Constructive Relationships Between Algebraic Thickness and Normality

Joan Boyar and Magnus Gausdal Find
Department of Mathematics and Computer Science
University of Southern Denmark
Campusvej 55, Odense, Denmark
{joan,magnusgf}@imada.sdu.dk

Abstract. We study the relationship between two measures of Boolean functions; algebraic thickness and normality. For a function $f$, the algebraic thickness is a variant of the sparsity, the number of nonzero coefficients in the unique $F_2$ polynomial representing $f$, and the normality is the largest dimension of an affine subspace on which $f$ is constant. We show that for $s < 3$, any function with algebraic thickness $n^s$ is constant on some affine subspace of dimension $\Omega\left(n^{\frac{s}{s-2}}\right)$. Furthermore, we give a randomized algorithm for finding such a subspace. In the literature, often so called $0$-restrictions are used. We show that finding the optimal $0$-restriction is $\mathbf{NP}$-hard. We give a construction which shows that for $2 < s < 3$, our algorithm finds an affine subspace that is at most a factor of $\Theta\left(\sqrt{n}\right)$ from the best guarantee, and restricted to the class of $0$-restrictions, it is at most a factor of $\Theta\left(\sqrt{\log n}\right)$ from the best guarantee.

1 Introduction and Known Results

Boolean functions play an important role in many fields of computer science. In cryptography, Boolean functions are sometimes classified according to some measure of complexity (also called cryptographic complexity [5], nonlinearity criteria [12] or nonlinearity measures [11]). Examples of such measures are nonlinearity, algebraic degree, normality, algebraic thickness and multiplicative complexity, and there are a number of results showing that if a function has a small value according to a certain measure, the function is vulnerable to a certain attack, (see [6] for a good survey).

A significant amount of work has been put into establishing relationships between these measures, e.g. considering questions of the form “if a function $f$ is simple (or complex) according to one measure, what does that say about $f$ according to some other measure”.

In this paper we focus on the relationship between algebraic thickness and normality. Both of these measures capture, each in their own way, how “different” functions are from being linear [4,5]. That is, there is an intuitive connection and it is natural to consider to which extent this relationship is formal. In fact, these two measures have been studied in the same papers previously (see e.g.
The relationship between these measures was initiated in the work of Cohen and Tal in [8], where they show that functions with a certain algebraic thickness have a certain normality. The proof is existential in nature and does not immediately imply an efficient algorithm - randomized nor deterministic - for witnessing this normality. For relatively small values of algebraic thickness, we tighten their bounds and furthermore present a randomized algorithm to witness this normality. The question of giving a constructive proof of normality is not just a theoretical one. Recently a generic attack on stream ciphers with low normality was successfully mounted in the work [14]. If it is possible to constructively compute a witness of normality given a function with low algebraic thickness, this implies that any function with low algebraic thickness is likely to be vulnerable to the attack in [14], as well as any other attack based on normality. Our work suggests that this is indeed possible for functions with small algebraic thickness.

2 Preliminaries and Known Results

Let $\mathbb{F}_2$ be the field of order 2, $\mathbb{F}_2^n$ the $n$-dimensional vector space over $\mathbb{F}_2$, and $[n] = \{1, \ldots, n\}$. A mapping from $\mathbb{F}_2^n$ to $\mathbb{F}_2$ is called a Boolean function. It is a well known fact that any Boolean function $f$ in the variables $x_1, \ldots, x_n$ can be expressed uniquely as a multilinear polynomial over $\mathbb{F}_2$ called the algebraic normal form (ANF) or the Zhegalkin polynomial. That is, there exist unique constants $c_{\emptyset}, \ldots, c_{\{1, \ldots, n\}}$, such that

$$f(x_1, \ldots, x_n) = \sum_{S \subseteq [n]} c_S \prod_{j \in S} x_j,$$

where arithmetic is in $\mathbb{F}_2$. In the rest of this paper, most arithmetic will be in $\mathbb{F}_2$, although we still need arithmetic in $\mathbb{R}$. If nothing is mentioned it should be clear from the context what field is referred to. The largest $|S|$ such that $c_S = 1$ is called the (algebraic) degree of $f$, and functions with degree 2 are called quadratic functions. We let $\log$ be the logarithm base two, $\ln$ the natural logarithm, and $\exp(\cdot)$ the natural exponential function with base $e$.

Algebraic Thickness For a Boolean function, $f$, let $\|f\| = \sum_{S \subseteq [n]} c_S$, with arithmetic in $\mathbb{R}$. This measure is sometimes called the sparsity of $f$ (e.g. [8]). The algebraic thickness [2,4] of $f$, denoted $\mathcal{T}(f)$ is defined as the smallest value of $\|f\|$ when some affine bijection has been applied to the inputs of $f$. More precisely, letting $A_n$ denote the set of affine, bijective operators on $\mathbb{F}_2^n$,

$$\mathcal{T}(f) = \min_{A \in A_n} \|f \circ A\|. \quad (1)$$

Algebraic thickness was introduced and first studied by Carlet in [2,4,3]. Affine functions have algebraic thickness at most 1, and Carlet showed that...
for any constant $c > \sqrt{\ln 2}$, for sufficiently large $n$ there exist functions with algebraic thickness
\[ 2^{n-1} - cn^{\frac{\ln 2}{n}} \]
and that a random Boolean function will have such high algebraic thickness with very high probability. Furthermore no function has algebraic thickness larger than $2^{n}$.

Carlet observes that algebraic thickness was also implicitly mentioned in [13, Page 208] and related to the so called “higher order differential attack” due to Knudsen [10] and Lai [11] in that they are dependant on the degree as well as the number of terms in the ANF of the function used.

Normality A $k$-dimensional flat is an affine (sub)space of $\mathbb{F}_2^n$ with dimension $k$. A function is $k$-normal if there exist a $k$-dimensional flat $E$ such that $f$ is constant on $E$ [12]. For simplicity define the normality of a function $f$, which we denote $N(f)$, as the largest $k$ such that $f$ is $k$-normal. We recall that affine functions have normality at least $n - 1$ (which is the largest possible for non-constant functions), while for any $c > 1$, a random Boolean function has normality $c \log n$ with high probability.

$k$-normal functions are often called affine dispersers of dimension $k$, and a great deal of work has been put into deterministic constructions of functions with low normality. Currently the asymptotically best known deterministic function, due to Shaltiel, has normality $2^{\log 0.9 n}$ [16].

Notice the asymmetry in our definitions: linear functions have very low algebraic thickness (0 or 1) but very high normality ($n$ or $n - 1$), whereas random functions, with high probability, have very high algebraic thickness (at least $2^{n-1} - 0.92 \cdot n \cdot 2^{-n^2}$) but low nonlinearity (less than $1.01 \log n$) [3].

Remark on Computational Efficiency In this paper, we say that something is efficiently computable if it is computable by a Turing machine in time polynomially bounded in the size of the input. Some algorithms in this paper will have a Boolean function with a certain algebraic thickness as input. In such cases, we assume that the function is represented by the ANF of the function witnessing this small algebraic thickness. That is, if a function $f$ with algebraic thickness $T(f) = T$ and $g = f \circ A$ for some affine bijection $A$, has $\|g\| = T$, the input to the algorithm is a representation of $A$ along with a description of the ANF of $g$. That is, in this setting, representing a function $f$ uses $\text{poly}(T(f) + n^2)$ bits. Besides that, we will not go further into details regarding the computational model or representation.

Quadratic Functions The normality and algebraic thickness of quadratic functions are well understood due to the following theorem due to Dickson [9] (see also [6] for a proof).

Theorem 1 (Dickson). Let $f : \mathbb{F}_2^n \to \mathbb{F}_2$ be quadratic. Then there exist an invertible $n \times n$ matrix $A$, a vector $b \in \mathbb{F}_2^n$, $t \leq \frac{n}{2}$, and $c \in \mathbb{F}_2$ such that for
\[ y = Ax + b \] one of the following two equations holds

\[ f(x) = y_1 y_2 + y_3 y_4 + \ldots + y_{t-1} y_t + c, \]  
\[ f(x) = y_1 y_2 + y_3 y_4 + \ldots + y_{t-1} y_t + y_{t+1}. \]

Furthermore, \( A, b \) and \( c \) can be found efficiently.

That is, any quadratic function is affine equivalent to some inner product function. We highlight a simple but useful consequence of Theorem 1. Simply by setting one variable in each of the degree two terms to zero, one gets:

**Proposition 1.** Let \( f : \mathbb{F}_2^n \to \mathbb{F}_2 \) be quadratic. Then \( N(f) \geq \left\lceil \frac{n+1}{2} \right\rceil \). Furthermore, a flat witnessing the normality of \( f \) can be found efficiently.

**Some Relationships** It was shown in [4] that normality and algebraic thickness are logically independent of (that is, not subsumed by) each other. Several other results relating these measures to algebraic thickness and normality are given in [4]. We mention a few relations to other measures.

Clearly, functions with degree \( d \) have algebraic thickness \( O(n^d) \), so having superpolynomial algebraic thickness requires superconstant degree. The fact that there exist functions with low degree and low normality has been established in [2] and [8] independently.

**Theorem 2 ([2,8]).** Let \( f \) on \( n \) variables be a random degree three polynomial (that is, all terms in the ANF occur independently with probability \( \frac{1}{2} \)). Then with high probability, \( f \) remains nonconstant on any subspace of dimension \( c\sqrt{n} \), for some fixed constant \( c > 0 \).

In fact, as mentioned in [8] it is not hard to generalize this to the fact that for any constant \( d \), a random degree \( d \) polynomial has normality \( O(n^{1/(d-1)}) \). The authors show that this is, in fact, tight. More precisely they give an elegant proof showing that any function with degree \( d \) has \( N(f) \in \Omega(n^{1/(d-1)}) \). They use this result to show the following relation between algebraic thickness and normality.

**Theorem 3 (Cohen and Tal [8]).** Let \( c \) be an integer and let \( f \) have \( T(f) \leq n^c \). Then \( N(f) \in \Omega(n^{1/(4c)}) \).

The proof goes in two steps: First they show by probabilistic methods that \( f \) has a restriction with a certain number of free variables and a certain degree, and after this they appeal to a relation between degree and normality. This latter result is nonconstructive in nature and does not naturally induce an algorithm.

### 3 A computational problem: Sparsest 0-restriction

In this section we will consider a particular kind of restriction of Boolean functions.

**Definition 1.** Let \( f : \mathbb{F}_2^n \to \mathbb{F}_2 \). Setting \( k < n \) of the bits to 0 results in a new function \( f' \) on \( n - k \) variables. We say that \( f' \) is a 0-restriction of \( f \).
By inspecting the proof in the next section and the proof of Theorem 3, one can see that most of the restrictions performed are in fact setting variables to 0. Furthermore, by inspecting the flat used for the attack performed in [14] (section 5.3), one can see that it is of this form as well. Given these observations it is natural to consider the computational complexity of the problem: Given a function represented by its ANF, reduce the number of terms as much as possible by setting (at most) \( k \) variables to 0. Formally we define the decision problem as follows:

**Definition 2.** Let the language 0-Restriction be as follows: A string is in 0-Restriction if it represents two integers \( \Delta, k \) and the ANF of a Boolean function, with the property that it is possible to set \( k \) variables to 0 and reduce the number of terms in the ANF by at least \( \Delta \).

This problem turns out to be \( \text{NP} \)-complete.

**Theorem 4.** 0-Restriction is \( \text{NP} \)-complete.

**Proof.** To see that the language is in \( \text{NP} \), notice that a certificate can simply be the list of variables set to 0. To show completeness, we reduce from Vertex Cover. Let the graph \( G = (V,E,k) \) be a vertex cover instance with vertices \( V \), edges \( E \) and the problem is to determine if there exists a set of \( k \) vertices such that every edge is incident to at least one such vertex. Let \( |V| = n \) and \( |E| = m \). From \( G \) we construct the ANF of a function on the variables \( x_1, \ldots, x_n \) as follows: for each edge \( (v_i, v_j) \) we include the term \( x_i x_j \).

We claim that \( G \) has a vertex cover of size \( k \) if and only if there exist \( k \) variables such that when set to 0 the number of terms is reduced by \( m \). Suppose \( \{v_{i_1}, \ldots, v_{i_k}\} \) is a vertex cover of \( G \). Setting the corresponding variables to 0 makes the function constantly zero, hence reduces the number of terms by \( m \). Conversely if there exist \( k \) variables such that when set to 0 they reduce the number of terms with \( m \), the corresponding vertices form a vertex cover in \( G \).

Interestingly, we notice that the instances produced in the reduction above are, in fact, quadratic functions. So finding a “small” 0-restriction that makes a quadratic function constant is \( \text{NP} \)-complete, whereas by Proposition 1 the corresponding question for general flats (as opposed to just 0-restrictions) is polynomial time solvable. To the best of our knowledge, the computational complexity of the problem of finding small flats for arbitrary functions represented by their ANF is open (see also [8]).

4 Algebraic Thickness and Normality

This section is devoted to showing that functions with algebraic thickness at most \( n^{3-\epsilon} \) are constant on flats of somewhat large dimensions. Furthermore our proof reveals a randomized polynomial time algorithm to find such a subspace. In the following, a term of degree at least 3 will be called a crucial term, and for a function \( f \), the number of crucial terms will be denoted \( T_{\geq 3}(f) \).
Our approach can be divided into two steps: First it uses 0-restrictions to obtain a quadratic function, and after this we can use Proposition 1. As implied by Theorem 4, finding the optimal 0-restrictions is indeed a computationally hard task. Nevertheless, as we shall show in this section, the following greedy algorithm gives reasonable guarantees. The greedy algorithm simply works by continually finding the variable that is contained in the largest number of terms, and sets this variable to 0. We begin with a simple proposition about the greedy algorithm that will be useful throughout the section, and it gives a tight bound.

**Proposition 2.** Let \( g : \mathbb{F}_2^n \rightarrow \mathbb{F}_2 \) have \( T^{\geq 3}(g) \geq m \). Then some variable \( x_j \) is contained in at least \( \lceil \frac{3m}{n} \rceil \) crucial terms.

**Proof.** Suppose for the sake of contradiction that each variable is contained in strictly less than \( \lceil \frac{3m}{n} \rceil \) crucial terms. Let \( v_i \) be the number of terms containing variable \( x_i \), and \( t_j \) be the number of variables contained in the \( j \)-th term, and suppose \( v^* = \max_{i \in [n]} v_i < \frac{3m}{n} \). By counting the number of variable occurrences in two different ways, we obtain the following inequalities:

\[
3m \leq \sum_{j=1}^{n} t_j = \sum_{i=1}^{n} v_i \leq n v^* < n \left( \frac{m}{n} \right) = 3m,
\]

arriving at the desired contradiction.

### 4.1 Case 1: Algebraic thickness at most \( O(n^2) \)

The following Lemma can be seen as a toy example of our main result of this section. The proof is simple but yet it contains the most important ideas for the subsequent results in the paper.

**Lemma 1.** Let \( c \leq \frac{4}{5} \) and let \( f : \mathbb{F}_2^n \rightarrow \mathbb{F}_2 \) have \( T^{\geq 3}(f) \leq cn \). Then \( f \) has a 0-restriction \( f' \) on \( n' \geq n - \lceil \frac{4c-1}{5}n \rceil \) variables with \( T^{\geq 3}(f') \leq \frac{c}{n} \).

**Proof.** Let the greedy algorithm run until a function \( f' \) on \( n' \) variables with \( T^{\geq 3}(f') \leq \frac{c}{n} \) is obtained. By Proposition 2 we eliminate at least 2 terms in each step. The number of algorithm iterations is at most \( \lceil \frac{4c-1}{5}n \rceil \). Indeed, let \( \lceil \frac{4c-1}{5}n \rceil = \frac{4c-1}{5}n + \delta \) for some \( 0 \leq \delta < 1 \). After this number of iterations the number of variables left is

\[
n' = n - \frac{3c - 1}{5} n - \delta = \frac{6 - 3c}{5} n - \delta
\]

and the number of critical terms is at most

\[
 cn - 2 \left( \frac{3c - 1}{5} n - \delta \right) = \frac{2 - c}{5} n - 2\delta.
\]

In particular \( \frac{n'}{n} \geq \frac{2 - c}{5} n - 2\delta \).
Lemma 1 is essentially tight. That is, for every rational $\frac{1}{3} \leq c \leq \frac{2}{3}$ there exist infinitely many functions where this number of restrictions is necessary. Let $\frac{1}{3} \leq c \leq \frac{2}{3}$ be fixed. Consider the function on 6 variables:

$$f(x) = x_1x_2x_3 + x_1x_4x_5 + x_2x_4x_6 + x_3x_5x_6.$$ 

Setting any one of the variables to 0 kills two terms and it is possible to kill all four terms by setting two variables to 0. Now consider the following function defined on $n = 30m$ variables. For convenience we index the variables by $x_{i,j}$ with $1 \leq i \leq 5m$, $1 \leq j \leq 6$.

$$g(x) = \sum_{i=1}^{5m} f(x_{i,1}, x_{i,2}, x_{i,3}, x_{i,4}, x_{i,5}, x_{i,6})$$

This function has $\mathcal{T}^{2,3}(g) = \frac{2}{3}n$. This shows that the lemma is tight for $c = \frac{2}{3}$.

To see that it is tight for $c < \frac{2}{3}$, consider the function the greedy algorithm has obtained at the point in time when the number of variables is $n'$ and $\mathcal{T}^{2,3}(g') = cn'$ (assuming $cn'$ is an integer). By the structure of this polynomial it is never possible to eliminate more than two terms per variable restriction, so the bound from Lemma 1 is tight.

Lemma 1 and its proof suggest the general principle: The more terms there are, the more terms are killed in each step of the greedy algorithm. This is the key idea in the proof of the following lemma.

**Lemma 2.** Let $f : \mathbb{F}_2^n \to \mathbb{F}_2$ have $\mathcal{T}^{2,3}(f) \leq tn$ for $t \leq \frac{n}{6}$. Then $f$ has a 0-restriction $f'$ on $n' \geq n \left(\frac{(3t-1)^2n-1}{6t-1}\right)^2 - 1$ variables with $\mathcal{T}^{2,3}(f') \leq n'\frac{2}{3}$.

**Proof.** We consider the greedy algorithm at the point where it has $n'$ free variables and at most $\frac{2}{3}n'$ crucial terms. For the sake of analysis, we divide the execution of the algorithm into phases. At any point in time let $f'$ (resp. $n'$) denote the current function (resp. the number of free variables). The algorithm is in phase $i$ when

$$\frac{(i-1)}{3}n' < \mathcal{T}^{2,3}(f') \leq \frac{i}{3}n',$$

thus we start in phase $3t$, then go to phase $3t - 1$, etc. We let $n_i$ be $n'$ when the algorithm enters phase $i$, so initially $n = n_{3t}$. A priori it is not even clear that the algorithm will ever leave phase $3t$. We will rely heavily on the following claim.

**Claim.** The number of steps in phase $i$ is at most $\left\lfloor \frac{n_i}{2i+1} \right\rfloor$.

**Proof (Proof of claim).** As long as the algorithm is in phase $i$, Proposition 2 gives that at least one variable, say $x_q$, occurs in at least $i$ different crucial terms. By setting $x_q = 0$, the number of crucial terms decreases by at least $i$. Let $0 \leq \delta < 1$ such that $\left\lceil \frac{n_i}{2i+1} \right\rceil = \frac{n_i}{2i+1} + \delta$. After $\left\lfloor \frac{n_i}{2i+1} \right\rfloor$ iterations the number of crucial terms left is at most

$$T \leq \frac{i}{3}n_i - i \left\lfloor \frac{n_i}{2i+1} \right\rfloor = \frac{n_i2(i-1)}{6i+3} - \delta i,$$
and the number of free variables left is
\[ n' = n_i - \left[ \frac{1}{2i+1} n_i \right] = n_i - \frac{n_i}{2i+1} - \delta = n_i \left( \frac{2i}{2i+1} \right) - \delta, \]
so in particular
\[ \frac{i - 1}{3} n' = \frac{n_i 2i(i - 1)}{6i + 3} - \delta \frac{i - 1}{3} \geq T, \]
with equality only if \( \delta = 0 \), proving the claim.

Notice that if we through the entire execution of the algorithm have that
\( 2i + 1 \) divides \( n_i \), the number of variables when entering \( n_1 \) would be at least
\[ n_1 \geq n \prod_{j=1}^{3t-1} \frac{2j}{2j+1} = n \frac{(3t-1)! 2^{3t-1} 2}{(6t-1)!}. \]

The question remaining to consider is when \( 2i + 1 \) does not divide \( n_i \), possibly several times during the execution. In terms of the behavior of the algorithm this corresponds to the fact that in the last variable restriction in a phase \( i \), the greedy algorithm kills more terms than what is needed to leave phase \( i \). We now give an amortized argument showing that for all but the last phase this is only better for the algorithm than in the case of Equation 2.

For each restriction of a variable \( x_a \) in a phase \( i > 1 \), if \( s \) terms are killed, we transfer \( \frac{s}{i} \) units of credit to \( x_a \). The goal is now to show that each variable killed receives at least 1 unit of credit. If each variable receives exactly 1 unit of credit, we meet the bound from Equation 2 with equality, and the more variables that receive more than 1 unit of credit, the more pessimistic the bound. Suppose at some point the division does not go up and the greedy algorithm restricts a variable \( x_a \), eliminates \( s \) terms and is left with \( (i - 1)n' - s_1 \) terms and \( n' \) variables for some \( 0 < s_1 < s \). We bookkeep this by letting \( s_1 \) terms count as if they were killed in phase \( i - 1 \), so \( x_a \) gets the transfer of
\[ \frac{s - s_1}{i} + \frac{s_1}{i - 1} > \frac{s}{i}. \]

By the nature of the greedy algorithm and Proposition 2, \( s \geq i \), so \( x_a \) receives strictly more than 1 unit of credit. By doing this analysis, and possibly rounding up in the last phase, we end up with the desired statement.

When \( t \) is superconstant this can be approximated by a simpler expression. In the proof of the next theorem we will rely on Stirling’s approximation [15] that for \( m \geq 1 \)
\[ \sqrt{2\pi mn^m e^{-m} e^{1/(12m+1)}} \leq m! \leq \sqrt{2\pi nm} e^{-m} e^{1/12m}, \]

Theorem 5. Let \( 5 \leq t \leq n/6 \), let \( T^{2/3}(f) \leq tn \). Then for sufficiently large \( n \), \( f \) has a 0-restriction \( f' \) on at least \( n' \geq \left\lceil \frac{n}{\sqrt{6m-1}} - \frac{1}{3} \right\rceil \) variables, which is constant. Furthermore this flat can be found by the greedy algorithm.
Proof. By Lemma 2 the greedy algorithm finds such a function with \( n' \geq n (3t-1)b^{3t-1} - 1 \). Using Equation 3

\[
\frac{n (3t-1)b^{3t-1}}{(6t-1)!} - 1 \geq n \left( \frac{\sqrt{2\pi(3t-1)(3t-1)e^{-(3t-1)}e^{1/(12(3t-1)+1)}2^{3t-1}}}{\sqrt{2\pi(6t-1)(6t-1)e^{-(6t-1)}e^{1/12(6t-1)}}} - 1 \right)
\]

\[
= n \frac{2\pi(3t-1)(3t-1)e^{-(3t-1)}e^{1/(12(3t-1)+1)}2^{3t-1}}{\sqrt{2\pi(6t-1)(6t-1)e^{-(6t-1)}e^{1/12(6t-1)}}} - 1
\]

\[
= n \sqrt{2\pi} \frac{1}{\sqrt{6t-1}} \left( \frac{3t-1}{6t-1} \right)^{6t-1} e^{2/(12(3t-1)+1)} e^{-6t+2e^{1/(12(3t-1)+1)}2^{3t-1}}
\]

\[
= n \frac{1}{2} \sqrt{2\pi} \frac{1}{\sqrt{6t-1}} \left( \frac{3t-1}{6t-1} \right)^{6t-1} e^{2/(12(3t-1)+1)} e^{-6t+2e^{1/(12(3t-1)+1)}2^{3t-1}} - 1.
\]

When \( t \geq 5 \) we have

\[
\left( \frac{3t-1}{6t-1} \right)^{6t-1} e^{2/(12(3t-1)+1)} \geq 0.36,
\]

so we have that the number of variables is at least \( ne^{1/2} \sqrt{2\pi} 0.36 e^{-6t+2e^{1/(12(3t-1)+1)}2^{3t-1}} \geq \frac{1}{2} \frac{1.22}{\sqrt{6t-1}} - 1 \). Now there are at most \( \frac{n}{\pi} \) crucial terms left. The greedy algorithm can kill all these terms with at most \( \frac{n}{\pi} \) variable restrictions, hence at the point in time when there are no crucial terms left, the number of variables is at least

\[
\left[ n \frac{1.22}{\sqrt{6t-1}} - \frac{1}{3} \right].
\]

Now one can use Proposition 1 to obtain a flat of dimension at least

\[
\left[ \frac{n - 0.40}{\sqrt{6t-1}} - \frac{1}{3} \right]
\]

on which \( f \) is constant, proving the theorem.

It should be noted that when \( t \) is large one can get a better bound than in Equation 1. More precisely similar calculations show that

\[
\lim_{t \to \infty} \frac{(3t-1)b^{3t-1}}{(6t-1)!} \sqrt{t} = \sqrt{\frac{\pi}{12}}.
\]

4.2 Case 2: Superquadratic Algebraic Thickness

The analysis from Lemma 2 does not immediately extend to the case where \( T^{\geq 3}(f) \geq 2n^2 \) simply because the number of phases alone is larger than \( n \). For this case we invoke a different analysis of the same greedy algorithm.

**Lemma 3.** Let \( f : \mathbb{F}_2^n \to \mathbb{F}_2 \) and \( T^{\geq 3}(f) \leq 2n^2 \). Then \( f \) has a 0-restriction \( f' \) on \( n' \) variables and \( T^{\geq 3}(f') \leq 4(n')^2 \) with

\[
n' \geq n \sqrt{\frac{2}{(t-3)(t-2)}} - 1.
\]
Proof. As in the proof of Lemma \[ \text{2} \], we divide the execution into phases, but now we say that the algorithm is in phase \( i \) when the number of crucial terms satisfy

\[ i n^2 < T^{2^3}(f') \leq (i + 1)n^2, \]

and as before we let \( n_i \) be the number of variables left when the algorithm enters phase \( i \). Our goal is to show that at the point in the time when the algorithm enters phase 3, the number of free variables is as desired. By Proposition \[ \text{2} \], in each iteration, the greedy algorithm reduces the number of terms by at least \( 3(\alpha n \geq \frac{\alpha^2 n^2 + 2\alpha n^2}{2}) \).

We leave the phase when this number is at most \( i ((1 - \alpha)n) \). Notice that this holds with equality when \( \alpha = \frac{i}{3 - \sqrt{i^2 - 2i}} = 1 - \sqrt{\frac{i - 2}{i}} \).

We conclude that for \( i \geq 3 \), after \( \left[ (1 - \sqrt{i^2 - 2i}) n \right] \) iterations the greedy algorithm leaves phase \( i \). Similar to the proof of Lemma \[ \text{2} \], we are interested in an upper bound on the number of iterations used by the greedy algorithm. Hence by the same reasoning we can assume that the number of iterations in all phases except for the last one is exactly \( (1 - \sqrt{i^2 - 2i}) n \) and the number of iterations in phase 4 is \( (1 - \sqrt{\frac{i - 2}{2}}) n + 1 \). Hence we have that

\[ n_3 \geq \prod_{j=3}^{t-1} \sqrt{\frac{j - 2}{j}} - 1 \geq n \sqrt{\frac{(t - 3)!}{(t - 1)!}} - 1 = n \sqrt{\frac{2}{(t - 3)(t - 2)}} - 1. \]

Which proves the Lemma.

We conclude this section by collecting the preceding results into one single algorithm that given a function with low algebraic thickness finds a large flat on which the function is constant. The algorithm is partially randomized, with zero error and expected polynomial running time.

**Theorem 6.** There exists a randomized algorithm with expected polynomial running time that given \( f : \mathbb{F}_2^n \to \mathbb{F}_2 \) with \( T^{2^3}(f) \leq n^s \) for \( 1 < s < 3 \) finds a flat of dimension \( \Omega \left( n^{2^{-s}} \right) \) on which \( f \) is constant.

**Proof.** If \( T^{2^3}(f) \leq \frac{n^2}{4} \), the result follows directly from Theorem \[ \text{5} \], so suppose \( T^{2^3}(f) > \frac{n^2}{4} \) and let \( \tau = s - 2 \). Now if \( T^{2^3}(f) > 4n^2 \), we use Lemma \[ \text{5} \] with \( t = n^\tau \) to obtain a 0-restriction \( f' \) with \( T^{2^3}(f') \leq 4n^2 \) and \( n' \in \Omega \left( n^{1-\tau} \right) \). This is too high in order to invoke Theorem \[ \text{5} \]. Now we set each variable independently
to 0 with probability 0.99. Every fixed crucial term survives this restriction with probability at most $0.01^3 = 0.000001$. So the expected number of variables after this restriction is $0.01n'$ while the number of expected crucial terms is at most $0.000004n'^2$. It follows from Chernoff bounds that for large $n$, the number of variables is $n'' \leq 0.0099n'$ with probability less than 0.01. Markov’s inequality implies that the probability of having $T \geq 3(f'') \geq 0.000005n'^2$ is at most 0.8.

That is, with probability at least 0.19 we obtain a function $f''$ where the number of crucial terms left is at most $0.000005n'^2$ and the number of variables left is at least $0.099n'$. In expectation, after $\Theta(1)$ attempts, such a function is obtained. In this case we have

$$T \geq 3(f'') \leq 0.000005n'^2 < \frac{(0.0099n')^2}{6} \approx 0.0000163n'^2,$$

so we can now apply Proposition 1 to find a flat of dimension

$$\Omega \left( \sqrt{n''} \right) = \Omega \left( n^{\frac{1}{2}} \right).$$

This improves on Theorem 3 for functions with algebraic thickness $n^s$ for $s \leq 2.83$, and the smaller $s$, the bigger the improvement, e.g. for $T(f) \leq n^2$, our bound guarantees $N(f) \in \Omega(n^1/2)$, compared to $\Omega(n^{1/8})$.

4.3 Normal Functions with low sparsity

There are at least two obvious questions to ask when faced with Theorem 6. First, can the algorithm can be modified to work with functions of higher algebraic thickness, that is, $T \geq 3(f) \in \Omega(n^3)$? Consider the function whose ANF contains all terms of degree 3. Setting any number of variables constant results in a function on $n'$ variables and $\binom{n'}{3}$ crucial terms. This already shows that if one wants to improve Theorem 6 to hold for the cubic case, one needs to do something more sophisticated than 0-restrictions.

Second, to what extent can the algorithm be improved to find flats of larger dimensions? The purpose of this section is first to show that the result from Theorem 6 is at most a factor of $\Theta(\sqrt{n})$ from being tight. More precisely, we show that for any $2 < s < 3$ there exist functions with thickness at most $n^s$ that are nonconstant on flats of dimension $O(n^{2-s})$. Notice that when $s = 3$, this result implies Theorem 2.

**Theorem 7.** For any $2 < s < 3$, for sufficiently large $n$, there exist functions with degree 3 and algebraic thickness at most $n^s$ that remain nonconstant on all flats of dimension $4.3n^{2-s}$.

**Proof.** The proof uses the probabilistic method. We endow the set of all degree 3 Boolean functions with a probability distribution $D$, and show that under this distribution a function has the promised normality with high probability.

The proof is divided into the following steps. First we describe the probability distribution $D$. Then, we fix an arbitrary $k$-dimensional flat $E$, and bound
the probability that a random $f$ chosen according to $\mathcal{D}$ is constant on $E$. We show that for $k = Cn^{2-s/2}$, where $C$ is a constant to be determined later, this probability is sufficiently small that a union bound over all possible choices of $E$ gives that there exist a function $f$ with normality at least $k$.

We define $\mathcal{D}$ by describing the probability distribution on its ANF. We let each possible degree 3 term be included with probability $\frac{1}{n^3 - s}$. The expected number of terms is thus $n^s - \frac{s}{2}$, and the probability of having more than $n^s$ terms is less than 0.001 for large $n$. Now let some $k$-dimensional flat $E$ be arbitrary but fixed.

One way to think of a function restricted to a $k$-dimensional flat is that it can be obtained by a sequence of $n - k$ affine variable substitutions of the form $x_i := \sum_{j \in S} x_j + c$. This changes the ANF of the function since $x_i$ is no longer a “free” variable. Assume without loss of generality that we substitute for the variables $x_n, \ldots, x_{k+1}$ in that order. Initially we start with the function $f$ given by

$$f(x) = \sum_{\{a,b,c\} \subseteq [n]} I_{abc} x_a x_b x_c,$$

where $I_{abc}$ is the indicator random variable, indicating whether the $x_a x_b x_c$ is contained in the ANF. Suppose we perform the $n - k$ restrictions and obtain the function $\tilde{f}$. The ANF of $\tilde{f}$ is given by

$$\tilde{f}(x) = \sum_{\{a,b,c\} \subseteq [k]} \left( I_{abc} + \sum_{s \in S_{abc}} I_s \right) x_a x_b x_c,$$

where $S_{abc}$ is some set of indicator random variables depending on the restrictions performed. It is important that the indicator random variable corresponding to $x_a x_b x_c$ for $\{a, b, c\} \subseteq [k]$ is only occurring at $x_a x_b x_c$. Hence we conclude that independently of the outcome of all the indicator random variables $I_{a'b'c'}$ with

$$\{a', b', c'\} \not\subseteq [k],$$

we have that the marginal probability for any $I_{abc}$ with $\{a, b, c\} \subseteq [k]$ occurring remains at least $\frac{1}{n^s}$.

Define $t = \binom{k}{3}$ random variables, $Z_1, \ldots, Z_t$, one for each potential term in the ANF of $\tilde{f}$, such that $Z_j = 1$ if and only if the corresponding term is present in the ANF, and 0 otherwise. The obtained function is only constant if there are no degree 3 terms, so the probability of $\tilde{f}$ being constant is thus at most
\[ \mathbb{P}[Z_1 = \ldots = Z_t = 0] \leq \left(1 - \frac{1}{n^{3-s}}\right)^t \]

\[ = \left(1 - \frac{1}{n^{3-s}}\right)^{\binom{t}{k}} \]

\[ \leq \left(1 - \frac{1}{n^{3-s}}\right)^{\frac{C^3}{27} (n^{6-3s/2})} \]

\[ = \left(\left(1 - \frac{1}{n^{3-s}}\right)^{n^{3-s}}\right)^{\frac{C^3}{27} (n^{3-s/2})} \]

\[ \leq \exp\left(-\frac{C^3}{27} n^{3-s/2}\right). \]

The number of choices for \( E \) is at most \( 2^n (k+1) \), so the probability that \( f \) becomes constant on some affine flat of dimension \( k \) is at most

\[ \exp\left(-C^3\frac{2^k}{27} n^{3-s/2}\right) \exp\left(\ln(2)nC(n^{2-s/2} + 1)\right) \]

\[ = \exp\left(-\frac{C^3}{27} n^{3-s/2} + C \ln(2)n^{3-s/2} + n\right). \]

Now if \( C > 3\sqrt{3} \ln(2) \approx 4.33 \), this quantity tends to 0. We conclude that with high probability the function obtained has algebraic thickness at most \( n^s \) and normality at least 4.33 \( n^{2-\frac{s}{2}} \).

There is factor of \( \Theta(\sqrt{n}) \) between the existence guaranteed by Theorem 6 and the upper bound given in Theorem 7 and we leave it as an interesting problem to close this gap.

The algorithm studied in this paper works by setting variables to 0 until all remaining terms have degree at most 2, and after that appealing to Theorem 1. It turns out that the construction from the proof of the above theorem shows that among such algorithms, the upper bound from Theorem 6 is very close to being asymptotically tight.

**Theorem 8.** For any \( 2 < s < 3 \), there exist functions with degree 3 and algebraic thickness at most \( n^s \) that remain nonconstant on any 0-restriction of dimension \( 3\sqrt{\ln n} n^{\frac{3-s}{2}} \).

**Proof.** We use the same proof strategy as in the proof of Theorem 7. Endow the set of all degree 3 Boolean functions with the same probability distribution \( \mathcal{D} \). For large \( n \), the number of terms is larger than \( n^s \) with probability at most 0.001. Now we set all but \( C\sqrt{\ln n} n^{\frac{s}{2}} \) of the variables to 0, and consider the probability of the function being constant under this fixed 0-restriction. We will show that this probability is so small that a union bound over all such choices gives that
with high probability the function is nonconstant under any such restriction. At the end of the proof we will see that setting $C = 3$ will suffice. There are $(C\sqrt{\ln n n^{3-s}})$ possible degree 3 terms on these remaining variables, and each one is included with probability $\frac{1}{n^{3-s}}$. The function is constant on this 0-restriction only if no such term is included, and this happens with probability

$$
\left(1 - \frac{1}{n^{3-s}}\right)^{(C\sqrt{\ln n n^{2-s}})} \leq \left(1 - \frac{1}{n^{3-s}}\right)^{\frac{C^3}{27}\left(\sqrt{\ln n} n^{2-s}\right)}
$$

$$
= \left(1 - \frac{1}{n^{3-s}}\right)^{n^{3-s}\left(\ln n\right)^{3/2}}
$$

$$
\leq \exp\left(\frac{C^3}{27}\left(\sqrt{\ln n} (\ln n)^{3/2}\right)\right),
$$

and the number of 0-restrictions with all but $C\sqrt{\ln n n^{3-s}}$ variables fixed is

$$
\left(\frac{n}{C\sqrt{\ln n n^{3-s}}}\right)^{n^{3-s}C\sqrt{\ln n n^{2-s}}} \leq \left(\frac{n^{3-s}C\sqrt{\ln n n^{2-s}}}{n^{3-s}C\sqrt{\ln n n^{2-s}}!}\right)
$$

$$
= \exp\left(\ln n^{3-s}C\sqrt{\ln n n^{2-s}} - \ln((C\sqrt{\ln n n^{2-s}})!)/n^{3-s}C\sqrt{\ln n n^{2-s}}!\right)
$$

$$
\leq \exp\left(\ln n^{3-s}C\sqrt{\ln n n^{2-s}} - 0.99\ln(C\sqrt{\ln n n^{2-s}})\right)
$$

$$
\leq \exp\left(\ln n^{3-s}C\sqrt{\ln n n^{2-s}} - \frac{3-s}{2}n^{3-s}C\sqrt{\ln n n^{2-s}}\right)
$$

$$
= \exp\left(\ln n^{3-s}C\sqrt{\ln n n^{2-s}}\left(1 - 0.98\frac{3-s}{2}\right)\right),
$$

where the last two inequalities hold for sufficiently large $n$. Now, letting $C \geq 3$ shows that for large enough $n$ this probability tends to zero, hence we have that with high probability the function does not have a 0-restriction on $3\sqrt{\ln n n^{2-s}}$ variables.

5 Open Questions

We end this paper with some open questions. Although the gap between the results in Theorem 6 and Theorem 8 is relatively small, it would be interesting to narrow it or close it: Is it possible to come up with an algorithm that finds a larger 0-restriction with no crucial terms? Even a stronger nonconstructive relationship would be interesting.

It seems even more interesting to close the gap between Theorem 6 and Theorem 7. We suspect that it could be possible to use the full strength of
affine restrictions to make an algorithm giving better guarantees than the one in Theorem 6. It might also be possible to improve the bound in Theorem 7.

The algorithm in Theorem 6 is randomized, but randomization is only used in one step. It would be interesting to give a completely deterministic version of the algorithm.

References

1. Joan Boyar, Magnus Find, and René Peralta. Four measures of nonlinearity. In Paul G. Spirakis and Maria J. Serna, editors, CIAC, volume 7878 of Lecture Notes in Computer Science, pages 61–72. Springer, 2013. Eprint with correction available at the Cryptology ePrint Archive, Report 2013/633, 2013, http://eprint.iacr.org/.

2. Claude Carlet. On the complexity of cryptographic boolean functions. In 6th Conference on Finite Fields and Applications (GL Mullen, H. Stichtenoth, and H. Tapia-Recillas, eds.) Springer-Verlag, pages 53–69, 2002.

3. Claude Carlet. On the algebraic thickness and non-normality of boolean functions. In Information Theory Workshop, 2003. Proceedings. 2003 IEEE, pages 147–150. IEEE, 2003.

4. Claude Carlet. On the degree, nonlinearity, algebraic thickness, and nonnormality of boolean functions, with developments on symmetric functions. IEEE Transactions on Information Theory, 50(9):2178–2185, 2004.

5. Claude Carlet. The complexity of boolean functions from cryptographic viewpoint. In Matthias Krause, Pavel Pudlák, Rüdiger Reischuk, and Dieter van Melkebeek, editors, Complexity of Boolean Functions, 12.03. - 17.03.2006, volume 06111 of Dagstuhl Seminar Proceedings. Internationales Begegnungs- und Forschungszentrum fuer Informatik (IBFI), Schloss Dagstuhl, Germany, 2006.

6. Claude Carlet. Boolean functions for cryptography and error correcting codes. In Yves Crama and Peter L. Hammer, editors, Boolean Models and Methods in Mathematics, Computer Science, and Engineering, chapter 8, pages 257–397. Cambridge, UK: Cambridge Univ. Press, 2010.

7. Pascale Charpin. Normal boolean functions. J. Complexity, 20(2-3):245–265, 2004.

8. Gil Cohen and Avishay Tal. Two structural results for low degree polynomials and applications. CoRR, abs/1404.0654, 2014.

9. Leonard Eugene Dickson. Linear Groups with and with an Exposition of the Galois Field Theory. Teubner’s Sammlung von Lehrbuchern auf dem Gebiete der matematischen Wissenschaften, VL:x+312, 1901.

10. Lars R Knudsen. Truncated and higher order differentials. In Fast Software Encryption, pages 196–211. Springer, 1995.

11. Xuejia Lai. Higher order derivatives and differential cryptanalysis. In Communications and Cryptography, pages 227–233. Springer, 1994.

12. Willi Meier and Othmar Staffelbach. Nonlinearity criteria for cryptographic functions. In Jean-Jacques Quisquater and Joos Vandewalle, editors, Advances in Cryptology - EUROCRYPT ’89, Workshop on the Theory and Application of of Cryptographic Techniques, Houthalen, Belgium, April 10-13, 1989, Proceedings, volume 434 of Lecture Notes in Computer Science, pages 549–562. Springer, 1989.

13. Alfred Menezes, Paul C. van Oorschot, and Scott A. Vanstone. Handbook of Applied Cryptography. CRC Press, 1996.
14. Miodrag J. Mihaljevic, Sugata Gangopadhyay, Goutam Paul, and Hideki Imai. Generic cryptographic weakness of $k$-normal boolean functions in certain stream ciphers and cryptanalysis of grain-128. *Periodica Mathematica Hungarica*, 65(2):205–227, 2012.

15. Herbert Robbins. A remark on Stirling’s formula. *American Mathematical Monthly*, pages 26–29, 1955.

16. Ronen Shaltiel. Dispersers for affine sources with sub-polynomial entropy. In Rafail Ostrovsky, editor, *IEEE 52nd Annual Symposium on Foundations of Computer Science, FOCS 2011, Palm Springs, CA, USA, October 22-25, 2011*, pages 247–256. IEEE, 2011.