Constructing and Counting Even-Variable
Symmetric Boolean Functions with
Algebraic Immunity not Less Than \( d \)

Yuan Li, Hui Wang and Haibin Kan

Abstract

In this paper, we explicitly construct a large class of symmetric Boolean
functions on \( 2k \) variables with algebraic immunity not less than \( d \), where
integer \( k \) is given arbitrarily and \( d \) is a given suffix of \( k \) in binary repre-
sentation. If let \( d = k \), our constructed functions achieve the maximum
algebraic immunity. Remarkably, \( 2^{\lceil \log_2 k \rceil + 2} \) symmetric Boolean functions
on \( 2k \) variables with maximum algebraic immunity are constructed, which
is much more than the previous constructions. Based on our construction,
a lower bound of symmetric Boolean functions with algebraic immunity
not less than \( d \) is derived, which is \( 2^{\lceil \log_2 d \rceil + 2(k - d + 1)} \). As far as we know,
this is the first lower bound of this kind.

1 Introduction

Algebraic attack has received a lot of attention in studying security of the cryp-
tosystems. If a Boolean function used in stream ciphers has low degree anni-
hilators, it will be easily attacked. This adds a new cryptographic property for
designing Boolean functions to be used as building blocks in cryptosystems
which is known as algebraic immunity (AI). Since then algebraic immunity, as
a property of Boolean functions, is widely studied.

Constructing Boolean functions with high AI is interesting and important.
A lot of general methods to construct Boolean functions with maximum alge-
braic immunity are proposed [4], [5], [10]. Results in [5], [11] show that the
number of general Boolean functions achieving maximum algebraic immunity
is large.

Among all Boolean functions, symmetric Boolean function is an interesting
class and their properties are well studied [9], [12], [13]. In [12], [13], the au-
thors proved that there are only two symmetric Boolean functions on odd num-
ber of variables with maximum AI. In Braeken’s thesis [15], some symmetric
Boolean functions on even variables with maximum AI are constructed. In [8],
more such functions are constructed, which generalizes results in [15]. In [14],
by using weight support technique, all \((2^m + 1)\)-variable symmetric Boolean
functions with submatrimal algebraic immunity \( 2^{m-1} \) are constructed.

In this paper, we focus on constructing symmetric Boolean functions with
high algebraic immunity on \( 2k \) variables, where \( k \) is given arbitrarily. For a
given \( d \), where \( d \) is a suffix of \( k \) in binary representation, we construct a large
class of Boolean functions with AI not less than \( d \). Particularly, if let \( d = k \), our constructed Boolean functions achieve maximum AI. Comparing with all the previous constructions of this kind, the number of our constructed Boolean functions is much larger. Furthermore, a lower bound of symmetric Boolean functions with algebraic immunity not less than \( d \) is derived.

2 Preliminaries

Let \( \mathbb{F}_2 \) be the finite field with only two elements. To prevent confusion with the usual sum, the sum over \( \mathbb{F}_2 \) is denoted by \( \oplus \). The Hamming weight of a vector \( \alpha = (\alpha_1, \ldots, \alpha_n) \) is defined by \( \text{wt}(\alpha) = \sum_{i=1}^{n} \alpha_i \).

A Boolean function on \( n \) variables may be viewed as a mapping from \( \mathbb{F}_2^n \) into \( \mathbb{F}_2 \). We denote by \( B_n \) the set of all \( n \)-variable Boolean functions. The Hamming weight (1)

\[
\text{wt}(f) = \text{size of the support } \text{supp}(f) = \{ x \in \mathbb{F}_2^n \mid f(x) = 1 \}.
\]

The support of \( f \) is also called the on set of \( f \), which is denoted by \( 1_f \). On the contrary, the off set of \( f \) is the set \( \{ x \in \mathbb{F}_2^n \mid f(x) = 0 \} \), which is denoted by \( 0_f \).

Any \( f \in B_n \) can be uniquely represented as

\[
f(x_1, x_2, \ldots, x_n) = \bigoplus_{\alpha \in \mathbb{F}_2^n} c_{\alpha} \prod_{i=1}^{n} x_{i}^{\alpha_i} = \bigoplus_{\alpha \in \mathbb{F}_2^n} c_{\alpha} x_{\alpha},
\]

This kind of expression of \( f \) is called the Algebraic Normal Form (ANF). The algebraic degree of \( f \) is the number of variables in the highest order term with nonzero coefficient, which is denoted by \( \text{deg}(f) \).

A Boolean function is said to be symmetric if its output is invariant under any permutation of its input bits. For a symmetric Boolean function \( f \) on \( n \) variables, we have

\[
f(x_1, x_2, \ldots, x_n) = f(x_{\sigma(1)}, x_{\sigma(2)}, \ldots, x_{\sigma(n)})
\]

for all permutations \( \sigma \) on \( \{1, 2, \ldots, n\} \).

This equivalently means that the output of \( f \) only depends on the weight of its input vector. As a consequence, \( f \) is related to a function \( \nu_f : \{0, 1, \ldots, n\} \mapsto \mathbb{F}_2 \) such that \( f(\alpha) = \nu_f(\text{wt}(\alpha)) \) for all \( \alpha \in \mathbb{F}_2^n \). The vector \( \nu_f = (\nu_f(0), \nu_f(1), \ldots, \nu_f(n)) \) is called the simplified value vector (SVV) of \( f \). The set of all \( n \)-variable Boolean functions are denoted by \( SB_n \).

Proposition 2.1. \cite{9} A Boolean function \( f \) on \( n \) variables is symmetric if and only if its ANF can be written as follows:

\[
f(x_1, x_2, \ldots, x_n) = \bigoplus_{i=0}^{n} \lambda_f(i) \bigoplus_{\alpha \in \mathbb{F}_2^n, \text{wt}(\alpha) = i} x_{\alpha} = \bigoplus_{i=0}^{n} \lambda_f(i) \sigma^n_i,
\]

where \( \sigma_i^n \) is the elementary symmetric polynomial of degree \( i \) on \( n \) variables.

Then, the coefficients of the ANF of \( f \) can be represented by a \((n + 1)\)-bit vector, \( \lambda_f = (\lambda_f(0), \lambda_f(1), \ldots, \lambda_f(n)) \), called the simplified algebraic normal form (SANF) vector of \( f \).
Proposition 2.2. Let \( f \) be a symmetric Boolean function on \( n \) variables. Then, its simplified value vector \( v_f \) and its simplified ANF vector \( \lambda_f \) are related by
\[
v_f(i) = \bigoplus_{k \geq i} \lambda_f(k) \quad \text{and} \quad \lambda_f(i) = \bigoplus_{k \geq i} v_f(k),
\]
for all \( i = 0, 1, \ldots, n \).

**Definition 2.3.** For a given \( f \in B_n \), a nonzero function \( g \in B_n \) is called an annihilator of \( f \) if \( fg = 0 \) and the algebraic immunity (AI) of \( f \) is the minimum degree of all annihilators of \( f \) or \( f \oplus 1 \), which is denoted by \( AI(f) \).

Note that \( AI(f) \leq \deg(f) \), since \( f(f \oplus 1) = 0 \). Therefore, a function with high AI will not have a low algebraic degree. It was known from [6] that for any \( f \in B_n \), \( AI(f) \leq \lceil \frac{n}{2} \rceil \).

Two Boolean functions \( f \) and \( g \) are said to be affine equivalent if there exist \( A \in GL_n(\mathbb{F}_2) \) and \( b \in \mathbb{F}_2^n \) such that \( g(x) = f(xA + b) \). Clearly, algebraic degree, algebraic immunity are affine invariant.

The binary representation of an integer \( a \) is denoted by \((a_m a_{m-1} \ldots a_0)_2\), such that
\[
a = \sum_{i=0}^{m} a_i 2^i.
\]
If integer \( b \) is ended by \( a_1 a_0 \) in binary, we often denote by \( b = (a_1 a_0)_2 \), where \( * \) represents some 01 string. For convenience of the description in the sequel, we introduce the following notation.

**Definition 2.4.** Let \( a, b \) be two nonnegative integers with their binary representations \((a_m a_{m-1} \ldots a_0)_2\) and \((b_i b_{i-1} \ldots b_0)_2\), \( m \leq n \). If \( a_i = b_i \) for all \( i = 0, 1, \ldots, m \), we say \( a \) is a suffix of \( b \) in binary and denote by \( a \preceq b \). Furthermore, if \( a < b \), we say \( a \) is a proper suffix of \( b \), which is denoted by \( a \prec b \).

### 3 Main Results

**Lemma 3.1.** Let \( f, g \in B_n \), integer \( 0 \leq d \leq n \). If \( f(\alpha) = \bigoplus_{0 \leq \beta \leq \alpha \leq d} g(\beta) \) for all \( \alpha \in \mathbb{F}_2^n \) with \( wt(\alpha) \leq d \), then \( g(\beta) = \bigoplus_{\alpha \preceq \beta} f(\alpha) \) for all \( \beta \in \mathbb{F}_2^n \) with \( wt(\beta) \leq d \).

**Proof.** By direct computation, for any \( \beta \in \mathbb{F}_2^n \) with \( wt(\beta) \leq d \), we have
\[
\bigoplus_{\alpha \preceq \beta} f(\alpha) = \bigoplus_{\alpha \preceq \beta} \bigoplus_{\gamma \preceq \alpha} g(\gamma) = \bigoplus_{\gamma \preceq \alpha \preceq \beta} \bigoplus_{\gamma \preceq \alpha} g(\gamma) = \bigoplus_{\gamma \preceq \beta} g(\gamma) = g(\beta),
\]
which completes our proof.

**Lemma 3.2.** Let \( f, g \in B_n \), integer \( 0 \leq d \leq n \). If \( f(\alpha) = 1 \) for all \( \alpha \in \mathbb{F}_2^n \) satisfying \( 0 \leq wt(\alpha) \leq d \) and \( g(\beta) = 1 \) for all \( \beta \in \mathbb{F}_2^n \) satisfying \( n - d \leq wt(\beta) \leq n \), then both \( f \) and \( g \) do not have annihilators with degree less than or equal to \( d \).
Proof. Let $g' = g(x_1 \oplus 1, x_2 \oplus 1, \ldots, x_n \oplus 1)$, which takes 1 on all points with weight not exceeding $d$. Since $g'$ is affine equivalent to $g$, $AI(g') = AI(g)$. Therefore, it suffices to prove $f$ has no annihilator with degree not greater than $d$.

Assuming there is a function $h \in B_n$ such that $fh = 0$ and $\deg(h) < d$, we will show that $h = 0$. Write $h$ in ANF

$$h = \bigoplus_{\alpha \in \mathbb{F}_2^n} c_\alpha x^\alpha.$$ 

Since for any $\alpha \in \mathbb{F}_2^n$ with $\wt(\alpha) \leq d$, we have $h(\alpha) = 0$, i.e., $\bigoplus_{\beta \preceq \alpha} c_\beta = 0$. By Lemma 3.1 for any $\beta \in \mathbb{F}_2^n$ with $\wt(\beta) \leq d$, $c_\beta = \bigoplus_{\alpha \preceq \beta} h(\alpha) = 0$. Combining with $\deg(h) \leq d$, we conclude $h = 0$.

The following theorem is our main result, which gives a sufficient condition for a function $f \in SB_{2k}$ to have algebraic immunity not less than $d$, where $d$ is a suffix of $k$ in binary.

**Theorem 3.3.** Let $f \in SB_n$, $n = 2k$, $d \leq k$ and $d \geq 2$. If for any integer $i, j$ with $0 \leq i \leq d - 1$, $n - d + 1 \leq j \leq n$ and

$$k - i \equiv j - k \equiv 2^t \mod 2^{t+1}$$

for some nonnegative integer $t$, $v_f(i) = v_f(j) \oplus 1$ holds, then $AI(f) \geq d$.

**Proof.** To prove $AI(f) \geq d$, we need to show $f$ or $f \oplus 1$ has no annihilator with degree less than $d$. Without loss of generality, we only need to prove $f$ has no annihilator with degree less than $d$, because it also satisfies the conditions in this theorem by replacing $f$ by $f \oplus 1$.

Assume there is a function $g \in B_n$, such that $fg = 0$ and $\deg(g) \leq d - 1$, our aim is to show $g = 0$. Write $g$ in ANF

$$g = \bigoplus_{\alpha \in \mathbb{F}_2^n} c_\alpha x^\alpha.$$ 

Since $\deg(g) \leq d - 1$, we have $c_\alpha = 0$ for all $\wt(\alpha) \geq d$. If $f(\alpha) = 1$, then $g(\alpha) = 0$, which is

$$\bigoplus_{\beta \preceq \alpha, \wt(\beta) \leq d-1} c_\beta = 0. \quad (7)$$

Denote equation (7) on point $\alpha$ by $s_\alpha = 0$. By Lemma 3.1 we know $c_\beta = \bigoplus_{\alpha \preceq \beta} s_\alpha$, for $\wt(\beta) \leq d - 1$. We need to prove that all the equations $s_\alpha = 0$, $\alpha \in 1_f$, on $\sum_{i=0}^{d-1} \binom{n}{d}$ variables $c_\beta$, $\wt(\beta) \leq d - 1$, has only zero solution.

To assist our proof, we introduce a decomposition of integers according to $k$. Let $k = (k_m, k_{m-1} \ldots k_0)_2$, then

$$C_p = \begin{cases} 
\{x | x - k \equiv 2^p \mod 2^{p+1}\}, & 0 \leq p \leq m, \\
\{x | x - k \equiv 0 \mod 2^{m+1}\}, & p = m + 1.
\end{cases} \quad (8)$$

In other words, $C_p$, $0 \leq p \leq m$ contains all integers with binary representation $(k_m k_{m-1} \ldots k_0)_2$ and $C_{m+1}$ contains all integers with binary representation $(k_m k_{m-1} \ldots k_0)_2$. It’s easy to see $C_p$, $p = 0, 1, \ldots, m + 1$ is a decomposition of all integers and $[0, d - 1] \cup [n - d + 1, n] \subseteq \bigcup_{i=0}^{d} C_i$. 


For convenience of the following description, we define some families of equations, say $A_i$, $B_i$ and $E_i$, where

\[
A_i = \{ s_\alpha = 0 \mid \alpha \in \mathbb{F}_2^n, \text{wt}(\alpha) \in [0, d-1] \text{ and } \text{wt}(\alpha) \in C_i \},
\]

\[
B_i = \{ s_\alpha = 0 \mid \alpha \in \mathbb{F}_2^n, \text{wt}(\alpha) \in [n-d+1, n] \text{ and } \text{wt}(\alpha) \in C_i \},
\]

\[
E_i \in \{ A_i, B_i \},
\]

for $i = 0, 1, \ldots, \lfloor \log_2 d \rfloor$. Now, we use math induction to prove that $A_0$ or $B_0$, union $A_1$ or $B_1$, \ldots, union $A_p$ or $B_p$, denoted by $\cup_{i=0}^p E_i$, has the same solution space with $\cup_{i=0}^p A_i$, i.e., $\text{span}(\cup_{i=0}^p E_i) = \text{span}(\cup_{i=0}^p A_i)$, for $p = 0, 1, \ldots, \lfloor \log_2 d \rfloor$. The induction parameter is $p$.

**Basis step:** $p = 0$. First, we will prove that the solution space of $A_0$ is a subspace of that of $B_0$ by representing all the equations in $B_0$ as linear combinations of equations in $A_0$. Take an arbitrary equation $s_\alpha = 0$ in $B_0$, expanding $s_\alpha$ as follows,

\[
s_\alpha = \bigoplus_{\beta \leq \alpha} c_\beta = \bigoplus_{\beta \leq \alpha} \bigoplus_{0 \leq \text{wt}(\beta) \leq d-1} s_{\gamma}
\]

\[
= \bigoplus_{0 \leq \text{wt}(\gamma) \leq d-1} \left(s_\gamma \bigoplus_{0 \leq \text{wt}(\beta) \leq d-1} 1 \right)
\]

\[
= \bigoplus_{0 \leq \text{wt}(\gamma) \leq d-1} \left(s_\gamma \bigoplus_{i=0}^{d-1-\text{wt}(\gamma)} \left(\text{wt}(\alpha) - \text{wt}(\gamma)\right) \right) .
\]

Considering $s_\gamma$ in the (10), where $\text{wt}(\gamma) \notin C_0$, we want to show the coefficient of $s_\gamma$ is 0. By Lucas’ formula, we know $(\text{wt}(\alpha) - \text{wt}(\gamma)) = 1$ over $\mathbb{F}_2$ if and only if $i \leq \text{wt}(\alpha) - \text{wt}(\gamma)$. Note that $\text{wt}(\alpha) - \text{wt}(\gamma) = \ast k_0 = \ast 1_2$ and $d - 1 - \text{wt}(\gamma) = \ast k_0 = \ast 1_2$. Hence, if $i = (\ast k_0 0 0)$ satisfies $i \leq \text{wt}(\alpha) - \text{wt}(\gamma)$ and $i \leq d - 1 - \text{wt}(\gamma)$, then $i + 1 = (\ast 1 0)$ also satisfies the above constraints and vice versa. We conclude that an $i$ ended by 0 in its binary representation satisfying $i \leq \text{wt}(\alpha) - \text{wt}(\gamma)$ must correspond with another $i$ ended by 1 in the inner sum of (10). Thus, $\bigoplus_{i=0}^{d-1-\text{wt}(\gamma)} (\text{wt}(\alpha) - \text{wt}(\gamma)) = 0$ when $\gamma \notin C_0$, and all equations in $B_0$ could be represented as linear combinations of those in $A_0$. Therefore a solution of equations $A_0$ is also a solution of $B_0$, which implies the solution space of $A_0$ is a subspace of that of $B_0$.

By Lemma 3.2 it’s easy to see equations in both $A_0$ and $B_0$ are linearly independent. Since they have the same size, the dimensions of both solution spaces are the same. Therefore, the solution spaces of $A_0$ and $B_0$ are the same, which completes the basis step for $p = 0$.

**Induction step:** assuming the proposition is true for $p = q - 1$, $q \geq 1$, we will prove it’s also true for $p = q$.

First, we will prove the solution space of $\cup_{i=0}^p A_i$ is a subspace of that of $\cup_{i=0}^{p-1} A_i \cup B_q$. Taking an arbitrary $s_\alpha = 0$ in $B_q$, we want to show $s_\alpha$ can be represented as linear combinations of equations in $\cup_{i=0}^p A_i$. Similar with the method in basis step, expand $s_\alpha$ as

\[
\bigoplus_{0 \leq \text{wt}(\gamma) \leq d-1} \left(s_\gamma \bigoplus_{i=0}^{d-1-\text{wt}(\gamma)} \left(\text{wt}(\alpha) - \text{wt}(\gamma)\right) \right) .
\]
as well as in $\bigcup_{i=0}^{d-1} − \text{wt}(\gamma)$, $\text{wt}(\alpha) − \text{wt}(\gamma) = 0$ when $\text{wt}(\gamma) \notin \bigcup_{i=0}^{d-1} C_i$. Take an arbitrary $\gamma$ such that $\text{wt}(\gamma) \notin \bigcup_{i=0}^{d-1} C_i$. Noting that $\text{wt}(\alpha) = (k_0 k_1 \ldots k_2)$, $\text{wt}(\gamma) = (k_0 k_1 \ldots k_2)$ and $d = (k_{[\log_2 d]} \ldots k_0 k_1 \ldots k_2) = 1$, we have $\text{wt}(\alpha) − \text{wt}(\gamma) = (10 \ldots 0) = (10 \ldots 0)$ and $d − 1 − \text{wt}(\gamma) = (01 \ldots 1)$. It’s easy to see that if there is an $i = (0i_q \ldots i_0)$, $0 \leq i \leq d − 1 − \text{wt}(\gamma)$, satisfying $\text{wt}(\alpha) − \text{wt}(\gamma) = 1$, i.e., $i \leq \text{wt}(\alpha) − \text{wt}(\gamma)$, then $i + 2^q = (1i_q \ldots i_0)$ also satisfies $i + 2^q \leq \text{wt}(\alpha) − \text{wt}(\gamma)$ and $i + 2^q \leq d − 1 − \text{wt}(\gamma)$ and vice versa. Since this correspondence is one on one, the 1’s in the inner sum of (11) can be divided into pairs. Therefore, $\bigcup_{i=0}^{d-1} − \text{wt}(\gamma) = 0$ and all equations in $B_q$ can be written as sums of equations in $\bigcup_{i=0}^{d-1} A_i$. We conclude that the solution space of $\bigcup_{i=0}^{d-1} A_i$ is a subspace of that of $\bigcup_{i=0}^{d-1} A_i \cup B_q$.

By induction hypothesis, 
$$\text{span}(\bigcup_{i=0}^{d-1} A_i \cup B_q) = \text{span}(\bigcup_{i=0}^{d-1} B_i \cup B_q) = \text{span}(\bigcup_{i=0}^{d-1} B_i).$$

And by Lemma 3.2 it’s not hard to see there is no linear dependence in $\bigcup_{i=0}^{d-1} B_i$ as well as in $\bigcup_{i=0}^{d-1} A_i$. Note that $|\bigcup_{i=0}^{d-1} A_i| = |\bigcup_{i=0}^{d-1} B_i|$, the dimensions of the solution spaces of $\bigcup_{i=0}^{d-1} A_i$ and $\bigcup_{i=0}^{d-1} A_i \cup B_q$ are the same. Combining with the fact that solution space of $\bigcup_{i=0}^{d-1} A_i$ is a subspace of that of $\bigcup_{i=0}^{d-1} A_i \cup B_q$, we claim these two solution spaces are exactly the same. Using induction hypothesis again, we have 
$$\text{span}(\bigcup_{i=0}^{d-1} A_i) = \text{span}(\bigcup_{i=0}^{d-1} B_i \cup B_q) = \text{span}(\bigcup_{i=0}^{d-1} B_i \cup B_q) = \text{span}(\bigcup_{i=0}^{d-1} B_i),$$

which completes the induction.

Now, let’s go back to the original problem that proving $g = 0$. By the conditions in this theorem, for any $\alpha \in F_2^n$, $\text{wt}(\alpha) \in C_i \cap [0, d − 1]$, we have $f(\alpha) = m$; for any $\alpha \in F_2^n$, $\text{wt}(\alpha) \notin C_i \cap [0, d − 1]$, we have $f(\alpha) = m \oplus 1$, where $m = 0$ or 1. If $m = 1$, we could list equations on the point $\alpha$, where $\text{wt}(\alpha) \notin C_i \cap [0, d − 1]$, which is exactly the equations set $A_i$. If $m = 0$, we could list equations on the point $\alpha$, where $\text{wt}(\alpha) \notin C_i \cap [n − d + 1, n]$, which is exactly the equations set $B_i$. If let $t$ run over from 0 to $[\log_2 d]$, we obtain equations $\bigcup_{i=0}^{[\log_2 d]} E_i$, which is equivalent to $\bigcup_{i=0}^{[\log_2 d]} A_i$. By Lemma 3.2, $\bigcup_{i=0}^{[\log_2 d]} A_i$ has only zero solution, thus $\bigcup_{i=0}^{[\log_2 d]} E_i$ has only zero solution. Therefore, $g = 0$ and the proof is complete.

**Construction 3.4.** Given two positive integers $k, d$, where $d \leq k$ and $2 \leq d \leq k$, we construct a function $f$ in $SB_{2k}$ as follows.

- Choose $[\log_2 d] + 1$ numbers in $F_2$ arbitrarily, denoted by $m_0, m_1, \ldots, m_{[\log_2 d]}$.

- Define a symmetric Boolean function $f$ through it’s simplified value vector, which is

\[
v_f(i) = \begin{cases} 
m_i, & i \in C_t \cap [0, d − 1], 
m_t \oplus 1, & i \in C_t \cap [n − d + 1, n], 
0 \text{ or } 1, & \text{otherwise.} \end{cases}
\]
By Theorem 3.3, \( \text{AI}(f) \geq d \) for \( f \) in Construction 3.4. We present an example here to illustrate our construction. Let \( k = 6 = (110)_2 \) and \( d = k \). We have \( C_0 = \{1, 3, 5, 7, 9, 11, \ldots \} \), \( C_1 = \{0, 4, 8, 12, \ldots \} \) and \( C_2 = \{2, 10, \ldots \} \). Therefore, constraints \( v_f(1) = v_f(3) = v_f(5) = v_f(7) \oplus 1 = v_f(9) \oplus 1 = v_f(11) \oplus 1 \), \( v_f(0) = v_f(4) = v_f(8) \oplus 1 = v_f(12) \oplus 1 \) and \( v_f(2) = v_f(10) \oplus 1 \) must be satisfied. Let \( m_0, m_1, m_2 \in \mathbb{F}_2 \) take over all the 8 combinations, we obtain the following 8 functions with maximum algebraic immunity in Table 1.

<table>
<thead>
<tr>
<th>( m_0m_1m_2 )</th>
<th>( \text{SVV}: v_f(0) \ldots v_f(12) )</th>
<th>( \text{SANF}: \lambda_f(0) \ldots \lambda_f(12) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>0000000111111</td>
<td>00000001110000</td>
</tr>
<tr>
<td>000</td>
<td>0000001111111</td>
<td>00000010100000</td>
</tr>
<tr>
<td>001</td>
<td>0010000111011</td>
<td>00110010100000</td>
</tr>
<tr>
<td>001</td>
<td>0010001111011</td>
<td>00110010100000</td>
</tr>
<tr>
<td>010</td>
<td>1000100101110</td>
<td>11100010100000</td>
</tr>
<tr>
<td>010</td>
<td>1000101101110</td>
<td>11100010100000</td>
</tr>
<tr>
<td>011</td>
<td>1010100101010</td>
<td>11000010100000</td>
</tr>
<tr>
<td>011</td>
<td>1010101101010</td>
<td>11000010100000</td>
</tr>
<tr>
<td>100</td>
<td>0101010010101</td>
<td>01000010100000</td>
</tr>
<tr>
<td>100</td>
<td>0101011101011</td>
<td>01000010100000</td>
</tr>
<tr>
<td>101</td>
<td>0111010010001</td>
<td>01100010100000</td>
</tr>
<tr>
<td>101</td>
<td>0111011100001</td>
<td>01100010100000</td>
</tr>
<tr>
<td>110</td>
<td>1101100000100</td>
<td>10100010100000</td>
</tr>
<tr>
<td>110</td>
<td>1101110000100</td>
<td>10100010100000</td>
</tr>
<tr>
<td>111</td>
<td>1111110000000</td>
<td>10000010100000</td>
</tr>
<tr>
<td>111</td>
<td>1111111000000</td>
<td>10000011110000</td>
</tr>
</tbody>
</table>

Corollary 3.5. The number of symmetric Boolean functions on \( 2k \) variables, with algebraic immunity greater than or equal to \( d \), \( d \geq 2 \) and \( d \prec k \), is not less than

\[
2^{|\log_2 d|+2(k-d+1)}.
\]

Proof. We prove this by enumerating all the functions in Construction 3.4. There are \(|\log_2 d| + 1 \) numbers on \( \mathbb{F}_2 \) which can be chosen arbitrarily. To show different choices will generate different functions, it’s sufficient to prove \( C_t \cap [0, d-1] \neq \emptyset \) if \( 0 \leq t \leq |\log_2 d| + 1 \). It’s obvious that \( \sum_k k^t k_0 \leq 0 \). If \( 0 \leq t \leq |\log_2 d| + 1 \), it’s obvious that \( \sum_k k^t k_0 \leq 0 \). If \( t = |\log_2 d| + 2 \), \( \sum_k k^t k_0 \leq 0 \). Because \( k_i = 1 \), we have \( \sum_k k^t k_0 \leq 0 \).

Since the number of all choices for \( m_0, m_1, \ldots, m_{|\log_2 d|} \) is \( 2^{|\log_2 d|+1} \) and \( v_f(i) \) could take either 0 or 1 when \( i \in [d, n-d] \), the total number of such of \( f \) can be constructed is

\[
2^{|\log_2 d|+1} + n-d+1 = 2^{|\log_2 d|+2(k-d+1)},
\]

which completes our proof. \( \square \)

We present another example here to illustrate our counting result. Let \( k = 13 = (1101)_2 \), \( d = 5 = (101)_2 \prec k \). Hence \( C_0 = \{0, 2, 4, 6, \ldots \} \), \( C_1 = \{3, 7, \ldots \} \) and \( C_2 = \{1, 9, \ldots \} \). For arbitrary \( m_0, m_1, m_2 \in \mathbb{F}_2 \), \( m_0 = v_f(0) = v_f(2) = \ldots \) and \( m_1, m_2 \) take over all the 8 combinations, we obtain the following 8 functions with maximum algebraic immunity in Table 1.
\( v_f(4) = v_f(26) \oplus 1 = v_f(24) \oplus 1 = v_f(22) \oplus 1, m_1 = v_f(3) = v_f(23) \oplus 1 \) and \( m_2 = v_f(1) = v_f(25) \oplus 1 \) must be satisfied, while the others bits could take 0 or 1 arbitrarily. Let \( m_0, m_1, m_2 \) run over all 8 combinations, \( 2^{20} \) functions \( \in SB_{26} \) are constructed and listed in Table 2.

<table>
<thead>
<tr>
<th>( m_0m_1m_2 )</th>
<th>SVV: ( v_f(0) \ldots v_f(26) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>00000??\ldots??11111</td>
</tr>
<tr>
<td>001</td>
<td>01000??\ldots??1101</td>
</tr>
<tr>
<td>010</td>
<td>00010??\ldots??10111</td>
</tr>
<tr>
<td>011</td>
<td>01010??\ldots??10101</td>
</tr>
<tr>
<td>100</td>
<td>10101??\ldots??01010</td>
</tr>
<tr>
<td>101</td>
<td>11101??\ldots??01000</td>
</tr>
<tr>
<td>110</td>
<td>10111??\ldots??00010</td>
</tr>
<tr>
<td>111</td>
<td>11111??\ldots??00000</td>
</tr>
</tbody>
</table>

**References**


