolean functions. *CoRR*, abs/1204.6447, 2012. 1.1, 4.3, 4.3

OS08. Ryan O'Donnell and Rocco Servedio. Learning monotone decision trees in polynomial time. *SIAM Journal on Computing*, 37(3):827–844, 2008. 1, 4.3

OW07. Ryan O'Donnell and Karl Wimmer. Approximation by DNF: examples and counterexamples. In *Proceedings of the 34th Annual Colloquium on Automata, Languages and Programming*, pages 195–206, 2007. 1

PPZ97. Ramamohan Paturi, Pavel Pudlák, and Francis Zane. Satisfiability coding lemma. In *Proceedings of the 38th IEEE Symposium on Foundations of Computer Science*, pages 566–574, 1997. 1, 2, 1, 2

RRS+12. Dana Ron, Ronitt Rubinfeld, Muli Safra, Alex Samorodnitsky, and Omri Weinstein. Approximating the influence of monotone boolean functions in $o(n)$ query complexity. *TOCT*, 4(4):11, 2012. 1

Shi00. Y. Shi. Lower bounds of quantum black-box complexity and degree of approximating polynomials by influence of boolean variables. *Inform. Process. Lett.*, 75(1-2):79–83, 2000. 1

Tra09. Patrick Traxler. Variable influences in conjunctive normal forms. In *SAT*, pages 101–113, 2009. 1, 1