Sensitivity of Boolean Functions, Abstract Harmonic Analysis, and Circuit Complexity *

A. Bernasconi† B. Codenotti‡

TR-93-030

June 1993

Abstract

We exploit the notion of sensitivity of Boolean functions to find complexity results. We first analyze the distribution of the average sensitivity over the set of all the n-ary Boolean functions, and show some applications of this analysis. We then use harmonic analysis on the cube to study how the average sensitivity of a Boolean function *propagates* if the function corresponds, e.g., to an oracle available to compute another function. To do this, we analyze the sensitivity of the composition of Boolean functions. We find the relation existing between a complexity measure for symmetric functions introduced in [FKPS 85] and the average sensitivity. We use this relation to prove that symmetric functions in AC^0 have exponentially decreasing average sensitivity. This is, in the special case of symmetric functions, a substantial improvement over the characterization given in [LMN 89].

---

*This work has been partly supported by the Italian National Research Council, under the "Progetto Finalizzato Sistemi Informatici e Calcolo Parallelo", subproject 2 "Processori dedicati".

†Istituto di Elaborazione dell'Informazione, Consiglio Nazionale delle Ricerche, Pisa (Italy). Supported by the Consorzio Pisa Ricerche. E-mail: bernasconi@iei.pi.cnr.it

‡International Computer Science Institute, Berkeley CA, and Istituto di Elaborazione dell'Informazione, Consiglio Nazionale delle Ricerche, Pisa (Italy). E-mail: brunoc@icsi.berkeley.edu or codenotti@iei.pi.cnr.it

1 Introduction

The Fourier transform of a Boolean function is a linear mapping of the values of the function onto a set of coefficients, known as Fourier coefficients. The nature of this transformation makes the Fourier coefficients informative about the *regularities* of the function, and thus about its computational complexity. This fact was well known since the early ages of *switching function theory*. In a review article of 1971 [Leh 71], Robert Lechner says that *"The representation of a switching functions as an n-dimensional abstract Fourier transform over the finite two-element field (...) has many valuable properties. These properties have inspired new algorithms for some classical problems of combinational logic synthesis ..."*

An important quantity related to the Fourier transform is the *sensitivity* which measures if the value of a function is likely to change in correspondence to arguments which are Hamming neighbors. The goal of this paper is to study Boolean functions by using information on their (average) sensitivity. As we will see, there have been only very preliminary results on this subject, and this is the first systematic approach to using the notion of sensitivity as a tool for a computational complexity investigation.

1.1 Overview of known results

In this section we review some results related to our work. Fourier analysis of a Boolean function allows one to evaluate the influence that a given subset of the variables has in the determination of the value of the function, where the influence of a subset of variables is the probability that the value of the function remains undetermined as long as the variables in the subset are not assigned values. Since the sum of the influences of all the variables defines the average sensitivity, Fourier analysis can be used to determine how much a function is sensitive to its arguments.

Chapter 2 in [SHB 68] is dedicated to the description of some mathematical background on error detection in digital machines. In particular, the *Boolean difference* function - which is the difference between the value of a function at a given argument w and the value at an argument which is a Hamming neighbor of w - is used to analyze error propagation. It turns out that the Boolean difference allows to define both influence and sensitivity. Karpovsky [Kar 76] proposes to use the number of nonvanishing Fourier coefficients of a Boolean function f as a measure of its complexity. Hurst et al. [HMM 82] relate the circuit complexity of a Boolean function to its power spectrum coefficients. Brandman et al. [BOH 90] establish a relationship between the Fourier coefficients of a Boolean function f and (i) the average size of any decision tree for f; (ii) the minimum number of $\wedge$ gates in a circuit computing f according to its disjuntive normal form. Kahan et al. [KKL 88] find connections between influence and harmonic analysis and use theorems on influence to prove results on rapidly mixing Markov chains. In addition, they relate the average sensitivity of functions to their Fourier coefficients. However this relation was already implicit from the work of [HMM 82]. Ben-Or and Linial [BL 89] study collective coin flipping, where the collective coin is viewed as a Boolean function. In this case, measuring influence corresponds to measuring how much the collective coin is sensitive to the presence of faults and the goal is to find Boolean functions on which the influence of each variable is as small as possible, to prevent that a small subset of variables, e.g. the set of faulty processors, takes control of the collective coin. Linial et al. [LMN 89] take advantage of the relation between the average sensitivity of Boolean functions and their Fourier transform to prove several facts, e.g., that sets in AC^0 have low average sensitivity. Bruck [Bru 90] and Bruck and Smolensky [BS 92] use abstract harmonic analysis to derive necessary and sufficient conditions for a Boolean function to be a polynomial threshold.

1.2 Results of this paper

We first analyze the distribution of the average sensitivity and show that almost all n-ary Boolean functions have average sensitivity in the vicinity of $\frac{n}{2}$ (see Lemma 1 in Section 3). This fact allows us to prove that at most a fraction of order $\frac{1}{n2^n}$ of all the n-ary Boolean functions belongs to AC^0. The symmetry of the distribution of the sensitivity and the corresponding spectral symmetry (see Lemma 3 in Section 3) are the basis to propose a natural extension of the class AC^0. We then use harmonic analysis of the Boolean cube to study how the sensitivity of a Boolean function *propagates* if the function is used, e.g., as an oracle, in the computation of another function. To do this, we analyze the composition of Boolean functions. More precisely, we find the linear transformation relating the Fourier coefficients of a Boolean function f to the Fourier coefficients of a Boolean function $h = f(g_1, \ldots, g_m)$, where the g_i's are Boolean functions. We give an exact evaluation of the norm of the matrix of the transformation (see Lemma 6 and Lemma 7 in Section 4). We also find upper bounds on the sensitivity of h with respect to the sensitivity of f and of g_i (see Theorems 8 and 9 in Section 4). This technique is particularly amenable to find lower bounds on the size of small depth circuits as a function of the sensitivity of the functions they compute, to analyze relativized complexity classes and to find properties of sets that reduce to sets with given sensitivity. We find the relation existing between a complexity measure for symmetric functions introduced by [FKPS 85] and the average sensitivity (see Lemma 11 in Section 5). We use this relation to prove that the average sensitivity of symmetric functions in AC^0 decreases exponentially (see Theorem 12 in Section 5). This is, in the special case of symmetric functions, a substantial improvement over the characterization given in [LMN 89]. We also prove that a family of Boolean functions has exponentially decreasing sensitivity if and only if the associated set is almost sparse or co-sparse. This allows us to conclude that sets in AC^0 whose characteristic function is symmetric are almost sparse or co-sparse. By applying counting arguments, we then prove that there are no more than $n^{polylogn}$ symmetric functions in AC^0. This confirms the intuition that almost all the 2^n symmetric functions of n variables are in $NC^1 - AC^0$. The result of Theorem 12, together with the characterization of [WWY 92], give a very clear *picture* of the very simple structure of symmetric functions in AC^0. We show that sets with a given sensitivity can not be complete in NC^1 under certain special reductions. In particular we use the notion of sensitivity to find another proof of the fact that $MAJORITY$ is not complete for NC^1 under *projections* (see Section 6). We furthermore give a technique for evaluating the average sensitivity of functions computable by read-once formulas. The idea is to take advantage of the independence of the variables to obtain very simple expressions for the average sensitivity. This special case is important because, e.g., every NC^1 function can be transformed, by projection, onto a read-once function.

1.3 Notations

Unless otherwise specified, the indexing of vectors and matrices starts from 0 rather than 1. The symbol e_1 denotes the first column of the identity matrix. A^T denotes the transpose of a matrix A. $\rho(B)$ denotes the spectral radius of a matrix B, i.e. the largest of the absolute values of the eigenvalues of B. The notation $\|x\|$ ($\|B\|$) without any subscript stands for the L_2-norm of a vector x (matrix B). The subscript 1 is used to specify the L_1-norm of vectors and matrices. All the logarithms are to the base 2. The notation *polylogn* stands for a function growing like a polynomial in the logarithm of n. Given a Boolean function f on k binary variables, we will often use its 2^k-tuple vector representation $f = (f_0\, f_1 \ldots f_{2^k-1})$, where $f_i = f(x(i))$ and $x(i)$ is the binary expansion of i. If x and y are two binary strings of the same length, then $d(x,y)$ and $x \oplus y$ denote their Hamming distance and the string obtained by computing the *exclusive or* of the bits of x and y, respectively. $|x|$ denotes the number of ones in a binary string x.

1.4 Organization of this paper

The rest of this paper is organized as follows. In section 2 we give some background on abstract harmonic analysis and its connections to sensitivity analysis. In section 3 we give a classification of Boolean functions according to their average sensitivity and we show that the corresponding spectral analysis suggests a natural generalization of the class AC^0. In section 4 we show how to evaluate the sensitivity of *composed* Boolean functions by analyzing their Fourier coefficients. In section 5 we find the connections between *average sensitivity*, sparseness and complexity, and we use these relations to prove some complexity results. In section 6 we present some applications of our results to complexity classes and reductions, and we study the special case of Boolean functions computable by read-once formulas. In section 7 we provide a framework for future research.

2 Abstract harmonic analysis and sensitivity

In this section we give some background on abstract harmonic analysis. Our main sources are [Leh 71] and [Loo 53]. Then we show, using a simple derivation based on [HMM 82], the relation between Fourier coefficients and sensitivity. At the end of the section we take into account the special cases of symmetric and monotone functions.

Let us consider the space $\mathcal{F}$ of all the two-valued functions on $\{0, 1\}^n$. The domain of $\mathcal{F}$ is a locally compact Abelian group and the elements of its range, i.e. 0 and 1, can be added and multiplied as complex numbers. The above properties allow one to analyze $\mathcal{F}$ by using tools from harmonic analysis. This means that it is possible to construct an orthogonal basis set of Fourier transform kernel functions for $\mathcal{F}$. The kernel functions of the Fourier transform are defined in terms of a group homomorphism from $\{0, 1\}^n$ to the direct product of n copies of the multiplicative subgroup $\{\pm 1\}$ on the unit circle of the complex plane. The functions $Q_w(x) = (-1)^{w_1 x_1}(-1)^{w_2 x_2} \ldots (-1)^{w_n x_n} = (-1)^{w^T x}$ are known as *group characters* or Fourier transform kernel functions [Lit 40]. The set of functions $\{Q_w | w \in \{0, 1\}^n\}$ is an orthogonal basis for $\mathcal{F}$.

We can now define the *Abstract Fourier Transform* of a Boolean function f as the rational valued function f^* which defines the coordinates of f with respect to the basis $\{Q_w(x), w \in \{0, 1\}^n\}$, i.e., $f^*(w) = 2^{-n} \sum_x Q_w(x) f(x)$. Then $f(x) = \sum_w Q_w(x) f^*(w)$ is the Fourier expansion of f.

Using the binary 2^n-tuple representation for the functions f and f^*, and considering the natural ordering of the n-tuples x and w, one can derive a convenient matrix formulation for the transform pair. Let us consider a $2^n \times 2^n$ matrix H_n whose (i, j)-th entry h_{ij} satisfies $h_{ij} = (-1)^{\underline{i}^T \underline{j}}$, where $\underline{i}^T \underline{j}$ denotes the inner product of the binary expansions of i and j. If $f = [f_0\, f_1 \ldots f_{2^n-1}]^T$ and $f^* = \left[f_0^*\, f_1^* \ldots f_{2^n-1}^*\right]^T$, then, from the fact that $H_n^{-1} = 2^{-n} H_n$, we get $f = H_n f^*$ and $f^* = 2^{-n} H_n f$.

Note that the matrix H_n is the Hadamard symmetric transform matrix [Leh 71] and can be recursively defined as

$$H_1 = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \qquad H_n = \begin{pmatrix} H_{n-1} & H_{n-1} \\ H_{n-1} & -H_{n-1} \end{pmatrix}.$$

We give now an interpretation to the Fourier coefficients. The coefficient f_0^* is the probability that f takes the value 1. The first order coefficients f_j^*, with $j = 2^i$, measure the correlation of the function f with its i-th variable, i.e. $f_{2^i}^* = 2^{-n} \sum_w (-1)^{w_i} f(w)$. The coefficients f_j^*, with $|j| > 1$, measure the correlation between the function and the parity of those bits whose position corresponds to a 1 in the binary expansion of j. There is no correlation if $f_j^* = 0$ and maximum correlation if $|f_j^*| = \frac{1}{2}$. The sign of the coefficient indicates if the correlation is actually with the parity ($f_j^* = -\frac{1}{2}$) or with its complement ($f_j^* = \frac{1}{2}$).

We can now describe the links between harmonic analysis and the notions of *influence* and *sensitivity*. Let $\mathcal{A}$ be a set of variables. The *influence* of $\mathcal{A}$ on f, denoted by $I_f(\mathcal{A})$, is the probability that f remains undetermined as long as the variables in $\mathcal{A}$ are not assigned values and the other variables are assigned at random according to the uniform distribution. The *sensitivity* $s_w(f)$ of a Boolean function f on a string $w \in \{0,1\}^n$ is the number of Hamming neighbors $\hat{w}$ of w such that $f(w) \neq f(\hat{w})$. The *average sensitivity* of f, $s(f)$, is the average of $s_w(f)$ over all $w \in \{0,1\}^n$. $s(f)$ can also be defined as the sum of the influences of all the variables on f.

Sometimes we will use the terminology *sensitivity of a set* as a shortcut for *sensitivity of the characteristic function of the elements of length* n *of a set*. By using the approach of [HMM 82], we now show the connection between average sensitivity and Fourier coefficients. We take advantage of two identities, i.e. *Parseval's identity*: $\sum_v f(v) = 2^n \sum_v (f^*(v))^2$, and *autocorrelation identity*: $\sum_v f(v)f(v \oplus u) = 2^n \sum_v Q_v(u)(f^*(v))^2$, which is a consequence of the orthogonality of the functions $Q_v(u)$.

We obtain, for the sensitivity $s_w(f)$ of a string w,

$$s_w(f) = \sum_{|u|=1} (f(w) - f(w \oplus u))^2 = \sum_{|u|=1} (f(w) + f(w \oplus u) - 2f(w)f(w \oplus u)),$$

from which

$$s_w(f) = nf(w) + \sum_{|u|=1} f(w \oplus u) - 2\sum_{|u|=1} f(w)f(w \oplus u).$$

For the average sensitivity, we get

$$\begin{aligned} s(f) = 2^{-n}\sum_w s_w(f) &= 2^{-n}\sum_w \left(nf(w) + \sum_{|u|=1} f(w \oplus u) - 2\sum_{|u|=1} f(w)f(w \oplus u)\right) \\ &= 2^{-n}\left(n\sum_w f(w) + \sum_{|u|=1}\sum_w f(w \oplus u) - 2\sum_{|u|=1}\sum_w f(w)f(w \oplus u)\right) = \\ &= 2^{-n+1}\left(n\sum_w f(w) - \sum_{|u|=1}\sum_w f(w)f(w \oplus u)\right). \end{aligned}$$

Since we have

$$\begin{aligned} \sum_{|u|=1}\sum_w f(w)f(w \oplus u) &= 2^n \sum_{|u|=1}\sum_w Q_w(u)(f^*(w))^2 = \\ &= 2^n \sum_{i=1}^{n}\sum_w (-1)^{w_i}(f^*(w))^2 = 2^n \sum_w (f^*(w))^2 \sum_{i=1}^{n}(-1)^{w_i} = \\ &= 2^n \sum_w (n - 2|w|)(f^*(w))^2 = n2^n \sum_w (f^*(w))^2 - 2^{n+1}\sum_w |w|(f^*(w))^2, \end{aligned}$$

then, by Parseval's equality, we obtain

$$s(f) = 2^{-n+1}\left(n\sum_w f(w) - n\sum_w f(w) + 2^{n+1}\sum_w |w|(f^*(w))^2\right) = 4\sum_w |w|(f^*(w))^2.$$

Two important special cases are *symmetric* and *monotone* Boolean functions. In these cases, Fourier coefficients and sensitivity can be evaluated in a simplified way.
Symmetric Functions. Let w be a binary string of length n.

$$f^*(w) = 2^{-n} \sum_{k=0}^{n} f_{2^k-1} \sum_{|x|=k} (-1)^{w^T x}; \; s(f) = 4n \sum_{i=1}^{n} \binom{n-1}{i-1} \left(f^*_{2^i-1}\right)^2.$$

Monotone Functions. The n coefficients which are sufficient to determine the sensitivity [KKL 88] can be computed as $f^* = 2^{-n}\hat{H}_n f$, where $\hat{H}_n$ is an $n \times 2^n$ matrix defined as

$$\hat{H}_1 = \begin{pmatrix} 1 & -1 \end{pmatrix}, \qquad \hat{H}_k = \begin{pmatrix} \hat{H}_{k-1} & \hat{H}_{k-1} \\ v_{k-1} & -v_{k-1} \end{pmatrix},$$

where $v_i = (1, 1 \ldots, 1)^T$. Then $s(f) = 2^{-n+1}\|\hat{H}_n f\|_1$.
Threshold Functions. In the case of monotone and symmetric functions, i.e. *thresholds*, we get

$$f^*(w) = 2^{-n} \sum_{k=h}^{n} \sum_{i=0}^{k} (-1)^i \binom{n-|w|}{k-i} \binom{|w|}{i}; \; s(f) = \frac{h}{2^{n-1}} \binom{n}{h},$$

where h is the positive integer which defines the threshold.

3 Distribution of the average sensitivity

We analyze the distribution of the average sensitivity of all Boolean functions defined on $\{0,1\}^n$ and we give an exact evaluation of its expected value and variance. Then we use Chebyshev's inequality to find an upper bound to the number of Boolean functions in AC^0.

First of all, note that, for any function f, $0 \le s(f) \le n$ and that $s(f)$ can assume only rational values.

Lemma 1 *The expected value of the average sensitivity of all the 2^{2^n} Boolean functions of n variables is equal to $\frac{n}{2}$.*

Proof The thesis easily follows from the linearity of the expected value. In fact:

$$\begin{aligned} E[s(f)] &= E\left[\frac{1}{2^n} \sum_{\omega} s_\omega(f)\right] = \frac{1}{2^n} \sum_{\omega} E[s_\omega(f)] = \\ &= \frac{1}{2^n} \sum_{\omega} \sum_{|u|=1} E[f(\omega) + f(\omega \oplus u) - 2f(\omega)f(\omega \oplus u)] = \\ &= \frac{1}{2^n} \sum_{\omega} \sum_{|u|=1} \left(E[f(\omega)] + E[f(\omega \oplus u)] - 2E[f(\omega)f(\omega \oplus u)]\right) = \\ &= \frac{1}{2^n} \sum_{\omega} \sum_{|u|=1} \left(\frac{1}{2} + \frac{1}{2} - 2 \cdot \frac{1}{4}\right) = \frac{n}{2}. \end{aligned}$$

Lemma 2 *The variance of the distribution of the average sensitivity is equal to* $\frac{n}{2^{n+2}}$.

Proof From the definition of variance we get

$$V[s(f)] = E[s^2(f)] - E^2[s(f)].$$

Then, we need evaluate the expected value of $s^2(f)$. First of all, we have

$$\begin{aligned} E[s^2(f)] &= E\left[\frac{1}{2^{2n}}\sum_{\omega}\sum_{\omega'} s_\omega(f)s_{\omega'}(f)\right] = \\ &= \frac{1}{2^{2n}}E\left[\sum_{\omega} s_\omega^2(f) + \sum_{\omega\neq\omega'} s_\omega(f)s_{\omega'}(f)\right] = \\ &= \frac{1}{2^{2n}}\left(\sum_{\omega} E[s_\omega^2(f)] + \sum_{\omega\neq\omega'} E[s_\omega(f)s_{\omega'}(f)]\right). \end{aligned}$$

Since

$$E[s_\omega^2(f)] = \sum_{i=1}^{n} E[d_i^2(\omega)] + \sum_{i\neq j} E[d_i(\omega)d_j(\omega)],$$

where $d_i(\omega) = f(\omega) + f(\omega \oplus i) - 2f(\omega)f(\omega \oplus i)$ and i is a string with the i-th bit equal to 1 and the others equal to 0, by multiplying $d_i(\omega)$ and $d_j(\omega)$ and evaluating the expected values, we obtain

$$E[s_\omega^2(f)] = \frac{n^2}{4} + \frac{n}{4}.$$

In the same way, we find

$$E[s_w(f)s_{\omega'}(f)] = \sum_{i,j} E[d_i(\omega)d_j(\omega')] = \frac{n^2}{4}.$$

Note that $E[d_i(\omega)d_j(\omega')]$ turns out to be equal to $1/4$ even if ω and ω' share a Hamming neighbor, i.e. $\omega \oplus i = \omega' \oplus j$ for some i and j, and even if ω is a Hamming neighbor of ω' , i.e. $\omega = \omega' \oplus j$ for some j (and *viceversa*).

Finally, we get

$$E[s^2(f)] = \frac{1}{2^{2n}}\left[2^n\left(\frac{n^2}{4} + \frac{n}{4}\right) + \left(2^{2n} - 2^n\right)\frac{n^2}{4}\right] = \frac{n}{2^{n+2}} + \frac{n^2}{4}$$

and the thesis follows from $E[s(f)] = \frac{n}{2}$.

Lemma 3 *The distribution of the average sensitivity of all the* 2^{2^n} *Boolean functions of* n *variables is symmetric with respect to the expected value* $\frac{n}{2}$, *i.e.*

$$\sharp\{f : \{0,1\}^n \to \{0,1\} \text{ s.t. } s(f) = \alpha\} = \sharp\{f : \{0,1\}^n \to \{0,1\} \text{ s.t. } s(f) = n - \alpha\}.$$

Proof We show that for any Boolean function f, with average sensitivity $s(f)$, there exists a function g s.t. $s(g) = n - s(f)$. We define g in the following way. For all $\omega \in \{0,1\}^n$

– if $|\omega|$ is odd, then $g(\omega) = 1 - f(\omega)$;

– if $|\omega|$ is even, then $g(\omega) = f(\omega)$.

The function g is such that, for all ω, $s_\omega(g) = n - s_\omega(f)$. In fact for all the strigs u, $|u| = 1$, ω and $\omega \oplus u$ have opposite parities, thus we obtain

$$s_\omega(g) = \sum_{|u|=1} |g(\omega) - g(\omega \oplus u)| =$$
$$= \sum_{|u|=1} (1 - |f(\omega) - f(\omega \oplus u)|) = n - s_\omega(f).$$

Hence

$$s(g) = \frac{1}{2^n} \sum_{\omega} s_\omega(g) = n - s(f).$$

Note that from the two constant functions $f(\omega) = 0$ and $f'(\omega) = 1 - f(\omega) = 1$ for all ω, which are the only two functions with average sensitivity equal to 0, we can use the proof of Lemma 3 to get the two functions with the maximum average sensitivity, i.e., the parity and its complement. The following two Theorems are applications of Chebyshev's inequality, Pr $\{|s(f) - E[s(f)]| \geq \varepsilon\} \leq \frac{V[s(f)]}{\varepsilon^2}$, to the distribution of average sensitivity.

Theorem 4 *For "almost all" Boolean functions on n variables, we have $\frac{n}{2} - \varepsilon \leq s(f) \leq \frac{n}{2} + \varepsilon$, where ε is a constant.*

Proof Follows from the Chebyshev's inequality for a constant ε. We get

$$\Pr \left\{ \left| s(f) - \frac{n}{2} \right| \geq \varepsilon \right\} \leq \frac{n}{2^{n+2}\varepsilon^2}.$$

Thus the number of Boolean function with average sensitivity ranging from $\frac{n}{2} - \varepsilon$ to $\frac{n}{2} + \varepsilon$ is $\Omega\left(2^{2^n}(1 - \frac{n}{2^{n+2}\varepsilon^2})\right)$.

Theorem 5 *The number of Boolean functions on n variables with $s(f) < k$, for $k < \frac{n}{2}$, is $O\left(\frac{2^{2^n}}{n2^n\left(1-\frac{2k}{n}\right)^2}\right)$.*

Proof Follows from the Chebyshev's inequality, with $\varepsilon = \frac{n}{2} - k$.

Since Boolean functions in AC^0 have average sensitivity $s(f) \leq \log^{O(1)} n$ [LMN 89], we can use Theorem 5 to prove that $\frac{2^{2^n}}{n2^n}\left(1 + \frac{4\log^{O(1)} n}{n}\right)$ is an upper bound to the number of functions in AC^0.

The proof of Lemma 3 suggests to define the complexity classes pAC^0 (*parity* AC^0) and sAC^0 (*symmetric* AC^0), where

$$pAC^0 = \{g|g = f \oplus PARITY\,,\ f \in AC^0\}, \quad sAC^0 = AC^0 \cup pAC^0.$$

These classes have the following properties. For any Boolean function $g \in pAC^0$ we have $s(g) = \Omega(n - \log^k n)$, for a constant k, i.e. functions in pAC^0 behave similarly to $PARITY$ or its complement (like functions in AC^0 behave similarly to the two constant functions). sAC^0 is a class which lies slightly above AC^0. In fact $sAC^0 \subset ACC$. In addition, we can use Theorems 2.2 and 2.8 in [Leh 71], to prove that, if $g = f \oplus PARITY$, then $g^*(w) = \frac{1}{2}\delta_{w,0} - \frac{1}{2}\delta_{w\oplus u,0} + f^*(w \oplus u)$, where u is the vector whose entries are all equal to 1, and $\delta_{i,j}$ is the Kronecker delta function. Thus the Fourier coefficients of order $|i|$ of g, $1 \leq |i| < n$, coincide with the Fourier coefficients of order $n - |i|$ of f, while $g_0^* = \frac{1}{2} + f_{2^n-1}^*$, and $g_{2^n-1}^* = -\frac{1}{2} + f_0^*$.

This last symmetric property of the Fourier coefficients allows us to adapt results on the Fourier coefficients of AC^0 functions to pAC^0 functions. As an example, we have that functions in pAC^0 have almost all their power spectrum on the high order coefficients.

More in general, it is interesting to compare the spectrum of a function f with that of a function g defined as $g(w) = f(w) \oplus p_m(w)$, where p_m is the parity of m bits, e.g. the first m bits. From the fact that $p_m(w) = w^T a (mod\ 2)$, where $a^T = [\,1\,,\,1,\ldots\,,\,1\,,0\,,\,\ldots\,,\,0]$, some algebra and the application of Theorem 2.8 in [Leh 71] yield $g^*(w_1\,,\,\ldots\,,w_n) = f^*(\bar{w_1}\,,\,\ldots\,,\bar{w_m}\,,\,w_{m+1}\,,\,\ldots\,,w_n)$, for $w \neq 0$ and $w \neq a$, and $g^*(0) = \frac{1}{2} + f^*(a)$, $g^*(a) = -\frac{1}{2} + f^*(0)$. Thus, all - but those for $w = 0$ and $w = a$ - the Fourier coefficients of f and g are the same, in different order. Some algebraic manipulation yields

$$s(g) = m - s(f) + 8 \sum_{w \neq 0, w \neq a} w^T(u - a)(f^*(w \oplus a))^2,$$

from which $s(g) \geq m - s(f)$.

The application of the above arguments to AC^0 functions gives rise to the definition of the complexity classes $p_iAC^0 = \{g|g = f \oplus p_i\,,\ f \in AC^0\}$, and $cAC^0 = \cup_{i=0}^{n} p_i AC^0$, where $p_0AC^0 = AC^0$. We have $AC^0 \subset cAC^0 \subset ACC$.

4 Evaluation of the Fourier Coefficients

Let $f : \{0\,,1\}^m \rightarrow \{0\,,1\}$ and $g_i\{0\,,1\}^{k_i} \rightarrow \{0\,,1\}$, $i = 1,\ldots,m$. Consider the Boolean function h given by $f(g_1(w_1), g_2(w_2), \ldots, g_m(w_m))$, where w_i is a k_i-tuple of Boolean variables, and k_i is the fan-in of the function g_i. Let f^* and h^* be the vectors of the Fourier coefficients of the functions f and h, respectively. H_i denotes the Hadamard matrix of size 2^i. We say that $h_i \asymp f_j$ if the m-tuple $g_1(w_1)\,g_2(w_2)\,\ldots g_m(w_m)$, where $w_1\,w_2\,\ldots\,w_m = x(i)$, is equal to $x(j)$, and $x(i)$ and $x(j)$ denote the binary expansions of i and j, respectively.

Lemma 6 *Let f, g_i and h be as above. Let $n = \sum_{i=1}^{m} k_i$, and V be a $2^n \times 2^m$ matrix defined as $2^{-n} H_n A H_m$, where A is a $2^n \times 2^m$ matrix whose entries α_{ij} satisfy*

$$\alpha_{ij} = \begin{cases} 1 & \text{if } h_i \asymp f_j \\ 0 & \text{otherwise} \end{cases}$$

Then $h^* = V f^*$.

Proof From the definition of the matrix A, we have $h = Af$. The thesis follows from the fact that $h^* = 2^{-n} H_n h$ and $f^* = 2^{-m} H_m f$.

Note that the columns of the matrix A are mutually orthogonal and thus the matrix $A^T A$ is a $2^m \times 2^m$ diagonal matrix whose (j,j)-th entry a_j is given by $\sum_{i=0}^{2^n-1} Pr\{h_i \asymp f_j\}$.

The matrix V contains all the information on the relationship between the Fourier coefficients of the functions f and h. The next Lemma states an important property of V.

Lemma 7 $\|V\| = \sqrt{2^{m-n}\,\rho(A^T A)} = \sqrt{2^m\, p}$, *where $p = \max_j p_j$, and $p_j = 2^{-n} a_j$.*

Note that p (or a sharp estimate for it) can be easily computed for many functions. Since $2^{-m} \leq p \leq 1$, we have that $\|V\| \leq 2^{\frac{m}{2}}$, which shows that $2^{\frac{m}{2}}$ is the maximum amplification factor for the Fourier coefficients. In addition, note that, if we interpret the set of the functions g_i as a single mapping $g : \{0\,,\,1\}^n \to \{0\,,\,1\}^m$, then a_j is the cardinality of the inverse image of the string $x(j)$ according to g, i.e. $a_j = \sharp\{w \in \{0\,,\,1\}^n | g(w) = x(j)\}$.

We now show two upper bounds on the average sensitivity of h as a function of the average sensitivity of g_i and f, respectively.

Theorem 8 $s(h) \leq \sum_{i=1}^{m} s(g_i)$.

Proof (Sketch) Follows from the definition of sensitivity as a sum of influences of the variables and by using some probabilistic arguments on the propagation of influences from the functions g_i's to the function f.

Theorem 9 $s(h) \leq n 2^{m-n} \rho(A^T A) s(f)$.

Proof Let D_n be a $2^n \times 2^n$ diagonal matrix whose (i,i)-th entry is defined as $|i|^{\frac{1}{2}}$. Then the average sensitivity of h can be written as

$$s(h) \;=\; 4\|D_n\, h^*\|^2 \;=\; 4\|D_n\, V\, f^*\|^2\,.$$

Now, if $\tilde{D}_m^{-1}$ is a $2^m \times 2^m$ diagonal matrix whose (i,i)-th entry is defined as $|i|^{-\frac{1}{2}}$, for $1 \leq i \leq 2^m$ and 0 for $i = 0$, we have that $\tilde{D}_m^{-1}\, D_m \;+\; e_1\, e_1^T \;=\; I$, from which

$$s(h) \;=\; 4\|D_n\, V\,(\,\tilde{D}_m^{-1}\, D_m \;+\; e_1\, e_1^T\,)\, f^*\|^2\,,$$

where D_m is defined as D_n. Since $D_n V e_1 = 0$ and $s(f) = 4\|D_m f^*\|^2$, we get $s(h) \leq \|D_n V \tilde{D}_m^{-1}\|^2 s(f)$, and the thesis now follows from Lemma 7 and from the equalities $\|D_n\|^2 = n$ and $\|\tilde{D}_m^{-1}\|^2 = 1$.

The upper bound of Theorem 8 is a starting point for understanding the strict connection between parallel complexity and sensitivity. As an example, we have that, if the function f is the OR (AND) function and the functions g_i are AND (OR) functions, and if the fan-in of the $\wedge$ ($\vee$) gates is of order n, then $s(h) = O(c(n)\frac{n}{2^n})$, where $c(n)$ is the number of $\wedge$ ($\vee$) gates, from which $c(n) = \Omega(\frac{2^n}{n} s(h))$. (See also [BOH 90] for another bound on $c(n)$.) Thus the notion of sensitivity *explains* why some functions can not be computed by circuits with polynomial size and small depth. Other applications will be shown in the full paper.

We consider now the upper bound of Theorem 9. There are two properties of composition which is worth analyzing, namely

- Since the average sensitivity cannot exceed n and there are functions in NC^1 which attain this bound, for a study of completeness within NC^1 with respect to many-to-one reductions, it is interesting to find under which conditions we have $n2^{m-n}\rho(A^T A)s(f) < n$.

- Consider circuits over the basis $\{AND, OR, NOT\}$. Since $s(AND) = s(OR) = \frac{n}{2^{n-1}}$ and since a polynomial number of AND/OR gates allows one to compute functions with sensitivity up to n, e.g., the parity of n bits, in general the amplification of the average sensitivity can have an exponential growth. Thus it is important to distinguish cases under which the composition of sensitivity has a polynomial, rather than exponential, amplification, i.e. $s(h) \leq n^{O(1)} s(f)$.

We take into account two (extreme) cases:

- **p is large.** This corresponds to saying that the function g maps many strings onto a few strings. In this case the upper bound is very large, but we can get direct information on h. In fact, if $p = 1$ then h is a constant function and thus $s(h) = 0$; if $p \geq 1 - [\alpha(n)]^{-1}$, where $\alpha(n)$ is any function increasing with n, then $s(h) = O(n[\alpha(n)]^{-1})$.

- **p is small.** This corresponds to saying that the cardinality of the inverse image of any string under g is approximately 2^{n-m}. If $2^{-m} \leq p \leq m^{O(1)} 2^{-m}$, then we have $s(h) \leq n^{O(1)} s(f)$. If $2^{-m} \leq p \leq [\alpha(n)]^{-1}$, then $ns(f) \leq n2^{m-n}\rho(A^T A)s(f) \leq n[\alpha(n)]^{-1} 2^m s(f)$. If $p \approx 2^{-m}$, then the functions f and h have approximately the same *density*, i.e. $2^{-m}\sum_w f(w) \approx 2^{-n}\sum_v h(v)$, so that $\frac{s(f)}{n} \leq s(h) \leq ns(f)$.

As a consequence of these properties, we can use the upper bound of Theorem 9 to see that a sparse set cannot be complete in NC^1 under many-to-one reductions if $p = O(\frac{1}{\beta(m)})$, where $\beta(m)$ is any function growing more than any polynomial in m.

5 Sensitivity: Between Structure and Complexity

In this section we first show that there is a correspondence between sparse - or co-sparse - sets and functions whose average sensitivity decreases exponentially. This fact is the basis for showing that low average sensitivity is a structural property of sets which generalizes sparsity in a natural way. Then we find some interesting relations between a measure of complexity for symmetric functions defined in [FKPS 85] and the average sensitivity. These relationships allow us to use a result of [CK 91] for proving that the average sensitivity of symmetric functions - and, more in general, of functions with polynomial index - in AC^0 decreases exponentially.

We say that a language over $\{0, 1\}$ is (i) *sparse* (*co-sparse*) if the strings of length at most n which belong to it are at most $n^{O(1)}$ (at least $2^n - n^{O(1)}$); (ii) *almost sparse* (*almost co-sparse*) if the strings of length at most n which belong to it are at most $n^{polylogn}$ (at least $2^n - n^{polylogn}$).

Lemma 10 *A set is sparse or co-sparse iff the average sensitivity of its characteristic function is* $O\left(\frac{n^{O(1)}}{2^n}\right)$. *A set is almost sparse or almost co-sparse iff the average sensitivity of its characteristic function is* $O\left(\frac{n^{polylogn}}{2^n}\right)$.

Proof From the definition of average sensitivity, we have that $s(f) \leq 2np$, where $p = Pr\{f(w) = 1\}$, and the probability is taken over all $w \in \{0, 1\}^n$. The *if* part of the Lemma follows from the fact that the characteristic function of a sparse set satisfies $p \leq \frac{n^{O(1)}}{2^n}$. It remains to prove that the functions with exponentially low average sensitivity are the characteristic functions of a sparse or co-sparse set. From Parseval's identity [Leh 71] and from the definition of f_0^* we obtain $p = \sum_i (f_i^*)^2$ and $p = f_0^*$. Then, we get

$$p = \sum_i (f_i^*)^2 = f_0^{*2} + \sum_{i \neq 0} (f_i^*)^2 \leq f_0^{*2} + \sum_{i \neq 0} |i| (f_i^*)^2 = f_0^{*2} + \frac{s(f)}{4},$$

i.e. $p^2 - p + \frac{s(f)}{4} \geq 0$. Solving the latter inequality for p and using the hypothesis that $s(f) \leq \frac{n^{O(1)}}{2^n}$ we obtain that either $p \leq \frac{n^{O(1)}}{2^n}$ or $p \geq 1 - \frac{n^{O(1)}}{2^n}$, i.e. f is the characteristic function of either a sparse or co-sparse set. The proof for almost sparse or almost co-sparse sets is similar.

From now on in this section f will be a symmetric Boolean function. We recall some definitions from [FKPS 85]. The minimum number of variables of f that have to be set to constant values so that f becomes a constant function is called *measure* of f and is denoted by $\mu(f)$. Let $w \in \{0, 1\}^{n+1}$ with elements w_i, where w_i is equal to the value of f when i variables are set to 1 and the other variables are set to 0. w is called *spectrum* of f. w_i is called i-th *character* of w. A *subword* of the spectrum is a string of the form $w_i \, w_{i+1} \, \ldots \, w_{i+k}$. [FKPS 85] show that $\mu(f)$ can be easily evaluated from the spectrum because $\mu(f) = n + 1 - \Gamma$, where Γ is the length of the longest constant subword of w.

Lemma 11 *Let f be a symmetric Boolean function defined on $\{0, 1\}^n$ with measure $\mu(f)$. Then*

$$\frac{n}{2^{n-1}} \sum_{k=0}^{\bar{k}} \left\{ \binom{n-1}{\lfloor \frac{\mu(f)}{2} \rfloor - k\Gamma - 1} + \binom{n-1}{\lceil \frac{\mu(f)}{2} \rceil - k\Gamma - 1} \right\} \leq s(f) \leq \frac{n}{2^{n-1}} \sum_{k=0}^{\mu(f)-1} \binom{n-1}{k},$$

where $\bar{k} = \lfloor \frac{\mu(f)}{2\Gamma} \rfloor$. Furthermore, both the lower and the upper bounds are tight.

Proof The influence of x_i on a symmetric function f is given by

$$I_f(x_i) = \frac{1}{2^{n-1}} \sum_{k=0}^{n-1} \binom{n-1}{k} |w_{k+1} - w_k|,$$

where w_k denotes the k-th character of the spectrum of f. Hence we have

$$s(f) = \frac{n}{2^{n-1}} \sum_{k=0}^{n-1} \binom{n-1}{k} |w_{k+1} - w_k|.$$

We need evaluate the maximum value which can be attained by the average sensitivity for functions whose measure is $\mu(f)$. By analyzing the behaviour of the binomial coefficients $\binom{n-1}{k}$ for $0 \leq k \leq n-1$, it is easy to see that this maximum can be detected by looking at spectra of one of the forms

$$w = \underbrace{1010101\ldots10}_{\mu}\underbrace{11111\ldots1}_{\Gamma};$$

$$w = \underbrace{11111\ldots1}_{\Gamma}\underbrace{0101010\ldots10}_{\mu}$$

$$w = \underbrace{1010101\ldots01}_{\mu}\underbrace{00000\ldots0}_{\Gamma};$$

$$w = \underbrace{00000\ldots0}_{\Gamma}\underbrace{1010101\ldots01}_{\mu}$$

Thus, the maximum average sensitivity of functions with measure $\mu(f)$ is

$$\frac{n}{2^{n-1}} \sum_{k=0}^{\mu(f)-1} \binom{n-1}{k}.$$

The lower bound on $s(f)$ can be evaluated similarly. In fact, the functions of measure $\mu(f)$ whose average sensitivity is minimal must have spectra of the form:

$$\underbrace{000\ldots0}_{\lfloor\frac{\mu(f)}{2}\rfloor - \bar{k}\Gamma}\underbrace{\underbrace{111\ldots1}_{\Gamma}\underbrace{000\ldots0}_{\Gamma}\ldots\underbrace{111\ldots1}_{\Gamma}}_{\bar{k}\Gamma}\underbrace{000\ldots0}_{\Gamma}\underbrace{\underbrace{111\ldots1}_{\Gamma}\ldots\underbrace{000\ldots0}_{\Gamma}\underbrace{111\ldots1}_{\Gamma}}_{\bar{k}\Gamma}\underbrace{000\ldots0}_{\lceil\frac{\mu(f)}{2}\rceil - \bar{k}\Gamma},$$

where $\bar{k} = \lfloor \frac{\mu(f)}{2\Gamma} \rfloor$. Hence there esist functions of average sensitivity

$$\frac{n}{2^{n-1}} \sum_{k=0}^{\bar{k}} \left\{ \binom{n-1}{\lfloor \frac{\mu(f)}{2} \rfloor - k\Gamma - 1} + \binom{n-1}{\lceil \frac{\mu(f)}{2} \rceil - k\Gamma - 1} \right\},$$

which is the minimum possible value for functions of measure μ.

The above Lemma has a very interesting consequence for symmetric functions in AC^0.

Theorem 12 *Let f be a symmetric function in AC^0. Then $s(f) = O(2^{-n+polylog\,n})$, and this is equivalent to saying that f is the characteristic function of an almost sparse - or almost co-sparse - language.*

Proof A consequence of Lemma 11, together with the characterization of [FKPS 85], is that symmetric functions in AC^0 have exponentially decreasing average sensitivity. In fact, since symmetric functions in AC^0 have measure bounded above by a polylog, from Lemma 11 we obtain

$$s(f) \leq \frac{n}{2^{n-1}} \sum_{k=0}^{polylogn} \binom{n-1}{k} \leq \frac{n^{polylogn}}{2^n}.$$

Then Lemma 10 implies that f is the characteristic function of either an almost sparse or almost co-sparse language.

Corollary 13 *The number of symmetric functions of n variables which are computable by polynomial size constant depth circuits is of order $n^{polylogn}$.*

Proof The upper bound follows from Theorem 1 in [WWY 92] and standard counting arguments. The lower bound is obtained by counting the number of functions for which $\mu(f) = O(polylog\,n)$.

6 Applications

In this section we present some applications of the results of section 4 and we study some special cases. We show that, in some cases, sets of high sensitivity can not reduce to sets of substantially smaller sensitivity. The main idea behind the proofs is to exploit a structural difference between AC^0 and NC^1, namely the fact that NC^1 functions have any possible average sensitivity, while AC^0 functions have sensitivity at most polylogarithmic. In particular we find another proof of the fact that $MAJORITY$ is not NC^1 complete under projections.

Finally we find a simple formula for expressing the average sensitivity of functions computable by read-once formulas. By using some relations between read-once formulas and circuits, this expression can be used, e.g., to find a lower bound to the number of $\wedge$ gates in a circuit computing f according to its disjuntive normal form. Using a different terminology which refers to the notions of *entropy* and *information content*, the same lower bound was proved in [BOH 90]. We first give a Lemma that links sensitivity to adjacency [Sub 90] and then a Lemma that states a property of the $MAJORITY$ function.

Lemma 14 *Let f and h be two Boolean functions over $\{0,1\}^n$. Let $g : \{0,1\}^n \rightarrow \{0,1\}^n$ be a function for which $d(x,\hat{x}) \leq 1$ implies $d(g(x), g(\hat{x})) \leq d$. If $h(x) = f(g(x))$, then $s(h) \leq s^{(d)}_{max}(f)$, where $s^{(d)}_{max}(f) = \max_w s^{(d)}_w(f)$, and $s^{(d)}_w(f)$ is the sensitivity of f on w taken by looking at strings whose Hamming distance from w is at most d.*

Proof (By Contradiction.) Assume $s(h) = k$ and $s^{(d)}_{max}(f) \leq k-1$. Then there exists at least one string w such that $s_w(h) \geq k$, i.e. there are at least k Hamming neighbors of w, w_i, for which $h(w) \neq h(w_i)$. This means that $f(g(w)) \neq f(g(w_i))$, $i = 1, 2, \ldots, k$, which is impossible because $d(g(w), g(w_i)) \leq d$ and $s^{(d)}_{max}(f) \leq k-1$.

Lemma 15 *Let f_n be the $MAJORITY$ function on n variables. Then*

$$s_{max}(f_n) = \max_w s_w(f_n) = \lfloor \frac{n}{2} \rfloor + 1$$

.

Proof Since $f_n = 1$ on strings with at least $\lfloor \frac{n}{2} \rfloor + 1$ bits equal to 1, then $s_w(f_n) = 0$ if w has at most $\lfloor \frac{n}{2} \rfloor - 1$ bits equal to 1 or at least $\lfloor \frac{n}{2} \rfloor + 2$ bits equal to 1. Thus we can restrict the analysis to strings with $\lfloor \frac{n}{2} \rfloor$ and $\lfloor \frac{n}{2} \rfloor + 1$ bits equal to 1. In both cases, we easily get that the number of Hamming neighbors for which the function changes value is $\lfloor \frac{n}{2} \rfloor + 1$.

Definition 1 *Let f and g be sets represented as infinite collections of Boolean functions, f_i and g_i. We say that g reduces to f by projections, denoted as $g \preceq_{pj} f$, if, for any n, $g_n(x_1, x_2, \ldots, x_n) = f_{p(n)}(\sigma_n(y_1), \sigma_n(y_2), \ldots, \sigma_n(y_{p(n)}))$, where $p(n)$ is a function bounded above by a polynomial in n, and $\sigma_n : \{y_1, y_2, \ldots, y_{p(n)}\} \rightarrow \{x_1, \bar{x_1}, x_2, \bar{x_2} \ldots x_n, \bar{x_n}, 0, 1\}$.*

The notion of *projection reducibility* was studied in [SV 81].

Lemma 16 *Let f be a set and $\{f_i, i = 1, \ldots\}$ be the set of Boolean functions representing it. If the maximal sensitivity of f_n, i.e. the maximum of the sensitivities of f_n on its arguments, is less than n, then f is not complete for NC^1 under reduction by projections.*

Proof Follows from the fact that $g \preceq_{pj} f$ implies $s(g) \leq s_{max}(f)$.

[SV 81] proved that $MAJORITY$ is not complete for the class NC^1 under reduction by projections. Lemma 16 allows us to find another proof of this fact.

Corollary 17 *MAJORITY is not complete for the class NC^1 under reduction by projections.*

Proof Follows from Lemma 16 together with Lemma 15.

We analyze now the role of sensitivity to characterize the relative complexity of problems in NC^1 using $DLOGTIME$ transformations which satisfy some constraints. In particular, a transformation will be called *adjacency preserving* if it does not increase the Hamming distance between the strings.

Lemma 18 *Let A be a language and $\{f_i, i = 1, \ldots\}$ be the set of its characteristic functions. If the maximal sensitivity of f_n is less than n, then A is not complete for the class NC^1 under adjacency preserving $DLOGTIME$ transformations.*

Proof Follows from Lemma 14 with $d = 1$.

Theorem 19 *MAJORITY is not complete for the class NC^1 under adjacency preserving $DLOGTIME$ transformations.*

Proof Follows from previous Lemma together with the fact that the maximal sensitivity of the characteristic function associated to $MAJORITY$ is $\lfloor \frac{n}{2} \rfloor + 1$.

Theorem 20 *Let A be a language and $\{f_i, i = 1, \ldots\}$ be the set of its characteristic functions. If the average sensitivity of f_n is asymptotically $o(n)$, then A is not complete for the class NC^1 under invertible $DLOGTIME$ transformations.*

The average sensitivity is a measure of how a function behaves assuming that all the possible arguments occur with the same probability. This is not the case, in general, when the function describes a portion of a circuit. In this case, its arguments can be viewed as the values computed by functions representing other parts of the circuit. The nature of these functions strongly influences the distribution of the arguments. To analyze this question we introduce a more general notion of sensitivity, which we call *on-line sensitivity.* We then apply this notion to evaluate the sensitivity of functions computable by read-once formulas. We also show the natural connection between the notion of on-line sensitivity and the technique of *amplification of Boolean formulas* - see [Bop 89], [DZ 92] for an introduction to this topic and for relevant results.

Definition 2 *The* on-line sensitivity $s_p(f)$ *of a Boolean function f is given by* $s_p(f) = \sum_w p_w s_w(f)$, *where p_w is the probability of occurrence of the argument w, and $s_w(f)$ is the sensitivity of f on w.*

Definition 3 *Let $\mathcal{A}$ be a set of variables. The* on-line influence *of $\mathcal{A}$ on f, $I_{f,p}(\mathcal{A})$, is the probability that f remains undetermined as long as the variables in $\mathcal{A}$ are not assigned values and the other variables are assigned according to a given probability distribution p.*

As for $s(f)$, one can evaluate $s_p(f)$ as the sum of the on-line influences of all the variables.

Definition 4 *The* amplification function $A_f(p)$ *of a Boolean function* f *is given by* $A_f(p) = Pr\{f(x_1, x_2, \ldots, x_n) = 1\}$, *if* $Pr\{x_i = 1\} = p$. *The* multivariate amplification function $A_f(p_1, \ldots, p_n)$ *of a Boolean function* f *is* $A_f(p_1, \ldots, p_n) = Pr\{f(x_1, x_2, \ldots, x_n) = 1\}$, *if* $Pr\{x_i = 1\} = p_i$.

On-line sensitivity is closely related to the amplification of sensitivity due to composition. The following example puts into evidence that functions with exponentially decreasing average sensitivity can combine together and rapidly produce functions of increasing average sensitivity. The reason is that the combination process allows one of the functions to produce strings on which the other one is very sensitive. In other words, on-line sensitivity is very useful to analyze the composition of functions whose sensitivity on different strings has a wide variance.

Example 1 *CNF formulas. Let*

$$f = \bigwedge_{i=1}^{m} w_i \,,\; w_i = \bigvee_{j=1}^{k} x_{i_j}.$$

It is easy to see that $I_{f,p}(w_i) = \prod_{j \neq i} p_j$, *where* p_j *is the probability that* $w_j = 1$. *Let* $n = mk$. *Since* $p_j = 1 - \frac{1}{2^k}$, *we obtain* $I_{f,p}(w_i) = (1 - \frac{1}{2^k})^{m-1}$ *and thus*

$$s_p(f) = \frac{n}{k}(1 - \frac{1}{2^k})^{\frac{n}{k}-1}.$$

Note that the maximum value of $s_p(f)$ *is of order* $\frac{n}{\log n}$ *and is attained for* $k = \log n$. *In addition, note that*

$$s_p(f) = \frac{n}{k}\left(A_{\bigvee_{i=1}^{k}}(1/2)\right)^{\frac{n}{k}-1}.$$

To study more general functions, it is convenient to use the multivariate amplification function. Let for example $h = f(g_1(w_1), g_2(w_2), \ldots, g_m(w_m))$. For $w \in \{0, 1\}^n$ we get

$$Pr\{h(w) = 1\} = \sum_{i=0}^{2^n-1} (h_i^*)^2 = A_f(p_1, \ldots, p_n),$$

where $Pr\{g_i(w_i) = 1\} = p_i$.

Further connections between amplification and Fourier coefficients are the following:

- *Evaluation of the on-line sensitivity.* The value of the weights that arise in the definition of on-line sensitivity can be computed according to a formula of the type $\prod_i A_{g_i} \prod_j (1 - A_{g_j})$.

- *Evaluation of the Fourier coefficients of the composition of Boolean functions.* The j-th entry of the diagonal matrix $A^T A$ (see Section 4) can be written as $2^n \prod_i A_{g_i} \prod_j (1 - A_{g_j})$.

Recall now that a *formula* is a Boolean circuit of fan-out 1 and a *read-once formula* is a formula in which each variable appears only once.

We turn to the analysis of the sensitivity of Boolean functions computable by read-once formulas (read-once functions from now). These functions are important, expecially for *low* complexity classes, because every NC^1 function on n variables can be viewed as the projection of a read-once function with $n^{O(1)}$ variables.

Lemma 21 *Let h be a read-once Boolean function defined as*

$$h(x) = f(g_1(w_1), g_2(w_2), \dots, g_m(w_m)),$$

where $w_i = x_{1i} x_{2i} \dots x_{k_i i}$.

(a) *The influence of x_{ij} on h is $I_h(x_{ij}) = I_{g_j}(z_i) I_{f,p}(z_j)$, where z_i and z_j denote the i-th and the j-th bits in input to g_j and f, respectively.*

(b) *The average sensitivity of h is* $s(h) = \sum_{j=1}^m (I_{f,p}(z_j) s(g_j))$.

Proof

(a) Follows from the definitions of influence and on-line influence.

(b) Follows from the definition of average sensitivity as sum of the influences of all the variables.

Corollary 22

1. $s(h) \le \sum_{j=1}^m s(g_j)$.
2. $s(h) \le \max_{1 \le j \le m}\{s(g_j)\} s_p(f) \le \max_{1 \le j \le m}\{s(g_j)\} s_{max}(f)$.
3. *If $g_j = g$ for all j, then* $s(h) = s(g) s_p(f)$.
4. *If $g_j \in \{\vee, \wedge\}$ for all j, then* $s(h) \le \frac{m k_{min}}{2^{k_{min}-1}} \le \frac{n}{2^{k_{min}-1}}$, *where* $k_{min} = \min_{1 \le i \le m} k_i$.
5. *If ϕ is a read-once Boolean function computable by a* layered *circuit of depth k, then $s(\phi) \le s(f_1) \prod_{i=2}^k s_p(f_i)$, where the functions f_i satisfy*
    - *The output of f_k is the value computed by ϕ*
    - *For $i \neq k$, f_i is the function of maximal on-line sensitivity whose output is an input for f_{i+1}.*

Proof

1. Follows from the fact that $I_{f,p}(z_j) \leq 1$ for $1 \leq j \leq m$.

2. Follows from

$$\begin{aligned} s(h) &= \sum_{j=1}^{m} I_{f,p}(z_j)\ s(g_j) \leq \max_{1\leq j\leq m}\{s(g_j)\} \sum_{j=1}^{m} I_{f,p}(z_j) = \max_{1\leq j\leq m}\{s(g_j)\} s_p(f) = \\ &= \max_{1\leq j\leq m}\{s(g_j)\} \sum_{w} p_w s_w(f) \leq \max_{1\leq j\leq m}\{s(g_j)\}\ \max_{w}\{s_w(f)\}. \end{aligned}$$

3. Follows from

$$s(h) = \sum_{j=1}^{m} I_{f,p}(z_j)\ s(g_j) = \sum_{j=1}^{m} I_{f,p}(z_j)\ s(g) = s(g) s_p(f).$$

4. Follows from 2 and from the following facts:

   - $\max_{1\leq j\leq m}\{s(g_j)\} = \max_{1\leq j\leq m}\left\{\frac{k_j}{2^{k_j-1}}\right\} = \frac{k_{min}}{2^{k_{min}-1}}$, with $k_{min} = \min_{1\leq i\leq m}\{k_i\}$.
   - $s_{max}(f) \leq m$.
   - $mk_{min} \leq \sum_{j=1}^{m} k_j = n$.

   Note that for $k_{min} = 2$, we get $s(h) \leq \frac{n}{2}$, and for $k_{min} = \log n + 1$, $s(h) \leq 1$.

5. Follows by repeated applications of 2.

The above corollary has several interesting consequences, e.g. the fact that the average sensitivity of functions computable by read-once formulas is upper bounded by $\frac{n}{2}$, and that, if the minimum fan-in of a gate is of order $\log n$, then the average sensitivity is not increasing.

7 Concluding Remarks

This paper is a step forward in the process of understanding the intriguing relations between sensitivity and quantitative aspects of computing. The influence of sensitivity on the efficiency of computational tasks has been studied in other areas, like, e.g., parallelism [CLR 93], dynamization [EGIN 92], and program checking [ABCG 93]. We believe that more general results are to come, expecially for what concerns the interplay between parallel complexity and sensitivity.

References

[ABCG 93] S. Ar, M. Blum, B. Codenotti, P. Gemmell. *Checking Approximate Computations over the Reals.* Proc. 25th STOC (1993), pp.786-795.

[BL 89] M. Ben Or, N. Linial. *Collective Coin Flipping.* In *Advances in Computing Research - Randomness and Computation*, Vol. 5, JAI Press (1989), pp.91-115.

[Bop 89] R. B. Boppana. *Amplification of Probabilistic Boolean Formulas.* In *Advances in Computing Research - Randomness and Computation*, Vol. 5, JAI Press (1989), pp.27-46.

[BOH 90] Y. Brandman, A. Orlitsky, J. Hennessy. *A Spectral Lower Bound Technique for the Size of Decision Trees and Two-Level AND/OR Circuits.* IEEE Trans. on Computers Vol. 39 (2) (1990), pp.282-287.

[Bru 90] J. Bruck. *Harmonic Analysis of Polynomial Threshold Functions.* SIAM Journal on Discrete Mathematics Vol. 3 (1990), pp.168-177.

[BS 92] J. Bruck, R. Smolensky. *Polynomial Threshold Functions, AC^0 Functions, and Spectral Norms.* SIAM Journal on Computing Vol. 21(1) (1992), pp.33-42.

[CK 91] P. Clote, E. Kranakis. *Boolean Functions, Invariance Groups, and Parallel Complexity.* SIAM Journal on Computing Vol. 20(3) (1991), pp.553-590.

[CLR 93] B. Codenotti, M. Leoncini, G. Resta. *Oracle Computations in Parallel Numerical Linear Algebra.* Theoretical Computer Science, to appear (1993).

[DZ 92] M. Dubiner, U. Zwick. *Amplification and Percolation.* Proc. 33rd FOCS (1992), pp.258-267.

[EGIN 92] D. Eppstein, Z. Galil, G.F. Italiano, A. Nissenzweig. *Sparsification - A Technique for Speeding up Dynamic Graph Algorithms.* Proc. 33rd FOCS (1992), pp.60-69.

[FKPS 85] R. Fagin, M.M. Klawe, N.J. Pippenger, L. Stockmeyer. *Bounded-Depth, Polynomial Size Circuits for Symmetric Functions.* Theoretical Computer Science Vol. 36 (1985), pp.239-250.

[HMM 82] S.L. Hurst, D.M. Miller, J.C. Muzio. *Spectral Method of Boolean Function Complexity.* Electronics Letters Vol. 18 (33) (1982), pp.572-574.

[KKL 88] J. Kahn, G. Kalai, N. Linial. *The Influence of Variables on Boolean Functions.* Proc. 29th FOCS (1988), pp.68-80.

[Kar 76] M.G. Karpovsky. *Finite Orthogonal Series in the Design of Digital Devices.* John Wiley and Son, New York (1976).

[Leh 71] R. J. Lechner. *Harmonic Analysis of Switching Functions.* In *Recent Development in Switching Theory*, Academic Press (1971), pp.122-229.

[LMN 89] N. Linial, Y. Mansour, N. Nisan. *Constant Depth Circuits, Fourier Transform, and Learnability.* Proc. 30th FOCS (1989), pp.574-579.

[Lit 40] D. Littlewood. *The Theory of Group Characters and the Matrix Representation of Groups.* Oxford University Press, London (1940).

[Loo 53] L.M. Loomis. *An Introduction to Abstract Harmonic Analysis.* Van Nostrand, Princeton, New Jersey (1953).

[SHB 68] F. Sellers, M.Y. Hsiao, L.W. Bearnson. *Error Detecting Logic for Digital Computers.* McGraw Hill (1968).

[SV 81] S. Skyum, L.G. Valiant. *A Complexity Theory Based on Boolean Algebra.* Proc. 22nd FOCS (1981), pp.244-253.

[Sub 90] A. Subramanian. *The Computational Complexity of the Circuit Value and Network Stability Problems.* PhD Thesis, Tech. Report No. STAN-CS-90-1311, Stanford University (1990).

[WWY 92] I. Wegener, N. Wurm, S.Z. Yi. *Symmetric Functions in* AC^0 *can be Computed in Constant Depth with Very Small Size.* In "Boolean Function Complexity", Edited by M.S. Paterson, Cambridge University Press (1992), pp.129-139.