Electronic Colloquium on Computational Complexity, Report No. 192 (2017)

Alternation, Sparsity and Sensitivity : Bounds and Exponential Gaps

Krishnamoorthy Dinesh Jayalal Sarma

Department of Computer Science & Engineering
Indian Institute of Technology Madras, Chennai, India
{kdinesh,jayalal}@cse.iitm.ac.in

Keywords : Boolean function parameters, Sensitivity Conjecture, XOR Log-Rank Conjecture

Abstract

The well-known Sensitivity Conjecture regarding combinatorial complexity measures on Boolean functions states that for any Boolean function $f : \{0,1\}^n \to \{0,1\}$, block sensitivity of f is polynomially related to sensitivity of f (denoted by $\mathsf{s}(f)$). From the complexity theory side, the Xor Log-Rank Conjecture states that for any Boolean function, $f : \{0,1\}^n \to \{0,1\}$ the communication complexity of a related function $f^{\oplus} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, (defined as $f^{\oplus}(x,y) = f(x \oplus y)$) is bounded by polynomial in logarithm of the sparsity of f (the number of non-zero Fourier coefficients for f, denoted by $\mathsf{sparsity}(f)$). Both the conjectures play a central role in the domains in which they are studied.

A recent result of Lin and Zhang (2017) implies that to confirm the above two conjectures it suffices to upper bound alternation of f (denoted $\mathsf{alt}(f)$) for all Boolean functions f by polynomial in $\mathsf{s}(f)$ and logarithm of $\mathsf{sparsity}(f)$, respectively. In this context, we show the following results:

- We show that there exists a family of Boolean functions for which $\mathsf{alt}(f)$ is at least *exponential* in $\mathsf{s}(f)$ and $\mathsf{alt}(f)$ is at least *exponential* in $\log \mathsf{sparsity}(f)$. En route to the proof, we also show an exponential gap between $\mathsf{alt}(f)$ and the decision tree complexity of f, which might be of independent interest.
- As our main result, we show that, despite the above exponential gap between $\mathsf{alt}(f)$ and $\log \mathsf{sparsity}(f)$, the Xor Log-Rank Conjecture is true for functions with the alternation upper bounded by $\mathsf{poly}(\log n)$. It is easy to observe that the Sensitivity Conjecture is also true for this class of functions.
- The starting point for the above result is the observation (derived from Lin and Zhang (2017)) that for any Boolean function f, $\mathsf{deg}(f) \leq \mathsf{alt}(f)\mathsf{deg}_2(f)\mathsf{deg}_m(f)$ where $\mathsf{deg}(f)$, $\mathsf{deg}_2(f)$ and $\mathsf{deg}_m(f)$ are the degrees of f over $\mathbb{R}$, $\mathbb{F}_2$ and $\mathbb{Z}_m$ respectively. We give three further applications of this bound: (1) We show that for Boolean functions f of constant alternation have $\mathsf{deg}_2(f) = \Omega(\log n)$. (2) Moreover, these functions also have high sparsity $(\Omega(\sqrt{\mathsf{deg}(f)}))$, thus partially answering a question of Kulkarni and Santha (2013). (3) We observe that our relation upper bounding real degree also improves the upper bound for influence to $\mathsf{deg}_2(f)^2 \cdot \mathsf{alt}(f)$ improving Guo and Komargodski (2017).

ISSN 1433-8092

1 Introduction

A central theme of research in Boolean function complexity is relating the complexity measures of Boolean functions (see [6] for a survey). For a Boolean function $f : \{0,1\}^n \to \{0,1\}$, *sensitivity* of f on $x \in \{0,1\}^n$, is the maximum number of indices $i \in [n]$, such that $f(x \oplus e_i) \neq f(x)$ where $e_i \in \{0,1\}^n$ with exactly the i^{th} bit as 1. The sensitivity of f (denoted by $\mathsf{s}(f)$) is the maximum sensitivity of f over all inputs. A related parameter is the *block sensitivity* of f (denoted by $\mathsf{bs}(f)$), where we allow disjoint blocks of indices to be flipped instead of a single bit. Another parameter is the *degree* (denoted by $\mathsf{deg}(f)$) of a multilinear polynomial over reals that agrees with f on Boolean inputs. If the polynomial is over $\mathbb{F}_2$, then the degree of the polynomial is denoted by $\mathsf{deg}_2(f)$.

Nisan and Szegedy conjectured that for an arbitrary function $f : \{0,1\}^n \to \{0,1\}$, $\mathsf{bs}(f) \leq \mathsf{poly}(\mathsf{s}(f))$ and this is popularly known as the Sensitivity Conjecture [26]. Though the measures $\mathsf{bs}(f)$ and $\mathsf{s}(f)$ were introduced to understand the Crew-Pram model of computation [9, 25], subsequent works [6, 3, 29] showed connection to other Boolean function parameters, in particular, $\sqrt{\mathsf{bs}(f)} \leq \mathsf{deg}(f) \leq \mathsf{bs}(f)^3$. Hence the Sensitivity Conjecture can equivalently be stated as : for any Boolean function f, $\mathsf{deg}(f) \leq \mathsf{poly}(\mathsf{s}(f))$. This question has been extensively studied in [28, 18, 1] (see [15] for a survey) and for various restricted classes of Boolean functions in [25, 31, 8, 7, 2]. There are also recent approaches to settle the conjecture via a formulation in terms of a communication game [11] and via a formulation in terms of distributions on the Fourier spectrum of Boolean functions [13].

For an $f : \{0,1\}^n \to \{-1,1\}$, define $f^{\oplus}(x,y) = f(x \oplus y)$. From the complexity theory side, the Xor Log-Rank Conjecture (proposed in [33]) says that the deterministic communication complexity of $f^{\oplus}$ (denoted by $\mathsf{CC}_{\oplus}(f)$) is polynomially upper bounded by the logarithm of sparsity of f (denoted by $\mathsf{sparsity}(f)$). The best known bound for any Boolean function f is due to Tsang et al., [30] who showed that the $\mathsf{CC}_{\oplus}(f)$ is $O(\sqrt{\mathsf{sparsity}(f)} \log \mathsf{sparsity}(f))$. The conjecture was proved for restricted classes of Boolean functions like monotone functions [23], symmetric functions [32], functions computable by constant depth polynomial size circuits [19], functions of small spectral norm [30] and read restricted polynomials over $\mathbb{F}_2$ [16].

Recently, Lin and Zhang [20] studied both the above stated conjectures in connection to alternation, a measure of non-monotonicity of f (denoted by $\mathsf{alt}(f)$, see Section 2 for definition), by proving that for any Boolean function f, $\mathsf{bs}(f) = O(\mathsf{s}(f)\mathsf{alt}(f)^2)$ and $\mathsf{CC}_{\oplus}(f) = O(\log \mathsf{sparsity}(f)\mathsf{alt}(f)^2)$. These results shows that to settle the Sensitivity Conjecture, it suffices to show that for any Boolean function f, $\mathsf{alt}(f) \leq \mathsf{poly}(\mathsf{s}(f))$ and to settle the Xor Log-Rank Conjecture, it suffices to show that for any Boolean function f, $\mathsf{alt}(f) \leq \mathsf{poly}(\log \mathsf{sparsity}(f))$.

Our Results : As a first step, we ask is it indeed true that for all Boolean functions f, $\mathsf{alt}(f) = O(\mathsf{poly}(s(f)))$ and for all Boolean functions f, $\mathsf{alt}(f) \leq \mathsf{poly}(\log \mathsf{sparsity}(f))$. We answer both of these questions in the negative by exhibiting a family of Boolean functions $\mathcal{F} = \{f_k \mid k \in \mathbb{N}\}$ (Definition 3.1) for which $\mathsf{alt}(f_k)$ is at least *exponential* in $\mathsf{s}(f_k)$ and $\mathsf{alt}(f_k)$ is at least *exponential* in $\log \mathsf{sparsity}(f_k)$.

Theorem 1.1. *There exists a family of Boolean functions* $\mathcal{F} = \{f_k : \{0,1\}^{n_k} \to \{0,1\} \mid k \in \mathbb{N}\}$ *such that* $\mathsf{alt}(f_k) \geq 2^{\mathsf{s}(f_k)} - 1$ *and* $\mathsf{alt}(f_k) \geq 2^{(\log \mathsf{sparsity}(f_k))/2} - 1$.

The main property of $f_k \in \mathcal{F}$ which we exploit to prove Theorem 1.1 is that $\mathsf{alt}(f_k) = 2^{\mathsf{DT}(f_k)} - 1$ (Theorem 3.2) where $\mathsf{DT}(f_k)$ is the depth of the optimal decision tree computing f_k (see Section 2

for definition). We also show an asymptotically matching upper bound for alternation of any Boolean function. More precisely, for any $f : \{0,1\}^n \to \{0,1\}$, we show that $\mathsf{alt}(f) \leq 2^{\mathsf{DT}(f)+1} - 1$ (Theorem 3.4).

Though the function family $\mathcal{F}$ rules out settling the Sensitivity Conjecture (Xor Log-Rank Conjecture resp.) via upper bounding alternation by a polynomial in sensitivity (polynomial in the logarithm of sparsity resp.) for all Boolean functions, it is partly unsatisfactory since both the conjectures are true for all $f_k \in \mathcal{F}$ (see Section 3.3 for details).

In fact, any $f : \{0,1\}^n \to \{0,1\}$ for which $\mathsf{alt}(f) = 2^{\Omega(\mathsf{DT}(f))}$, must satisfy $\mathsf{DT}(f) = O(\log n)$. In addition, if f depends on all the input variables, the Sensitivity Conjecture is true for f. Notice that, for all $f_k \in \mathcal{F}$, $\mathsf{DT}(f_k) = \log n_k$ and f_k depends on all the n_k variables. Hence a natural question is, does there exist another family of functions f where $\mathsf{alt}(f)$ is at least super-polynomial in $\mathsf{s}(f)$, but $\mathsf{DT}(f)$ is not logarithmic in n. To this end, we exhibit a family of Boolean functions $\mathcal{G}$, such that for all $g \in \mathcal{G}$, $\mathsf{alt}(g)$ is *super-linear* in $\mathsf{s}(g)$ and $\mathsf{DT}(g)$ is $\omega(\log n)$ where n is the number of variables in g.

Theorem 1.2. *There exists a family of Boolean functions $\mathcal{G} = \{g_k : \{0,1\}^{n_k} \to \{0,1\} \mid k \in \mathbb{N}\}$ such that $\mathsf{alt}(g_k) \geq \mathsf{s}(g_k)^{\log_3 5}$ while $\mathsf{DT}(g_k)$ is $\Omega(n_k^{\log_6 3})$.*

The main tool used in proving Theorem 1.2 is a bound on the alternation of composed Boolean functions (Lemma 3.5).

As mentioned before, Lin and Zhang [20] showed that Xor Log-Rank Conjecture is true for all Boolean functions satisfying $\mathsf{alt}(f) \leq \mathsf{poly}(\log \mathsf{sparsity}(f))$. As our main result, we further strengthen this when $\mathsf{sparsity}(f) < n$.

Theorem 1.3 (Main)**.** *For large enough n, the Xor Log-Rank Conjecture is true for all $f : \{0,1\}^n \to \{0,1\}$, such that $\mathsf{alt}(f) \leq \mathsf{poly}(\log n)$ where f depends on all its inputs.*

Our starting point in proving the above result is a relation connecting deg, deg_2, deg_m and alt. For all Boolean functions f,

$$\mathsf{deg}(f) \leq \mathsf{alt}(f) \cdot \mathsf{deg}_2(f) \cdot \mathsf{deg}_m(f) \tag{1}$$

We remark that for special cases, Eq. (1) is known to be true. For instance, if f is a monotone, it can be shown[1] that $\mathsf{deg}(f) \leq \mathsf{deg}_2(f)^2$. However, there are functions of large alternation where $\mathsf{deg}_2(f)$ is constant while $\mathsf{deg}(f)$ is n (for instance, parity on n bits). Hence, we cannot upper bound degree by $\mathbb{F}_2$-degree in general but Eq. (1) (for $m = 2$) says that we can indeed upper bound $\mathsf{deg}(f)$ by $\mathsf{deg}_2(f)$ using $\mathsf{alt}(f)$. This case ($m = 2$) is implicit in [20]. We now give three further applications of Eq. (1) (see Section 5).

As our first application, we show that using a related result due to Gopalan et al. [12] and Eq. (1) (with $m = 2$), all Boolean functions of bounded alternation ($\mathsf{alt}(f) = O(1)$) must have $\mathsf{deg}_2(f) = \Omega(\log n)$ (see Corollary 5.1 for a generalization).

As a second application, we show that Boolean functions with bounded alternation have high sparsity. Kulkarni and Santha [19] had studied the relation between $\log \mathsf{sparsity}(f)$ and $\mathsf{deg}(f)$ in the case of restricted families of monotone functions and asked if they are linearly related in the case of monotone functions. In this direction, we show the following lower bound for $\log_2 \mathsf{sparsity}(f)$ in terms of $\mathsf{deg}(f)$.

[1] When f is monotone, it is known that $\mathsf{DT}(f) \leq \mathsf{s}(f)^2$ and $\mathsf{s}(f) \leq \mathsf{deg}_2(f)$ (Corollary 5 and Proposition 4 resp., of [6]). Proposition 4 of [6] though states that $\mathsf{s}(f) \leq \mathsf{deg}(f)$ for any monotone f, the argument is valid for $\mathsf{deg}_2(f)$ also. Since, $\mathsf{deg}(f) \leq \mathsf{DT}(f)$ (cf. [6]), $\mathsf{deg}(f) \leq \mathsf{deg}_2(f)^2$.

Theorem 1.4. *For Boolean functions f with* $\mathsf{alt}(f) = O(1)$, $\log \mathsf{sparsity}(f) = \Omega(\sqrt{\mathsf{deg}(f)})$

As a third application, we observe that Eq. (1) implies an improved upper bound for influence (denoted by $\mathbf{I}[f]$, see Section 2 for a definition) to $\mathsf{deg}_2(f)^2 \cdot \mathsf{alt}(f)$. This improves the result of Guo and Komargodski [14] who showed that $\mathbf{I}[f] = O(\mathsf{alt}(f)\sqrt{n})$, thus giving faster learning algorithms for functions of bounded alternation in the PAC learning model.

2 Preliminaries

We introduce the notations and definitions used in this paper. All logarithms are to the base 2 unless otherwise stated. Let $[n] \stackrel{\text{def}}{=} \{1, 2, \ldots, n\}$. For $i \in [n]$, define e_i to be an n bit Boolean string with one in i^{th} location and zero elsewhere. A Boolean function $f : \{0,1\}^n \to \{0,1\}$ is *monotone* if $\forall x, y \in \{0,1\}^n$, $x \prec y \implies f(x) \leq f(y)$ where, $x \prec y$ iff $\forall i \in [n], x_i \leq y_i$. The *alternation* of a Boolean function is a measure of non-monotonicity of the Boolean function. More precisely, if we define a collection of distinct inputs $x_0, x_1, x_2 \ldots, x_n \in \{0,1\}^n$ satisfying $0^n = x_0 \prec x_1 \prec x_2 \prec \cdots \prec x_n = 1^n$ as a *chain* in the Boolean hypercube $\mathcal{B}_n$ then, alternation of f (denoted by $\mathsf{alt}(f)$) is defined as $\max\{\mathsf{alt}(f, \mathcal{C}) \mid \mathcal{C} \text{ is a chain in } \mathcal{B}_n\}$ where $\mathsf{alt}(f, \mathcal{C}))$ is $|\{i \mid f(x_{i-1}) \neq f(x_i), x_i \in \mathcal{C}, i \in [n]\}|$. Indeed for a monotone f, $\mathsf{alt}(f) = 1$.

Any chain $\mathcal{C}$ of a Boolean hypercube over $\{0,1\}^n$ is uniquely determined by a permutation $\sigma \in S_n$ and vice versa. An $x \in \{0,1\}^n$ belongs to a chain $\mathcal{C}$ defined by $\sigma \in S_n$ iff $x = 0^n$ or $x = \bigvee_{i=1}^{wt(x)} e_{\sigma(i)}$ where the OR is taken coordinate wise and $wt(x)$ is the number of ones in x. If a chain is defined using a permutation σ, we use σ to denote the chain $\mathcal{C}$.

For a Boolean function f on m variables and g on n variables, we denote $f \circ g$ as a function on mn variables $\{x_{11}, \ldots, x_{mn}\}$ defined as $f(g(x_{11}, \ldots, x_{1n}), g(x_{21}, \ldots, x_{2n}), \ldots, g(x_{m1}, \ldots, x_{mn}))$. We define $g^{\circ k}$ as the Boolean function on n^k variables as $g^{\circ(k-1)} \circ g$ for $k > 1$ and g for $k = 1$.

Given a Boolean function, there always exists a unique n variable multilinear polynomial over $\mathbb{R}[x_1, \ldots, x_n]$ such that the evaluation agrees with the function on $\{0,1\}^n$. The *degree* of function f is the degree of such a polynomial (denoted by $\mathsf{deg}(f)$). If we consider the polynomial to be over $\mathbb{Z}_m[x_1, \ldots, x_n]$ for an integer $m > 1$ instead, we get the $\mathbb{Z}_m$-*degree* of f (denoted by $\mathsf{deg}_m(f)$). If $m = 2$, it is the $\mathbb{F}_2$-degree of f (denoted by $\mathsf{deg}_2(f)$).

A deterministic Boolean *decision tree* is a rooted tree where the leaves are labeled 0 or 1 and non-leaf nodes labeled by a variable having two outgoing edges (corresponding to the value taken by the variable). A decision tree is said to compute a Boolean function f, if for all inputs x, the path from root to the leaf determined by x is labeled $f(x)$. Define $\mathsf{DT}(f)$ as the depth of the smallest depth decision tree computing f.

For an $x, y \in \{0,1\}^n$, we denote by $x \oplus y$, the input obtained by taking bitwise parity of x and y. For $B \subseteq [n]$, e_B denotes the characteristic vector of B. For $f : \{0,1\}^n \to \{0,1\}$ and $x \in \{0,1\}^n$, define the sensitivity of f on x (denoted by $\mathsf{s}(f, x)$) as $|\{i \mid f(x \oplus e_i) \neq f(x), i \in [n]\}|$. We define the block sensitivity of f on x (denoted by $\mathsf{bs}(f, x)$) as the size of maximal collection of disjoint non-empty sets $\{B_i\}$ where each $B_i \subseteq [n]$ in the collection satisfy $f(x \oplus e_{B_i}) \neq f(x)$. *Certificate complexity* of f on input x (denoted by $\mathsf{C}(f, x)$) is the size of the smallest certificate $S \subseteq [n]$ such that $\forall y \in \{0,1\}^n$ with $y|_S = x|_S \implies f(y) = f(x)$. The *sensitivity* of f (denoted by $\mathsf{s}(f)$) is defined as $\max_{x \in \{0,1\}^n} \mathsf{s}(f, x)$. The *influence* of a Boolean function f (denoted by $\mathbf{I}[f]$) is defined as $\mathbf{E}_{x \in \{0,1\}^n}[s(f, x)]$. The *block sensitivity* of f (denoted by $\mathsf{bs}(f)$) is $\max_{x \in \{0,1\}^n} \mathsf{bs}(f, x)$. Note that $\mathbf{I}[f] \leq \mathsf{s}(f) \leq \mathsf{bs}(f)$. It is also known that $\mathbf{I}[f] \leq \mathsf{deg}(f) \leq \mathsf{DT}(f)$ (cf. [6]).

For $x \in \{0,1\}^n$ and $S \subseteq [n]$, define $\chi_S(x) = (-1)^{\sum_{i \in S} x_i}$. Any $f : \{0,1\}^n \to \{-1,1\}$ can be uniquely expressed as $\sum_{S \subseteq [n]} \widehat{f}(S)\chi_S(x)$ where $\widehat{f}(S) \in \mathbb{R}$, indexed by $S \subseteq [n]$, denotes the *Fourier coefficients* of f which is $\frac{1}{2^n}\sum_x f(x)\chi_S(x)$ (see [27] for more details). The *sparsity* of a Boolean function f (denoted by $\mathsf{sparsity}(f)$) is the number of non-zero Fourier coefficients of f. For Boolean functions f whose range is $\{0,1\}$, we define sparsity of f to be the sparsity of the function $1-2f$ in this paper.

3 Alternation vs Sensitivity and Alternation vs Logarithm of Sparsity

In this section, we show that there exists a family of function $\mathcal{F} = \{f_k \mid k \in \mathbb{N}\}$ with $\mathsf{alt}(f_k)$ is at least exponential in $\mathsf{s}(f_k)$, $\mathsf{DT}(f_k)$ and $\log \mathsf{sparsity}(f_k)$ respectively (Section 3.1). Complementing this, we show that for any Boolean function f, $\mathsf{alt}(f)$ can be at most exponential in $\mathsf{DT}(f)$ (Section 3.2). We prove a bound on the alternation of composed Boolean functions and use it to obtain a family of functions with super-linear gap between alternation and sensitivity with large decision tree depth unlike functions in $\mathcal{F}$ (Section 3.3).

3.1 Exponential Gaps : Alternation vs Decision Tree Depth

We prove Theorem 1.1 in this section. We first show that there exists a family of function $\mathcal{F} = \{f_k \mid k \in \mathbb{N}\}$ with $\mathsf{alt}(f_k)$ equals $2^{\mathsf{DT}(f_k)} - 1$ (Theorem 3.2). Since for any Boolean function f, $\mathsf{s}(f) \leq \mathsf{DT}(f)$ (cf. [6]), we have, $\mathsf{alt}(f_k) = 2^{\mathsf{DT}(f_k)} - 1 \geq 2^{\mathsf{s}(f_k)} - 1$ and since for any Boolean function f, $\log \mathsf{sparsity}(f) \leq 2\mathsf{deg}(f) \leq 2\mathsf{DT}(f)$ (cf. [27]), we get that for f_k, $\mathsf{alt}(f_k) = 2^{\mathsf{DT}(f_k)} - 1 \geq 2^{0.5 \log \mathsf{sparsity}(f_k)} - 1$ thereby proving Theorem 1.1.

Hence, one cannot hope to show that for all Boolean functions f, alternation is upper bounded polynomially by sensitivity of f or polynomially by logarithm of sparsity of f. We now define our family $\mathcal{F}$ of Boolean functions.

Definition 3.1. Let $\mathcal{F} = \{f_k \mid k \in \mathbb{N}\}$ be a family of Boolean functions where for every $k \in \mathbb{N}$, $f_k : \{0,1\}^{2^k-1} \to \{0,1\}$ is defined by the decision tree which is a full binary tree of depth k with each of the $2^k - 1$ internal node querying a distinct variable and each of the nodes at level k have left leaf child labeled 0 and right leaf child labeled 1.

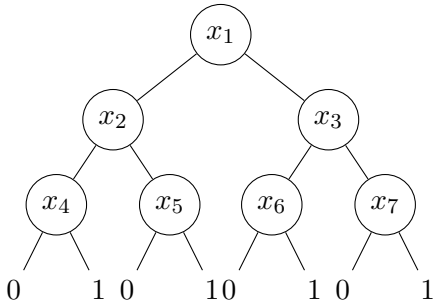
A Boolean function $f_3 \in \mathcal{F}$ is described using a decision tree in Fig. 1a.

Theorem 3.2. *For every $k \geq 1$, $f_k \in \mathcal{F}$, $\mathsf{alt}(f_k) = 2^{DT(f_k)} - 1$.*

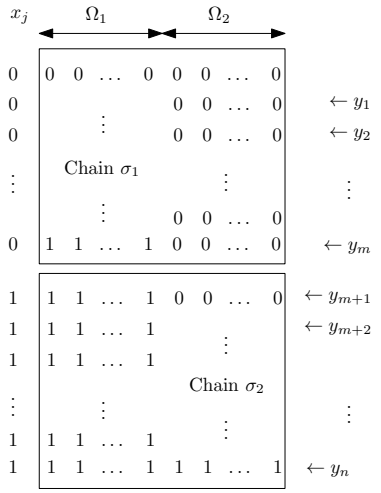
Proof. By definition, f_k is computed by a decision tree of depth k. We show that for $k \geq 1$, $\mathsf{alt}(f_k) \geq 2^k - 1$. Since f_k is defined on $2^k - 1$ variables, we get that $\mathsf{alt}(f_k) = 2^k - 1$ thereby completing the proof.

Since we need to work with functions whose variable set is not necessarily numbered from 1 to n, we associate bijections (instead of permutations) with chains. For any set Ω, let B_Ω be defined as $\{\sigma : [|\Omega|] \to \Omega \mid \sigma \text{ is a bijection}\}$. For a Boolean function f defined on the variables $\{x_{i_1}, x_{i_2}, \ldots, x_{i_n}\}$, $var(f)$ be defined as $\{i_1, i_2, \ldots, i_n\}$.

We now show that $\mathsf{alt}(f_k) \geq 2^k - 1$ by induction on k. For $k = 1$, since f depends only on 1 variable the result holds.



(a) Boolean function $f_3 \in \mathcal{F}$



(b) The chain σ constructed in the proof of Theorem 3.2. Note that Ω_1 and Ω_2 need not be contiguous.

Figure 1

Suppose that the result holds for $f_k \in \mathcal{F}$. For $f_{k+1} \in \mathcal{F}$ on $n = 2^{k+1}-1$ variables $\{x_1, x_2, \ldots, x_n\}$, let T be the decision tree (as in Definition 3.1) computing f_{k+1} of depth $k+1$ with x_j being the root variable for some $j \in [n]$. Let T_1 and T_2 be the left and right subtree of the root node x_j. Consider the Boolean function f_1 (resp. f_2) computed by the decision tree T_1 (resp. T_2). Note that since T_1 and T_2 are obtained from T in this way, by Definition 3.1, f_1 and f_2 belongs to $\mathcal{F}$ and computes the same function upto variable renaming. Also note that both f_1 and f_2 are on $m = 2^k - 1$ variables. Since both T_1 and T_2 are of depth k, by inductive hypothesis, $\mathsf{alt}(f_1) \geq m$ and $\mathsf{alt}(f_2) \geq m$. Using this, we now construct a chain for f_{k+1} of alternation $2^{k+1} - 1$.

Let $\Omega_1 = var(f_1), \Omega_2 = var(f_2)$. Let $\sigma_1 \in B_{\Omega_1}$ and $\sigma_2 \in B_{\Omega_2}$ be such that $\mathsf{alt}(f_1, \sigma_1) = 2^k - 1$ and $\mathsf{alt}(f_2, \sigma_2) = 2^k - 1$. We now define a $\sigma \in B_{var(f)} = B_{\Omega_1 \cup \Omega_2 \cup \{j\}}$ as $\sigma(i) = \sigma_1(i)$ if $i \in \{1, 2, \ldots, m\}$, $\sigma(m+1) = j$ and $\sigma(m+1+i) = \sigma_2(i)$ if $i \in \{1, 2, \ldots, m\}$. By definition, σ is indeed a bijection. The σ obtained is pictorially represented in Fig. 1b for clarity of exposition.

We now claim that $\mathsf{alt}(f, \sigma) = 2^{k+1} - 1$. To show this, consider the chain corresponding to σ given by $0^n \prec y_1 \prec \ldots y_m \prec y_{m+1} \prec \ldots \prec y_n = 1^n$ all belonging to $\{0,1\}^n$. By definition of σ, for $i \in [m]$ since j^{th} bit of y_i is 0, $f(y_i) = f_1(y_i|_{\Omega_1})$ and for $i \in \{0\} \cup [m]$, since j^{th} bit of y_{m+1+i} is 1, $f(y_{m+1+i}) = f_2(y_{m+1+i}|_{\Omega_2})$. Again by definition of σ, for $i \in [m]$, 0^m along with the elements $y_i|_{\Omega_1}$ for $i \in [m]$ (in that order) is a chain witnessing f_1 alternating m times and $y_{m+1+i}|_{\Omega_2}$ for $i \in \{0\} \cup [m]$ (in that order) is a chain witnessing f_2 alternating m times. Now observe that $f(y_m) \neq f(y_{m+1})$. This is because $f(y_m)$ is f_1 evaluated on $x_i = 1$ for all $i \in \Omega_1$ is the rightmost child of T_1 which is 1 and $f(y_{m+1})$ is f_2 evaluated on $x_i = 0$ for all $i \in \Omega_2$ is the leftmost child of T_2 which is 0. Hence $\mathsf{alt}(f, \sigma) = \mathsf{alt}(f_1, \sigma_1) + \mathsf{alt}(f_2, \sigma_2) + 1 = 2m + 1 = 2^{k+1} - 1$ completing the induction. □

In the next section, we show that for any Boolean function f, we can indeed upper bound alternation of f by an exponential in decision tree depth of f.

3.2 Alternation is at most Exponential in Decision Tree Depth

In this section, we show that for all Boolean functions f, $\mathsf{alt}(f) \leq 2^{\mathsf{DT}(f)+1} - 1$. Markov [22] studied a parameter closely related to $\mathsf{alt}(f)$ defined as *decrease* (denoted by $\mathsf{dc}(f)$) where the definition is same as alternation except that the flips in the chain from 1 to 0 (corresponding to a decrease in the function value) alone are counted. Hence for any Boolean function f, $\mathsf{alt}(f) \in \{2\mathsf{dc}(f) - 1, 2\mathsf{dc}(f), 2\mathsf{dc}(f) + 1\}$. Markov [22] showed the following tight connection between negations needed to compute a Boolean function and its decrease.

Theorem 3.3 (Markov [22])**.** *Let* $\mathsf{negs}(f)$ *be the minimum number of negations needed in any circuit computing* f*. Then, for an* $f : \{0,1\}^n \to \{0,1\}$, $\mathsf{negs}(f) = \lceil \log(1 + \mathsf{dc}(f)) \rceil$.

We also use the notion of a *connector* for two Boolean functions which is a crucial idea used by Markov in proving the above theorem. A connector of two Boolean functions f_0 and f_1 is a function $g(b_0, b_1, x)$ on $n + 2$ bits such that $g(0, 1, x) = f_0(x)$ and $g(1, 0, x) = f_1(x)$. Markov showed the following remarkable bound that $\mathsf{negs}(g) \leq \max\{\mathsf{negs}(f_0), \mathsf{negs}(f_1)\}$ (for a proof, see Jukna [17]). We now prove our result which follows from an inductive application of Theorem 3.3.

Theorem 3.4. *For any* $f : \{0,1\}^n \to \{0,1\}$, $\mathsf{alt}(f) \leq 2^{\mathsf{DT}(f)+1} - 1$.

Proof. Since $\mathsf{alt}(f) \leq 2\mathsf{dc}(f)+1$, it suffices to show that $\mathsf{dc}(f) \leq 2^{\mathsf{DT}(f)}-1$. Proof is by induction on $\mathsf{DT}(f)$. For $\mathsf{DT}(f) = 1$, the function depends on at most 1 variable, giving $\mathsf{dc}(f) \leq 1 = 2^{\mathsf{DT}(f)} - 1$. For any f with $\mathsf{DT}(f) \geq 2$ computed by a decision tree T of depth k, let x_i be the variable queried at the root of T for some $i \in [n]$. Define f_0 as f restricted to $x_i = 0$ and f_1 as f restricted to $x_i = 1$. Removing the node x_i from T gives two decision trees which computes f_0 and f_1 (respectively) giving $\mathsf{DT}(f_0) \leq \mathsf{DT}(f) - 1$ and $\mathsf{DT}(f_1) \leq \mathsf{DT}(f) - 1$. Hence by induction, $\mathsf{dc}(f_0) \leq 2^{\mathsf{DT}(f_0)} - 1 \leq 2^{\mathsf{DT}(f)-1} - 1$ and similarly, $\mathsf{dc}(f_1) \leq 2^{\mathsf{DT}(f)-1} - 1$.

Applying Theorem 3.3, we get, $\mathsf{negs}(f_0) = \lceil \log(\mathsf{dc}(f_0) + 1) \rceil$ (and similarly for f_1). Let g be the connector for f_0 and f_1. Since $f(x) = x_i \wedge f_1(x) \vee \neg x_i \wedge f_0(x)$, $f(x) = g(\neg x_i, x_i, x)$. Applying Markov's result on the number of negations needed in computing g, we get $\mathsf{negs}(f) \leq \mathsf{negs}(g) + 1 \leq 1 + \max\{\mathsf{negs}(f_0), \mathsf{negs}(f_1)\}$. Since, $\max\{\mathsf{negs}(f_0), \mathsf{negs}(f_1)\} \leq \lceil \log(2^{\mathsf{DT}(f)-1} - 1 + 1) \rceil$ which is $\mathsf{DT}(f) - 1$, $\mathsf{negs}(f) \leq \mathsf{DT}(f)$. Applying Theorem 3.3, on f completes the induction. □

Note that for the family of functions $\mathcal{F}$, Theorem 3.2 shows that the above result is asymptotically tight. In Appendix A.1, we present a simpler proof of a slightly weaker result (in terms of an additive constant) that for all f, $\mathsf{alt}(f) \leq 2^{\mathsf{DT}(f)+1} + 1$.

3.3 Super Linear Gaps Between Alternation and Sensitivity

In this section, we exhibit a family of functions with a super-linear gap between alternation and sensitivity with high decision tree depth. We start by giving a motivation for this study.

In Section 3.1, we showed the existence of a family $\mathcal{F}$ of Boolean functions with alternation at least exponential in sensitivity and alternation is at least logarithm of sparsity. Hence this family $\mathcal{F}$ rules out the possibility of upper bounding alternation by a polynomial sensitivity or a polynomial in logarithm of sparsity for all Boolean functions which would have settled the Sensitivity Conjecture and Xor Log-Rank Conjecture by the results of Lin and Zhang [20]. However, this is still unsatisfactory as the above mentioned conjectures holds true for the family $\mathcal{F}$ as argued below.

Sensitivity Conjecture is true for all $f_k \in \mathcal{F}$: To argue this, we first observe that as f_k depends on all its inputs, $s(f_k) = \Omega(\log n_k)$ [28]. Hence we can conclude that $\mathsf{DT}(f_k) = O(\mathsf{s}(f_k))$. Since $\mathsf{deg}(f_k) \leq \mathsf{DT}(f_k)$ (cf. [6]), we get $\mathsf{deg}(f_k) = O(\mathsf{s}(f_k))$. Note that, this argument is valid for any function f that depends on n variables, and $\mathsf{DT}(f) = O(\log n)$.

XOR **Log-Rank Conjecture is true for all $f_k \in \mathcal{F}$:** Let T_k be the decision tree computing f_k (as per Definition 3.1). Consider the Fourier expansion of $f_k : \{0,1\}^{n_k} \to \{-1,1\}$ given by,

$$\begin{aligned} f_k(x) &= \sum_{a \in \{0,1\}^k} val(\ell_a) \prod_{i_j \in var(\ell_a)} \frac{1 + (-1)^{x_{i_j}} (-1)^{a_j}}{2} \\ &= \frac{-1}{2^k} \sum_{a \in \{0,1\}^k} val(\ell_a) \sum_{S \subseteq var(\ell_a)} \chi_S(x) \prod_{j : i_j \in S} (-1)^{a_j} \end{aligned}$$

where, for the leaf ℓ_a of T_k, $a \in \{0,1\}^k$ is the value taken by the variables with indices $var(\ell_a) = \{i_1, i_2, \ldots, i_k\}$ in the path to ℓ_a and $val(\ell_a) \in \{-1,1\}$ is the value of f_k on the input a. For $a, b \in \{0,1\}^k$, if $a \neq b$ then $var(\ell_a) \neq var(\ell_b)$ as the variable set in each path of T_k are different. Hence $\chi_S(x)$ for $S = var(\ell_a)$ is never canceled in the above expansion and we can conclude that $\mathsf{sparsity}(f_k) \geq 2^k$. Since $\mathsf{DT}(f_k) = k$, we have $\mathsf{DT}(f_k) \leq \log \mathsf{sparsity}(f_k)$. Using the fact that for any Boolean function f, $\mathsf{CC}_{\oplus}(f) \leq 2\mathsf{DT}(f)$ [23], $\mathsf{CC}_{\oplus}(f_k) = O(\log \mathsf{sparsity}(f_k))$ thereby concluding that XOR LOG-RANK CONJECTURE is true for all $f_k \in \mathcal{F}$.

The key property of $f \in \mathcal{F}$ due to which both the conjectures are true is that the decision tree complexity of $f \in \mathcal{F}$ is logarithmic in the number of variables of f. In fact for any $f : \{0,1\}^n \to \{0,1\}$ which satisfy $\mathsf{alt}(f) \geq 2^{\mathsf{DT}(f)}$ must have $\mathsf{DT}(f) = O(\log n)$ as $\mathsf{alt}(f) \leq n$. Hence, we ask if there exists a family of functions f where $\mathsf{alt}(f)$ grows faster than $\mathsf{s}(f)$ but $\mathsf{DT}(f)$ is not very small.

Super-linear Gap between Alternation and Sensitivity :

In the rest of this section, we answer this question by exhibiting a family of Boolean functions $\mathcal{G} = \{g_k : \{0,1\}^{n_k} \to \{0,1\} \mid k \in \mathbb{N}\}$ with $\mathsf{alt}(g_k) = \omega(\mathsf{s}(g_k))$ and $\mathsf{DT}(g_k)$ is $\Omega(n_k^{\log_6 3})$. Before proceeding, we show a lower bound on the alternation of composition of two Boolean functions in terms of its alternation.

Lemma 3.5. *For any $g : \{0,1\}^n \to \{0,1\}$, with $g(0^n) \neq g(1^n)$ and any $f : \{0,1\}^m \to \{0,1\}$, $\mathsf{alt}(f \circ g) \geq \mathsf{alt}(f) \cdot \mathsf{alt}(g)$.*

Proof. Without loss of generality, assume $g(0^n) = 0$ and $g(1^n) = 1$ (otherwise work with $\neg g$ as $\mathsf{alt}(g) = \mathsf{alt}(\neg g)$). Let $A = (0^n = z_0 \prec z_1 \prec \cdots \prec z_n = 1^n)$ be a chain on $\{0,1\}^n$ such that the alternation of g is maximum. Consider any maximum alternation chain of f and let σ be the permutation associated with the chain. We exhibit a chain $B = (y_0, y_1, \ldots, y_{nm})$ on $\{0,1\}^{nm}$ with $\mathsf{alt}(f) \cdot \mathsf{alt}(g)$ many alternations.

We divide the inputs in the chain B into m blocks of size n each. We say that for a $k \in [nm]$, the input $y_k \in \{0,1\}^{nm}$ belongs to the *block* b if $b = \lceil \frac{k}{n} \rceil$. We define the *position* of y_k in its block, $pos(k)$, as n if $n \mid k$ and $(k \mod n)$ otherwise. Let $y_k = (x_1^k, x_2^k, \ldots, x_m^k)$ where $x_i^k \in \{0,1\}^n$. For $k = 0$, define $x_i^k = 0^n$ for all $i \in [m]$ and for $k = nm$, define $x_i^k = 1^n$ for all $i \in [m]$. For the remaining values of k, x_i^k for $i \in [m]$ is defined as

$$x_i^k = \begin{cases} z_n = 1^n & \text{if } i \in \{\sigma(1), \sigma(2), \ldots, \sigma(b-1)\} \text{ and } b \geq 2 \\ z_{pos(k)} & \text{if } i = \sigma(b) \\ z_0 = 0^n & \text{otherwise} \end{cases}$$

We can see that $y_0 = 0^{nm}$ and $y_{nm} = 1^{nm}$ and for $k \in [nm]$, from the above definition, $y_{k-1} \prec y_k$ as $\forall i \in [m]\ x_i^{k-1} \prec x_i^k$. We now argue that $f \circ g$ alternates at least $\mathsf{alt}(f) \cdot \mathsf{alt}(g)$ times in the chain B. Consider the input $(g(x_1^k), g(x_2^k), \ldots, g(x_m^k))$ to the function f and let y_k belong to the block b.

Consider the case when $b = 1$. In this case, all of x_i^k except $i = \sigma(b)$ is 0^n. As long as y_k stays within the block b, the input the function f changes its value only at $x^k_{\sigma(b)} = x^k_{\sigma(1)}$. Since $g(x^k_{\sigma(b)})$ changes its value $\mathsf{alt}(g)$ times, $f \circ g$ will also alternate $\mathsf{alt}(g)$ times if value of f changes on flipping location $\sigma(b)$ in its input.

For k such that $b > 1$, by definition of y_k, $x^k_{\sigma(1)}, \ldots, x^k_{\sigma(b-1)}$ is 1^n and $x^k_{\sigma(b+1)} \ldots, x^k_{\sigma(m)}$ is 0^n. Since $g(0^n) = 0$ and $g(1^n) = 1$, the input to f will be either $r_1 = \vee_{i=1}^{b} e_{\sigma(i)}$ or $r_0 = \vee_{i=1}^{b-1} e_{\sigma(i)}$. Since g alternates $\mathsf{alt}(g)$ times thereby changing the input to f between r_0 and r_1, $f \circ g$ will also alternate $\mathsf{alt}(g)$ times if value of f changes on flipping location $\sigma(b)$.

Thus in both cases, if f alternates once, $f \circ g$ alternates $\mathsf{alt}(g)$ in the chain B. Since f alternates $\mathsf{alt}(f)$ times on σ, $f \circ g$ alternates $\mathsf{alt}(f) \cdot \mathsf{alt}(g)$ times in the chain B. □

For $f = \vee_n$ and g being a parity on m bits for any odd integer m, $\mathsf{alt}(f \circ g) \geq mn$ by Lemma 3.5 while $\mathsf{alt}(f) \cdot \mathsf{alt}(g) = m$. Thus, in general, it is not true that $\mathsf{alt}(f \circ g) \leq \mathsf{alt}(f) \cdot \mathsf{alt}(g)$ and hence Lemma 3.5 is not tight. Using Lemma 3.5, we prove the following Corollary.

Corollary 3.6. *For any $h : \{0,1\}^n \to \{0,1\}$, with $h(0^n) \neq h(1^n)$, for any $k \geq 2$, $\mathsf{alt}(h^{\circ k}) \geq \mathsf{alt}(h)^k$.*

Proof. By induction on k. For $k = 2$, applying Lemma 3.5 with $f = g = h$, we get that $\mathsf{alt}(h \circ h) \geq \mathsf{alt}(h)^2$. Now for $k > 2$, applying Lemma 3.5 with $f = h^{\circ k-1}$ and $g = h$ and by inductive hypothesis, $\mathsf{alt}(h^{\circ k-1} \circ h) \geq \mathsf{alt}(h^{\circ k-1}) \cdot \mathsf{alt}(h) \geq \mathsf{alt}(h)^{k-1} \cdot \mathsf{alt}(h)$. Hence $\mathsf{alt}(h^{\circ k}) = \mathsf{alt}(h^{\circ k-1} \circ h) \geq \mathsf{alt}(h)^k$. □

We use Corollary 3.6 to exhibit a family of Boolean functions for which alternation is super-linear in sensitivity.

Theorem 1.2 *There exists a family of Boolean functions $\mathcal{G} = \{g_k : \{0,1\}^{n_k} \to \{0,1\} \mid k \in \mathbb{N}\}$ such that $\mathsf{alt}(g_k) \geq \mathsf{s}(g_k)^{\log_3 5}$ while $\mathsf{DT}(g_k)$ is $\Omega(n_k^{\log_6 3})$.*

Proof. Consider the address function $\mathsf{ADDR}_t : \{0,1\}^{t+2^t} \to \{0,1\}$ defined as $\mathsf{ADDR}_t(x_1, x_2, \ldots, x_t, y_0 y_1, y_2, \ldots, y_{2^t-1}) = y_{int(x_1 x_2 \ldots x_t)}$ where $int(x)$ is the integer corresponding to the binary string x. Consider the chain $(000000, 001000, 101000, 101010, 111010, 111011, 111111)$. Since, ADDR_2 changes value 5 times along this chain, $\mathsf{alt}(\mathsf{ADDR}_2) \geq 5$ while $\mathsf{s}(\mathsf{ADDR}_2) = 3$. We consider the family of functions $\mathcal{G} = \{g_k \mid k \in N\}$ obtained by composing ADDR_2 k times. Since sensitivity of composed function is at most the product of their sensitivity [29], $\mathsf{s}(g_k) \leq \mathsf{s}(\mathsf{ADDR}_2)^k = 3^k$. Since $g_1 = \mathsf{ADDR}_2$ is 0 on all zero input and 1 on all ones input, applying Corollary 3.6, $\mathsf{alt}(g_k) \geq 5^k \geq \mathsf{s}(g_k)^{\log_3 5}$. Note that $\mathsf{DT}(g_k) = \mathsf{DT}(\mathsf{ADDR}_2)^k$ (as decision tree depth multiplies under composition [29]). Hence $\mathsf{DT}(g_k) = 3^k$ which is $n_k^{\log_6 3}$ where n_k is the number of variables of g_k and hence does not grows logarithmic in n_k. □

Remark 3.7. *We observe that for the family of functions $\mathcal{G} = \{g_k : \{0,1\}^{n_k} \to \{0,1\} \mid k \in \mathbb{N}\}$ of Theorem 1.2, $\log \mathsf{sparsity}(g_k) \leq 2\mathsf{deg}(g_k) = 2 \cdot 3^k$ [27]. In Theorem 1.2, we also showed that $\mathsf{alt}(g_k) \geq 5^k$ which is at least $(0.5 \log \mathsf{sparsity}(g_k))^{\log_3 5}$. Hence the same family also exhibits a super-linear gap between $\mathsf{alt}(g_k)$ and $\log \mathsf{sparsity}(g_k)$.*

4 Xor Log-Rank Conjecture for Bounded Alternation Boolean Functions

In this section, we prove the Xor Log-Rank Conjecture for f when $\mathsf{alt}(f)$ is at most $\mathsf{poly}(\log n)$. Before proceeding, we prove that for all Boolean functions f, $\mathsf{deg}(f) \leq \mathsf{alt}(f)\mathsf{deg}_2(f)\mathsf{deg}_m(f)$ (Eq. (1) from Introduction) in Lemma 4.1. We remark that the case $m = 2$ is already observed by Lin and Zhang (Theorem 14, [20]).

Lemma 4.1. *For any Boolean function $f : \{0,1\}^n \to \{0,1\}$, and $m > 1$,*

$$\mathsf{deg}(f) \leq \mathsf{alt}(f) \cdot \mathsf{deg}_2(f) \cdot \mathsf{deg}_m(f) \tag{1}$$

Proof. Proof of this lemma closely follows the argument of Lin and Zhang (Theorem 14, [20]) Fix any Boolean function f. Buhrman and de Wolf [6] showed that given a certificate of f on 0^n of size $\mathsf{C}(f, 0^n)$ and a polynomial representation of f over reals with degree of $\mathsf{deg}(f)$, $\mathsf{DT}(f) \leq \mathsf{C}(f, 0^n) \cdot \mathsf{deg}(f)$. Their idea is that any certificate of f must set at least one variable in all monomials of maximum degree in the polynomial representation of f. Hence, querying variables in a certificate must reduce the degree of the function by at least once. Observe that the same argument holds even if the polynomials are represented over $\mathbb{Z}_m$. Hence $\mathsf{DT}(f) \leq \mathsf{C}(f, 0^n) \cdot \mathsf{deg}_m(f)$.

Applying a result of Lin and Zhang, who showed that $\mathsf{C}(f, 0^n) \leq \mathsf{alt}(f)\mathsf{deg}_2(f)$ (Lemma 12(2) of [20]), we get that, $\mathsf{DT}(f) \leq \mathsf{alt}(f)\mathsf{deg}_2(f)\mathsf{deg}_m(f)$. Observing that $\mathsf{deg}(f) \leq \mathsf{DT}(f)$ (cf. [6]) completes the argument. □

Applying Lemma 4.1 with $m = 2$, we have for any Boolean function f,

$$\mathsf{deg}(f) \leq \mathsf{alt}(f) \cdot \mathsf{deg}_2(f)^2 \tag{2}$$

As mentioned above, Eq. (2) is implicit in the result of [20]. We now proceed to prove the main result of this section. As a first step towards showing $\mathsf{alt}(f) \leq \mathsf{poly}(\log n)$ implies that the Xor Log-Rank Conjecture holds for f, we prove the following bound on the weighted average of the Fourier coefficients, weighted by the number of elements.

Proposition 4.2. *For an $f : \{0,1\}^n \to \{-1,1\}$ that depends on all its inputs, $\sum_S |\widehat{f}(S)||S| \geq n$*

Proof. It suffices to show that for every $i \in [n]$, $\sum_{S:i\in S} |\widehat{f}(S)| \geq 1$. Fix an $i \in [n]$. Since

$$f(x) = \sum_S \widehat{f}(S)\chi_S(x) = \sum_{S\subseteq[n]\setminus\{i\}} (\widehat{f}(S) + \widehat{f}(S \cup \{i\})(-1)^{x_i}) \prod_{j\in S}(-1)^{x_j}$$

for any $S \subseteq [n] \setminus \{i\}$ and $b \in \{0,1\}$, $\widehat{f|_{x_i=b}}(S) = \widehat{f}(S) + (-1)^b \cdot \widehat{f}(S \cup \{i\})$. Hence we conclude that for any x, $f_{x_i=0}(x) - f_{x_i=1}(x) = \sum_{S\subseteq[n]\setminus\{i\}} 2\widehat{f}(S \cup \{i\}) \prod_{j\in S}(-1)^{x_j}$. Now taking absolute values on both sides and applying triangle inequality,

$$\left|\frac{f_{x_i=0} - f_{x_i=1}}{2}\right| = \left|\sum_{S\subseteq[n]\setminus\{i\}} \widehat{f}(S \cup \{i\}) \prod_{j\in S}(-1)^{x_j}\right| \leq \sum_{S\subseteq[n]\setminus\{i\}} \left|\widehat{f}(S \cup \{i\})\right|$$

Since f is sensitive at i on some input $a \in \{0,1\}^n$, for the input a' obtained by removing i^{th} bit from a, $|f_{x_i=0}(a') - f_{x_i=1}(a')| = 2$ implying $\sum_{S:i\in S} |\widehat{f}(S)| \geq 1$ by the above equation which completes the proof. □

We show that if $\mathsf{alt}(f) \leq \mathsf{poly}(\log n)$, then $\mathsf{deg}(f) \leq \mathsf{poly}(\log \mathsf{sparsity}(f))$. This implies that the Xor Log-Rank Conjecture holds[2] for f. As a first step, using Proposition 4.2, we show that if $\mathsf{deg}(f) \leq \mathsf{poly}(\log n)$, then $\mathsf{deg}(f) \leq \mathsf{poly}(\log \mathsf{sparsity}(f))$ (Lemma 4.3). We then argue using Eq. (2) that $\mathsf{alt}(f) \leq \mathsf{poly}(\log n)$ implies that $\mathsf{deg}(f) \leq \mathsf{poly}(\log n)$ proving Theorem 1.3.

Lemma 4.3. *For an $f : \{0,1\}^n \to \{0,1\}$ which depends on all its inputs and for large enough n, if $\mathsf{deg}(f) \leq (\log n)^c$ for some $c > 0$, then $\mathsf{deg}(f) \leq (\log \mathsf{sparsity}(f))^c$.*

Proof. Since the f depends on all the inputs, applying Proposition 4.2 to $g(x) = 1 - 2f(x)$, $n \leq \sum_S |\widehat{g}(S)||S| \leq \mathsf{deg}(f) \sum_S |\widehat{g}(S)| \leq \mathsf{deg}(f)\sqrt{\mathsf{sparsity}(f)}$. In concluding this, we used the fact that maximum sized index $|S|$ for which $\widehat{g}(S) \neq 0$ is $\mathsf{deg}(f)$ and $\sum_S |\widehat{g}(S)| \leq \sqrt{\mathsf{sparsity}(f)}$ [27]. Thus, $\sqrt{\mathsf{sparsity}(f)} \cdot \mathsf{deg}(f) \geq n$. Since $\mathsf{deg}(f) \leq (\log n)^c$, we have $\sqrt{\mathsf{sparsity}(f)} \geq \frac{n}{(\log n)^c} \geq \sqrt{n}$ for large enough n. Hence $\mathsf{deg}(f) \leq (\log n)^c \leq (\log \mathsf{sparsity}(f))^c$. □

Theorem 1.3 *For large enough n, the Xor Log-Rank Conjecture is true for all $f : \{0,1\}^n \to \{0,1\}$, such that $\mathsf{alt}(f) \leq \mathsf{poly}(\log n)$ where f depends on all its inputs.*

Proof. If $\mathsf{deg}_2(f) = 1$, then f is a parity function and the Xor Log-Rank Conjecture holds for f. Hence we can assume that $\mathsf{deg}_2(f) > 1$. If $\mathsf{alt}(f) \leq \mathsf{deg}_2(f)$, then by Eq. (2), we have $\mathsf{deg}(f) \leq \mathsf{deg}_2(f)^3 \leq \log \mathsf{sparsity}(f)^3$ (since $\mathsf{deg}_2(f) > 1$, $\mathsf{deg}_2(f) \leq \log \mathsf{sparsity}(f)$ [4]). Hence the Xor Log-Rank Conjecture holds for f. If $\mathsf{alt}(f) > \mathsf{deg}_2(f)$, then by Eq. (2), $\mathsf{deg}(f) < \mathsf{alt}(f)^3$. Since $\mathsf{alt}(f) \leq \mathsf{poly}(\log n)$, we have $\mathsf{deg}(f) \leq \mathsf{poly}(\log n)$. Applying Lemma 4.3, we get that $\mathsf{deg}(f) \leq \mathsf{poly}(\log \mathsf{sparsity}(f))$. □

Remark 4.4. *It should be noted that for f satisfying conditions of Theorem 1.3, the Sensitivity Conjecture is true. This is because for f that depends on all its inputs, $\mathsf{s}(f) = \Omega(\log n)$ [28] implying that $\mathsf{alt}(f) \leq \mathsf{poly}(\log n) \leq \mathsf{poly}(\mathsf{s}(f))$. Hence the Sensitivity Conjecture is true for f by the result of Lin and Zhang [20].*

We conclude this section, by giving an alternate proof for the result of Lin and Zhang [20] that if for all Boolean functions f, $\mathsf{alt}(f) \leq \mathsf{poly}(\log \mathsf{sparsity}(f))$, then the Xor Log-Rank Conjecture is true by making use of Eq. (2). Then, we give a trade-off between sensitivity and sparsity of Boolean functions.

Theorem 4.5. *For any Boolean function f, $\mathsf{alt}(f) \leq (\log \mathsf{sparsity}(f))^c$ then Xor Log-Rank Conjecture holds.*

Proof. For any Boolean function f with $\mathsf{alt}(f) \leq (\log \mathsf{sparsity}(f))^c$ we have the following exhaustive cases.

[Case 1 : $\mathsf{alt}(f) = 1$] : When $\mathsf{alt}(f) = 1$, f is unate. Hence the Xor Log-Rank Conjecture holds due to the result of [23].

[Case 2: $\mathsf{deg}_2(f)$ is constant] : When $\mathsf{deg}_2(f)$ is constant, the Xor Log-Rank Conjecture holds due to the result of [30].

[2]Using the facts that $\mathsf{CC}_{\oplus}(f) \leq 2\mathsf{DT}(f)$ [23] and $\mathsf{DT}(f) \leq \mathsf{deg}(f)^4$ [6], $\mathsf{CC}_{\oplus}(f) = O(\mathsf{deg}(f)^4)$ implying $\mathsf{CC}_{\oplus}(f) = O(\mathsf{poly}(\log \mathsf{sparsity}(f))$.

[**Case 3 :** $1 < \mathsf{alt}(f) < \mathsf{deg}_2(f)$] : For this case, when $\mathsf{alt}(f) < \mathsf{deg}_2(f)$, by Eq. (2), $\mathsf{deg}(f) \leq \mathsf{deg}_2(f)^3$ which is upper bounded by $(\log \mathsf{sparsity}(f))^3$ (since $\mathsf{deg}_2(f) > 1$, $\mathsf{deg}_2(f) \leq \log \mathsf{sparsity}(f)$ [4]). Thus $\mathsf{deg}(f) \leq (\log \mathsf{sparsity}(f))^3$. On the other hand, $\mathsf{CC}_{\oplus}(f) \leq 2\mathsf{DT}(f)$ [23]. Also it is known that $\mathsf{DT}(f) \leq \mathsf{deg}(f)^4$ [6]. Hence $\mathsf{CC}_{\oplus}(f) = O(\mathsf{deg}(f)^4)$ giving $\mathsf{CC}_{\oplus}(f) = O((\log \mathsf{sparsity}(f))^{12})$.

[**Case 4 :** $\mathsf{deg}_2(f) \leq \mathsf{alt}(f) \leq (\log \mathsf{sparsity}(f))^c$] : we have $\mathsf{CC}_{\oplus}(f) = O(\mathsf{deg}(f)^{\ell}) \leq O(\mathsf{alt}(f)^{3\ell})$ where the last inequality follows by Eq. (2) in this case. By assumption, $\log \mathsf{sparsity}(f) \geq alt(f)^{1/c}$. Thus $\mathsf{CC}_{\oplus}(f) = O(\log \mathsf{sparsity}(f))^{3\ell/c})$.

□

From Proposition 4.2, we derive a trade-off between sensitivity and sparsity of Boolean functions.

Corollary 4.6. *For any Boolean function f which depends on all its n bits,* $\mathsf{s}(f)\sqrt{\mathsf{sparsity}(f)} \geq n$

Proof. Applying Cauchy-Schwartz inequality to $\sum_S |\widehat{f}(S)||S| \geq n$ from Proposition 4.2, we get

$$\sum_S |\widehat{f}(S)||S| \leq \sqrt{\sum_S |S|^2 \widehat{f}(S)^2 \cdot \mathsf{sparsity}(f)} \leq \mathsf{s}(f)\sqrt{\mathsf{sparsity}(f)}$$

The last inequality follows since $\sum_S |S|^2 \widehat{f}(S)^2 = \frac{1}{2^n} \sum_x \mathsf{s}(f,x)^2 \leq \mathsf{s}(f)^2$ (cf. [27]). □

5 Three Further Applications of the deg vs deg_2 Relation

We showed that for all Boolean functions f, $\mathsf{deg}(f) \leq \mathsf{alt}(f)\mathsf{deg}_2(f)^2$ (Eq. (2)) in Section 4. We now give three applications of this result. Firstly, we shows that Boolean functions of bounded alternation must have $\mathbb{F}_2$ degree $\Omega(\log n)$. Secondly, we partially answer a question raised by Kulkarni and Santha [19] on the sparsity of monotone Boolean functions by show a variant of their statement. Thirdly, we observe that Eq. (2) improves a bound on $\mathbf{I}[f]$ due to Guo and Komargodski [14].

$\mathbb{F}_2$-degree of Bounded Alternation Functions : For an n bit monotone Boolean function f, we have $\mathsf{deg}(f) \leq \mathsf{deg}_2(f)^2$. If f depends on all its n bits then $\mathsf{deg}(f) \geq \log n - O(\log \log n)$ [26]. This implies that for monotone functions, $\mathsf{deg}_2(f) \geq \Omega(\sqrt{\log n})$. We now present a short argument improving this bound using Eq. (2).

Suppose $f : \{0,1\}^n \to \{0,1\}$ and it depends on all its input bits. Gopalan et al. [12] showed that for any such Boolean function, $\mathsf{deg}(f) \geq n/2^{\mathsf{deg}_2(f)}$. Along with Eq. (2), this implies that for an f with $\mathsf{alt}(f) \leq n^{\epsilon}$ where $0 < \epsilon < 1$, $\mathsf{deg}_2(f)2^{\mathsf{deg}_2(f)} \geq n^{1-\epsilon}$. Hence, $\mathsf{deg}_2(f) \geq (1-\epsilon)\log n - 2(\log\left(\frac{1-\epsilon}{2}\log n\right))$. This gives us the following corollary.

Corollary 5.1. *Fix any $0 < \epsilon < 1$. Let $f : \{0,1\}^n \to \{0,1\}$ be such that $\mathsf{alt}(f) \leq n^{\epsilon}$ and it depends on all its inputs. Then,*

$$\mathsf{deg}_2(f) \geq (1-\epsilon)\log n - O\left(\log\left(\frac{1-\epsilon}{2}\log n\right)\right)$$

Hence, Boolean functions whose alternation is at most n^{ϵ}, cannot have a constant $\mathbb{F}_2$-degree. However, this need not be the case if we allow $\mathsf{alt}(f)$ to be n (for instance, parity on n bits).

Fourier Spectrum for Bounded Alternation Functions : Kulkarni and Santha [19] studied certain special Boolean functions which are indicator functions $f_{\mathcal{M}}$ of a bridgeless matroids $\mathcal{M}$ on ground set $[n]$. While it is known that for any f, $\log \mathsf{sparsity}(f) \leq 2 \cdot \mathsf{deg}(f)$ [27], Kulkarni and Santha showed that this upper bound is asymptotically tight for $f = f_{\mathcal{M}}$. They observed that $f_{\mathcal{M}}$ is a monotone function (by virtue of the underlying support set being a matroid) and asked if a similar statement holds for the general class of monotone Boolean functions. More precisely, they asked whether $\log \mathsf{sparsity}(f) = \Omega(\mathsf{deg}(f))$ for every monotone Boolean function f.

We show that for functions of constant alternation (which includes monotone functions), $\log \mathsf{sparsity}(f)$ is relatively large. This can be seen as a variant of the question posed by Kulkarni and Santha.

Theorem 1.4 *For Boolean functions f with $\mathsf{alt}(f) = O(1)$, $\log \mathsf{sparsity}(f) = \Omega(\sqrt{\mathsf{deg}(f)})$.*

Proof. Observe that for $\mathsf{deg}_2(f) = 1$, f is parity of constant number of variables or its negation as $\mathsf{alt}(f) = O(1)$. Hence the result holds for this case. For $\mathsf{deg}_2(f) > 1$, by Eq. (2) and the result that $\mathsf{deg}_2(f) \leq \log \mathsf{sparsity}(f)$ when $\mathsf{deg}_2(f) > 1$ [4], we get that $\mathsf{deg}(f) = O(\mathsf{deg}_2(f)^2) = O(\log \mathsf{sparsity}(f)^2)$. □

Note that Theorem 1.4 does show that logarithm of sparsity of monotone functions is nearly close to the upper bound possible but does not completely settles the question of Kulkarni and Santha.

Improved Upper Bound for $\mathbf{I}[f]$: For an n bit Boolean function, the best known upper bound of $\mathbf{I}[f]$ in terms of $\mathsf{alt}(f)$ is $\mathbf{I}[f] \leq O(\mathsf{alt}(f)\sqrt{n})$ due to Guo and Komargodski [14] using a probabilistic argument. In Appendix A.2 we give a simpler proof using a recent characterization of alternation due to Blais et al. [5]. Since $\mathbf{I}[f] \leq \mathsf{deg}(f)$ [27], Eq. (2) gives an improvement over the known bound on $\mathbf{I}[f]$ when $\mathsf{deg}_2(f) < \sqrt[4]{n}$.

Proposition 5.2. *For any $f : \{0,1\}^n \to \{0,1\}$, $\mathbf{I}[f] \leq \mathsf{alt}(f) \cdot \mathsf{deg}_2(f)^2$.*

This immediately gives improved learning algorithms for functions of bounded alternation in the PAC learning model.

Blais et al. [5] gave a uniform learning algorithm for the class of functions C_t computable by circuits with at most t negations that can learn an $f \in C_t$ from random examples with error $\epsilon > 0$ in time $n^{O(2^t\sqrt{n})/\epsilon}$ where $t \leq O(\log n)$. In terms of alternation, the runtime is $n^{O(\mathsf{alt}(f)\sqrt{n})/\epsilon}$. The main tool used in this area is the low degree learning algorithm due to Linial, Mansour and Nisan [21] using which the following result is derived in [27].

Lemma 5.3 (Corollary 3.22 and Theorem 3.36 [27])**.** *For $t \geq 1$, let*

$$\mathcal{A}_t = \{f \mid f : \{-1,1\}^n \to \{-1,1\}, \mathbf{I}[f] \leq t\}$$

and

$$\mathcal{B}_t = \{f \mid f : \{-1,1\}^n \to \{-1,1\}, \mathsf{deg}(f) \leq t\}$$

Then A_t can be learned from random examples with error ϵ in time $n^{O(t/\epsilon)}$ for any $\epsilon \in (0,1)$ and B_t can be exactly learned from random examples in time $n^t\mathsf{poly}(n, 2^t)$.

The claimed result follows from Proposition 5.2. Applying Lemma 5.3, we obtain

- an exact learning algorithm from random examples with a runtime of $n^{O(k)}\mathsf{poly}(n, 2^{O(k)})$ for $k = \mathsf{alt}(f)\mathsf{deg}_2(f)^2$ and

- an ϵ error learning algorithm from random examples with a runtime $n^{O(\mathsf{alt}(f)\mathsf{deg}_2(f)^2/\epsilon)}$

thereby removing the dependence on the parameter n in the exponent and improving the runtime for those f such that $\mathsf{deg}_2(f) < \sqrt[4]{n}$.

6 Discussion and Open Problems

In this paper, we showed a limitation of alternation as a Boolean function parameter in settling the Sensitivity Conjecture and Xor Log- Rank Conjecture. In spite of this limitation, we derived that both the above conjectures are true for functions whose alternation is upper bounded by $\mathsf{poly}(\log n)$. En route the proof, we showed that the degree can be upper bounded in terms of $\mathbb{F}_2$ degree and alternation (Eq. (2)) and demonstrated its use with three applications. In conclusion, we propose the following three directions for further exploration.

Parameter Trade-offs : The family of Boolean functions $\mathcal{F}$ (in Definition 3.1) have the drawback that their decision tree depth is very small while exhibiting an exponential gap between alternation and sensitivity, and alternation and logarithm of sparsity. On the other hand, the family of Boolean functions $\mathcal{G}$ obtained (in Theorem 1.2) have a large decision tree depth but could achieve the same with only a super linear gap between the above mentioned parameters. These two family of functions seems to be at the two extremes ends in terms of the gap achievable and the decision tree depth. Thus, an open problem would be to show a trade-off between the lower bound on the decision tree depth and the gap that can be proven for these parameters.

Monotone functions have dense spectrum : Can we show that for every monotone function f, $\log \mathsf{sparsity}(f) = \Omega(\mathsf{deg}(f))$. Note that Theorem 1.4 shows that $\log \mathsf{sparsity}(f) = \Omega(\sqrt{\mathsf{deg}(f)})$ and hence only partially settle this question of Kulkarni and Santha [19].

Improving upper bound on $\mathsf{deg}(f)$: Eq. (1) says that for any Boolean function f and $m \geq 2$, $\mathsf{deg}(f) \leq \mathsf{alt}(f) \cdot \mathsf{deg}_2(f) \cdot \mathsf{deg}_m(f)$. Note that the upper bound is weak if $\mathsf{deg}_2(f)$ is large. Hence, we ask if it is possible to improve Eq. (1) by showing $\mathsf{deg}(f) \leq \mathsf{alt}(f) \cdot \mathsf{deg}_m(f) \cdot \mathsf{deg}_{m'}(f)$ for $m \neq m'$.

Acknowledgments

The authors would like to thank the anonymous reviewers for pointing out errors in the earlier version and for providing constructive comments which improved the presentation of the paper.

References

[1] Andris Ambainis, Krišjānis Prūsis and Jevgēnijs Vihrovs. Sensitivity versus certificate complexity of Boolean functions. In *Computer Science - Theory and Applications - 11th International Computer Science Symposium in Russia, CSR 2016, St. Petersburg, Russia, June 9-13, 2016, Proceedings*, pages 16–28, 2016.

[2] Mitali Bafna, Satyanarayana V. Lokam, Sébastien Tavenas, and Ameya Velingker. On the sensitivity conjecture for read-k formulas. In *41st International Symposium on Mathematical Foundations of Computer Science, MFCS 2016 - Kraków, Poland*, pages 16:1–16:14, 2016.

[3] Robert Beals, Harry Buhrman, Richard Cleve, Michele Mosca, and Ronald de Wolf. Quantum lower bounds by polynomials. *Journal of the ACM*, 48(4):778–797, 2001.

[4] Anna Bernasconi and Bruno Codenotti. Spectral analysis of Boolean functions as a graph eigenvalue problem. *IEEE Transactions on Computers*, 48(3):345–351, 1999.

[5] Eric Blais, Clément L. Canonne, Igor Carboni Oliveira, Rocco A. Servedio, and Li-Yang Tan. Learning circuits with few negations. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2015, August 24-26, 2015, Princeton, NJ, USA*, pages 512–527, 2015.

[6] Harry Buhrman and Ronald de Wolf. Complexity measures and decision tree complexity: a survey. *Theoretical Computer Science*, 288(1):21–43, 2002.

[7] Karthik C. S. and Sébastien Tavenas. On the sensitivity conjecture for disjunctive normal forms. In *36th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2016, Chennai, India*, pages 15:1–15:15, 2016.

[8] Sourav Chakraborty. On the sensitivity of cyclically-invariant Boolean functions. *Discrete Mathematics & Theoretical Computer Science*, 13(4):51–60, 2011.

[9] Stephen Cook, Cynthia Dwork, and Rdiger Reischuk. Upper and lower time bounds for parallel random access machines without simultaneous writes. *SIAM Journal on Computing*, 15(1):87–97, 1986.

[10] E. I. Nechiporuk. On the complexity of schemes in some bases containing non-trivial elements with zero weights. *Problemy Kibernetiki*, 8:123–160, 1962.

[11] Justin Gilmer, Michal Koucký, and Michael E. Saks. A new approach to the sensitivity conjecture. In *Proceedings of the 2015 Conference on Innovations in Theoretical Computer Science*, ITCS '15, pages 247–254, New York, NY, USA, 2015. ACM.

[12] Parikshit Gopalan, Shachar Lovett, and Amir Shpilka. On the complexity of Boolean functions in different characteristics. In *Proceedings of the 24th Annual IEEE Conference on Computational Complexity, CCC 2009, Paris, France, 15-18 July 2009*, pages 173–183, 2009.

[13] Parikshit Gopalan, Rocco A. Servedio, and Avi Wigderson. Degree and sensitivity: Tails of two distributions. In *31st Conference on Computational Complexity, CCC 2016, Tokyo, Japan*, pages 13:1–13:23, 2016.

[14] Siyao Guo and Ilan Komargodski. Negation-limited formulas. *Theoretical Computer Science*, 660:75–85, 2017.

[15] Pooya Hatami, Raghav Kulkarni, and Denis Pankratov. *Variations on the Sensitivity Conjecture*. Number 4 in Graduate Surveys. Theory of Computing Library, 2011.

[16] Jen-Chun Chang and Hsin-Lung Wu. The Log-Rank Conjecture for Read-k XOR Functions. *Journal of Information Science and Engineering*, 34(2):To appear, March 2018.

[17] Stasys Jukna. *Boolean function complexity : Advances and Frontiers*. Algorithms and combinatorics. Springer, Berlin, Heidelberg, 2012.

[18] Claire Kenyon and Samuel Kutin. Sensitivity, block sensitivity, and ℓ-block sensitivity of Boolean functions. *Information and Computation*, 189(1):43 – 53, 2004.

[19] Raghav Kulkarni and Miklos Santha. Query complexity of matroids. In *Algorithms and Complexity, 8th International Conference, CIAC 2013, Barcelona, Spain, May 22-24, 2013. Proceedings*, pages 300–311, 2013.

[20] Chengyu Lin and Shengyu Zhang. Sensitivity Conjecture and Log-Rank Conjecture for Functions with Small Alternating Numbers. In *44th International Colloquium on Automata, Languages, and Programming (ICALP 2017)*, volume 80, pages 51:1–51:13, Dagstuhl, Germany, 2017.

[21] Nathan Linial, Yishay Mansour, and Noam Nisan. Constant depth circuits, fourier transform, and learnability. *Journal of the ACM*, 40(3):607–620, 1993.

[22] A. A. Markov. On the inversion complexity of a system of functions. *Journal of the ACM*, 5(4):331–334, October 1958.

[23] Ashley Montanaro and Tobias Osborne. On the communication complexity of XOR functions. *CoRR*, abs/0909.3392, 2009.

[24] Hiroki Morizumi. Limiting negations in formulas. In *Automata, Languages and Programming, 36th International Colloquium, ICALP 2009, Rhodes, Greece, July 5-12, 2009, Proceedings, Part I*, pages 701–712, 2009.

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

[26] Noam Nisan and Mario Szegedy. On the degree of Boolean functions as real polynomials. In *Proceedings of the 24th Annual ACM Symposium on Theory of Computing*, STOC '92, pages 462–467, New York, NY, USA, 1992. ACM.

[27] Ryan O'Donnell. *Analysis of Boolean Functions*. Cambridge University Press, New York, NY, USA, 2014.

[28] Hans-Ulrich Simon. A tight $\omega(\log \log n)$-bound on the time for parallel ram's to compute nondegenerated Boolean functions. In *Proceedings of the International Conference on Fundamentals of Computation Theory, FCT*, pages 439–444, London, UK, UK, 1983. Springer-Verlag.

[29] Avishay Tal. Properties and applications of Boolean function composition. In *Proceedings of the 4th Conference on Innovations in Theoretical Computer Science*, ITCS '13, pages 441–454, New York, NY, USA, 2013. ACM.

[30] Hing Yin Tsang, Chung Hoi Wong, Ning Xie, and Shengyu Zhang. Fourier sparsity, spectral norm, and the log-rank conjecture. In *54th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2013, 26-29 October, 2013, Berkeley, CA, USA*, pages 658–667, 2013.

[31] Gyorgy Turán. The critical complexity of graph properties. *Information Processing Letters*, 18(3):151 – 153, 1984.

[32] Zhiqiang Zhang and Yaoyun Shi. Communication complexities of symmetric XOR functions. *Quantum Information & Computation*, 9(3):255–263, 2009.

[33] Zhiqiang Zhang and Yaoyun Shi. On the parity complexity measures of Boolean functions. *Theoretical Computer Science*, 411(26-28):2612–2618, 2010.

A Appendix

A.1 Second Proof of Exponential Upper Bound on Alternation

Nechiporuk [10], also discovered independently by Morizumi [24], related the decrease of a function f to the minimum number of negations needed in any formula computing f.

Theorem A.1 (Nechiporuk, Morizumi)**.** *Let* $\mathsf{negs}_F(f)$ *be the minimum number of negations needed in any formula computing* f*. Then* $\mathsf{negs}_F(f) = \mathsf{dc}(f)$*.*

We now give a weaker version (in terms of the additive constant) of Theorem 3.4.

Theorem A.2. *For any Boolean function* f*,* $\mathsf{alt}(f) \leq 2^{DT(f)+1} + 1$

Proof. Since $\mathsf{alt}(f) \leq 2\mathsf{dc}(f) + 1$, it suffices to show that $\mathsf{dc}(f) \leq 2^{\mathsf{DT}(f)}$ to complete the proof. Let T be a decision tree of depth $\mathsf{DT}(f)$ computing f. Given such a decision tree T, we can always obtain a formula computing f with at most $2^{\mathsf{DT}(f)}$ leaves. Hence, $\mathsf{negs}_F(f)$ is upper bounded by $2^{\mathsf{DT}(f)}$ as all internal negations can be pushed to the leaves. Applying Theorem A.1, we get that $\mathsf{dc}(f) \leq 2^{\mathsf{DT}(f)}$ which completes the proof. □

A.2 Simpler Proof of $\mathbf{I}[f] \leq \mathsf{alt}(f)\sqrt{n}$

Suppose $f : \{0,1\}^n \to \{0,1\}$ is a parity of k monotone functions. Hence, in any chain from 0^n to 1^n in $\mathcal{B}_n$, f changes its value at most k times. Blais et al. [5] recently showed that the converse of the statement is also true.

Proposition A.3 (Characterization of Alternation [5])**.** *Let* $f : \{0,1\}^n \to \{0,1\}$*. Then there exists* $k = \mathsf{alt}(f)$ *monotone functions* $g_1, \ldots, g_k$ *each from* $\{0,1\}^n$ *to* $\{0,1\}$ *such that*

$$f(x) = \begin{cases} \oplus_{i=1}^{k} g_i & \text{if } f(0^n) = 0 \\ \neg \oplus_{i=1}^{k} g_i & \text{if } f(0^n) = 1 \end{cases}$$

Let f be such that $\mathbf{I}[f] = \Omega(n)$. Observe (cf. [27]) that an $\Omega(n)$ lower bound on influence implies that a *constant* fraction of edges in the Boolean hypercube must have f evaluating to different values on their end points. Blais et al. [5] used this idea and a probabilistic argument to prove that the expected alternation in a random chain of the hypercube must be large – inferring that $\mathsf{alt}(f) = \Omega(\sqrt{n})$. Later Guo and Komargodski [14], again using a similar probabilistic argument, generalized it to show that $\mathbf{I}(f) = O(\mathsf{alt}(f)\sqrt{n})$.

We now give a simpler proof of the statement $\mathbf{I}[f] \leq O(\mathsf{alt}(f)\sqrt{n})$.

Lemma A.4. *For any* n *variable Boolean function* f*,* $\mathbf{I}[f] \leq \mathsf{alt}(f)\sqrt{n}$*.*

Proof. For any Boolean function f_1, f_2 on n bits and for any $x \in \{0,1\}^n$, it is easy to see that $\mathsf{s}(f_1 \oplus f_2, x) \leq \mathsf{s}(f_1, x) + \mathsf{s}(f_2, x)$ (cf. Lemma 28 of [7]). Hence,

$$\mathbf{I}[f_1 \oplus f_2] = \mathbf{E}_x(\mathsf{s}(f_1 \oplus f_2, x)) \leq \mathbf{E}_x(\mathsf{s}(f_1, x)) + \mathbf{E}_x(\mathsf{s}(f_2, x)) = \mathbf{I}[f_1] + \mathbf{I}[f_2] \tag{3}$$

Suppose $f(0^n) = 0$ and $\mathsf{alt}(f) = k$. Applying Proposition A.3, we get that there exists k monotone functions, $g_1, g_2, \ldots, g_k$ all on n variables such that $f(x) = g_1(x) \oplus g_2(x) \ldots \oplus g_k(x)$. Repeatedly applying Eq. (3) on f, we get that $\mathbf{I}[f] \leq \sum_{i=1}^{k} \mathbf{I}[g_i]$. Since g_is are monotone, $\mathbf{I}[g_i] \leq \sqrt{n}$ [27]. Hence $\mathbf{I}[f] \leq k\sqrt{n} = \mathsf{alt}(f)\sqrt{n}$. For the case when $f(0^n) = 1$ in the decomposition of Proposition A.3 all g_is will still be monotone except one which will be negation of a monotone function. But even for such functions influence is upper bounded by $\sqrt{n}$ [27]. Hence a similar argument holds in this case too. □

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il