The degree of a Boolean function and some algebraic properties of its support

J.-J. Climent[1], F. J. García[2] & V. Requena[3]
[1]*Departament d'Estadística i Investigació Operativa, Universitat d'Alacant, Spain*
[2]*Departament de Mètodes Quantitatius i Teoria Econòmica, Universitat d'Alacant, Spain*
[3]*Departamento de Estadística, Matemáticas e Informática, Universidad Miguel Hernández de Elche, Spain*

Abstract

In this paper, the support of a Boolean function is used to establish some algebraic properties. These properties allow the degree of a Boolean function to be obtained without having to calculate its algebraic normal form. Furthermore, some algorithms are derived and the average time computed to obtain the degree of some Boolean functions from its support.
Keywords: Boolean function, support, weight, algebraic normal form, degree.

1 Introduction

Boolean functions are used in cryptographic applications such as block ciphers, stream ciphers, hash functions [1–3], and coding theory [4–6], among others. One of the basic requirements relative to the Boolean functions used in stream ciphers is that they allow to increase the linear complexity [7–9], which is obtained if these functions have a high algebraic degree.

Boolean functions can be represented in many ways; the most commonly used are the algebraic normal form or the truth table. The algebraic normal form of a Boolean function provides its degree directly, but not its weight; on the other hand, if we know the truth table, then we know its weight, but do not know its degree.

In this paper we introduce some properties that allow us to compute the degree of a Boolean function without computing all the coefficients of its algebraic normal form.

WIT Transactions on Information and Communication Technologies, Vol 45, © 2013 WIT Press
www.witpress.com, ISSN 1743-3517 (on-line)
doi:10.2495/DATA130031

The rest of the paper is organized as follows. In Section 2 we introduce some basic definitions and notations that are used hereafter. In Section 3, we introduce some linear algebra properties that allow us to improve the algorithm introduced in Section 2. Section 4 is devoted to numerical results and finally, in Section 5 we present the conclusions.

2 Preliminaries

We denote by $\mathbb{F}_2$ the Galois field of two elements, 0 and 1, with the addition (denoted by $\oplus$) and the multiplication (denoted by juxtaposition). For any positive integer n, it is well-known that $\mathbb{F}_2^n$ is a linear space of dimension n over $\mathbb{F}_2$ with the usual addition (denoted also by $\oplus$). We denote by $\operatorname{Span}\{\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_k\}$ the linear subspace of $\mathbb{F}_2^n$ generated by the vectors $\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_k \in \mathbb{F}_2^n$. If we denote by $\boldsymbol{i}$ the binary expansion of n digits of the integer i, for $i = 0, 1, 2, \ldots, 2^n - 1$, then

$$\mathbb{F}_2^n = \{\boldsymbol{i} \mid 0 \leq i \leq 2^n - 1\}.$$

With this notation, the standard basis of $\mathbb{F}_2^n$ is $\{\mathbf{2^{n-1}}, \mathbf{2^{n-2}}, \ldots, \mathbf{2^2}, \mathbf{2}, \mathbf{1}\}$.

For a vector $\boldsymbol{u} \in \mathbb{F}_2^n$ we denote by $S(\boldsymbol{u})$ the linear subspace of $\mathbb{F}_2^n$ spanned by the vectors of the standard basis of $\mathbb{F}_2^n$ corresponding to the nonzero components of $\boldsymbol{u}$; that is, if we assume that $u_{i_1}, u_{i_2}, \ldots, u_{i_k}$, with $1 \leq i_1 < i_2 < \cdots < i_k \leq n$, are the nonzero components of $\boldsymbol{u} = (u_1, u_2, \ldots, u_n)$, then

$$\boldsymbol{u} = \mathbf{2^{n-i_1}} \oplus \cdots \oplus \mathbf{2^{n-i_k}} \quad \text{and} \quad S(\boldsymbol{u}) = \operatorname{Span}\left\{\mathbf{2^{n-i_1}}, \ldots, \mathbf{2^{n-i_k}}\right\}.$$

We say that k, that is, the number of nonzero components of vector $\boldsymbol{u}$, is the **weight** of $\boldsymbol{u}$. It is evident that $\dim S(\boldsymbol{u}) = w(\boldsymbol{u})$.

If $F \subseteq \mathbb{F}_2^n$ and $\boldsymbol{a} \in \mathbb{F}_2^n$, then $\boldsymbol{a} \oplus F = \{\boldsymbol{a} \oplus \boldsymbol{u} \mid \boldsymbol{u} \in F\}$. When F is a k-dimensional linear subspace of $\mathbb{F}_2^n$ we say that $\boldsymbol{a} \oplus F$ is the **k-dimensional affine subspace** of $\mathbb{F}_2^n$ passing through $\boldsymbol{a}$ in the direction of F.

For $1 \leq k < n$, we consider that $\mathbb{F}_2^n = \mathbb{F}_2^k \times \mathbb{F}_2^{n-k}$. So, if $\boldsymbol{u} \in \mathbb{F}_2^n$, then $\boldsymbol{u} = (\boldsymbol{v}, \boldsymbol{w})$ with $\boldsymbol{v} \in \mathbb{F}_2^k$ and $\boldsymbol{w} \in \mathbb{F}_2^{n-k}$. In particular, if $\boldsymbol{a} \in \mathbb{F}_2^{n-k}$, we also denote by $\boldsymbol{a}$ the vector $(\mathbf{0}, \boldsymbol{a}) \in \mathbb{F}_2^n$.

A **Boolean function** of n variables is a map $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$. The set of all Boolean functions of n variables is denoted by $\mathcal{B}_n$; it is well known that $\mathcal{B}_n$, with the usual addition of functions (that we also denote by $\oplus$), is a linear space of dimension 2^n over $\mathbb{F}_2$, so $|\mathcal{B}_n| = 2^{2^n}$. The complementary function of $f \in \mathcal{B}_n$ is the Boolean function $1 \oplus f$ given by $(1 \oplus f)(\boldsymbol{x}) = 1 \oplus f(\boldsymbol{x})$ for all $\boldsymbol{x} \in \mathbb{F}_2^n$.

If $f \in \mathcal{B}_n$, we call **truth table** of f (see, for example, [10, 11]) the binary sequence of length 2^n given by $\boldsymbol{\xi} = (f(\mathbf{0}), f(\mathbf{1}), \ldots, f(\mathbf{2^n - 1}))$. We call **weight** of f, denoted by $w(f)$, the number of 1s of the truth table of f.

The **support** of f, denoted by $\operatorname{Supp}(f)$, is the set of vectors of $\mathbb{F}_2^n$ whose image by f is 1, that is, $\operatorname{Supp}(f) = \{\boldsymbol{a} \in \mathbb{F}_2^n \mid f(\boldsymbol{a}) = 1\}$ and therefore, $w(f) = |\operatorname{Supp}(f)|$. Obviously, f is the null function if and only if $\operatorname{Supp}(f) = \emptyset$ and then $w(f) = 0$; analogously, f is the constant function 1 if and only if $\operatorname{Supp}(f) = \mathbb{F}_2^n$ and, in this case, $w(f) = 2^n$.

It can be checked that if $f, g \in \mathcal{B}_n$, then $\mathrm{Supp}\,(f \oplus g) = \mathrm{Supp}\,(f)\,\Delta\,\mathrm{Supp}\,(g)$, where Δ denote the symmetric difference of sets and, as a consequence,

$$w(f \oplus g) \equiv w(f) + w(g) \pmod 2.$$

In general, if $f_j \in \mathcal{B}_n$, for $j = 1, 2, \ldots, m$, then

$$\mathrm{Supp}\left(\bigoplus_{j=1}^{m} f_j\right) = \bigtriangleup_{j=1}^{m} \mathrm{Supp}\,(f_j) \tag{1}$$

and, therefore,

$$w\left(\bigoplus_{j=1}^{m} f_j\right) \equiv \sum_{j=1}^{m} w(f_j) \pmod 2. \tag{2}$$

Furthermore, $\mathrm{Supp}\,(1 \oplus f) = \mathbb{F}_2^n \setminus \mathrm{Supp}\,(f)$ and $w(1 \oplus f) = 2^n - w(f)$.

We say that f is **balanced** if $w(f) = 2^{n-1}$. It is evident that f is balanced if and only if $1 \oplus f$ is balanced.

Now assume that $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$ where each x_j, for $j = 1, 2, \ldots, n$, is a binary variable. If $f \in \mathcal{B}_n$, then we can write $f(\boldsymbol{x})$ uniquely as (see, for example, [5, 10–13])

$$f(\boldsymbol{x}) = \bigoplus_{\boldsymbol{u} \in \mathbb{F}_2^n} \mu_f(\boldsymbol{u}) \boldsymbol{x}^{\boldsymbol{u}} \tag{3}$$

where $\mu_f(\boldsymbol{u}) \in \mathbb{F}_2$, and if $\boldsymbol{u} = (u_1, u_2, \ldots, u_n)$, then

$$\boldsymbol{x}^{\boldsymbol{u}} = x_1^{u_1} x_2^{u_2} \cdots x_n^{u_n} \quad \text{with} \quad x_j^{u_j} = \begin{cases} x_j, & \text{if } u_j = 1, \\ 1, & \text{if } u_j = 0. \end{cases}$$

Expression (3), where each term $\boldsymbol{x}^{\boldsymbol{u}}$ is called a **monomial**, is known as the **Algebraic Normal Form** (ANF) of $f(\boldsymbol{x})$. Note that μ_f is also a Boolean function of n variables, called the **Möbius transformation** of f.

For $f \in \mathcal{B}_f$, we call **degree** of f, denoted by $\deg(f)$, the maximum of the degrees of the monomials of its ANF. So,

$$\deg(f) = \max\{w(\boldsymbol{u}) \mid \mu_f(\boldsymbol{u}) = 1\}. \tag{4}$$

Obviously, $\deg(1 \oplus f) = \deg(f)$ and $\deg(1) = 0$. As it is usual we say that $\deg(0) = -\infty$.

We say that $f \in \mathcal{B}_n$ is an **affine function** if $\deg(f) = 1$; in this case expression (3) becomes

$$f(\boldsymbol{x}) = a_0 \oplus a_1 x_1 \oplus a_2 x_2 \oplus \cdots \oplus a_n x_n$$

with $a_j \in \mathbb{F}_2$, for $j = 0, 1, 2, \ldots, n$ and $a_l \neq 0$ for some $l = 1, 2, \ldots, n$. In particular, if $a_0 = 0$, we say that f is a **linear function**. It can be checked that any affine function is balanced, although the converse is not true.

Furthermore, if $f \in \mathcal{B}_n$ and for all $\boldsymbol{a} \in \mathbb{F}_2^n$ we consider $g_{\boldsymbol{a}} \in \mathcal{B}_n$ defined by $g_{\boldsymbol{a}}(\boldsymbol{x}) = f(\boldsymbol{x} \oplus \boldsymbol{a})$, then $\deg(g_{\boldsymbol{a}}) = \deg(f)$, for all $\boldsymbol{a} \in \mathbb{F}_2^n$, and it is not difficult to see that

$$\mathrm{Supp}(g_{\boldsymbol{a}}) = \boldsymbol{a} \oplus \mathrm{Supp}(f), \quad \text{for all } \boldsymbol{a} \in \mathbb{F}_2^n.$$

The following theorem is a reformulation of a well-known result for people working in coding theory (see [14, Theorem 1, page 372]).

Theorem 1: *If $f \in \mathcal{B}_n$, then the coefficients $\mu_f(\boldsymbol{u})$ of the ANF of f can be computed as*

$$\mu_f(\boldsymbol{u}) = \bigoplus_{\boldsymbol{a} \in S(\boldsymbol{u})} f(\boldsymbol{a}), \quad \textit{for all } \boldsymbol{u} \in \mathbb{F}_2^n. \tag{5}$$

As a consequence of the previous result we have the following corollary.

Corollary 1: *Let $f \in \mathcal{B}_n$.*

1. *Assume that $\boldsymbol{u} \in \mathbb{F}_2^n$. The monomial $\boldsymbol{x}^{\boldsymbol{u}}$ is in the ANF of f if and only if*

$$|\mathrm{Supp}(f) \cap S(\boldsymbol{u})| \equiv 1 \pmod 2.$$

2. $\deg(f) = \max\{w(\boldsymbol{u}) \mid \boldsymbol{u} \in \mathbb{F}_2^n$ *and* $|\mathrm{Supp}(f) \cap S(\boldsymbol{u})| \equiv 1 \pmod 2\}$.

Let $\boldsymbol{u} = (1, 1, \ldots, 1) \in \mathbb{F}_2^n$. Since $S(\boldsymbol{u}) = \mathbb{F}_2^n$, from Corollary 1.1 we have that the degree of a Boolean function $f \in \mathcal{B}_n$ is n if and only if $w(f)$ is an odd number.

As another immediate consequence of Corollary 1 we have the following algorithm that computes the degree of the Boolean function whose support is a given set.

Algorithm 1: Assume that F is a subset of $\mathbb{F}_2^n$ and let $f \in \mathcal{B}_n$ such that $F = \mathrm{Supp}(f)$. This algorithm computes $\deg(f)$.

1. If $|F|$ is odd, then $\deg(f) = n$. Go to step 4.
2. For $\boldsymbol{u} \in \mathbb{F}_2^n$ compute $w(\boldsymbol{u})$ and $|F \cap S(\boldsymbol{u})|$.
3. $\deg(f) = \max\{w(\boldsymbol{u}) \mid \boldsymbol{u} \in \mathbb{F}_2^n$ and $|F \cap S(\boldsymbol{u})| \equiv 1 \pmod 2\}$
4. End.

It is evident that the number of Boolean functions of n variables whose support has an odd number of elements is 2^{2^n-1}. So, exactly half of the Boolean functions of n variables have degree n and, for such functions, determining their degree from their support is immediate (see step 1 of Algorithm 1).

In an earlier paper [15] we introduced the following results that we quote here for completeness.

Theorem 2 (Theorem 5 of [15]): *Let $f \in \mathcal{B}_n$ such that $|\mathrm{Supp}(f)|$ is an even number. If $\bigoplus_{\boldsymbol{a} \in \mathrm{Supp}(f)} \boldsymbol{a} = \boldsymbol{0}$, then $\deg(f) \leq n - 2$.*

Theorem 3 (Theorem 7 of [15]): *Assume that $f \in \mathcal{B}_n$. If $1 \leq k < n$, then*

$$f(\boldsymbol{y}, \boldsymbol{x}) = \bigoplus_{\boldsymbol{b} \in \mathbb{F}_2^k} \left(\bigoplus_{\boldsymbol{a} \in S(\boldsymbol{b})} f_{\boldsymbol{a}}(\boldsymbol{x}) \right) \boldsymbol{y}^{\boldsymbol{b}} \tag{6}$$

where $f_{\boldsymbol{a}} \in \mathcal{B}_k$, for $\boldsymbol{a} \in \mathbb{F}_2^k$, satisfies $f_{\boldsymbol{a}}(\boldsymbol{x}) = f(\boldsymbol{a}, \boldsymbol{x})$. Furthermore,

1. $\mathrm{Supp}(f_{\boldsymbol{a}}) = \{\boldsymbol{v} \in \mathbb{F}_2^{n-k} \mid (\boldsymbol{a}, \boldsymbol{v}) \in \mathrm{Supp}(f)\}$,

2. $\mathrm{Supp}\left(\bigoplus_{\boldsymbol{a}\in S(\boldsymbol{b})} f_{\boldsymbol{a}}\right) = \bigtriangleup_{\boldsymbol{a}\in S(\boldsymbol{b})} \mathrm{Supp}\,(f_{\boldsymbol{a}})$, *for all* $\boldsymbol{b} \in \mathbb{F}_2^k$,
3. $\deg(f) = \max_{\boldsymbol{b}\in\mathbb{F}_2^k}\left\{\deg\left(\bigoplus_{\boldsymbol{a}\in S(\boldsymbol{b})} f_{\boldsymbol{a}}\right) + w(\boldsymbol{b})\right\}$.

The above results allow us to establish the following algorithm to determine the degree of any $f \in \mathcal{B}_n$ from $\mathrm{Supp}\,(f)$.

Algorithm 2 (Algorithm of [15]): Assume that we know $\mathrm{Supp}\,(f)$ for a given $f \in \mathcal{B}_n$. This algorithm provides $\deg(f)$.

1. If $|\mathrm{Supp}\,(f)|$ is odd, then $\deg(f) = n$. Go to step 5.
2. Let $\boldsymbol{s} = \bigoplus_{\boldsymbol{a}\in\mathrm{Supp}(f)} \boldsymbol{a}$.
3. If $\boldsymbol{s} \neq \boldsymbol{0}$, then $\deg(f) = n-1$. Go to step 5.
4. Let $\mathrm{maxdeg}\,(f) = n-2$ be the maximum value that $\deg(f)$ can take and assume that $k = 1$.
   (a) For $\boldsymbol{b} \in \mathbb{F}_2^k$ do the following:
      i. Let $g_{\boldsymbol{b}} = \bigoplus_{\boldsymbol{a}\in S(\boldsymbol{b})} f_{\boldsymbol{a}}$ and obtain $\mathrm{Supp}\,(g_{\boldsymbol{b}})$ according to parts 1 and 2 of Theorem 3.
      ii. If $|\mathrm{Supp}\,(g_{\boldsymbol{b}})|$ is odd, then $\deg(g_{\boldsymbol{b}}) = n-k$. Go to step 4(a)vi.
      iii. Let $\boldsymbol{s}_{\boldsymbol{b}} = \bigoplus_{\boldsymbol{a}\in\mathrm{Supp}(g_{\boldsymbol{b}})} \boldsymbol{a}$.
      iv. If $\boldsymbol{s}_{\boldsymbol{b}} \neq \boldsymbol{0}$, then $\deg(g_{\boldsymbol{b}}) = n-k-1$. Go to step 4(a)vi.
      v. In other case, let $\mathrm{maxdeg}\,(g_{\boldsymbol{b}}) = n-k-2$ the maximum value that $\deg(g_{\boldsymbol{b}})$ can take.
      vi. End for.
   (b) If $\mathrm{maxdeg}\,(f) = \max_{\boldsymbol{b}\in\mathbb{F}_2^k}\{\deg(g_{\boldsymbol{b}}) + w(\boldsymbol{b})\}$, then $\deg(f) = \mathrm{maxdeg}\,(f)$. Go to step 5.
   (c) In other case, do $\mathrm{maxdeg}\,(f) = \max_{\boldsymbol{b}\in\mathbb{F}_2^k}\{\deg(g_{\boldsymbol{b}}) + w(\boldsymbol{b}), \mathrm{maxdeg}\,(g_{\boldsymbol{b}}) + w(\boldsymbol{b})\}$, increase k in one unit and go to step 4a.
5. End.

3 Some linear algebra properties

In this section we introduce some results that allow us to improve Algorithm 2 introduced in the previous section.

The following result establishes that any k-dimensional linear subspace of $\mathbb{F}_2^n$ (or the complementary set of any k-dimensional linear subspace of $\mathbb{F}_2^n$) is the support of a Boolean function of n variables with degree $n-k$.

Theorem 4: *Assume that* $1 \leq k \leq n$. *If* F *or* $\mathbb{F}_2^n \setminus F$ *is a* k*-dimensional linear subspace of* $\mathbb{F}_2^n$, *then there exists* $f \in \mathcal{B}_n$ *such that* $\deg(f) = n-k$ *and* $F = \mathrm{Supp}\,(f)$.

PROOF: Firstly, assume that F is a k-dimensional linear subspace of $\mathbb{F}_2^n$. Clearly, the map $f : \mathbb{F}_2^n \to \mathbb{F}_2$ given by

$$f(\boldsymbol{x}) = \begin{cases} 1, & \text{if } \boldsymbol{x} \in F, \\ 0, & \text{if } \boldsymbol{x} \notin F, \end{cases}$$

is a Boolean function of n variables whose support is F.

Assume that $n-k+1 \leq l \leq n$ and that the ANF of $f(\boldsymbol{x})$ contains the monomial $x_{i_1}x_{i_2}\cdots x_{i_l}$; then, by Corollary 1.1,

$$\left|F \cap \operatorname{Span}\left\{\mathbf{2^{n-i_1}}, \mathbf{2^{n-i_2}}, \ldots, \mathbf{2^{n-i_l}}\right\}\right| \equiv 1 \pmod 2.$$

Nevertheless, since

$$\begin{aligned}\dim &\left(F \cap \operatorname{Span}\left\{\mathbf{2^{n-i_1}}, \mathbf{2^{n-i_2}}, \ldots, \mathbf{2^{n-i_l}}\right\}\right)\\ &= \dim F + \dim \operatorname{Span}\left\{\mathbf{2^{n-i_1}}, \mathbf{2^{n-i_2}}, \ldots, \mathbf{2^{n-i_l}}\right\}\\ &\qquad - \dim\left(F + \operatorname{Span}\left\{\mathbf{2^{n-i_1}}, \mathbf{2^{n-i_2}}, \ldots, \mathbf{2^{n-i_l}}\right\}\right)\\ &\geq k + l - n \geq 1,\end{aligned}$$

necessarily

$$\begin{aligned}&\left|F \cap \operatorname{Span}\left\{\mathbf{2^{n-i_1}}, \mathbf{2^{n-i_2}}, \ldots, \mathbf{2^{n-i_l}}\right\}\right|\\ &= 2^{\dim\left(F \cap \operatorname{Span}\left\{\mathbf{2^{n-i_1}}, \mathbf{2^{n-i_2}}, \ldots, \mathbf{2^{n-i_l}}\right\}\right)} \equiv 0 \pmod 2.\end{aligned}$$

So, we have a contradiction. Therefore, the ANF of $f(\boldsymbol{x})$ does not contain any monomial of degree l and, consequently, $\deg(f) \leq n-k$.

Now assume that $\{\boldsymbol{b_1}, \boldsymbol{b_2}, \ldots, \boldsymbol{b_k}\}$ is a basis of F and complete such basis, with the vectors of the standard basis, to obtain a new basis

$$\left\{\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_k, \mathbf{2^{n-i_1}}, \mathbf{2^{n-i_2}}, \ldots, \mathbf{2^{n-i_{n-k}}}\right\}$$

of $\mathbb{F}_2^n$. Clearly $F \cap \operatorname{Span}\left\{\mathbf{2^{n-i_1}}, \mathbf{2^{n-i_2}}, \ldots, \mathbf{2^{n-i_{n-k}}}\right\} = \{\mathbf{0}\}$ and, by Corollary 1.1, the ANF of $f(\boldsymbol{x})$ contains the monomial $x_{i_1}x_{i_2}\cdots x_{i_{n-k}}$; so $\deg(f) \geq n-k$.

Now, from this inequality and the previous one, we have that $\deg(f) = n-k$.

On the other hand, if $G = \mathbb{F}_2^n \setminus F$ is a k-dimensional linear subspace of $\mathbb{F}_2^n$, then, from the above part, there exists $g \in \mathcal{B}_n$ such that $\deg(g) = n-k$ and $G = \operatorname{Supp}(g)$. Let $f \in \mathcal{B}_n$ such that $f = 1 \oplus g$. Clearly, $\deg(f) = \deg(g)$ and $\operatorname{Supp}(f) = \mathbb{F}_2^n \setminus \operatorname{Supp}(g)$; that is, $\deg(f) = n-k$ and $F = \operatorname{Supp}(f)$.

Note that as a consequence of this theorem, any $[n,k]$ binary code is the support of a Boolean function of n variables of degree $n-k$ (see, for example, [6, 16, 17] when the authors construct Boolean functions with some properties using linear codes).

Next result establishes that Theorem 4 also holds if we change "linear subspace" by "affine subspace".

Corollary 2: *Assume that $1 \leq k \leq n$. If F or $\mathbb{F}_2^n \setminus F$ is a k-dimensional affine subspace of $\mathbb{F}_2^n$, then there exists $f \in \mathcal{B}_n$ such that $\deg(f) = n-k$ and $F = \operatorname{Supp}(f)$.*

PROOF: Firstly, assume that F is a k-dimensional affine subspace of $\mathbb{F}_2^n$. Then $F = \boldsymbol{a} \oplus G$ with G a k-dimensional linear subspace of $\mathbb{F}_2^n$ and $\boldsymbol{a} \in \mathbb{F}_2^n \setminus G$. Therefore, by Theorem 4, there exists $g \in \mathcal{B}_n$ such that $\deg(g) = n-k$ and $G = \operatorname{Supp}(g)$.

Now, let $f \in \mathcal{B}_n$ such that $f(\boldsymbol{x}) = g(\boldsymbol{x} \oplus \boldsymbol{a})$ for all $\boldsymbol{x} \in \mathbb{F}_2^n$. It is evident that $\deg(f) = n - k$ and $\mathrm{Supp}(f) = \boldsymbol{a} \oplus \mathrm{Supp}(g) = \boldsymbol{a} \oplus G = F$.

On the other hand, it $G = \mathbb{F}_2^n \setminus F$ is a k-dimensional affine subspace of $\mathbb{F}_2^n$, then, from the above part, there exists $g \in \mathcal{B}_n$ such that $\deg(g) = n - k$ and $G = \mathrm{Supp}(g)$. Let $f \in \mathcal{B}_n$ such that $f = 1 \oplus g$. Clearly, $\deg(f) = \deg(g)$ and $\mathrm{Supp}(f) = \mathbb{F}_2^n \setminus \mathrm{Supp}(g)$; that is, $\deg(f) = n - k$ and $F = \mathrm{Supp}(f)$.

As an immediate consequence of Theorem 4 and Corollary 2 we have the following remark.

Remark 1: Assume that $f \in \mathcal{B}_n$.

1. If $|\mathrm{Supp}(f)| = 2$ and $\mathbf{0} \in \mathrm{Supp}(f)$ (respectively, $\mathbf{0} \notin \mathrm{Supp}(f)$), then $\mathrm{Supp}(f)$ is a 1-dimensional linear subspace (respectively, affine subspace) of $\mathbb{F}_2^n$ and consequently, $\deg(f) = n - 1$.
2. If $|\mathrm{Supp}(f)| = 2^n - 2$ and $\mathbf{0} \notin \mathrm{Supp}(f)$ (respectively, $\mathbf{0} \in \mathrm{Supp}(f)$), then $\mathbb{F}_2^n \setminus \mathrm{Supp}(f)$ is a 1-dimensional linear subspace (respectively, affine subspace) of $\mathbb{F}_2^n$ and consequently $\deg(f) = n - 1$.

The converse of Theorem 4 is not true in general. Nevertheless, if $k = n$, then $F = \mathbb{F}_2^n$ is the support of the constant function $f(\boldsymbol{x}) = 1$ whose degree is 0. Furthermore, if $k = n - 1$, then the converse of Theorem 4 also holds as we can see in the following result.

Theorem 5: *Assume that $F \subseteq \mathbb{F}_2^n$. Then F or $\mathbb{F}_2^n \setminus F$ is an $(n-1)$-dimensional linear subspace of $\mathbb{F}_2^n$ if and only if there exists $f \in \mathcal{B}_n$ such that $\deg(f) = 1$ and $F = \mathrm{Supp}(f)$.*

PROOF: If F or $\mathbb{F}_2^n \setminus F$ is an $(n-1)$-dimensional linear subspace of $\mathbb{F}_2^n$, by Theorem 4, there exists $f \in \mathcal{B}_n$ such that $\deg(f) = 1$ and $F = \mathrm{Supp}(f)$.

Conversely, let $f \in \mathcal{B}_n$ such that $\deg(f) = 1$ and $F = \mathrm{Supp}(f)$. On the one hand

$$f(\boldsymbol{x}) = a_0 \oplus a_1 x_1 \oplus a_2 x_2 \oplus \cdots \oplus a_n x_n$$

for some $a_0, a_1, a_2, \ldots, a_n \in \mathbb{F}_2$, with $a_l \neq 0$ for some $l = 1, 2, \ldots, n$, and clearly

$$S = \{(u_1, u_2, \ldots, u_n) \in \mathbb{F}_2^n \mid a_1 u_1 \oplus a_2 u_2 \oplus \cdots \oplus a_n u_n = 0\}$$

is an $(n-1)$-dimensional linear subspace of $\mathbb{F}_2^n$. On the other hand, it is easy to see that $S = F$, if $a_0 = 1$, and $S = \mathbb{F}_2^n \setminus F$, if $a_0 = 0$.

It is also possible to get a similar result to Theorem 5 considering affine subspaces.

Corollary 3: *Assume that $F \subseteq \mathbb{F}_2^n$. Then F or $\mathbb{F}_2^n \setminus F$ is an $(n-1)$-dimensional affine subspace of $\mathbb{F}_2^n$ if and only if there exists $f \in \mathcal{B}_n$ such that $\deg(f) = 1$ and $F = \mathrm{Supp}(f)$.*

This result may not be true if $\dim F = k \neq n - 1$, because $2^n - 2^k = 2^k(2^{n-k} - 1)$ is not a power of 2.

The following example shows how we can use the above results to improve the process described in the above section.

Example 1: Let $f \in \mathcal{B}_5$ such that

$$\mathrm{Supp}\,(f) = \{\mathbf{2}, \mathbf{3}, \mathbf{8}, \mathbf{9}, \mathbf{16}, \mathbf{20}, \mathbf{22}, \mathbf{23}, \mathbf{25}, \mathbf{26}, \mathbf{27}, \mathbf{28}, \mathbf{30}, \mathbf{31}\}.$$

Note that $|\mathrm{Supp}\,(f)| = 14$ and $\left|\mathbb{F}_2^5 \setminus \mathrm{Supp}\,(f)\right| = 18$, so neither $\mathrm{Supp}\,(f)$ nor $\mathbb{F}_2^5 \setminus \mathrm{Supp}\,(f)$ can be a linear subspace nor an affine subspace of $\mathbb{F}_2^5$.

Since $|\mathrm{Supp}\,(f)|$ is even and

$$\mathbf{2} \oplus \mathbf{3} \oplus \mathbf{8} \oplus \mathbf{9} \oplus \mathbf{16} \oplus \mathbf{20} \oplus \mathbf{22} \oplus \mathbf{23} \oplus \mathbf{25} \oplus \mathbf{26} \oplus \mathbf{27} \oplus \mathbf{28} \oplus \mathbf{30} \oplus \mathbf{31} = \mathbf{0},$$

from Theorem 2, $\deg(f) \leq 5 - 2 = 3$. Now, according to part 3 of Theorem 3, we have that

$$\deg(f) = \max\{\deg(f_0), \deg(f_0 \oplus f_1) + 1\}$$

with $f_0, f_1 \in \mathcal{B}_4$ such that

$$f_0(x_2, x_3, x_4, x_5) = f(0, x_2, x_3, x_4, x_5),$$
$$f_1(x_2, x_3, x_4, x_5) = f(1, x_2, x_3, x_4, x_5),$$

and, from part 1 of Theorem 3,

$$\begin{aligned} \mathrm{Supp}\,(f_0) &= \{\mathbf{2}, \mathbf{3}, \mathbf{8}, \mathbf{9}\}, \\ \mathrm{Supp}\,(f_1) &= \{\mathbf{0}, \mathbf{4}, \mathbf{6}, \mathbf{7}, \mathbf{9}, \mathbf{10}, \mathbf{11}, \mathbf{12}, \mathbf{14}, \mathbf{15}\}. \end{aligned} \tag{7}$$

Therefore,

$$\begin{aligned} \mathrm{Supp}\,(f_0 \oplus f_1) &= \mathrm{Supp}\,(f_0)\,\Delta\,\mathrm{Supp}\,(f_1) \\ &= \{\mathbf{0}, \mathbf{2}, \mathbf{3}, \mathbf{4}, \mathbf{6}, \mathbf{7}, \mathbf{8}, \mathbf{10}, \mathbf{11}, \mathbf{12}, \mathbf{14}, \mathbf{15}\}. \end{aligned} \tag{8}$$

In addition, since $\mathbf{2} \oplus \mathbf{3} \oplus \mathbf{8} \oplus \mathbf{9} = \mathbf{0}$ and

$$\mathbf{0} \oplus \mathbf{2} \oplus \mathbf{3} \oplus \mathbf{4} \oplus \mathbf{6} \oplus \mathbf{7} \oplus \mathbf{8} \oplus \mathbf{10} \oplus \mathbf{11} \oplus \mathbf{12} \oplus \mathbf{14} \oplus \mathbf{15} = \mathbf{0},$$

from Theorem 2,

$$\deg(f_0) \leq 4 - 2 = 2 \quad \text{and} \quad \deg(f_0 \oplus f_1) \leq 4 - 2 = 2$$

and, therefore $\deg(f) \leq \max\{2, 2 + 1\} = 3$.

It is easy to check (see expressions (7) and (8) that none of the sets $\mathrm{Supp}\,(f_0)$, $\mathbb{F}_2^4 \setminus \mathrm{Supp}\,(f_0)$, $\mathrm{Supp}\,(f_0 \oplus f_1)$ and $\mathbb{F}_2^4 \setminus \mathrm{Supp}\,(f_0 \oplus f_1)$ can be linear subspaces of $\mathbb{F}_2^4$. Nevertheless

$$\mathrm{Supp}\,(f_0) = \mathbf{2} \oplus \{\mathbf{0}, \mathbf{1}, \mathbf{10}, \mathbf{11}\} = \mathbf{2} \oplus \mathrm{Span}\,\{\mathbf{1}, \mathbf{10}\},$$
$$\mathbb{F}_2^4 \setminus \mathrm{Supp}\,(f_0 \oplus f_1) = \{\mathbf{1}, \mathbf{5}, \mathbf{9}, \mathbf{13}\} = \mathbf{1} \oplus \{\mathbf{0}, \mathbf{4}, \mathbf{8}, \mathbf{12}\} = \mathbf{1} \oplus \mathrm{Span}\,\{\mathbf{4}, \mathbf{8}\}$$

are affine subspaces of dimension 2. So, by Corollary 2,

$$\deg(f_0) = 4 - 2 = 2 \quad \text{and} \quad \deg(f_0 \oplus f_1) = 4 - 2 = 2$$

and, therefore $\deg(f) = \max\{2, 2 + 1\} = 3$.

Finally, we modify Algorithm 2 to check, when the cardinal of the support of the function we are considering is a power of 2, if that support is a vector or affine subspace.

Algorithm 3: Assume that we know $\mathrm{Supp}\,(f)$ for a given $f \in \mathcal{B}_n$. This algorithm provides $\deg(f)$.

1. If $|\mathrm{Supp}\,(f)|$ is odd, then $\deg(f) = n$. Go to step 7.
2. If $|\mathrm{Supp}\,(f)| = 2$ or $|\mathrm{Supp}\,(f)| = 2^n - 2$, then $\deg(f) = n-1$. Go to step 7.
3. If $|\mathrm{Supp}\,(f)|$ is a power of 2, for example 2^r (with $r \geq 2$), check if $\mathrm{Supp}\,(f)$ is a linear or affine subspace. If this is the case, then $\deg(f) = n - r$. Go to step 7.
4. Let $\boldsymbol{s} = \bigoplus_{\boldsymbol{a} \in \mathrm{Supp}(f)} \boldsymbol{a}$.
5. If $\boldsymbol{s} \neq \mathbf{0}$, then $\deg(f) = n - 1$. Go to step 7.
6. Let $\mathrm{maxdeg}\,(f) = n - 2$ be the maximum value that $\deg(f)$ can take and assume that $k = 1$.
   (a) For $\boldsymbol{b} \in \mathbb{F}_2^k$ do the following:
      i. Let $g_{\boldsymbol{b}} = \bigoplus_{\boldsymbol{a} \in S(\boldsymbol{b})} f_{\boldsymbol{a}}$ and obtain $\mathrm{Supp}\,(g_{\boldsymbol{b}})$ according to parts 1 and 2 of Theorem 3.
      ii. If $|\mathrm{Supp}\,(g_{\boldsymbol{b}})|$ is odd, then $\deg(g_{\boldsymbol{b}}) = n-k$. Go to step 6(a)viii.
      iii. If $|\mathrm{Supp}\,(g_{\boldsymbol{b}})| = 2$ or $|\mathrm{Supp}\,(g_{\boldsymbol{b}})| = 2^{n-k} - 2$, then $\deg(f) = n - k - 1$. Go to step 6(a)viii.
      iv. If $|\mathrm{Supp}\,(g_{\boldsymbol{b}})|$ is a power of 2, for example 2^r (with $r \geq 2$), check if $\mathrm{Supp}\,(g_{\boldsymbol{b}})$ is a linear or affine subspace. If this is the case, then $\deg(g_{\boldsymbol{b}}) = n - k - r$. Go to step 6(a)viii.
      v. Let $\boldsymbol{s}_{\boldsymbol{b}} = \bigoplus_{\boldsymbol{a} \in \mathrm{Supp}(g_{\boldsymbol{b}})} \boldsymbol{a}$.
      vi. If $\boldsymbol{s}_{\boldsymbol{b}} \neq \mathbf{0}$, then $\deg(g_{\boldsymbol{b}}) = n - k - 1$. Go to step 6(a)viii.
      vii. In other case, let $\mathrm{maxdeg}\,(g_{\boldsymbol{b}}) = n-k-2$ the maximum value that $\deg(g_{\boldsymbol{b}})$ can take.
      viii. End for.
   (b) If $\mathrm{maxdeg}\,(f) = \max_{\boldsymbol{b} \in \mathbb{F}_2^k}\{\deg(g_{\boldsymbol{b}}) + w(\boldsymbol{b})\}$, then $\deg(f) = \mathrm{maxdeg}\,(f)$. Go to step 7.
   (c) In other case, do

$$\mathrm{maxdeg}\,(f) = \max_{\boldsymbol{b} \in \mathbb{F}_2^k}\{\deg(g_{\boldsymbol{b}}) + w(\boldsymbol{b}), \mathrm{maxdeg}\,(g_{\boldsymbol{b}}) + w(\boldsymbol{b})\},$$

   increase k in one unit and go to step 6a.
7. End.

4 Numerical results

For a given n the number of Boolean functions of n variables is 2^{2^n}. For different values of n (with $8 \leq n \leq 14$) we obtained, on a standard personal computer, randomly 1000 subsets of $\mathbb{F}_2^n$; that is, 1000 supports of Boolean functions of n variables. For these supports we compute the degree of the corresponding Boolean function by computing the ANF from expression (5) and using Algorithms 1, 2

Table 1: Times obtained in the computation of the degree of a Boolean function of n variables from its support

n	ANF	Algorithm 1	Algorithm 2	Algorithm 3
8	0.0075	0.0748	0.0033	0.0038
9	0.0229	0.1489	0.0059	0.0069
10	0.0696	0.3564	0.0132	0.0159
11	0.2151	0.8105	0.0278	0.0348
12	0.7766	1.7892	0.0527	0.0767
13	− − −	5.0492	0.1172	0.2177
14	− − −	9.4359	0.2360	0.5370

Table 2: Times obtained in the computation of the degree of a Boolean function of n variables whose support is a linear subspace

n	Algorithm 2	Algorithm 3
8	0.0036	0.0004
9	0.0069	0.0007
10	0.0151	0.0015
11	0.0324	0.0036
12	0.0714	0.0087
13	0.1761	0.0196
14	0.4998	0.0542

and 3. Columns 2, 3, 4 and 5 of Table 1 show the average times (in seconds) we obtained in each case. For $n = 13, 14$ we have not enough memory in our computer to obtain the ANF. In general, Algorithm 2 is faster than Algorithm 3. Nevertheless, if the supports considered are linear subspaces, then Algorithm 3 is much faster than Algorithm 2. Table 2 shows the average times (in seconds) we obtained to compute the degree of the Boolean functions corresponding to 1000 linear subspaces of $\mathbb{F}_2^n$ for different values of n.

5 Conclusions

In this paper we present some properties of the support of a Boolean function that allow us to obtain the different terms of the algebraic normal form. In particular, when we know the support of a Boolean function, we can obtain its degree without computing its algebraic normal form. For example, if $f(\boldsymbol{x})$ is a Boolean function of n variables, we prove that the degree of f is n if and only if the support of f

have an even number of elements; note that it is very easy to check this property; in fact, half of the Boolean functions have degree n. Furthermore, if the support of a Boolean function of n variables is a k-dimensional linear or affine subspace of $\mathbb{F}_2^n$, then we obtain that its degree is $n - k$.

As a consequence of these properties, we also introduce different algorithms to compute the degree of a Boolean function from its support, without computing its ANF, and present some numerical results using these algorithms.

Acknowledgements

The work of the first author was partially supported by Spanish grant MTM2011-24858 of the Ministerio de Economía y Competitividad of the Gobierno de España. The work of the third author was partially supported by the research project UMH-Bancaja with reference IPZS01.

References

[1] Braeken, A., Nikov, V., Nikova, S. & Preneel, B., On Boolean functions with generalized cryptographic properties. *Progress in Cryptology – INDOCRYPT 2004*, eds. A. Canteaut & K. Viswanathan, Springer-Verlag: Berlin, volume 3348 of *Lecture Notes in Computer Science*, pp. 120–135, 2004.

[2] Carlet, C. & Tarannikov, Y., Covering sequences of Boolean functions and their cryptographic significance. *Designs, Codes and Cryptography*, **25**, pp. 263–279, 2002.

[3] Kurosawa, K. & Matsumoto, R., Almost security of cryptographic Boolean functions. *IEEE Transactions on Information Theory*, **50(11)**, pp. 2752–2761, 2004.

[4] Borissov, Y., Braeken, A., Nikova, S. & Preneel, B., On the covering radii of binary Reed-Muller codes in the set of resilient Boolean functions. *IEEE Transactions on Information Theory*, **51(3)**, pp. 1182–1189, 2005.

[5] Kurosawa, K., Iwata, T. & Yoshiwara, T., New covering radius of Reed-Muller codes for t-resilient functions. *IEEE Transactions on Information Theory*, **50(3)**, pp. 468–475, 2004.

[6] Matsumoto, R., Kurosawa, K., Itoh, T., Konno, T. & Uyematsu, T., Primal-dual distance bounds of linear codes with application to cryptography. *IEEE Transactions on Information Theory*, **52(9)**, pp. 4251–4256, 2006.

[7] Meier, W. & Staffelbach, O., Nonlinearity criteria for cryptographic functions. *Advances in Cryptology – EUROCRYPT'89*, eds. J. Quisquater & J. Vandewalle, Springer-Verlag: Berlin, volume 434 of *Lecture Notes in Computer Science*, pp. 549–562, 1990.

[8] Rueppel, R.A., *Analysis and Design of Stream Ciphers*. Springer Verlag: New York, NY, 1986.

[9] Rueppel, R.A. & Staffelbach, O.J., Products of linear recurring sequences with maximum complexity. *IEEE Transactions on Information Theory*, **33(1)**, pp. 124–131, 1987.
[10] Olejár, D. & Stanek, M., On cryptographic properties of random Boolean functions. *Journal of Universal Computer Science*, **4(8)**, pp. 705–717, 1998.
[11] Pasalic, E. & Johansson, T., Further results on the relation between nonlinearity and resiliency for Boolean functions. *Crytography and Coding*, ed. M. Walker, Springer-Verlag: Berlin, volume 1746 of *Lecture Notes in Computer Science*, pp. 35–44, 1999.
[12] Preneel, B., Van Leekwijck, W., Van Linden, L., Govaerts, R. & Vandewalle, J., Propagation characteristics of Boolean functions. *Advances in Cryptology – EUROCRYPT'90*, ed. I.B. Damgard, Springer-Verlag: Berlin, volume 473 of *Lecture Notes in Computer Science*, pp. 161–173, 1991.
[13] Qu, C., Seberry, J. & Pieprzyk, J., On the symmetric property of homogeneous Boolean functions. *Proceedings of the Australasian Conference on Information Security and Privacy – ACISP'99*, eds. J. Pieprzyk, R. Safavi-Naini & J. Seberry, Springer-Verlag: Berlin, volume 1587 of *Lecture Notes in Computer Science*, pp. 26–35, 1999.
[14] MacWilliams, F.J. & Sloane, N.J.A., *The Theory of Error-Correcting Codes*. North-Holland: Amsterdam, 6th edition, 1988.
[15] Climent, J.J., García, F.J. & Requena, V., Computing the degree of a Boolean function from its support. *Proceedings of the 2010 International Symposium on Information Theory and its Applications (ISITA2010)*, eds. C.C. Chao & R. Kohno, IEEE Press, pp. 123–128, 2010.
[16] Dawson, E. & Wu, C.K., Construction of correlation immune Boolean functions. *Information and Communications Security – ICIS '97*, eds. Y. Han, T. Okamoto & S. Qing, Springer-Verlag: Berlin, volume 1334 of *Lecture Notes in Computer Science*, pp. 170–180, 1997.
[17] Kurosawa, K. & Satoh, T., Design of $SAC/PC(l)$ of order k Boolean functions and three other cryptographic criteria. *Advances in Cryptology – EUROCRYPT '97*, ed. W. Fumy, Springer-Verlag: Berlin, volume 1233 of *Lecture Notes in Computer Science*, pp. 434–449, 1997.