On the Power of Circuits with Gates of Low
$L_1$ Norms

Vince Grolmusz *

August 3, 1995

Abstract

We examine the power of Boolean functions with low $L_1$ norms in
several settings. In large part of the recent literature, the degree of
a polynomial which represents a Boolean function in some way was
designed to be the measure of the complexity of the Boolean function
(see, e.g. [1], [3], [5], [36], [27], [34], [26], [16]). However, some functions
with low communicational complexity (AND, OR, PARITY, ID) have
high degree, but small $L_1$ norms. So, in conjunction with communica-
tion complexity, instead of the degree, the $L_1$ norm can be an
important measure of hardness. We conjecture that the randomized
communication complexity of any Boolean function is bounded by the
polylogarithm of its $L_1$ norm.

We can prove only a weaker statement: we present a two-party,
randomized, common-coin communication protocol for computing func-
tions with $O(L_1^2\delta)$ bits of communication, with error-probability of
$\exp(-c\delta)$, (even with large degree or exponential number of terms).
Then we present several applications of this theorem for circuit lower
bounds (both for bounded- and unbounded depth), and a decision-tree
lower bound.

*Department of Computer Science, Eötvös University, Budapest, Address: Múzeum
krt.6-8, H-1088 Budapest, HUNGARY; E-mail: grolmusz@cs.elte.hu
1 INTRODUCTION

Methods in communication complexity have become standard tools in circuit complexity theory ([19], [23], [22], [26], [29], [12], [15], [10]). These methods are also used with success for giving lower bounds for the depth of decision trees with linear or low-degree test functions [11], [26], [37].

Another important tool in examining Boolean function complexity is representing the Boolean functions by polynomials above some field or ring, which facilitates using algebraic or analytical methods (see, e.g. [1], [3], [4], [6], [8], [7], [24], [27], [34]).

The previous two approaches are unified, i.e. communication complexity tools are applied to the polynomial representations of Boolean functions in [26], [10], [16], or in the full version of [12]. In the present work, communication complexity tools will be applied to polynomials, intimately related to the Fourier expansions of Boolean functions.

1.1 Fourier Expansions

The Fourier-expansion of Boolean functions [24], [8], [20], [27] are defined as follows:

Let us represent Boolean function \( f \) as a function \( f : \{-1,1\}^n \rightarrow \{-1,1\} \) where \(-1\) stands for “true”. The set of all real valued functions over \( \{-1,1\}^n \) forms a \( 2^n \) dimensional vector-space over the reals. Let us define for \( \alpha = (\alpha_1, \alpha_2, ..., \alpha_n) \in \{0,1\}^n \)

\[
X^\alpha = \prod_{i=1}^{n} x_i^{\alpha_i}.
\]

The monomials \( X^\alpha \) for \( \alpha \in \{0,1\}^n \) form an basis in this \( 2^n \)-dimensional vector space; consequently, any function \( h : \{-1,1\}^n \rightarrow \mathbb{R} \) can be uniquely expressed as

\[
h(x_1, x_2, ..., x_n) = \sum_{\alpha \in \{0,1\}^n} a_\alpha X^\alpha \quad (1)
\]

The right-hand-side of (1) is called the Fourier-expansion of \( h \), and numbers \( a_\alpha \) for \( \alpha \in \{0,1\}^n \) are called the spectral (or Fourier-) coefficients of \( h \). The
L₁ norm of h is:
\[
L₁(h) = \sum_{a \in \{0,1\}^n} |a_α|,
\]
We are especially interested in the Fourier-expansions of Boolean functions.

1.1.1 Examples

- The PARITY function in this setting is \(x_1x_2...x_n\), its L₁ norm is 1, while its degree is \(n\).
- It is easy to verify that
  \[
  \bigwedge_{i=1}^n x_i = -\frac{1}{2^{n-1}} \left(2^{n-1} - \prod_{i=1}^n (x_i + 1)\right) =
  \]
  \[
  = -\frac{1}{2^{n-1}} \left(2^{n-1} - (1 + x_1 + x_2 + ... + x_n + x_1x_2 + ... + x_1x_2...x_n)\right);
  \]
  and

  \[
  \bigvee_{i=1}^n x_i = \frac{1}{2^{n-1}} \left(2^{n-1} - \prod_{i=1}^n (1 - x_i)\right) =
  \]
  \[
  = \frac{1}{2^{n-1}} \left(2^{n-1} - (1 - x_1 - x_2 - ... - x_n + x_1x_2 + ... + (-1)^n x_1x_2...x_n)\right).
  \]

Let us observe that both the \(n\)-fan-in OR and AND have exponentially many non-zero Fourier-coefficients, their degree is \(n\), while their L₁ norms are less than three.

- The inner product mod 2 function (IP) is defined as follows:
  \[
  IP(x_1, x_2, ..., x_{2n}) = \prod_{i=1}^n (x_{2i-1} \land x_{2i}).
  \]
  It is easy to verify that \(L₁(IP)\) is the highest possible for any \(2n\) variable Boolean functions: \(2^n\).

- The set-disjointness function (DISJ) is defined as
  \[
  DISJ(x_1, x_2, ..., x_{2n}) = \bigvee_{i=1}^n (x_{2i-1} \land x_{2i}).
  \]
  Its degree is \(2n\), and its L₁ norm is \(\Omega((3/2)^n)\).
• The ID (identity) function is defined as follows:

\[ ID(x_1, x_2, \ldots, x_{2n}) = \bigwedge_{i=1}^{n} (-x_{2i-1} x_{2i}), \]

i.e. it is TRUE exactly when \( x_{2i-1} = x_{2i} \) for all \( i = 1, 2, \ldots, n \). Its degree is \( 2n \), it has exponentially many non-zero Fourier-coefficients, and its \( L_1 \) norm is the same as that of the \( n \)-fan-in AND; less than three.

We can get further examples by negating an arbitrary set of the variables in the previous ones. This operation will not affect the degree or the \( L_1 \) norm in the previous examples, but we can get further non-symmetric examples for functions with exponentially many terms and small \( L_1 \) norms from AND, OR or ID.

1.2 \( L_1 \) Norm and Communication Complexity

Those functions in the previous example, which have constant \( L_1 \) norms – AND, OR, PARITY – are known to have constant (two-party) communication complexity, while the IP function and the DISJ function with exponential \( L_1 \) norms have linear (both deterministic and probabilistic) communication complexity [9], [21], [31]. The deterministic communication complexity of the ID function is \( n \), but it has a random, common-coin protocol with \( O(1) \) communication (cf. [26]). These observations motivate the following conjecture:

**Conjecture 1** There exists a constant \( c > 0 \) such that the 2-party, probabilistic, common-coin communication complexity of \( f : \{-1,1\}^n \to \{-1,1\} \) is at most

\[ \left( \log(L_1(f)) \right)^c. \]

Conjecture 1 is analogous with the following conjecture of Lovász and Saks [25]:

4
Conjecture 2 There exists a constant $c > 0$ such that the 2-party (deterministic) communication complexity of $f : \{0, 1\}^n \rightarrow \{0, 1\}$ is at most

$$
\left( \log(\text{rank } C_f) \right)^c,
$$

where $C_f$ is the communication-matrix of function $f$.

Nisan and Wigderson [28] have several nice results concerning Conjecture 2, however, it remained open.

We have shown in [13] and [14] that the multi-party version of Conjecture 1 is true for Boolean functions of at least linear $L_1$ norm, even in the deterministic setting:

Theorem 3 There exists a $c > 0$ such that for any Boolean function $f$ of $n$ variables, with $L_1(f) \geq n$, and with $k = c \log L_1(f)$, the $k$-party communication complexity of $f$ is at most

$$
O(\log^3 L_1(f)).
$$

\\[\square\]

1.2.1 The Main Result

Let $f : \{-1, 1\}^{2n} \rightarrow \{-1, 1\}$ be a Boolean function and let

$$
f(x) = \sum_{\alpha \in I} a_\alpha X^\alpha
$$

be its Fourier-expansion, where $a_\alpha \neq 0$ for $\alpha \in I$. Suppose that the $2n$ variables of $f$ are partitioned between two players, Alice and Bob, and Alice does not know the values of Bob’s variables, and Bob does not know the values of Alice’s variables, and they want to compute the value of $f$. Suppose that they also know the Fourier-expansion (2) of $f$. Since every monomial $X^\alpha$ can be evaluated by communicating 1 bit, $|I|$ bits are enough for computing $f$. When $|I| > n$ then this is worse than the trivial $n$-bit communication protocol. We have seen in Section 1.1.1, that simple functions, even with small $L_1$ norms, may have exponentially many non-zero Fourier-coefficients.

By a famous result of Bruck and Smolensky [8], there exists a polynomial $G$ for $f$, such that $G$ can be written as a sum of $O(nL_1^2)$ monomials, each
with coefficient 1, and the sign of $f$ and $G$ coincides for all inputs. If Alice and Bob try to evaluate $G$'s monomials one-by-one, the resulting protocol of $O(nL_1^2(f))$ communication is also worse than the trivial $n$-bit protocol, even for functions of low $L_1$ norm.

Here we prove an improvement of this trivial protocol. Our result is still very far from the bound of Conjecture 1, but it has numerous applications for circuit- and decision-tree lower bounds (cf. Sections 1.3, 1.4, 1.5).

**Theorem 4** Let $f : \{-1, 1\}^{2^n} \rightarrow \{-1, 1\}$ be a Boolean function, and let $\delta = \delta(n) > 0$. Then there exists a two-party, randomized, common-coin communication protocol, which computes $f$ with

$$O(L_1^2(f) \delta)$$

communication, and for every fixed input, it is correct with probability at least

$$1 - \exp(-c\delta).$$

Setting $\delta = \log^2 n$, the communication is $O(L_1^2(f) \log^2 n)$, which is small for any small $L_1$ norm, and the inverse of the error is still super-polynomial.

The proof of Theorem 4 is based on a generalization of the result of Bruck and Smolensky [8] (our Lemma 9), and on a probabilistic common-coin communication protocol (Section 2).

Goldmann, Håstad and Razborov [10] also used the result of Bruck and Smolensky [8] for gaining an $O(L_1^{-1}(f))$ advantage (relative to simple guessing the value of $f$) in a communication protocol; however, their method seems to be inappropriate for simultaneous evaluation of more than one Boolean functions, which is the main application of our Theorem 4.

### 1.3 Circuit-Applications: Unbounded Depth

While several famous lower bound proofs can be found in the literature for small-depth circuits ([38], [18], [30], [34]), lower bounds for the size of general unbounded depth circuits are rare and generally much weaker than the small-depth results.

Smolensky [35] proved an $\Omega(n/\log n)$ lower bound for circuits of arbitrary symmetric gates, computing an explicit function of $n$ variables.
Razborov [31] gave a linear lower bound for circuits of linear threshold gates with arbitrary weights, computing the inner product function.

Nisan [26] called a Boolean function \( f : \{-1,1\}^n \to \{-1,1\} \) a threshold gate of degree \( d \) (or \( d \)-threshold gate), if \( f \) can be expressed as a sign of a real polynomial of degree at most \( d \). Then he has built a random \((d+1)\)-party protocol — using the results of [19] — which evaluates the \( d \)-threshold gates with a small number of communicated bits, and then, using the BNS-lower bound [2], the size--lower bound of \( \Omega(e^{2n}/\log^2 n) \) follows for \( d = O(\log n) \).

We — instead of symmetricity or degree conditions — require the \( L_1 \) norms of the gate functions to be small, so Boolean gates with non-zero large-degree coefficients are also allowed.

We say that a Boolean gate has \( L_1 \) norm of \( L \) if it computes a function of \( L_1 \) norm of \( L \). Let function \( 1P \) be defined as in Section 1.1. Then

**Theorem 5** Let \( C_n \) be a circuit of gates with \( L_1 \) norm of at most \( n^\nu \), where \( 0 < \nu < \frac{1}{2} \). If \( C_n \) computes \( 1P(x) \) for all \( x \in \{0,1\}^{2n} \), then

\[
\text{size}(C_n) = \Omega \left( \frac{n^{1-2\nu}}{\log n} \right).
\]

Let us note that the restriction on the \( L_1 \) norms of the gates are logarithmic, relative to the maximum \( 2^n \). Similarly, the restrictions made by Nisan [26] on the degree of the gate functions are also logarithmic, relative to the maximum \( n \).

### 1.4 Circuit-Applications: Bounded Depth

In the recent literature one can find very interesting lower bounds and techniques for bounded-depth circuits with hard-to-handle gates (e.g. \( \text{MOD} \) gates, \( \text{MAJORITY} \) gates, etc.). See for example [6], [8], [34], [10], [12], [15], [4], or see [5] for a survey.

Hajnal, Maass, Pudlák, Szegedy and Turán [17] proved an exponential lower bound for the size of depth-2 circuits with a \( \text{MAJORITY} \) gate at the top, and linear threshold gates of small weights on the bottom.

Hästad and Goldmann [19] generalized it to circuits with a \( \text{MAJORITY} \) gate at the top and \( d \)-threshold gates with small weights at the bottom, \( (d = O(\log n)) \), using the BNS-lower bound [2].
Nisan [26], generalizing the results of [17] and [19], also gives an exponential lower bound to the size of those depth–2 circuits, which compute GIP, with a MAJORITY gate at the top, and several \(d\)-threshold gates of arbitrary weights at the bottom, for \(d = O(\log n)\).

We prove here an exponential lower bound in the case when on the bottom the gates compute Boolean functions of arbitrary degree but with small \(L_1\) norm, while on the top there is a MAJORITY gate.

\[\text{Theorem 6} \quad \text{Let } C_n \text{ be a depth–2 circuit with a MAJORITY gate on the top, and gates with } L_1 \text{ norm at most } n^\nu, \text{ with } \nu < \frac{1}{2}, \text{ on the bottom. If } C_n \text{ computes IP, then} \]
\[\text{size}(C_n) = \exp(n^\varepsilon), \]
\[\text{for some } \varepsilon = \varepsilon(\nu) > 0.\]

### 1.5 Further Applications: Decision Trees

Most of the work done in the Boolean decision tree model deals with test functions of the form “Is the \(i\)th input bit = 1?” (simple decision trees), these trees appear in evasiveness problems. Less is known about decision trees, where each test function may depend on all the variables.


Valan [37] proved a near-linear lower bound for decision trees with \(d\)-threshold test functions \((d = O(\log n))\) and small integer weights, computing the GIP function.

Nisan [26] proved an \(\Omega(c_d n/\log^2 n)\) lower bound for the maximum depth of decision trees with \(d\)-threshold functions of arbitrary weights and \(d = O(\log n)\), computing the GIP function.

We allow test functions of arbitrary degree, but their \(L_1\) norms are required to be small.

**Theorem 7** Let \(T_n\) be a decision tree computing IP with test functions of \(L_1\) norm at most \(n^\nu\), where \(0 < \nu < \frac{1}{2}\). Then the maximum depth of \(T_n\) is
\[\Omega\left(\frac{n^{1-2\nu}}{\log n}\right).\]
2 The Proof of the Main Result

**Definition 8** [8] Let \( f : \{-1,1\}^n \rightarrow \{-1,1\} \) be a Boolean function, and let
\[
f(x) = \sum_{\alpha \in \{0,1\}^n} a_\alpha X^\alpha
\]  
(3)
be its Fourier expansion. Random monomials \( Z_i \) are defined as follows:
\[
Z_i = \text{sgn}(a_\alpha)X^\alpha \quad \text{with probability} \quad \frac{|a_\alpha|}{L_1(f)}.
\]
For any \( \delta > 0 \), let the \( G_\delta(x) \) random polynomial be the sum of \( N_\delta = \lfloor \frac{16}{3}\delta L_1^2(f) \rfloor \) independently chosen monomials \( Z_i \):
\[
G_\delta(x) = \sum_{i=1}^{N_\delta} Z_i.
\]
The following lemma is a generalization of a lemma of Bruck and Smolensky [8].

**Lemma 9** Let \( f : \{-1,1\}^n \rightarrow \{-1,1\} \) be a Boolean function, and suppose that \( L_1(f) \geq 2 \). Let \( \delta > 0 \). Then, for any fixed \( x \in \{-1,1\}^n \),
\[
\Pr(\text{sgn}(G_\delta(x)) \neq \text{sgn}(f(x))) \leq \exp(-c\delta).
\]

**Proof.** The expectation of \( Z_i \):
\[
E(Z_i(x)) = \sum_{\alpha \in \{0,1\}^n} \frac{|a_\alpha|}{L_1(f)} \text{sgn}(a_\alpha)X^\alpha = \frac{f(x)}{L_1(f)},
\]
where we used that \( \text{sgn}(v)|v| = v \).

The expectation of \( G_\delta(x) \):
\[
E(G_\delta(x)) = \frac{N_\delta f(x)}{L_1(f)}.
\]
(4)

The variance of \( Z_i \):
\[
\text{Var}(Z_i(x)) = E(Z_i^2) - E^2(Z_i) = 1 - \frac{1}{L_1^2(f)}.
\]

The variance of \( G_\delta(x) \):

\[
\text{Var}(G_\delta(x)) = N_\delta \left( 1 - \frac{1}{L_1(f)} \right).
\]

Since \( L_1(f(x)) \leq 2 \):

\[
\frac{3N_\delta}{4} \leq \text{Var}(G_\delta(x)) \leq N_\delta
\]

or

\[
\sqrt{\frac{3N_\delta}{4}} \leq D(G_\delta(x)) \leq \sqrt{N_\delta}, \tag{5}
\]

where \( D(G_\delta(x)) = \sqrt{\text{Var}(G_\delta(x))} \), the standard deviation of \( G_\delta(x) \).

From (4), the sign of \( E(G_\delta(x)) \) is the same as the sign of \( f(x) \). Consequently,

\[
\Pr \left( \text{sgn}(G_\delta(x)) \neq \text{sgn}(f(x)) \right) = \Pr \left( \text{sgn}(G_\delta(x)) \neq \text{sgn}(E(G_\delta(x))) \right) \leq
\]

\[
\leq \Pr \left( |G_\delta(x) - E(G_\delta(x))| \geq \frac{N_\delta}{L_1(f)} \right).
\]

From the Bernstein–inequality (see [33], or [32]), (or from the Central Limit Theorem), with \( D = D(G_\delta(x)) \), we have:

\[
\Pr(|G_\delta(x) - E(G_\delta(x))| \geq \mu D) \leq 2 \exp \left( -\frac{\mu^2}{2(1 + \frac{\mu^2}{D})} \right), \tag{6}
\]

where \( \mu \) must satisfy: \( 0 < \mu < \frac{D}{2} \).

Because of (5), we can set \( \mu = \sqrt{\delta} \). On the other hand,

\[
\mu D \leq \frac{N_\delta}{L_1(f)},
\]

so, from (6):

\[
\Pr \left( \text{sgn}(G_\delta(x)) \neq \text{sgn}(f(x)) \right) < e^{-\delta},
\]

for some positive constant \( \epsilon \).
2.1 The proof of Theorem 4:

Suppose first, that $L_1(f) < 2$. Then

$$L_2(f) = <f, f> = 2^{-n} \sum_{x \in \{-1, 1\}^n} f^2(x) = \sum_{\alpha \in \{0,1\}^n} a_\alpha^2 = 1,$$

using the Parseval identity. So we can write:

$$2 > \sum_{\alpha \in \{0,1\}^n} |a_\alpha| > \sum_{\alpha \in \{0,1\}^n} a_\alpha^2 = 1.$$

Consequently, there exists an $\alpha$ such that $|a_\alpha| > \frac{1}{2}$.

- Now, if for all $x \in \{-1, 1\}$:

  $$\text{sgn}(a_\alpha X^\alpha) = \text{sgn}(f),$$

  then the players can evaluate $f$ simply by communicating the value of $X^\alpha$ with 1 bit, and we are done.

- Otherwise, the other Fourier coefficients of $f$ should compensate for $|a_\alpha|$, consequently, the sum of their absolute values is at least $3/2$. So

  $$\sum_{\alpha \in \{0,1\}^n} |a_\alpha| \geq 2,$$

  but this contradicts with $L_1(f) < 2$.

So we may assume that $L_1(f) \geq 2$. Then by Lemma 9, for a $c > 0$ and for any $\delta > 0$, there exists a polynomial

$$G_{\delta, f} = \sum_{i=1}^{N_{\delta}} Z_i$$

such that for any fixed $x$:

$$\Pr(\text{sgn}(G_{\delta, f}(x)) \neq \text{sgn}(f(x))) \leq \exp(-c\delta).$$

Let us consider now the following communication game. Two players, Alice and Bob want to evaluate function $f$. First they randomly generate
(using public coins) polynomial $G_{\delta,f}(x)$, without any communication. From Lemma 9, with $\exp(-c\delta)$ probability of error, the sign of $f$ and $G_{\delta,f}$ coincides. Consequently, if the players evaluate polynomial $G_{\delta,f}$, then they will know the value of $f$. $G_{\delta,f}$ contains at most $N_{\delta} = O(\delta L^2(f))$ monomials.

With the same number of bits Alice sends to Bob the sign of the products of her own variables of each monomial, and from these Bob computes the sign of each one, and from these signs the sign of polynomial $G_{\delta,f}$. The total number of communicated bits is $N_{\delta}$. The probability of error is $\exp(-c\delta)$. □

3 Applications

In this section we give the proofs of the application-results, i.e. Theorems 5, 6, and 7.

3.1 Proof of Theorem 5

Let us consider the following communication game: Alice is given a $u = (x_1, x_3, ..., x_{2n-1})$, Bob is given a $v = (x_2, x_4, ..., x_{2n})$, and they want to compute $IP(x_1, x_2, ..., x_{2n})$. Since, by assumption, $C_n$ computes $IP$, they will get the value of $IP(x_1, x_2, ..., x_{2n})$ by computing the output of $C_n$. For this, it is enough to compute every gate of $C_n$. A gate of $L_1$ norm $n^\nu$ can be computed by communicating $O(n^{2\nu}\delta)$ bits by Theorem 4, so the output of $C_n$ can be computed by communicating

$$c_0 \text{ size}(C_n)n^{2\nu}\delta$$

bits, and the error is at most $O(\text{size}(C_n)\exp(-c\delta))$, where $c_0$ is a positive constant.

We can apply here a lower bound result of Chor and Goldreich [9]:

**Theorem 10 [9]:** Suppose that probabilistic protocol $P$, computing $IP(x)$, has an average success probability at least

$$\frac{1}{2} + \epsilon$$

for some $\epsilon > \frac{1}{2e - 2}$, and the protocol communicates — for fixed $\epsilon$ and for fixed $n$ — always $\gamma_\epsilon(n)$ bits. Then

$$\gamma_\epsilon(n) > n - 3 - 3 \log \frac{1}{\epsilon}.$$
Now, setting $\delta = d \log n$ (with a large enough $d > 0$), from Theorem 10 and from (7):

$$\text{size}(C_n) = \Omega\left(\frac{n^{1-2\epsilon}}{\log n}\right).$$

\[\square\]

### 3.2 Proof of Theorem 6.

**Definition 11** [26] The randomized $\epsilon$-error complexity of Boolean function $f$, $R_{\epsilon}(f)$, is defined to be the cost of the best randomized protocol for $f$, which computes the correct answer with probability $1-\epsilon$. For a family $F$ of Boolean functions $f$, let

$$R_{\epsilon}(F) = \max_{f \in F} R_{\epsilon}(f).$$

We use for proving Theorem 6 the following lemma of Nisan ([26], Lemma 5) (it is stated here only in the 2-players case):

**Lemma 12** [26] Let $G$ be a family of Boolean functions. If $f$ can be computed as the MAJORITY of $s$ functions from $G$, then $R_{1/2+1/4\alpha}(f) \leq R_{1/4\alpha}(G)$.

\[\square\]

Now, let $f = \text{IP}$, let $G$ be the family of Boolean functions of $2n$ variables with $L_1$ norm of at most $n^\alpha$. Then, by Theorem 4,

$$R_{1/4\alpha}(G) = O(n^{2\alpha} \log s),$$

consequently, from Lemma 12:

$$R_{1/2+1/4\alpha}(\text{IP}) = O(n^{2\alpha} \log s),$$

and this, using Theorem 10, implies the statement of Theorem 6.\[\square\]
3.3 Proof of Theorem 7.

The proof of Theorem 7 is very similar to the proof of Theorem 5. The details are left to the reader.

Acknowledgment. The author is indebted to György Turán for a very useful discussion, and to Chi-Jen Lu for helpful comments. The author also acknowledges the support from grants OTKA 4271, F014919.

References


