THEORY OF COMPUTING LIBRARY
GRADUATE SURVEYS 4 (2011), pp. 1–27
www.theoryofcomputing.org

Variations on the Sensitivity Conjecture

Pooya Hatami Raghav Kulkarni Denis Pankratov

Received: November 2, 2010; published: June 22, 2011.

Abstract: The *sensitivity* of a Boolean function f of n Boolean variables is the maximum over all inputs x of the number of positions i such that flipping the i-th bit of x changes the value of $f(x)$. Permitting to flip disjoint blocks of bits leads to the notion of *block sensitivity*, known to be polynomially related to a number of other complexity measures of f, including the decision-tree complexity, the polynomial degree, and the certificate complexity. A long-standing open question is whether sensitivity also belongs to this equivalence class. A positive answer to this question is known as the *Sensitivity Conjecture*. We present a selection of known as well as new variants of the Sensitivity Conjecture and point out some weaker versions that are also open. Among other things, we relate the problem to Communication Complexity via recent results by Sherstov (QIC 2010). We also indicate new connections to Fourier analysis.

ACM Classification: F.0, F.1, F.2

AMS Classification: 68-02, 68Q05, 68Q25

Key words and phrases: sensitivity, block sensitivity, complexity measures of Boolean functions

1 The Sensitivity Conjecture

For Boolean strings $x, y \in \{0,1\}^n$ let $x \oplus y \in \{0,1\}^n$ denote the *coordinate-wise exclusive or* of x and y. Let $e_i \in \{0,1\}^n$ denote an n-bit Boolean string whose i^{th} bit is 1 and the rest of the bits are 0.

Let $f : \{0,1\}^n \to \{0,1\}$ be a Boolean function. On an input $x \in \{0,1\}^n$, the i^{th} bit is said to be *sensitive* for f if $f(x \oplus e_i) \neq f(x)$, i. e., flipping the i^{th} bit results in flipping the output of f. The sensitivity of f on input x, denoted by $s(f,x)$, is the number of bits that are sensitive for f on input x.

Definition 1.1. The *sensitivity* of a Boolean function f, denoted by $s(f)$, is the maximum value of $s(f,x)$ over all choices of x.

DOI: 10.4086/toc.gs.2011.004

For $B \subseteq [n] = \{1,2,\ldots,n\}$ let $e_B \in \{0,1\}^n$ denote the characteristic vector of B, i. e., the i^{th} bit of e_B is 1 if $i \in B$ and 0 otherwise. We write $\bar{x}$ for $x \oplus e_{[n]}$. We say that a "block" B is sensitive for f on x if $f(x \oplus e_B) \neq f(x)$. The block sensitivity of f on x, denoted by $bs(f,x)$, is the maximum number of *pairwise disjoint* sensitive blocks of f on x.

Definition 1.2. The *block sensitivity* of a Boolean function f, denoted by $bs(f)$, is the maximum possible value of $bs(f,x)$ over all choices of x.

The study of sensitivity of Boolean functions originated from Stephen Cook and Cynthia Dwork [10] and Rüdiger Reischuk [29]. They showed an $\Omega(\log s(f))$ lower bound on the number of steps required to compute a Boolean function f on a CREW PRAM. A CREW PRAM, abbreviated from Consecutive Read Exclusive Write Parallel RAM, is a collection of synchronized processors computing in parallel with access to a shared memory with no write conflicts. The minimum number of steps required to compute a function f on a CREW PRAM is denoted by $\mathrm{CREW}(f)$. After Cook, Dwork and Reischuk introduced sensitivity, Noam Nisan [23] found a way to modify the definition of sensitivity to characterize $\mathrm{CREW}(f)$ exactly. Nisan introduced the notion of *block sensitivity* and proved that $\mathrm{CREW}(f) = \Theta(\log bs(f))$ for every Boolean function f [23].

Obviously, for every Boolean function f,

$$s(f) \leq bs(f)\,.$$

Block sensitivity turned out to be polynomially related to a number of other complexity measures (see Section 2); however, to this day it remains unknown whether block sensitivity is bounded above by a polynomial in sensitivity. This problem was first stated by Nisan and Mario Szegedy [24].

Problem 1.3 (Nisan and Szegedy)**.** Is it true that for every Boolean function f,

$$bs(f) \leq \mathrm{poly}(s(f))\,?$$

In fact, Nisan and Szegedy even suggested the possibility

$$bs(f) = O(s(f)^2)\,. \tag{1.1}$$

Decades of failed attempts at producing a super-quadratic gap between these quantities have led the community to lean toward a positive answer. We refer to the positive answer to Problem 1.3 as the **Sensitivity Conjecture**.

The rest of the paper is organized as follows. In Section 2 we describe complexity measures of Boolean functions polynomially related to block sensitivity. In Section 3 we review progress on the Sensitivity Conjecture. In Section 4 we give a brief introduction to the field of communication complexity. In Section 5 we present alternative formulations of the Sensitivity Conjecture and point out weaker versions that are also open. Along the way we encounter important examples of Boolean functions. We present these functions in Section 6.

To the best of our knowledge the results stated in this paper without attribution have not appeared previously in the literature.

2 Measures related to block sensitivity

Block sensitivity is polynomially related to several other complexity measures of Boolean functions, some of which we describe in this section.

A *deterministic decision tree* on n variables $x_1, \dots, x_n$ is a rooted binary tree, whose internal nodes are labeled with variables, and the leaves are labeled 0 or 1. There are exactly two edges leaving each internal node, one labeled 0 and one labeled 1. To evaluate such a tree on input x, start at the root and query the corresponding variable, then move to the next node along the edge labeled with the outcome of the query. Repeat until a leaf is reached, at which point the label of the leaf is declared to be the output of the evaluation. A decision tree computes a Boolean function f if it agrees with f on all inputs.

Definition 2.1. The *deterministic decision-tree complexity* of a Boolean function f, denoted by $D(f)$, is the depth of a minimum-depth decision tree that computes f.

One way to extend the deterministic decision tree model is to add randomness to the computation. A randomized decision tree computing a Boolean function f with error probability at most $1/3$ is given by a probability distribution μ on all deterministic decision trees, such that for all x the probability that a tree drawn from μ outputs $f(x)$ is at least $2/3$. The *depth of a randomized decision tree* defined by μ is the maximum depth of a deterministic decision tree T with $\mu(T) > 0$.

Definition 2.2. The *bounded-error randomized decision-tree complexity* of a Boolean function f, denoted by $R_2(f)$, is the minimum depth of a randomized decision tree computing f with error probability at most $1/3$. (The subscript 2 refers to permitting 2-sided error.)

We denote the *quantum decision-tree complexity with bounded error* of a Boolean function f by $Q_2(f)$. Discussion of quantum complexity is outside the scope of this paper. For an introduction to quantum complexity see a survey by de Wolf [37].

A *certificate* of a Boolean function f on input x is a subset $S \subset [n]$ such that

$$(\forall y \in \{0,1\}^n)(x|_S = y|_S \Rightarrow f(x) = f(y)) .$$

The *certificate complexity* of a Boolean function f on input x, denoted by $C(f,x)$, is the minimum size of a certificate of f on x.

Definition 2.3. The *certificate complexity* of a Boolean function f, also known as the *non-deterministic decision-tree complexity* of f and denoted by $C(f)$, is the maximum of $C(f,x)$ over all choices of x.

Definition 2.4. A polynomial $p : \mathbb{R}^n \to \mathbb{R}$ *represents* f if

$$(\forall x \in \{0,1\}^n)(p(x) = f(x)) .$$

The *degree* of a Boolean function f, denoted by $\deg(f)$, is the degree of the *unique multilinear* polynomial that represents f.

Definition 2.5. A polynomial $p : \mathbb{R}^n \to \mathbb{R}$ *approximately represents* f if

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

The *approximate degree* of a Boolean function f, denoted by $\widetilde{\deg}(f)$, is the minimum degree of a polynomial that approximately represents f.

	$bs(f)$	$D(f)$	$\deg(f)$	$C(f)$
$bs(f)$	1(1)	1 [*](1 [*])	2 [24]($\log_3 6$ [25][3])	1 [*](1 [*])
$D(f)$	3 [3, 23](2 [][4])	1(1)	3 [21]($\log_3 6$ [25][3])	2 [3][2](2 [][4])
$\deg(f)$	3 [3, 23](2 [][4])	1 [*](1 [*])	1(1)	2 [3](2 [][4])
$C(f)$	2 [23]($\log_4 5$ [5, 1][1])	1 [*](1 [*])	3 [21]($\log_3 6$ [25][3])	1(1)

Table 1: Known polynomial relations between various complexity measures. An entry in the table shows the polynomial upper bound on the measure from a row in terms of a measure from a column and the biggest known gap between two measures. The references to the papers, where the corresponding results can be found, are given in square brackets.

Definition 2.6. Complexity measures A and B are *polynomially related* if there exist polynomials p_1, p_2 over $\mathbb{R}$

$$(\forall f)\,[A(f) \leq p_1(B(f)) \text{ and } B(f) \leq p_2(A(f))]\,.$$

Theorem 2.7 ([3],[23],[24])**.** *The following complexity measures of Boolean functions are all polynomially related:*

$$bs(f), \quad D(f), \quad R_2(f), \quad C(f), \quad \deg(f), \quad \widetilde{\deg}(f), \quad Q_2(f)\,.$$

Table 1 presents a quick summary of the known polynomial relations between complexity measures that play a prominent role in this paper. An entry from the table shows the smallest known degree of a polynomial in the corresponding measure from the column that gives an upper bound on the corresponding measure from the row, as well as the degree of the biggest known gap between two measures. An entry also contains references to papers where the result can be found. References of the form [*] indicate that the result is immediate from the definitions of complexity measures. For example, entry 3 [21]($\log_3 6$ [25]) in the second row and third column means that $D(f) = O(\deg(f)^3)$ (see [21]) and that there is a Boolean function f, for which $D(f) = \Omega(\deg(f)^{\log_3 6})$ (see [25]). For a thorough treatment of polynomial relations between various complexity measures of Boolean functions (including variants of quantum query complexity) we refer to the survey by Buhrman and de Wolf [6]. That survey includes a full proof of Theorem 2.7 and proofs of most of the relations in Table 1.

Using Theorem 2.7, one immediately obtains many equivalent formulations of the Sensitivity Conjecture. For instance, "is $\deg(f) \leq \mathrm{poly}(s(f))$ for every Boolean function f?" The following stronger version of this inequality "is conjectured" according to Gotsman and Linial [13]:

$$\deg(f) = O(s(f)^2)\,. \tag{2.1}$$

[1]The construction appeared in [5] before the notion of block sensitivity was introduced. The analysis of $C(f)$ and $bs(f)$ of the example appears in [1].

[2]The result is due to [4, 15, 35].

[3]The example is due to Kushilevitz and appears in footnote 1 on p. 560 of the Nisan-Wigderson paper [25]. See Example 6.3.2 of this paper.

[4]These gaps are demonstrated by an AND-of-ORs function; see Example 6.2.1 in this paper.

Currently inequalities (1.1) and (2.1) are not comparable. A function constructed by David Rubinstein (see Example 6.1.1) shows that the quadratic bound would be best possible in both cases; for that function f, we have $\deg(f) = n$, $bs(f) = n/2$, $s(f) = \sqrt{n}$.

The purpose of this paper is to point out some nontrivial variations on the Sensitivity Conjecture that, to our knowledge, have not been stated explicitly in the literature. We also propose several weaker versions of the Sensitivity Conjecture which might provide starting points.

We introduce the following pictorial notation to indicate relations between the statements appearing in this paper and the Sensitivity Conjecture:
☺ - a consequence of the Sensitivity Conjecture.
☹ - implies the Sensitivity Conjecture, but the reverse implication is not known. These might be good candidates for refutation.
😐 - equivalent to the Sensitivity Conjecture.
😉 - conditionally equivalent to the Sensitivity Conjecture.

3 Progress on the Sensitivity Conjecture

The progress on the Sensitivity Conjecture has been limited. We say that a Boolean function $f : \{0,1\}^n \to \{0,1\}$ *depends on the* i^{th} *variable* if for some input x the i^{th} bit is sensitive for f on x. Hans-Ulrich Simon [34] proved the first non-trivial lower bound on the sensitivity of an arbitrary Boolean function.

Theorem 3.1 (H.-U. Simon)**.** *For every Boolean function $f : \{0,1\}^n \to \{0,1\}$ we have*

$$s(f)4^{s(f)} \geq n'$$

where n' is the number of variables on which f depends.

Now we present Simon's proof. First we introduce some notation. Let Q_n denote the n-cube graph, i. e., $V(Q_n) = \{0,1\}^n$ and two vertices are adjacent if the corresponding strings differ in exactly one position. We denote the minimum degree of the graph G by $\delta(G)$.

Lemma 3.2. *For a non-empty subgraph $G = (V,E)$ of Q_n, we have*

$$|V| \geq 2^{\delta(G)}.$$

Proof. We will proceed by induction on n. The claim is trivially true for $n = 1$. Now, in the inductive step, for $j \in \{0,1\}$, let $G_j = (V_j, E_j)$ be the induced subgraph of G on the set $V_j = \{v \in V \mid v_1 = j\}$, where v is the string $v_1 v_2 \cdots v_n$. If G_j is empty then $G = G_{1-j}$, so $\delta(G_{1-j}) = \delta(G)$ and the claim follows by induction. Otherwise, since a vertex $v \in V_j$ can have at most one neighbor in V_{1-j}, we have $\delta(G_j) \geq \delta(G) - 1$ for $j \in \{0,1\}$. From the inductive hypothesis it follows that $|V| = |V_0| + |V_1| \geq 2^{\delta(G_0)} + 2^{\delta(G_1)} \geq 2^{\delta(G)}$. □

Proof of Theorem 3.1. Let $C_G(v)$ denote the connected component of the graph G containing vertex v. To simplify the proof, we will assume that f depends on all variables, i. e., $n = n'$. The case $n \neq n'$ follows immediately. For $i \in [n]$, pick x such that $f(x) \neq f(x \oplus e_i)$. Define the following induced subgraphs of Q_n:

$$\begin{array}{lll}
G_0 = (V_0, E_0) & \text{where} & V_0 = \{v \in V(Q_n) \mid v_i = x_i, f(v) = f(x)\}, \\
G_1 = (V_1, E_1) & \text{where} & V_1 = \{v \in V(Q_n) \mid v_i = 1 - x_i, f(v) = f(x \oplus e_i)\}, \\
\widetilde{G_0} = (\widetilde{V_0}, \widetilde{E_0}) & \text{where} & \widetilde{V_0} = \{v \in V_0 \mid v \in C_{G_0}(x), v \oplus e_i \in C_{G_1}(x \oplus e_i)\}, \\
\widetilde{G_1} = (\widetilde{V_1}, \widetilde{E_1}) & \text{where} & \widetilde{V_1} = \{v \in V_1 \mid v \oplus e_i \in \widetilde{V_0}\}.
\end{array}$$

Clearly, $x \in \widetilde{V_0}$ and for any $\widetilde{v} \in \widetilde{V_0}$ the i^{th} bit is sensitive for f on $\widetilde{v}$. Now, for a vertex $\widetilde{v} \in \widetilde{V_0}$ at most $s(f)$ neighbors of $\widetilde{v}$ in Q_n lie in $\widetilde{V_1}$. Among the rest of the neighbors of $\widetilde{v}$ in Q_n, at most $s(f) - 1$ do not become neighbors of $\widetilde{v} \oplus e_i$ in $\widetilde{G_1}$ when their i^{th} variable is flipped. Hence the remaining $n - 2s(f) + 1$ neighbors of $\widetilde{v}$ belong to $\widetilde{V_0}$. This shows that $\delta(\widetilde{G_0}) \geq n - 2s(f) + 1$.

The total number of inputs $y \in \{0,1\}^n$ such that the i^{th} bit is sensitive for f on y is at least $|\widetilde{V_0}| \geq 2^{n-2s(f)+1}$ by Lemma 3.2. Then the total number of sensitive bits over all inputs is at least $n2^{n-2s(f)+1}$. Therefore, $n2^{n-2s(f)+1} \leq s(f)2^n$, since the number of sensitive bits for a particular input is at most $s(f)$. This completes the proof. □

A simple calculation leads to the following corollary.

Corollary 3.3 (H.-U. Simon)**.** *For every Boolean function $f : \{0,1\}^n \to \{0,1\}$ we have*

$$s(f) \geq \frac{1}{2}\log n' - \frac{1}{2}\log\log n' + \frac{1}{2}$$

where n' is the number of variables on which f depends.

Obviously, $bs(f) \leq n'$, so Theorem 3.1 implies that

$$bs(f) \leq s(f)4^{s(f)} \tag{3.1}$$

for every Boolean function f. This upper bound was later improved by Claire Kenyon and Samuel (Sandy) Kutin [17]. They introduced a notion of *ℓ-block sensitivity*, which considers only sensitive blocks of size at most ℓ. Let $bs_\ell(f)$ denote the *ℓ-block sensitivity* of a Boolean function f.

Theorem 3.4 (Kenyon and Kutin)**.** *For any Boolean function f, we have*

$$s(f) \geq (bs_\ell(f)(\ell-1)!/e)^{1/\ell}.$$

In particular, this theorem implies that sensitivity is polynomially related to ℓ-block sensitivity for any constant ℓ. It is easy to see that the size of any minimal sensitive block of f is at most $s(f)$. Hence $bs(f) = bs_{s(f)}(f)$ and the following corollary follows immediately from Stirling's approximation.

Corollary 3.5 (Kenyon and Kutin)**.** *For any Boolean function f, we have*

$$bs(f) \leq \left(\frac{e}{\sqrt{2\pi}}\right) e^{s(f)} \sqrt{s(f)}.$$

This is the best known upper bound on block sensitivity in terms of sensitivity. On the other hand, no gap greater than Rubinstein's mentioned quadratic gap (Example 6.1.1) has been found.

Gaps between sensitivity and some other complexity measures are surveyed by Buhrman and de Wolf [6].

Nisan [23] showed that the Sensitivity Conjecture is true for monotone Boolean functions.

Proposition 3.6 (Nisan). *For a monotone Boolean function f, we have $C(f) = s(f) = bs(f)$.*

Proof. It suffices to show that $C(f) \leq s(f)$, since $s(f) \leq bs(f) \leq C(f)$. Fix x such that $C(f,x) = C(f)$ and let S be a minimum certificate of f on x, i. e., $|S| = C(f,x)$. Without loss of generality assume that $f(x) = 1$. Consider $y \in \{0,1\}^n$, such that $y_i = 1$ if $i \in S$ and $y_i = 0$ otherwise. For each $i \in S$, the i^{th} bit is sensitive for f on y, since otherwise $S - \{i\}$ would be a certificate for f on x contradicting the minimality of S. Hence $s(f) \geq C(f)$. □

A Boolean function $f : \{0,1\}^n \to \{0,1\}$ is *invariant under a permutation* $\sigma : [n] \to [n]$, if for any string x, $f(x_1, \dots, x_n) = f(x_{\sigma(1)}, \dots, x_{\sigma(n)})$. The set of all permutations under which f is invariant forms a group, called the *invariance group* of f. A Boolean function is said to be *transitive* if its invariance group Γ is *transitive*, i. e., for each $i, j \in [n]$ there is a permutation $\sigma \in \Gamma$ such that $\sigma(i) = j$.

For a graph G on v vertices, let $\langle G \rangle$ denote a string of length $n = \binom{v}{2}$ over alphabet $\{0,1\}$ encoding the adjacency relation. A *graph property* is a Boolean function $f : \{0,1\}^n \to \{0,1\}$, such that for any two isomorphic graphs G and G' we have $f(\langle G \rangle) = f(\langle G' \rangle)$. Clearly, a graph property is a transitive function. György Turán [36] proved that any property of v-vertex graphs has sensitivity $\Omega(v) = \Omega(\sqrt{n})$. Turán asked if this result generalizes to transitive functions, i. e., if every transitive function on n variables has sensitivity at least $\Omega(\sqrt{n})$. Sourav Chakraborty [7] answered this question in the negative by constructing a transitive function with sensitivity $\Theta(n^{1/3})$ and block sensitivity $\Theta(n^{2/3})$ (see Example 6.4.1). We propose the following modification of Turán's question:

Question 3.7. If $f : \{0,1\}^n \to \{0,1\}$ is transitive and $f(\mathbf{0}) \neq f(\mathbf{1})$, is then $s(f) = \Omega(\sqrt{n})$?

Remark 3.8. Ronald Rivest and Jean Vuillemin [30] proved that if n is a prime power and $f(\mathbf{0}) \neq f(\mathbf{1})$ then $D(f) = n$. From their proof it can be immediately inferred that the stronger statement $\deg(f) = n$ also holds. This implies that if conjecture (2.1) is true, it would give an affirmative answer to Question 3.7 for n that are prime powers.

4 Communication complexity

In Section 5 we shall examine connections between sensitivity and other complexity measures. Much of that material will either be directly related to the communication complexity (Sections 5.1 and 5.3) or use communication complexity as a tool (Section 5.2). In this section we give the communication complexity background we shall need.

The two-party communication model was introduced by Andrew Chi-Chih Yao [39] in 1979. In this model, two parties, traditionally called Alice and Bob, are trying to collaboratively compute a known Boolean function $F : X \times Y \to \{0,1\}$. Each party is computationally unbounded; however, Alice is only given input $x \in X$ and Bob is only given $y \in Y$. In order to compute $F(x,y)$, Alice and Bob communicate in accordance with an agreed-upon *communication protocol* $\mathcal{P}$. Protocol $\mathcal{P}$ specifies as a function of transmitted bits only whether the communication is over and, if not, who sends the next bit. Moreover, $\mathcal{P}$ specifies as a function of the transmitted bits and x the value of the next bit to be sent by Alice. Similarly for Bob. The communication is over as soon as one of the parties knows the value of $F(x,y)$. The cost of the protocol $\mathcal{P}$ is the number of bits exchanged on the worst input.

Definition 4.1. The *deterministic communication complexity* of F, denoted by $DC(F)$, is the cost of an optimal communication protocol computing F.

A Boolean function $F : X \times Y \to \{0,1\}$ can be described by the *communication matrix* of F defined as the $|X| \times |Y|$ matrix M with entries $M_{x,y} = F(x,y)$. We write $\operatorname{rank}(F)$ to denote the rank of the communication matrix of F over $\mathbb{R}$. A classical result due to Kurt Mehlhorn and Erik Schmidt [20] says that for all F

$$\log \operatorname{rank}(F) \leq DC(F)\,. \tag{4.1}$$

It is easy to see that for every Boolean function F we have $DC(F) \leq \operatorname{rank}(F)$. The gap between the two bounds is exponential, and reducing this gap or showing that the gap can be achieved is a major open problem. László Lovász and Michael Saks [19] propose the following question, a positive answer to which is known as the **Log-rank Conjecture**.

Question 4.2 (Lovász and Saks)**.** Is $DC(F) \leq \operatorname{poly}(\log \operatorname{rank}(F))$ for every Boolean function $F : X \times Y \to \{0,1\}$?

The largest known gap between the two measures is due to Nisan, Avi Wigderson, and Eyal Kushilevitz [25]. They exhibited a function F, for which we have $DC(F) = \Omega(n)$ and $\log \operatorname{rank}(F) = O(n^{\log_6 3})$. We explain the details in Example 6.3.2.

There are several ways in which the deterministic communication model can be extended to include randomization. In the *public-coin* model, Alice and Bob have access to a shared random string r chosen according to some probability distribution. The only difference in the definition of a protocol is that now the protocol $\mathcal{P}$ specifies the next bit to be sent by Alice as a function of x, the already transmitted bits, and a random string r. Similarly for Bob. In the *private-coin* model, Alice has access to a random string r_A hidden from Bob, and Bob has access to a random string r_B hidden from Alice.

Definition 4.3. The *bounded-error randomized communication complexity of F with public coins (private coins)*, denoted by $RC_2(F)$ ($RC_2^{\text{pri}}(F)$), is the minimum cost of a public-coin (private-coin) randomized protocol that computes F correctly with probability at least $2/3$ on every input. (The subscript 2 refers to permitting 2-sided error.)

Clearly, for every Boolean F we have $RC_2(F) \leq RC_2^{\text{pri}}(F)$. Ilan Newman [22] showed that the two measures are identical up to constant multiplicative factors and logarithmic additive terms.

Theorem 4.4 (Newman)**.** *For every Boolean function* $F : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ *we have*

$$RC_2^{\text{pri}}(F) = O(RC_2 + \log n)\,.$$

Ramamohan Paturi and Janos Simon [27] introduced the notion of *sign-rank* to give a characterization of *unbounded-error probabilistic communication complexity*, which considers the minimum cost *private-coin* protocol with success probability strictly greater than $1/2$. It is important that the notion of unbounded-error probabilistic communication complexity is defined with respect to private-coin protocols, because every Boolean function admits a 2-bit unbounded-error public-coin protocol.

Definition 4.5. The *sign-rank* of Boolean function $F : X \times Y \to \{0,1\}$ of two arguments, denoted by $\mathrm{rank}_{\pm}$, is defined as

$$\mathrm{rank}_{\pm}(F) = \min_{L} \left\{ \mathrm{rank}\,(L) \;\middle|\; (\forall x,y) \;\; \left((-1)^{F(x,y)} L_{x,y} > 0\right) \right\}$$

where the minimum ranges over all $|X| \times |Y|$ matrices L with real entries.

For a Boolean function $F : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, let M_F denote the matrix

$$M_F = \left((-1)^{F(x,y)}\right)_{x,y \in \{0,1\}^n}.$$

The following remarkable lower bound on the sign-rank was proved by Jürgen Forster [11].

Theorem 4.6 (Forster)**.** *For Boolean function* $F : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$*, we have*

$$\mathrm{rank}_{\pm}(F) \geq \frac{2^n}{\|M_F\|},$$

where $\|M_F\|$ *denotes the spectral norm of* M_F*.*

A $k \times k$ matrix H is called an *Hadamard matrix* if its entries are ± 1 and rows are orthogonal. It is easy to see that $\|H\| = \sqrt{k}$. Hence, the following corollary.

Corollary 4.7 (Forster)**.** *For every Boolean function* $F : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$*, if* M_F *is an Hadamard matrix then* $\log \mathrm{rank}_{\pm}(F) \geq n/2$*.*

In Conjecture 5.7 we propose to relate sign-rank to sensitivity. For more information on communication complexity see [18].

Following the pioneering work by Alexander Sherstov [32], in the next section we shall study a Boolean function f by studying the communication problems $F(x,y) = f(x \circ y)$ where $\circ$ is bitwise $\wedge$, $\vee$ or $\oplus$. Notice that for every Boolean function f we have

$$DC(f(x \vee y)) \leq 2D(f) \quad \text{and} \quad DC(f(x \wedge y)) \leq 2D(f). \tag{4.2}$$

The inequalities follow from the observation that Alice and Bob can solve communication problems $f(x \vee y)$ and $f(x \wedge y)$ by simulating any decision tree for f with just two bits of communication per queried variable.

5 Sensitivity vs. other complexity measures

Unlike the complexity measures mentioned in Section 2, some of the complexity measures in this section are not polynomially related to block sensitivity and yet proving an inequality between these measures and $\mathrm{poly}(s(f))$ turns out to be equivalent to proving a polynomial relation between block sensitivity and sensitivity, i. e., the Sensitivity Conjecture itself.

5.1 Log-rank vs. sensitivity

In this section we present some implications of the following recent result by Sherstov, which appears as Theorem 6.4 in [32].

Theorem 5.1 (Sherstov)**.** *For every Boolean function f,*

$$\max\{\log\mathrm{rank}(f(x\wedge y)), \log\mathrm{rank}(f(x\vee y))\} = \Omega(\deg(f)).$$

Conjecture 5.2. ☺ *For every Boolean function f,*

$$\log\mathrm{rank}[f(x\wedge y)] \le \mathrm{poly}(s(f)).$$

Proposition 5.3. *Conjecture 5.2 is equivalent to the Sensitivity Conjecture.*

Remark 5.4. We note that this equivalence should be somewhat surprising, because $\log\mathrm{rank}(f(x\wedge y))$ and $s(f)$ are *not* polynomially related. Indeed, for the AND function $f:\{0,1\}^n\to\{0,1\}$ defined by $f(x)=\bigwedge_{i=1}^n x_i$, there is a simple protocol to compute $f(x\wedge y)$ with just 1 bit of communication. Thus, by Equation (4.1) we have $\log\mathrm{rank}(f(x\wedge y))\le DC(f(x\wedge y))\le 1$, and yet f has sensitivity n.

Proof of Proposition 5.3.

(a) Assume the Sensitivity Conjecture. Beals *et al.* [3] showed that $D(f)\le C(f)bs(f)$ and Nisan [23] proved that $C(f)\le bs(f)^2$. Therefore, we get $D(f)\le bs(f)^3$ as a simple corollary. Since $DC(f(x\wedge y))\le 2D(f)$ (4.2) and for all F we have $\log\mathrm{rank}(F(x,y))\le DC(F)$ (4.1), Conjecture 5.2 follows.

(b) Assume Conjecture 5.2. For a Boolean function f, define $g(x)=f(\bar{x})$. Clearly, $s(g)=s(f)$ and

$$|\log\mathrm{rank}(g(x\wedge y)) - \log\mathrm{rank}(f(x\vee y))| \le 1.$$

Applying the hypothesis of Conjecture 5.2 to both g and f, we get that

$$\max\{\log\mathrm{rank}(f(x\vee y)), \log\mathrm{rank}(f(x\wedge y))\} \le \mathrm{poly}(s(f)).$$

It follows that $\deg(f)\le\mathrm{poly}(s(f))$ by Theorem 5.1. This implies the Sensitivity Conjecture, since $bs(f)\le 2\deg(f)^2$ [24].

□

Conjecture 5.5. ☺ *For every Boolean function f,*

$$\log\mathrm{rank}[f(x\oplus y)] \le \mathrm{poly}(s(f)).$$

Arbitrarily large separations between $\log\operatorname{rank}(f(x \oplus y))$ and $s(f)$ are known. For instance, let $f : \{0,1\}^n \to \{0,1\}$ be the parity of n bits. Analogous to the argument in Remark 5.4, we have $\log\operatorname{rank}(f(x \oplus y)) \le 1$. For the parity function we also have $s(f) = n$. Nonetheless, we have the following equivalence.

Corollary 5.6. *Conjecture 5.5 is equivalent to the Sensitivity Conjecture.*

Proof.

(a) Showing that the Sensitivity Conjecture implies Conjecture 5.5 is similar to part (a) in Proposition 5.3.

(b) Assume Conjecture 5.5. For a Boolean function $f : \{0,1\}^n \to \{0,1\}$, define $F : \{0,1\}^{2n} \to \{0,1\}$ by $F(x,y) = f(x \wedge y)$, where $x, y \in \{0,1\}^n$. Applying the hypothesis of Conjecture 5.5 to F we obtain

$$\log\operatorname{rank}(F((x,y) \oplus (x',y'))) \le \operatorname{poly}(s(F)) .$$

We can rewrite $F((x,y) \oplus (x',y'))$ as $F(x \oplus x', y \oplus y') = f((x \oplus x') \wedge (y \oplus y'))$. Hence the communication matrix of $F((x,y) \oplus (x',y'))$ contains the communication matrix of $f(x \wedge y)$ as a submatrix. In particular,

$$\operatorname{rank}(f(x \wedge y)) \le \operatorname{rank}(F((x,y) \oplus (x',y'))) .$$

The above two inequalities together with the easy observation that $s(f) \le s(F) \le 2s(f)$ imply that $\log\operatorname{rank}(f(x \wedge y)) \le \operatorname{poly}(s(f))$. The result now follows from Proposition 5.3. $\square$

Since $\operatorname{rank}_\pm(F) \le \operatorname{rank}(F)$ for every F, we propose a possibly weaker version of Conjecture 5.5 stated for the sign-rank (see Definition 4.5).

Conjecture 5.7. ☺ *For every Boolean function f,*

$$\log\operatorname{rank}_\pm(f(x \oplus y)) \le \operatorname{poly}(s(f)) .$$

We note that the same function f, namely, the parity, that supplied an arbitrary large separation between $\log\operatorname{rank}(f(x \oplus y))$ and $s(f)$ also provides an arbitrary separation between $\log\operatorname{rank}_\pm(f(x \oplus y))$ and $s(f)$. We ask the following question inspired by Proposition 5.3 and Corollary 5.6.

Question 5.8. Does Conjecture 5.7 imply Conjecture 5.5? I. e., is Conjecture 5.7 equivalent to the Sensitivity Conjecture?

5.2 Parity decision trees

Parity decision trees are similar to decision trees; the difference is that instead of querying only one variable at a time, one may query the sum modulo 2 of an arbitrary subset of variables. (See [40] for a brief introduction to parity decision trees.) The parity decision-tree complexity of a Boolean function f is denoted by $D_\oplus(f)$. Obviously, $D_\oplus(f) \le D(f)$. Similarly to Equation (4.2) we have

$$DC(f(x \oplus y)) \le 2D_\oplus(f) . \tag{5.1}$$

In this section, we explore the relationship between $D_\oplus$ and sensitivity.

Conjecture 5.9. ☺ *For every Boolean function f,*

$$D_{\oplus}(f) \le \mathrm{poly}(s(f)).$$

Note that parity decision trees are strictly more powerful than decision trees. For instance, deciding the parity of n bits requires a decision tree of depth n whereas a parity decision tree of depth 1 suffices. For the parity function we also have $s(f) = n$. The seemingly weaker Conjecture 5.9 actually turns out to be equivalent to the Sensitivity Conjecture.

Proposition 5.10. *Conjecture 5.9 is equivalent to the Sensitivity Conjecture.*

Proof.

(a) The Sensitivity Conjecture implies Conjecture 5.9 since $D_{\oplus}(f) \le D(f)$ (5.1) and $D(f) \le bs(f)^3$.

(b) Assume Conjecture 5.9. The Sensitivity Conjecture follows from Corollary 5.6 and the fact that $\log \mathrm{rank}(f(x \oplus y)) \le DC(f(x \oplus y)) \le 2D_{\oplus}(f)$.

□

In Example 6.2.1 we exhibit a quadratic gap between $D_{\oplus}$ and sensitivity.

5.3 Fourier-analytic setting

In the previous sections, we considered Boolean functions from $\{0,1\}^n$ to $\{0,1\}$. For the purpose of studying the Fourier spectrum of Boolean functions, it is convenient to use the range $\{+1,-1\}$, replacing 0 with $+1$ and 1 with -1. This replacement preserves polynomial relations between the complexity measures studied in this section. So, in Sections 5.3 and 5.4 only, the term "Boolean function" will refer to functions $\{0,1\}^n \to \{+1,-1\}$. For a brief introduction to Fourier Analysis on the Boolean cube, we refer to the survey by de Wolf [38].

The functions $\{0,1\}^n \to \mathbb{R}$ form an inner product space $\mathcal{V}$ with the inner product defined by

$$\langle f, g \rangle = \mathop{\mathbb{E}}_{x \in \{0,1\}^n} [f(x)g(x)] \tag{5.2}$$

where the expectation is over the uniform distribution.

Definition 5.11. For $S \subseteq [n]$, the *character* $\chi_S : \{0,1\}^n \to \{-1,1\}$ is defined as

$$\chi_S(x) = (-1)^{\sum_{i \in S} x_i}.$$

The set of characters forms an orthonormal basis for $\mathcal{V}$. Hence, every function $f : \{0,1\}^n \to \mathbb{R}$ can be written uniquely as

$$f = \sum_S \langle f, \chi_S \rangle \chi_S. \tag{5.3}$$

Equation (5.3) is referred to as the *Fourier expansion* of f. The *Fourier coefficient* of f corresponding to S is defined as

$$\widehat{f}(S) = \langle f, \chi_S \rangle.$$

Equation (5.3) immediately implies *Parseval's Identity*,

$$\langle f, f\rangle = \sum_S \widehat{f}(S)^2 . \tag{5.4}$$

Note that for Boolean functions f we have $\langle f, f\rangle = 1$ by (5.2).

Recall that every Boolean function f admits a unique multilinear polynomial representation (see Definition 2.4). For $S \subseteq [n]$ define

$$\lambda_S(x) = \prod_{i\in S} x_i .$$

Then f can be written uniquely as

$$f = \sum_S c_S \lambda_S , \tag{5.5}$$

for some $c_S \in \mathbb{R}$. Since $f(e_S) = \sum_{B\subseteq S} c_B$, it is easy to see by induction on $|S|$ that, in fact, we have $c_S \in \mathbb{Z}$. The following proposition establishes connections between the coefficients of the multilinear representation and the Fourier coefficients.

Proposition 5.12. *Let f be a Boolean function and $f = \sum_S c_S \lambda_S$ be its multilinear expansion where $c_S \in \mathbb{Z}$. We have*

1. $c_S = (-2)^{|S|} \sum_{B:S\subseteq B\subseteq [n]} \widehat{f}(B)$ *and*
2. $\widehat{f}(S) = \sum_{B\subseteq [n], B'\subseteq S} c_B (-1)^{|B'|} 2^{-|B\setminus B'|}$.

Proof.

1. We can express

$$\chi_B(x) = (-1)^{\sum_{i\in B} x_i} = \prod_{i\in B}(1-2x_i) = \sum_{S\subseteq B} (-2)^{|S|}\lambda_S(x) .$$

Thus

$$f = \sum_{B\subseteq [n]} \widehat{f}(B)\chi_B = \sum_{B\subseteq [n]} \widehat{f}(B) \sum_{S\subseteq B} (-2)^{|S|}\lambda_S = \sum_{S\subseteq [n]} (-2)^{|S|} \sum_{B:S\subseteq B\subseteq [n]} \widehat{f}(B)\lambda_S .$$

The claim follows.

2. Observe that $\langle \lambda_S, \lambda_T\rangle = 2^{-|S\cup T|}$. We have

$$\begin{aligned}\widehat{f}(S) = \langle f, \chi_S\rangle &= \Big\langle \sum_{B\subseteq [n]} c_B\lambda_B, \sum_{B'\subseteq S} (-2)^{|B'|}\lambda_{B'} \Big\rangle \\ &= \sum_{B\subseteq [n], B'\subseteq S} c_B \langle \lambda_B, \lambda_B' \rangle = \sum_{B\subseteq [n], B'\subseteq S} c_B (-1)^{|B'|} 2^{|B'|-|B\cup B'|} .\end{aligned}$$

The claim follows.

□

Corollary 5.13. *Every Fourier coefficient of a Boolean function f is an integer multiple of $2^{-\deg(f)}$.*

Proof. The claim follows from part 2 of Proposition 5.12 and the fact that $|B| \leq \deg(f)$. □

Conjecture 5.14. 😐 *For every Boolean function f,*

$$|\{S \mid \hat{f}(S) \neq 0\}| \leq 2^{\mathrm{poly}(s(f))}.$$

The proof that the above conjecture is equivalent to the Sensitivity Conjecture relies on the following characterization of the number of nonzero Fourier coefficients.

Lemma 5.15. *For every Boolean function f,*

$$|\{S \mid \hat{f}(S) \neq 0\}| = \mathrm{rank}(f(x \oplus y)).$$

Proof. Consider the matrix M with entries $M_{x,y} = f(x \oplus y)$. It is easy to check that for each $S \subseteq [n]$, the vector $(\chi_S(y))_{y \in \{0,1\}^n}$ is an eigenvector of M with a corresponding eigenvalue $2^n \widehat{f}(S)$. Since the χ_S form a basis of $\mathcal{V}$, the values $2^n \widehat{f}(S)$ are all the eigenvalues of M. Noting that the rank of a symmetric matrix is the number of its nonzero eigenvalues, we obtain $|\{S \mid \hat{f}(S) \neq 0\}| = \mathrm{rank}(M)$. □

The equivalence of Conjecture 5.14 and the Sensitivity Conjecture is immediate from this lemma and Corollary 5.6. □

Conjecture 5.16. 😐 *For every Boolean function f,*

$$\min_{S : \widehat{f}(S) \neq 0} |\widehat{f}(S)| \geq 2^{-\mathrm{poly}(s(f))}.$$

We show that Conjecture 5.16 is also equivalent to the Sensitivity Conjecture.

Proposition 5.17. *Conjecture 5.16 is equivalent to the Sensitivity Conjecture.*

Proof.

(a) Assume the Sensitivity Conjecture. For every Boolean function f we have

$$\min_{S : \widehat{f}(S) \neq 0} |\widehat{f}(S)| \geq 2^{-\deg(f)} \geq 2^{-\mathrm{poly}(s(f))},$$

where the first inequality follows from Corollary 5.13 and the second inequality follows from the Sensitivity Conjecture. This shows Conjecture 5.16 holds.

(b) Assume Conjecture 5.16. Let $\alpha = \min_{S : \widehat{f}(S) \neq 0} |\widehat{f}(S)|$. By Parseval's Identity (5.4), the number of non-zero Fourier coefficients is at most α^{-2}. By Lemma 5.15, α^{-2} is also an upper bound on $\mathrm{rank}(f(x \oplus y))$. Hence, $\alpha \geq 2^{-\mathrm{poly}(s(f))}$ implies $\mathrm{rank}(f(x \oplus y) \leq 2^{\mathrm{poly}(s(f))}$. By Corollary 5.6, this implies the Sensitivity Conjecture. □

The following consequence of the Sensitivity Conjecture appears to be open.

Conjecture 5.18. ☺ 😕 *For every Boolean function f,*

$$\sum_S |\widehat{f}(S)| \leq 2^{\mathrm{poly}(s(f))}.$$

Proposition 5.19. *Conjecture 5.14 implies Conjecture 5.18. In particular, the Sensitivity Conjecture implies Conjecture 5.18.*

Proof. Observe that $\widehat{f}(S) \in [-1,1]$ for all S, so $\sum_S |\widehat{f}(S)| \leq |\{S \mid \widehat{f}(S) \neq 0\}|$. □

We shall see that Conjecture 5.18 is equivalent to the Sensitivity Conjecture, assuming the following conjecture due to Vince Grolmusz [14] holds.

Conjecture 5.20 (Grolmusz)**.** *Let $F : \{0,1\}^m \times \{0,1\}^n \to \{-1,1\}$. Then*

$$RC_2(F) \leq \mathrm{poly}\big(\log \sum_{S \subseteq [m+n]} |\widehat{F}(S)|\big)\,.$$

To prove the equivalence claimed above, we need the following result by Sherstov [32, Theorem 5.1].

Theorem 5.21 (Sherstov)**.** *Let $F_1(x,y) := f(x \wedge y)$ and $F_2(x,y) := f(x \vee y)$, then*

$$\max\{RC_2(F_1), RC_2(F_2)\} = \Omega(bs(f)^{1/4})\,.$$

Proposition 5.22. *If Grolmusz's conjecture holds then Conjecture 5.18 is equivalent to the Sensitivity Conjecture.*

Proof. By Proposition 5.19, it suffices to show that Conjecture 5.18 implies the Sensitivity Conjecture. Consider two Boolean functions F_1 and F_2 on $2n$ variables defined as in Theorem 5.21. It is easy to check that $s(f) \leq s(F_1)$ and $s(F_2) \leq 2s(f)$. Assuming Conjecture 5.18 we get that

$$\log \sum |\widehat{F_1}(S)| \leq \mathrm{poly}(s(f)) \quad \text{and} \quad \log \sum |\widehat{F_2}(S)| \leq \mathrm{poly}(s(f))\,.$$

Now $bs(f) \leq \mathrm{poly}(s(f))$ follows from Theorem 5.21 assuming Conjecture 5.20. □

5.4 Fourier entropy

In this section, we outline some connections between measures introduced in Section 5.3 and Information Theory.

Let P be a probability distribution on a sample space Ω. Let $\mathrm{supp}(P)$ denote the support of P.

Definition 5.23. The *entropy* of P is defined as

$$H(P) = \sum_{\omega \in \Omega} P(\omega) \log \frac{1}{P(\omega)}\,,$$

the *max-entropy* of P is

$$H^{\max}(P) = \max_{\omega \in \mathrm{supp}(P)} \log \frac{1}{P(\omega)}\,,$$

and the *min-entropy* of P is

$$H^{\min}(P) = \min_{\omega \in \Omega} \log \frac{1}{P(\omega)}\,.$$

For every distribution P the following inequalities hold

$$H^{\min}(P) \leq H(P) \leq H^{\max}(P).$$

By Parseval's Identity (5.4), we can view $\{\widehat{f}(S)^2 \mid S \subseteq [n]\}$ as a probability distribution on $\Omega = 2^{[n]}$, the power set of $[n]$. We denote this probability distribution by $\widehat{f}^2$.

The following Conjecture is equivalent to the Sensitivity Conjecture. In fact, it is a restatement of Conjecture 5.16 in the terms of Definition 5.23 and the probability distribution $\widehat{f}^2$.

Conjecture 5.24. ☺ *For every Boolean f we have*

$$H^{\max}(\widehat{f}^2) \leq \text{poly}(s(f)).$$

Arbitrarily large separations between $H^{\max}(\widehat{f}^2)$ and $s(f)$ are known. For example, consider the parity function $f : \{0,1\}^n \to \{-1,1\}$, which can be expressed as $f(x) = (-1)^{\sum_{i=1}^n x_i}$. The whole Fourier spectrum is concentrated on a single coefficient, namely, $\widehat{f}([n])$. Hence $H^{\max}(\widehat{f}^2) = 0$, yet $s(f) = n$.

Definition 5.25. The *average sensitivity* (also called *influence* or *total influence*) of a Boolean function $f : \{0,1\}^n \to \{-1,1\}$ is defined as

$$as(f) = \mathbb{E}_{x \in \{0,1\}^n}(s(f,x)),$$

where the expectation is over the uniform distribution.

Compare Conjecture 5.24 with the **Fourier Entropy-Influence Conjecture** due to Ehud Friedgut and Gil Kalai [12].

Conjecture 5.26 (Friedgut and Kalai)**.** *There exists a constant C such that for every Boolean function f we have*

$$H(\widehat{f}^2) \leq C \cdot as(f).$$

While Conjecture 5.26 seems only analogous to Conjecture 5.24, the authors are unaware of any formal connections between the two conjectures. Recently, Ryan O'Donnell, John Wright, and Yuan Zhou [26] verified the Fourier Entropy-Influence Conjecture for certain classes of Boolean functions, including symmetric functions.

5.5 Shi's characterization of sensitivity

In this section we present some applications of Yaoyun Shi's work [33], which contains an interesting characterization of the sensitivity of Boolean functions.

Let f be a Boolean function $\{0,1\}^n \to \{0,1\}$. With some abuse of notation, we use f to denote the unique multilinear extension $f : \mathbb{R}^n \to \mathbb{R}$ of this function (cf. Definition 2.4).

Definition 5.27. For $a, b \in [0,1]^n$, the *linear restriction* of f on the line segment $\ell = [a,b]$, denoted by $f_\ell : [0,1] \to \mathbb{R}$, is defined as

$$f_\ell(t) := f((1-t)a + tb) \quad (t \in [0,1]).$$

Denote the supremum norm of a function $g : [0,1] \to \mathbb{R}$ by $\|g\|_\infty = \sup_{t\in[0,1]} |g(t)|$. Let g' denote the derivative of g. Shi [33] gave the following characterization of sensitivity.

Theorem 5.28 (Shi)**.** *For every Boolean function f, $s(f) = \sup_\ell \|f'_\ell\|_\infty$.*

Proof. It is easy to check that it suffices to consider the lines that join two points of the Boolean cube.

For $x \in [0,1]^n$, let $x^{(i,1)}$ ($x^{(i,0)}$) denote a vector whose i^{th} coordinate is 1 (0) and the other coordinates match with those of x. Let $a,b \in \{0,1\}^n$ and $\ell = (a,b)$ be the line joining a and b.

$$f'_\ell(t) = \sum_{i=1}^{n} (b_i - a_i) \cdot \frac{\partial f}{\partial x_i}((1-t)a + tb)\,.$$

Since f is multilinear, we have:

$$\frac{\partial f}{\partial x_i}(x) = f(x^{(i,1)}) - f(x^{(i,0)})\,.$$

Thus we have:

$$|f'_\ell(t)| \le \mathop{\mathbb{E}}_{p\in D_t}\left[\sum_{i=1}^{n}\left|f(p^{(i,1)}) - f(p^{(i,0)})\right|\right], \tag{5.6}$$

where D_t denotes the following probability distribution on the Boolean cube: for each k, $\Pr(p_k = 1) = (1-t)a_k + tb_k$. Notice that the right hand side of (5.6) is at most $s(f)$.

For the other direction let $a \in \{0,1\}^n$ and b be obtained from a by flipping each bit. It is easy to check that:

$$f'_\ell(0) = \sum |f(a\oplus e_i) - f(a)| = s(f,a)\,.$$

Choosing a vector a with maximum sensitivity completes the proof. □

Combining Theorem 5.28 with Conjecture 5.16 puts the Sensitivity Conjecture into an analytic setting.

Definition 5.29. The *approximate degree of linear restrictions* of a Boolean function f is defined as follows:

$$\overline{\deg}(f) = \max_\ell \min\left\{\deg(g) \mid g \in \mathbb{R}[t], \|f_\ell - g\|_\infty \le 1/3\right\}.$$

The connection between $\overline{\deg}(f)$ and $s(f)$ was established by Shi [33].

Theorem 5.30 (Shi)**.** *The complexity measures $\overline{\deg}(f)$ and $s(f)$ are polynomially related.*

Observe that, unlike all previous equivalence results, Theorem 5.30 gives a complexity measure polynomially related to $s(f)$ rather than $bs(f)$. Theorem 5.30 implies that the following conjecture due to Shi [33] is equivalent to the Sensitivity Conjecture.

Conjecture 5.31 (Shi)**.** 😐 *For every Boolean function f,*

$$\widetilde{\deg}(f) \le \mathrm{poly}\left(\overline{\deg}(f)\right).$$

5.6 Subgraphs of the n-cube

Recall that Q_n denotes the n-cube graph (see Section 3). Denote the maximum degree of graph G by $\Delta(G)$. For an induced subgraph G of a graph H let $H-G$ denote the subgraph of H induced on the vertex set $V(H)\setminus V(G)$. Let $\Gamma(G)=\max\{\Delta(G),\Delta(H-G)\}$. Craig Gotsman and Nati Linial [13] proved the following remarkable equivalence.

Theorem 5.32 (Gotsman and Linial)**.** *The following are equivalent for any monotone function $h:\mathbb{N}\to\mathbb{R}$:*

(a) *For any induced subgraph G of Q_n with $|V(G)|\neq 2^{n-1}$ we have $\Gamma(G)\geq h(n)$.*

(b) *For any Boolean function f we have $s(f)\geq h(\deg(f))$.*

Proof. Statement **(b)** is equivalent to the following:

(b') For any Boolean function f with $\deg(f)=n$ we have $s(f)\geq h(n)$.

Clearly, **(b)** implies **(b')**. To prove the reverse implication, let f be a Boolean function of degree d. Fix a monomial of degree d of the representing polynomial of f. Without loss of generality we may assume the monomial is $x_1\cdots x_d$. Define $g(x_1,\ldots,x_d):=f(x_1,\ldots,x_d,0,\ldots,0)$. Then, $s(f)\geq s(g)\geq h(d)$, as desired.

(a) $\Rightarrow$ **(b')** We prove the contrapositive. Given a Boolean function f with $s(f)<h(n)$, consider an induced subgraph G of Q_n with

$$V(G)=\{x\in\{0,1\}^n \mid f(x)p(x)=+1\},$$

where $p(x)=(-1)^{\sum x_i}$ is the parity function (as in Section 5.3, we take the range of Boolean functions to be $\{+1,-1\}$). Observe that $\widehat{f}(I)=\widehat{fp}([n]-I)$ for any subset $I\subseteq[n]$. Hence, $\widehat{fp}(\emptyset)=\widehat{f}([n])\neq 0$, since $\deg(f)=n$. Straight from the definition of Fourier coefficients, $\mathbb{E}_x[f(x)p(x)]=\widehat{fp}(\emptyset)\neq 0$, so $|V(G)|\neq 2^{n-1}$. Furthermore, $s(fp,x)=n-s(f,x)$ and $\deg_G(x)=n-s(fp,x)=s(f,x)$. By a similar argument, $\deg_{Q_n-G}(x)=s(f,x)$ for all x in $V(Q_n)\setminus V(G)$. Thus, $\Gamma(G)\leq s(f)<h(n)$.

(a) $\Leftarrow$ **(b')** Observe that the steps in the proof of **(a)** $\Rightarrow$ **(b')** are reversible. □

The proof of Theorem 5.32 translates a Boolean function with a polynomial gap between degree and sensitivity into a graph with the same polynomial gap between Γ and n, and vice versa. For example, observe that Rubinstein's function (see Example 6.1.1) has sensitivity $\sqrt{n}$ and full degree, which can be easily verified by a direct computation of $\widehat{f}([n])$. Therefore, Rubinstein's function can be used to obtain a graph G with the surprising property $\Gamma(G)=\Theta(\sqrt{n})$. Fan Chung, Zoltán Füredi, Ronald Graham, and Paul Seymour [9] independently constructed a graph G with $\Gamma(G)<\sqrt{n}+1$. Their example can be also obtained from Theorem 5.32 by applying the reduction in the proof of **(a)** $\Rightarrow$ **(b')** to the AND-of-ORs function (see Example 6.2.1), but note that the Gotsman-Linial theorem was not available at the time when Chung *et al.* gave their construction.

Theorem 5.32 immediately implies that the following conjecture is equivalent to the Sensitivity Conjecture.

Conjecture 5.33. ☺ *There is a constant $c>0$ such that for every induced subgraph G of Q_n with $|V(G)|\neq 2^{n-1}$ we have $\Gamma(G)\geq n^c$.*

5.7 Two-colorings of integer lattices

We call lattice points $a, b \in \mathbb{Z}^d$ "neighbors" if $\|a-b\|_1 = 1$. We say that a two-coloring C of $\mathbb{Z}^d$ with colors red and blue is non-trivial if the origin is colored red, and there is a point colored blue on each of the coordinate axes. The sensitivity of a point $a \in \mathbb{Z}^d$ under coloring C, denoted by $S(a,C)$, is the number of neighbors of a that are colored differently from a.

Definition 5.34. The *sensitivity of a coloring* is defined by $S(C) = \max_a S(a,C)$.

Scott Aaronson [2] stated the following question, a positive answer to which would imply the Sensitivity Conjecture.

Question 5.35 (Aaronson)**.** ☹ Does every non-trivial coloring of $\mathbb{Z}^d$ have sensitivity at least $d^{\Omega(1)}$?

Proposition 5.36 (Aaronson)**.** *A positive answer to Question 5.35 implies the Sensitivity Conjecture.*

We present Andrew Drucker's version of Aaronson's proof.

Proof. Given a Boolean function f on n variables, let x be an input on which f achieves the highest block sensitivity $b = bs(f)$. Let $S_1, \ldots, S_b$ be pairwise disjoint sensitive blocks of f on x, and let $R = [n] - (\bigcup_i S_i)$. Take a closed walk w on the Boolean cube $Q_{|S_i|}$ passing through $x|_{S_i}$ and $\bar{x}|_{S_i}$. Let $\gamma_i : \mathbb{Z} \to \{0,1\}^{|S_i|}$ denote a periodic function with period $|w|$ such that $\gamma_i|_{\{0,\ldots,|w|-1\}}$ is the walk w and $\gamma_i(0) = x|_{S_i}$. Consider the following mapping $\phi : \mathbb{Z}^b \to \{0,1\}^n$: a point $a \in \mathbb{Z}^b$ is mapped to the Boolean string $y \in \{0,1\}^n$ with $y|_{S_i} = \gamma_i(a_i)$ and $y|_R = x|_R$. Finally, obtain coloring C of $\mathbb{Z}^b$ by composing f with ϕ. By construction, we have $bs(f) = b$. Since C is non-trivial, we can apply the hypothesis of Question 5.35 to obtain $b \le \mathrm{poly}(s(C))$. It is easy to see that $s(C) \le 2s(f)$. Putting it all together, we have $bs(f) = b \le \mathrm{poly}(s(C)) \le \mathrm{poly}(s(f))$. □

6 Some Boolean functions

In this section we present some interesting examples of Boolean functions. They provide lower or upper bounds for various complexity measures, and some of them appear in more than one context.

6.1 Rubinstein's function

The following function was constructed by David Rubinstein [31]. It was discussed in Section 2, Section 3 and Section 5.6 of this paper.

Example 6.1.1 (Rubinstein's function)**.** Assume we have $n = k^2$ variables (k even), which are divided into k blocks with k variables each. The value of the function is 1 if there is at least one block with exactly two consecutive 1s in it, and it is 0 otherwise.

The block sensitivity of Rubinstein's function is $n/2$ (hence, the certificate complexity and the decision-tree complexity both are at least $n/2$) and the sensitivity is $\sqrt{n}$.

Proposition 6.1.2. *Rubinstein's function has full degree.*

Proof. It suffices to show that $\widehat{f}([n]) \neq 0$. We will actually compute $\widehat{f}([n])$ and find that

$$\widehat{f}([n]) = (-1)^k (k-1)^k / 2^{k^2}.$$

By definition, we have

$$\widehat{f}([n]) = \mathbb{E}_{x \in \{0,1\}^n} \left((-1)^{\sum_{i=1}^n x_i + f(x)} \right) = \frac{1}{2^{k^2}} \sum_x (-1)^{\sum_{i=1}^n x_i + f(x)}.$$

We shall prove the following statement by induction on $k - j$.

$$\sum_{x \,:\, x_1 = \cdots = x_{kj} = 0} (-1)^{\sum_{i=kj+1}^n x_i + f(x)} = (-1)^{k-j} (k-1)^{k-j}.$$

The equation holds for $j = k$. For the inductive step, define

$\mathcal{A} = \{x \mid x_1 = \cdots = x_{kj} = 0$ and there exists exactly one pair of consecutive 1s among $x_{kj+1}, \ldots, x_{k(j+1)}\}$

and

$$\mathcal{B} = \{x \mid x_1 = \cdots = x_{kj} = 0\} \setminus \mathcal{A}.$$

Then we have

$$\sum_{x \,:\, x_1 = \cdots = x_{kj} = 0} (-1)^{\sum_{i=kj+1}^n x_i + f(x)} = \sum_{x \in \mathcal{A}} (-1)^{\sum_{i=kj+1}^n x_i + f(x)} + \sum_{x \in \mathcal{B}} (-1)^{\sum_{i=kj+1}^n x_i + f(x)}.$$

Observe that $f(x) = 1$ for every $x \in \mathcal{A}$, so by symmetry the first term in the equation is 0. For $x \in \mathcal{B}$ the value of $f(x)$ does not depend on $x_{kj+1}, \ldots, x_{k(j+1)}$. Thus we obtain

$$\begin{aligned}
\sum_{x \in \mathcal{B}} (-1)^{\sum_{i=kj+1}^n x_i + f(x)} &= \left(\sum_{j=0}^{k} (-1)^j \binom{k}{j} - (k-1) \right) \sum_{x \,:\, x_1 = \cdots = x_{k(j+1)} = 0} (-1)^{\sum_{i=k(j+1)+1}^n x_i + f(x)} \\
&= -(k-1) \sum_{x \,:\, x_1 = \cdots = x_{k(j+1)} = 0} (-1)^{\sum_{i=k(j+1)+1}^n x_i + f(x)} \\
&= (-1)^{k-j} (k-1)^{k-j},
\end{aligned}$$

where the last step follows by induction.

For the case $j = 0$ we obtain $\sum_x (-1)^{\sum_{i=1}^n x_i + f(x)} = (-1)^k (k-1)^k$, completing proof of the claim. □

6.2 AND-of-ORs

The following folklore example was discussed in Section 2 and Section 5.6 of this paper.

Example 6.2.1 (AND-of-ORs function)**.** AND-of-ORs function is defined on k blocks of k variables each:

$$f(x_{11}, \ldots, x_{kk}) = \bigwedge_{i=1}^{k} \bigvee_{j=1}^{k} x_{ij}.$$

The block sensitivity and sensitivity of AND-of-ORs function on $n = k^2$ variables is k. AND-of-ORs has full degree and hence its decision-tree complexity is also n. The certificate complexity of AND-of-ORs function is k. To see that $D_{\oplus}(f) = n$, consider the mod 2 degree of f defined as follows.

Definition 6.2.2. The *mod* 2 *degree* of a Boolean function f, denoted by $\deg_{\oplus}(f)$, is the degree of the unique multilinear polynomial over $\mathbb{F}_2$ (the field of two elements) that represents f.

Observe that the OR function (and consequently the AND function) has full mod 2 degree. It follows that AND-of-ORs has full mod 2 degree, which shows that $D_{\oplus}(f) = n$, since for any Boolean function f, $\deg_{\oplus}(f) \le D_{\oplus}(f)$.

6.3 Kushilevitz's function

We begin with a definition of a *composition function*.

Definition 6.3.1. For a Boolean function $f : \{0,1\}^m \to \{0,1\}$ and a Boolean function $g : \{0,1\}^n \to \{0,1\}$, we define the *composition function* $f \diamond g$ on mn variables as follows:

$$(f \diamond g)(x_{11}, \ldots, x_{mn}) = f\big(g(x_{11}, \ldots, x_{1n}), \ldots, g(x_{m1}, \ldots, x_{mn})\big) .$$

Eyal Kushilevitz constructed a function f that provides the largest known gap in the exponent of a polynomial in $\deg(f)$ that gives an upper bound on $bs(f)$. Never published by Kushilevitz, the function appears in footnote 1 of the Nisan-Wigderson paper [25]. It was discussed in Section 2 and Section 4 of this paper.

Example 6.3.2 (Kushilevitz's function)**.** Define an auxiliary function h on 6 variables:

$$\begin{aligned} h(z_1, \ldots, z_6) \quad = \quad & \textstyle\sum_i z_i - \sum_{ij} z_i z_j + z_1z_3z_4 + z_1z_2z_5 + z_1z_4z_5 + z_2z_3z_4 + \\ & z_2z_3z_5 + z_1z_2z_6 + z_1z_3z_6 + z_2z_4z_6 + z_3z_5z_6 + z_4z_5z_6 \, . \end{aligned}$$

Kushilevitz's function is defined as $f_k = h \diamond h \diamond \cdots \diamond h$ (k times).

Observe that h is a function of 6 variables, has degree 3 and full sensitivity on the 0 input. Kushilevitz's function f_k is obtained by composing h with itself k times. It is defined on $n = 6^k$ variables and has full sensitivity, block sensitivity, decision-tree complexity, and certificate complexity, yet its degree is $3^k = n^{\log_6 3} \approx n^{0.613}$.

Kushilevitz's function gives rise to the communication problem $f_k(x \wedge y)$ with the largest known separation between the logrank and the deterministic communication complexity. In fact, this gap exists between the logrank and the bounded-error randomized communication complexity. This connection was made by Nisan and Wigderson [25].

Theorem 6.3.3 (Nisan and Wigderson)**.** *For Kushilevitz's function f_k on $n = 6^k$ variables we have*

$$DC(f_k(x \wedge y)) = \Omega(n), \quad RC_2(f_k(x \wedge y)) = \Omega(n), \quad \log \operatorname{rank}(f_k(x \wedge y)) = O(n^{\log_6 3} \log n).$$

In the proof of this result, we shall refer to the "disjointness function," denoted by UDISJ : $\{0,1\}^n \times \{0,1\}^n \to \{0,1\}$. It is a *partial function* defined by

$$\mathrm{UDISJ}(x,y) = \begin{cases} 0 & \text{if there is unique } i \text{ such that } x_i = y_i = 1, \\ 1 & \text{if for all } i \text{ we have } x_i = 0 \text{ or } y_i = 0. \end{cases}$$

A communication protocol $\mathcal{P}$ is said to compute a partial function F if the outputs of $\mathcal{P}$ and F agree on the inputs where F is defined.

The lower bound in Theorem 6.3.3 will follow from the $\Omega(n)$ lower bound on the bounded-error randomized communication complexity of UDISJ due to Bala Kalyanasundaram and Georg Schnitger [16] (see also [28] and [18, Chapter 4.6]).

Theorem 6.3.4 (Kalyanasundaram and Schnitger)**.**

$$RC_2(\mathrm{UDISJ}) = \Omega(n)\,.$$

In particular, $DC(\mathrm{UDISJ}) = \Omega(n)$.

Corollary 6.3.5 (Nisan-Wigderson)**.** *For every* $f : \{0,1\}^n \to \{0,1\}$ *such that* $f(0) = 0$ *and* $s(f,0) = n$ *we have* $DC(f(x \wedge y)) = \Omega(n)$ *and* $RC_2(f(x \wedge y)) = \Omega(n)$.

Proof. Observe that if f has properties $f(0) = 0$ and $s(f,0) = n$, then any protocol for $\neg f(x \wedge y)$ directly solves the UDISJ problem. □

The following general result will provide the upper bound on the rank in Theorem 6.3.3.

Proposition 6.3.6 (Nisan and Wigderson)**.** *Let* f *be the polynomial representation of a Boolean function* $\{0,1\}^n \to \{0,1\}$. *Let* m *denote the number of monomials in* f. *Then* $\mathrm{rank}(f(x \wedge y)) = m$. *In particular,*

$$\mathrm{rank}(f(x \wedge y)) \le \sum_{i=0}^{\deg(f)} \binom{n}{i} = 2^{O(\deg(f)\log n)}\,.$$

Proof. Let $f = \sum_S c_S \lambda_S$ be a multilinear expansion of f (5.5). Define $C = \{S \mid c_S \neq 0\}$. Note that $m = |C|$. For each $S \subseteq [n]$ define the column vector $v_S = (\lambda(x))_x$. Clearly $\{v_S\}$ is a linearly independent set of vectors. Let M be the communication matrix of $f(x \wedge y)$. We have $M(x,y) = f(x \wedge y) = \sum_S c_S \lambda_S(y) \lambda_S(x)$. Thus each column of M belongs to the span of $\{v_S \mid S \in C\}$. Consequently, $\mathrm{rank}(f(x \wedge y)) \le m$. Since $M(x, e_B) = \sum_{S \subseteq B} c_S \lambda_S(x)$, it is easy to see by induction on $|S|$ that the column space of M contains v_S for each $S \in C$. This in turn implies that $\mathrm{rank}(f(x \wedge y)) \ge m$. □

Corollary 6.3.5 shows that for Kushilveitz's function f_k on $n = 6^k$ variables we have $DC(f_k(x \wedge y)) = \Omega(n)$ and $RC_2(f_k(x \wedge y)) = \Omega(n)$, while Proposition 6.3.6 shows that $\log \mathrm{rank}(f_k(x \wedge y)) = O(n^{\log_6 3} \log n)$. This completes the proof of Theorem 6.3.3. □

6.4 Chakraborty's function

The following function was constructed by Sourav Chakraborty [7]. It was discussed in Section 3 of this paper.

Example 6.4.1 (Chakraborty's function)**.** Define an auxiliary function h on k^2 variables by a regular expression:

$$h(z_{11},\dots,z_{kk}) = 1 \iff z \in 110^{k-2}(11111(0+1)^{k-5})^{k-2}11111(0+1)^{k-8}111\,.$$

Chakraborty's function f on $n \geq k^2$ variables is defined as follows:

$$f(x_0,\dots,x_{n-1}) = 1 \iff \big(\exists i \in [n]\big)\big(g(x_i,x_{(i+1)},\dots,x_{(i+k^2)}) = 1\big)\,,$$

where indices in the arguments of function g are taken modulo n.

Chakraborty shows that for $n = k^3$ his function has sensitivity $\Theta(n^{1/3})$ [7], block sensitivity $\Theta(n^{2/3})$ and certificate complexity $\Theta(n^{2/3})$ [8].

Acknowledgments

We would like to thank Laci Babai for reviewing preliminary versions of this paper, giving helpful comments, and introducing previous generations of University of Chicago students to the Sensitivity Conjecture, including David Rubinstein, Sandy Kutin, and Sourav Chakraborty. Just as Laci seemed to be giving up on popularizing the conjecture, Sasha Razborov rekindled the flame. We would like to thank Sasha for introducing us to the Sensitivity Conjecture. We also thank the anonymous reviewers for their helpful comments that led to a considerably improved presentation of the material.

References

[1] SCOTT AARONSON: Quantum certificate complexity. In *Proc. 18th IEEE Conf. Comp. Compl. (CCC)*, pp. 171–178. IEEE Comp. Soc. Press, 2003. [doi:10.1109/CCC.2003.1214418] 4

[2] SCOTT AARONSON: The "sensitivity" of 2-colorings of the d-dimensional integer lattice. `http://mathoverflow.net/questions/31482/`, July 2010. 19

[3] ROBERT BEALS, HARRY BUHRMAN, RICHARD CLEVE, MICHELE MOSCA, AND RONALD DE WOLF: Quantum lower bounds by polynomials. *J. ACM*, 48(4):778–797, 2001. [doi:10.1145/502090.502097] 4, 10

[4] MANUEL BLUM AND RUSSELL IMPAGLIAZZO: Generic oracles and oracle classes. In *Proc. 28th FOCS*, pp. 118–126. IEEE Comp. Soc. Press, 1987. [doi:10.1109/SFCS.1987.30] 4

[5] SIEGFRIED BUBLITZ, UTE SCHÜRFELD, INGO WEGENER, AND BERND VOIGT: Properties of complexity measures for PRAMs and WRAMs. *Theoret. Comput. Sci.*, 48(1):53–73, 1986. [doi:10.1016/0304-3975(86)90083-6] 4

[6] HARRY BUHRMAN AND RONALD DE WOLF: Complexity measures and decision tree complexity: A survey. *Theoret. Comput. Sci.*, 288(1):21–43, 2002. [doi:10.1016/S0304-3975(01)00144-X] 4, 7

[7] SOURAV CHAKRABORTY: On the sensitivity of cyclically-invariant Boolean functions. In *Proc. 20th Ann. IEEE Conf. Comput. Complexity (CCC)*, pp. 163–167. IEEE Comp. Soc. Press, 2005. [doi:10.1109/CCC.2005.38] 7, 23

[8] SOURAV CHAKRABORTY: Sensitivity, block sensitivity and certificate complexity of Boolean functions. Master's thesis, University of Chicago, USA, 2005. 23

[9] FAN R. K. CHUNG, ZOLTÁN FÜREDI, RONALD L. GRAHAM, AND PAUL SEYMOUR: On induced subgraphs of the cube. *J. Combin. Theory Ser. A*, 49(1):180–187, 1988. [doi:10.1016/0097-3165(88)90034-9] 18

[10] STEPHEN COOK AND CYNTHIA DWORK: Bounds on the time for parallel RAM's to compute simple functions. In *Proc. 14th STOC*, pp. 231–233, New York, NY, USA, 1982. ACM Press. [doi:10.1145/800070.802196] 2

[11] JÜRGEN FORSTER: A linear lower bound on the unbounded error probabilistic communication complexity. *J. Comput. System Sci.*, 65:612–625, December 2002. [doi:10.1016/S0022-0000(02)00019-3] 9

[12] EHUD FRIEDGUT AND GIL KALAI: Every monotone graph property has a sharp threshold. *Proc. AMS*, 124(10):2993–3002, 1996. [doi:10.1090/S0002-9939-96-03732-X] 16

[13] CRAIG GOTSMAN AND NATHAN LINIAL: The equivalence of two problems on the cube. *J. Combin. Theory Ser. A*, 61(1):142–146, 1992. [doi:10.1016/0097-3165(92)90060-8] 4, 18

[14] VINCE GROLMUSZ: On the power of circuits with gates of low L_1 norms. *Theoret. Comput. Sci.*, 188(1–2):117–128, 1997. [doi:10.1016/S0304-3975(96)00290-3] 15

[15] JURIS HARTMANIS AND LANE A. HEMACHANDRA: One-way functions, robustness and the non-isomorphism of NP-complete sets. In *Proc. 2nd IEEE Struct. Complex. Theory Conf.*, pp. 160–174. IEEE Comp. Soc. Press, 1987. 4

[16] BALA KALYANASUNDARAM AND GEORG SCHINTGER: The probabilistic communication complexity of set intersection. *SIAM J. Discrete Math.*, 5:545–557, 1992. [doi:10.1137/0405044] 22

[17] CLAIRE KENYON AND SANDY KUTIN: Sensitivity, block sensitivity, and l-block sensitivity of Boolean functions. *Inform. and Comput.*, 189(1):43, 2004. [doi:10.1016/j.ic.2002.12.001] 6

[18] EYAL KUSHILEVITZ AND NOAM NISAN: *Communication Complexity*. Cambridge University Press, 1997. 9, 22

[19] LÁSZLÓ LOVÁSZ AND MICHAEL SAKS: Communication complexity and combinatorial lattice theory. *J. Comput. System Sci.*, 47(2):322–349, 1993. [doi:10.1016/0022-0000(93)90035-U] 8

[20] KURT MEHLHORN AND ERIK M. SCHMIDT: Las Vegas is better than determinism in VLSI and distributed computing. In *Proc. 14th STOC*, pp. 330–337, New York, NY, USA, 1982. ACM Press. [doi:10.1145/800070.802208] 8

[21] GATIS MIDRIJANIS: Exact quantum query complexity for total Boolean functions. Technical report, Cornell University ArXiv.org, 2004. [arXiv:quant-ph/0403168] 4

[22] ILAN NEWMAN: Private vs. common random bits in communication complexity. *Inform. Process. Lett.*, 39:67–71, July 1991. [doi:10.1016/0020-0190(91)90157-D] 8

[23] NOAM NISAN: CREW PRAMs and decision trees. In *Proc. 21st STOC*, pp. 327–335, New York, NY, USA, 1989. ACM Press. [doi:10.1145/73007.73038] 2, 4, 7, 10

[24] NOAM NISAN AND MARIO SZEGEDY: On the degree of Boolean functions as real polynomials. *Comput. Complexity*, 4:462–467, 1992. [doi:10.1007/BF01263419] 2, 4, 10

[25] NOAM NISAN AND AVI WIGDERSON: On rank vs. communication complexity. *Combinatorica*, 15:557–565, 1995. [doi:10.1007/BF01192527] 4, 8, 21

[26] RYAN O'DONNELL, JOHN WRIGHT, AND YUAN ZHOU: The Fourier entropy-influence conjecture for certain classes of Boolean functions. *To appear in ICALP'11*, 2011. 16

[27] RAMAMOHAN PATURI AND JANOS SIMON: Probabilistic communication complexity. *J. Comput. System Sci.*, 3(1):106–123, 1986. [doi:10.1016/0022-0000(86)90046-2] 9

[28] ALEXANDER A. RAZBOROV: On the distributional complexity of disjointness. *Theoret. Comput. Sci.*, 106:385–390, 1992. [doi:10.1016/0304-3975(92)90260-M] 22

[29] RÜDIGER REISCHUK: A lower time-bound for parallel random access machines without simultaneous writes. Technical Report RJ3431, IBM, New York, 1982. 2

[30] RONALD L. RIVEST AND JEAN VUILLEMIN: On recognizing graph properties from adjacency matrices. *Theoret. Comput. Sci.*, 3(3):371–384, 1976. [doi:10.1016/0304-3975(76)90053-0] 7

[31] DAVID RUBINSTEIN: Sensitivity vs. block sensitivity of Boolean functions. *Combinatorica*, 15(2):297–299, 1995. [doi:10.1007/BF01200762] 19

[32] ALEXANDER SHERSTOV: On quantum-classical equivalence for composed communication problems. *Quantum Inf. Comput.*, 10(5&6):435–455, 2010. 9, 10, 15

[33] YAOYUN SHI: Approximating linear restrictions of Boolean functions. Technical report, Manuscript, 2002. 16, 17

[34] HANS-ULRICH SIMON: A tight $\Omega(\log\log n)$-bound on the time for parallel RAMs to compute non-degenerate Boolean functions. In *Proc. 4th Int. Symp. Fundam. Comput. Theory (FCT)*, volume 158 of *LNCS*, pp. 439–444. Springer, 1983. 5

[35] GÁBOR TARDOS: Query complexity, or why is it difficult to separate $NP^A \cap coNP^A$ from P^A by random oracles A? *Combinatorica*, 9:385–392, 1989. [doi:10.1007/BF02125350] 4

[36] GYÖRGY TURÁN: The critical complexity of graph properties. *Inform. Process. Lett.*, 18:151–153, 1984. [doi:10.1016/0020-0190(84)90019-X] 7

[37] RONALD DE WOLF: Quantum communication and complexity. *Theoret. Comput. Sci.*, 287(1):337–353, 2002. [doi:10.1016/S0304-3975(02)00377-8] 3

[38] RONALD DE WOLF: A Brief Introduction to Fourier Analysis on the Boolean Cube. *Theory Comput. Library, Grad. Surv.*, 1, 2008. [doi:10.4086/toc.gs.2008.001] 12

[39] ANDREW CHI-CHIH YAO: Some complexity questions related to distributive computing. In *Proc. 11th STOC*, pp. 209–213, New York, NY, USA, 1979. ACM Press. [doi:10.1145/800135.804414] 7

[40] ZHIQIANG ZHANG AND YAOYUN SHI: On the parity complexity measures of Boolean functions. *Theoret. Comput. Sci.*, 411(26–28):2612–2618, 2010. [doi:10.1016/j.tcs.2010.03.027] 11

AUTHORS

Pooya Hatami
Ph. D. student
University of Chicago
pooya@cs.uchicago.edu
http://people.cs.uchicago.edu/~pooya

Raghav Kulkarni
Postdoctoral researcher
LIAFA, Paris 7
raghav@cs.uchicago.edu
http://people.cs.uchicago.edu/~raghav

Denis Pankratov
Ph. D. student
University of Chicago
pankratov@cs.uchicago.edu
http://people.cs.uchicago.edu/~pankratov

ABOUT THE AUTHORS

POOYA HATAMI is a Ph. D. student in computer science at the University of Chicago, advised by Alexander Razborov. He is interested in complexity theory and combinatorics. He enjoys sports such as basketball and volleyball, arts such as photography and sculpture, and music, especially Jazz and Blues.

RAGHAV KULKARNI did his undergraduate studies in Mathematics at the Chennai Mathematical Institute (India). He obtained his Ph. D. in Computer Science from the University of Chicago (USA) under the supervision of Janos Simon and Alexander Razborov. The title of his thesis is "Computational Complexity: Counting, Evasiveness, and Isolation." Currently he is a postdoc at LIAFA, Paris 7 in the "Algorithms and Complexity" group lead by Miklos Santha. The main focus of his current research is Decision-Tree Complexity.

DENIS PANKRATOV received his Bachelor's degree from the University of Toronto. His interest in the theory of computation was sparked by Allan Borodin, who gave inspiring lectures and supervised Denis's summer research project. Currently, Denis is a Ph. D. student at the University of Chicago supervised by Laci Babai. His main research interest is in complexity theory, and in particular, in communication complexity. In his spare time, he likes to lift heavy weights at the university gym.