The Weight and Nonlinearity of 2-rotation Symmetric Cubic Boolean Function

Hongli Liu

\* Institute of Mathematics and Statistics, Zhejiang University of Finance and Economics, China

Correspondence: Hongli Liu, Institute of Mathematics and Statistics, Zhejiang University of Finance and Economics, 18 XueYuan Street, Xiasha Higher Education Zone, Hangzhou 310018, China. Tel: 86-137-5827-3108. E-mail: ooohlhl@163.com

Received: March 25, 2015 Accepted: April 8, 2015 Online Published: May 23, 2015
doi:10.5539/jmr.v7n2p187 URL: http://dx.doi.org/10.5539/jmr.v7n2p187

The research is financed by the National Natural Science Foundation of China (Grant No:11302188), Zhejiang Provincial Natural Science Foundation of China (Grant No. LY14F010015).

Abstract

The conceptions of $\chi$-value and $K$-rotation symmetric Boolean functions are introduced by Cusick. $K$-rotation symmetric Boolean functions are a special rotation symmetric functions, which are invariant under the $k - th$ power of $\rho$. In this paper, we discuss cubic 2-value 2-rotation symmetric Boolean function with $2n$ variables, which denoted by $F^2n(\chi^{2n})$. We give the recursive formula of weight of $F^2n(\chi^{2n})$, and prove that the weight of $F^2n(\chi^{2n})$ is the same as its nonlinearity.

Keywords: Rotation symmetric Boolean function, Nonlinearity, Weight, $\chi$–value

1. Introduction

Boolean functions have many applications in coding theory and cryptography. Rotation symmetric Boolean functions(RSBF) as invariant Boolean functions under rotation transform have been widely studied. Higher nonlinearity is a very important character of Boolean functions which are widely used in coding theory and S-box design. Rotation symmetric Boolean functions as a subclass of $K – rotation symmetric$ have not higher nonlinearity. So, $K – rotation symmetric$ Boolean functions which are the generalization of notion of rotation symmetric function were proposed by Selçuk Kavut. The applications of the $k – rotation symmetric(k \geq 2)$ to coding theory and S-box design can be found in some papers. Cusick gave the definition of cubic 2 – rotation symmetric Boolean functions and used the notation $[2 – (1, r, s)]_{2n} : 2n \geq s$ as the cubic monomial 2-rotation symmetric functions (denoted by $2 – functions$). Cusick also described the affine equivalence of cubic MRS 2-rotation symmetric, and proved that the sequence of Hamming weights of $[2 – (1, r, s)]_{2n} : 2n \geq s$ satisfies a linear recursion with integer coefficients. In this paper, we will give the recursion formula of Hamming weight of $[2 – (1, 2, 3)]_{2n}(2n \geq 10)$ and prove that the nonlinearity of $[2 – (1, 2, 3)]_{2n}(2n \geq 10)$ is the same as its weight.

2. Preliminaries

Let $\mathbb{F}_2 = \{0, 1\}$ be the binary field, $\mathbb{F}_2^n$ be the $n$–dimensional vector space of over $\mathbb{F}_2$. A Boolean function in $n$ variables can be defined as a map from $\mathbb{F}_2^n$ into $\mathbb{F}_2$, denoted by $f^n(x^n)$, or $f^n$ in brief, where $x^n = (x_1, x_2, \cdots, x_n)$. Every Boolean function $f^n$ has a unique polynomial representation (usually called the algebraic normal form (ANF)), and the degree of $f^n$ is the degree of this polynomial(deg($f^n$) in brief). If every term in the algebraic normal form of $f^n$ has the same degree, then the function is said to be homogeneous. A Boolean function $f^n$ is called affine, if $deg(f^n) = 1$. If $f^n$ is affine and homogeneous(i.e.the constant term is 0), $f^n$ is said to be linear. The truth table of $f^n$ is defined to be the binary sequence $v_1, v_2, \cdots, v_{2^n}$, where the bits $v_1 = f((0, 0, \cdots, 0)), v_2 = f((0, 0, \cdots, 1)), \cdots, v_{2^n} = f((1, 1, \cdots, 1))$. The Hamming weight of a Boolean function $f^n$ is defined as the number of nonzero coordinates in its truth table, denoted by $wt(f^n)$. The Hamming distance $d(f^n, g^n)$ between two Boolean functions $f^n$ and $g^n$ is defined as the number of their different coordinates, which equals the Hamming weight of their sum $f + g$, where $+$ denotes the addition on $\mathbb{F}_2$. Two Boolean functions $f^n$ and $g^n$ in $n$ variables are said to be affine equivalent if there exists an invertible matrix $A$ with entries in $\mathbb{F}_2$ and $b \in \mathbb{F}_2^n$ such that $f^n(x) = g^n(Ax + b)$. 187
Definition 1 The nonlinearity \(NL(f^n)\) of a Boolean function \(f^n(x^n)\) is defined as

\[
NL(f^n) = \text{Min} \{d(f^n(x^n), c^n \cdot x^n) | c^n \in \mathbb{F}_2^n\},
\]

where \(\cdot\) is the vector dot product.

It is easy to see that if \(f^n\) and \(g^n\) are affine equivalent, then \(wt(f^n) = wt(g^n)\) and \(NL(f^n) = NL(g^n)\). We say that the weight and nonlinearity are affine invariants.

Definition 2 For a Boolean function \(f^n(x^n)\). The Fourier transform of \(f^n\) at \(c^n \in \mathbb{F}_2^n\) is defined as

\[
\hat{f}^n(c^n) = \sum_{x^n \in \mathbb{F}_2^n} (-1)^{f^n(x^n) + c^n \cdot x^n}.
\]

Definition 3 A Boolean function \(f^n(x^n)\) is called rotation symmetric if

\[
f^n(x_1, x_2, \ldots, x_n) = f^n(\rho(x_1, x_2, \ldots, x_n)), \text{ for all } (x_1, x_2, \ldots, x_n) \in \mathbb{F}_2^n,
\]

where \(\rho(x_1, x_2, \ldots, x_{n-1}, x_n) = (x_n, x_1, x_2, \ldots, x_{n-1})\).

If a monomial \(x_1x_2x_3\) appears in a rotation symmetric Boolean function as a term then all monomials in the orbit of \(x_1x_2x_3\) should appear in the function as terms. The rotation symmetric function is said to be monomial rotation symmetric (MRS) if it is generated by applying powers of \(\rho\) to a single monomial. We use the notation \((1, r, s)_n\) for the cubic MRS function in \(n\) variables generated by the monomial \(x_1x_rx_s\). A Boolean function is said to be \(k - \text{rotation symmetric}\) if it is invariant under the \(k - \text{th}\) power of \(\rho\) but not under any smaller power. A Boolean function is said to be monomial \(k - \text{rotation symmetric}\) if it is generated by applying powers of \(\rho^k\) to a single monomial. For brevity, we will refer to these functions as \(k - \text{functions}\). In this paper, the cubic 2-functions shall be discussed. We use the notation \(2 - (1, r, s)_{2n}\) for the cubic 2-function in \(2n\) variables generated by the monomial \(x_1x_rx_s\). If we assume \(r < s \leq 2n\) then the formula

\[
2 - (1, r, s)_{2n} = x_1x_rx_s + x_3x_{r+2}x_{s+2} + \cdots + x_{2n-1}x_{r-2}x_{s-2}.
\]

is called a standard form of the above 2-function.

A monomial \([a, b, c]\) in a cubic 2-function is said to be pure form, if \(a, b, c\) are all even or odd. A monomial that is not pure form is said to be mixed form. It is obvious that every monomial of \(2 - (1, r, s)_{2n}\) has the same form. A 2-function is said to be mixed form 2-function if its terms are mixed form. Otherwise, it is said to be pure form 2-function.

Definition 4 (\(\chi - \text{value}\)) Let \(2 - (1, r, s)_{2n}\) be a mixed form 2-function with monomial \([a, b, c]\)(\(a < b < c\)). Assume \(a\) is even(odd) and \(b, c\) are odd(even). Then the \(\chi\) value for \(2 - (1, r, s)_{2n}\) is defined as \(\chi = c - b\).

Theorem 1 Two 2-functions \(2 - (1, r, s)_{2n}\) and \(2 - (1, p, q)_{2n}\) are affine equivalent by some permutation for all \(n\) if and only if their \(\chi\)-values are equal.

Theorem 1 tells us that all 2-values functions with \(2n\) variables have the same weights and nonlinearity. So, in the following section, we will discuss the weight and nonlinearity of 2-values function \(2 - (1, 2, 3)_{2n}\).

3. The Weight of 2-values Function \(F^{2n}(x^{2n})\)

In this section, we shall study the recursive formula for weight of \(2 - (1, 2, 3)_{2n}\). Firstly, we give the standard form of 2-values function \(2 - (1, 2, 3)_{2n}\), denoted by \(F^{2n}(x^{2n})\) or \(F^{2n}\).

\[
F^{2n}(x^{2n}) = x_1x_2x_3 + x_3x_4x_5 + \cdots + x_{2n-3}x_{2n-2}x_{2n-1} + x_{2n-1}x_{2n}x_1.
\]

If \(T\) is a string, then \(\overline{T}\) denotes the complemented string with 0 and 1 interchanged. If \(X\) is a 4-bit block or a string of blocks, then \((X)_s\) or \(X_s\) is the string obtained by concatenation of \(s\) copies of \(X\). The concatenation of two strings \(u, v\) will be denoted by \(uv\) or \(u|v\). Now we define two sets of 4-bit strings

\[
T_1 = \{A = 0, 0, 1, 1; \overline{A} = 1, 1, 0, 0; B = 0, 1, 0, 1; \overline{B} = 1, 0, 1, 0; C = 0, 1, 1, 0; \overline{C} = 1, 0, 0, 1; D = 0, 0, 0, 0; \overline{D} = 1, 1, 1, 1\}
\]

and

\[
T_2 = \{U = 1, 0, 0, 0; \overline{U} = 0, 1, 1, 1; V = 0, 0, 0, 1; \overline{V} = 1, 1, 1, 0; X = 0, 1, 0, 0; \overline{X} = 1, 0, 1, 1; Y = 0, 0, 1, 0; \overline{Y} = 1, 1, 0, 1\}.
\]
We give the following result about the truth tables of monomials for $F^{2n}(x^{2n})$.

**Lemma 2** The truth table of any monomial for $F^{2n}(x^{2n})$ is
\[ x_i x_{i+1} x_{i+2} = (D_2^{2n-2} (D_2^{2n-3} (D_2^{2n-4} D_2^{2n-5}))))_{2i+1} \quad 1 \leq i \leq 2n - 5, \text{ and } i \text{ is odd.} \]
\[ x_{2n-1} x_{2n-2} x_{2n-1} = (D D D A)_{2n-1}. \]
\[ x_{2n-1} x_{2n} = D_{2n-3} V_{2n-1}. \]

From Lemma 2, we give the following algorithm as the output of truth table for $F^{2n}(x^{2n})$.

**Algorithm 1**

Step 5: $h^5_i \leftarrow D D D A D D A, h^3_i \leftarrow V V Y V Y V Y$.

Step 6: $h^s_i \leftarrow (h_i^{s-2} || h_i^{s-3})_2, i = 1, 2$, for odd $s$.

Output: $H_1 \leftarrow h^{2n-1}_1, H_2 \leftarrow \overline{h^{2n-1}_2}$, where $\overline{h^i}$ is the string obtained from $h^i$ by complementing its last $2^s-2$ bits. Write $F^{2n} = H_1 \parallel H_2$.

From the above algorithm, we give the recursive relationship of weight for $F^{2n}(x^{2n})$.

**Theorem 3** The weight of Boolean function $F^{2n}(x^{2n})$ satisfy
\[ \text{wt}(F^{2n}) = 2 \text{wt}(F^{2n-2}) + 4 \text{wt}(F^{2n-4}) + 2^{2n-3}. \]

**Proof.** Using Algorithm 1, we have
\[ \text{wt}(F^{2n}(x^{2n})) = \text{wt}(H_1) + \text{wt}(H_2) = \text{wt}(h^{2n-1}_1) + \text{wt}(\overline{h^{2n-1}_2}) \]
and
\[ h^{2n-1}_1 = h^{2n-3}_1 h^{2n-3}_1 h^{2n-3}_1 \]
\[ h^{2n-1}_2 = h^{2n-3}_1 h^{2n-3}_1 h^{2n-3}_1 \]
\[ h^{2n-3}_1 = h^{2n-5}_1 h^{2n-5}_1 h^{2n-5}_1 \]
\[ \overline{h^{2n-3}_1} = h^{2n-5}_1 h^{2n-5}_1 h^{2n-5}_1. \]

Therefore,
\[ \text{wt}(h^{2n-1}_1) = 2(\text{wt}(h^{2n-5}_1) + \text{wt}(\overline{h^{2n-5}_1})) = 2(4\text{wt}(h^{2n-5}_1) + 2\text{wt}(h^{2n-5}_1) + 2^{2n-5}) = 2(2\text{wt}(h^{2n-5}_1) + 2\text{wt}(h^{2n-5}_1) + 2^{2n-5}) \]
\[ = 2(2\text{wt}(h^{2n-5}_1) + \text{wt}(h^{2n-3}_1) + 2^{2n-5}) = 4\text{wt}(h^{2n-5}_1) + 2\text{wt}(h^{2n-3}_1) + 2^{2n-4}. \]

Similarly, we have
\[ \text{wt}(\overline{h^{2n-1}_1}) = 4\text{wt}(h^{2n-5}_1) + 2\text{wt}(h^{2n-3}_1) + 2^{2n-4}. \]

From (1), (2) and (3), we have
\[ \text{wt}(F^{2n}(x^{2n})) = \text{wt}(h^{2n-1}_1) + \text{wt}(\overline{h^{2n-1}_2}) = 4\text{wt}(h^{2n-5}_1) + 2\text{wt}(h^{2n-3}_1) + 2^{2n-4} + 4\text{wt}(h^{2n-5}_1) + 2\text{wt}(\overline{h^{2n-3}_2}) + 2^{2n-4} = 4\text{wt}(F^{2n-4}) + 2\text{wt}(F^{2n-2}) + 2^{2n-3}. \]

**4. The Nonlinearity of $F^{2n}(x^{2n})$**

Cusick and Stânica conjectured that the nonlinearity of cubic 1-values function $F^n(x)$ is the same as the weight, and Zhang et al. proved the conjecture. In this section, we shall prove the same result for $F^{2n}(x^{2n})$, that is,
\[ \text{wt}(F^{2n}) = NL(F^{2n}). \]
By the definitions of Fourier transform and Hamming weight, we can easily deduce that
\[ \text{wt}(F^{2n}(x^{2n})) = \frac{1}{2}(2^{2n} - F^{2n}(0)). \]

Therefore, we can restate (4) as
\[ \widehat{F^{2n}(0)} = \text{Max}(|\widehat{F^{2n}(c^{2n})}|c^{2n} \in \mathbb{F}^{2n}). \tag{5} \]

On the other hand, the recursion formula of \( F^{2n}(0) \) can be obtained by applying the recursion formula of \( \text{wt}(F^{2n}(x^{2n})) \).

\[
\begin{align*}
F^{2n}(0) &= 2^{2n} - \text{wt}(F^{2n}) \\
&= 2^{2n} - 2 \cdot [2\text{wt}(F^{2n-2}) + 4\text{wt}(F^{2n-4}) + 2^{2n-3}] \\
&= 2[2^{2n-1} - 2\text{wt}(F^{2n-2}) - 4\text{wt}(F^{2n-4}) - 2^{2n-3}] \\
&= 2[2^{2n-2} - 2\text{wt}(F^{2n-2}) + 2 \cdot 2^{2n-4} - 4\text{wt}(F^{2n-4})] \\
&= 2[F^{2n-2}(0) + 2F^{2n-4}(0)]. \tag{6}
\end{align*}
\]

Before giving the proof of (5), we need some notation:

\[ l_{2n-1} = \sum_{1 \leq i \leq 2n-3, \text{i is odd}} x_i x_{i+1} x_{i+2}, \]

\[ f_{2}^{2n-1}(x_1, x_2, \ldots, x_{2n-1}) = l_{2n-1}, \]

\[ f_{3}^{2n-1}(x_1, x_2, \ldots, x_{2n-1}) = l_{2n-1} + x_1, \]

\[ f_{4}^{2n-1}(x_1, x_2, \ldots, x_{2n-1}) = l_{2n-1} + x_{2n-1}, \]

\[ f_{5}^{2n-1}(x_1, x_2, \ldots, x_{2n-1}) = l_{2n-1} + x_{2n-1} x_1. \]

Firstly, we give the following recursive relations about \( f_{i}^{2n-1}(c^{2n-1}) \).

**Lemma 4** For every \( c^{2n-1} = (c_1, c_2, \ldots, c_{2n-1}) \in \mathbb{F}^{2n-1} \), we have

\[
\begin{align*}
f_{2}^{2n-1}(c^{2n-1}) &= (1 + (-1)c_{2n-1} + (-1)c_{2n-2})f_{1}^{2n-3}(c^{2n-3}) + (-1)c_{2n-2} c_{2n-3} f_{1}^{2n-3}(c^{2n-3}), \quad i = 1, 2, \\
f_{3}^{2n-1}(c^{2n-1}) &= (1 - (-1)c_{2n-1} + (-1)c_{2n-2})f_{1}^{2n-3}(c^{2n-3}) - (-1)c_{2n-2} c_{2n-3} f_{1}^{2n-3}(c^{2n-3}), \quad i = 3, 4, \\
f_{5}^{2n-1}(c^{2n-1}) &= (1 + (-1)c_{2n-2})f_{1}^{2n-3}(c^{2n-3}) - (-1)c_{2n-2} f_{3}^{2n-3}(c^{2n-3}) + (-1)c_{2n-2} c_{2n-3} f_{4}^{2n-3}(c^{2n-3}), \quad i = 5,
\end{align*}
\]

where \( c^{2n-2} \) and \( c^{2n-3} \) are the first \( 2n - 2 \) and \( 2n - 3 \) bits of \( c^{2n-1} \).

**proof** We prove the relation for \( i = 1 \), since the proof of the others are similar.

\[
\begin{align*}
f_{1}^{2n-1}(c^{2n-1}) &= \sum_{x_{2n-1}=0, x_{2n-2}=0} (-1)f_{1}^{2n-1}(x_{2n-1} c_{2n-1} c_{2n-2}) + \sum_{x_{2n-1}, x_{2n-2}=1} (-1)f_{1}^{2n-1}(x_{2n-1} c_{2n-1} c_{2n-2}) \\
&+ \sum_{x_{2n-1}=0, x_{2n-2}=1} (-1)f_{1}^{2n-1}(x_{2n-1} c_{2n-1} c_{2n-2}) + \sum_{x_{2n-1}, x_{2n-2}=1} (-1)f_{1}^{2n-1}(x_{2n-1} c_{2n-1} c_{2n-2}) \\
&= \sum_{x_{2n-1}} (-1)f_{1}^{2n-1}(x_{2n-1} c_{2n-1} c_{2n-2}) + \sum_{x_{2n-1}} (-1)f_{1}^{2n-1}(x_{2n-1} c_{2n-1} c_{2n-2}) \\
&+ \sum_{x_{2n-1}} (-1)f_{1}^{2n-1}(x_{2n-1} c_{2n-1} c_{2n-2}) + \sum_{x_{2n-1}} (-1)f_{1}^{2n-1}(x_{2n-1} c_{2n-1} c_{2n-2}) \\
&= f_{1}^{2n-3}(c^{2n-3}) + (-1)f_{1}^{2n-3}(c^{2n-3}) + (-1)f_{1}^{2n-3}(c^{2n-3}) + (-1)f_{1}^{2n-3}(c^{2n-3}) \\
&= (1 + (-1)c_{2n-1})f_{1}^{2n-3}(c^{2n-3}) + (-1)f_{1}^{2n-3}(c^{2n-3}) + (-1)f_{1}^{2n-3}(c^{2n-3}) + (-1)f_{1}^{2n-3}(c^{2n-3}).
\end{align*}
\]
From lemma 4, we can easily deduce the following corollary.

**Corollary 5** For every \( c^{2n-1} = (c_1, c_2, \ldots, c_{2n-1}) \in \mathbb{Z}_2^{2n-1} \), we have

\[
\begin{align*}
\hat{f}_i^{2n-1}(c^{2n-1}) &= 3 \hat{f}_i^{2n-3}(c^{2n-3}) + \hat{f}_i^{2n-3}(c^{2n-3}) - \frac{1}{2} \hat{f}_i^{2n-3}(c^{2n-3}) - \frac{1}{2} \hat{f}_i^{2n-3}(c^{2n-3}) \quad & i = 1, 2 \\
\hat{f}_i^{2n-1}(c^{2n-1}) &= \hat{f}_i^{2n-3}(c^{2n-3}) - \frac{1}{2} \hat{f}_i^{2n-3}(c^{2n-3}) - \frac{1}{2} \hat{f}_i^{2n-3}(c^{2n-3}) - \frac{1}{2} \hat{f}_i^{2n-3}(c^{2n-3}) \quad & i = 3, 4 \\
\end{align*}
\]

if \( c_{2n-2} = 0, c_{2n-1} = 0 \);

if \( c_{2n-2} = 0, c_{2n-1} = 1 \);

if \( c_{2n-2} = 1, c_{2n-1} = 0 \);

if \( c_{2n-2} = 1, c_{2n-1} = 1 \).

Table 1. The values of \( NL(F^{2n}) \).

<table>
<thead>
<tr>
<th>( 2n )</th>
<th>( 8 )</th>
<th>( 10 )</th>
<th>( 12 )</th>
<th>( 14 )</th>
<th>( 16 )</th>
<th>( 18 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( f )</td>
<td>72</td>
<td>336</td>
<td>1472</td>
<td>6336</td>
<td>26752</td>
<td>111616</td>
</tr>
</tbody>
</table>

Table 2. The values of \( F^{2n}(0) \).

<table>
<thead>
<tr>
<th>( 2n )</th>
<th>( 8 )</th>
<th>( 10 )</th>
<th>( 12 )</th>
<th>( 14 )</th>
<th>( 16 )</th>
<th>( 18 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( f )</td>
<td>112</td>
<td>352</td>
<td>1152</td>
<td>3712</td>
<td>12032</td>
<td>38912</td>
</tr>
</tbody>
</table>

The following lemma give the properties of \( F^{2n}(0) \).

**Lemma 6** \( F^{2n}(0) \) satisfies the relationship: \( F^{2n}(0) > 0 \) and \( 2F^{2n}(0) < F^{2n+2}(0) \).

**proof** We prove it by math induction. From Table 2, we can see the two results are true for \( 2n = 6, 8, 10, 12, 14, 16, 18 \). Assume that, for an arbitrary \( 2n \), the result is also true. Let’s derive the correctness of conclusion for \( 2n + 2 \) from this assumption.

From (6) and the assumption of induction, we have

\[
2F^{2n+2}(0) = 2(2F^{2n}(0) + 4F^{2n-2}(0)) < 2F^{2n+2}(0) + 4F^{2n}(0) = F^{2n+4}(0)
\]

and

\[
F^{2n+2}(0) = 2F^{2n}(0) + 4F^{2n-2}(0) > 0.
\]

Which exactly means that the result holds for \( 2n + 2 \).

**Lemma 7** Let \( c^{2n-1} = (c_1, c_2, \ldots, c_{2n-1}) \in \mathbb{Z}_2^{2n-1} \). If \( c_1 = 1 \), then

\[
|\hat{f}_i^{2n-1}(c^{2n-1})| < \frac{1}{2} F^{2n}(0), (i = 1, 5), \quad |\hat{f}_i^{2n-1}(c^{2n-1})| < \frac{1}{4} F^{2n+2}(0), (i = 2, 3, 4).
\]

\[
|\hat{f}_i^{2n-1}(c^{2n-1})| < \frac{1}{10} F^{2n+2}(0), \quad |\hat{f}_i^{2n-1}(c^{2n-1})| < \frac{3}{40} F^{2n+4}(0).
\]

**proof**
When $c_{2n-2} = 0, c_{2n-1} = 0$, we have
\[
\begin{align*}
\widetilde{f_{1}^{2n-1}}(c^{2n-1}) &= 3\widetilde{f_{1}^{2n-3}}(c^{2n-3}) + \widetilde{f_{3}^{2n-3}}(c^{2n-3}) \\
\widetilde{f_{2}^{2n-1}}(c^{2n-1}) &= 3\widetilde{f_{2}^{2n-3}}(c^{2n-3}) + \widetilde{f_{4}^{2n-3}}(c^{2n-3}) \\
\widetilde{f_{3}^{2n-1}}(c^{2n-1}) &= \widetilde{f_{1}^{2n-3}}(c^{2n-3}) - \widetilde{f_{3}^{2n-3}}(c^{2n-3}) \\
\widetilde{f_{4}^{2n-1}}(c^{2n-1}) &= \widetilde{f_{2}^{2n-3}}(c^{2n-3}) + \widetilde{f_{4}^{2n-3}}(c^{2n-3}) \\
\widetilde{f_{5}^{2n-1}}(c^{2n-1}) &= 2\widetilde{f_{1}^{2n-3}}(c^{2n-3}) + \widetilde{f_{2}^{2n-3}}(c^{2n-3}) + \widetilde{f_{4}^{2n-3}}(c^{2n-3}).
\end{align*}
\]

We prove it by math induction. The maximum values of $|\widetilde{f_{i}^{11}}(c^{11})|(i = 1, \ldots, 5)$ can be obtained with the help of Matlab soft, which are 352, 672, 672, 672, 352. From Table 2, we can see $\widetilde{f_{i}^{11}}(c^{11})(i = 1, 2, 3, 4) < \frac{1}{4} F_{14}(0), \widetilde{f_{1}^{11}}(c^{11}) < \frac{1}{10} F_{14}(0)$, and $\widetilde{f_{2}^{11}}(c^{11}) < \frac{3}{40} F_{16}(0)$.

Suppose the results are true for $2n - 1(n \geq 6)$, we prove that it is true for $2n + 1$.
\[
\begin{align*}
|\widetilde{f_{1}^{2n+1}}(c^{2n+1})| &= |3\widetilde{f_{1}^{2n-1}}(c^{2n-1}) + \widetilde{f_{3}^{2n-1}}(c^{2n-1})|
= |2\widetilde{f_{1}^{2n-1}}(c^{2n-1}) + \widetilde{f_{3}^{2n-1}}(c^{2n-1}) + \widetilde{f_{1}^{2n-3}}(c^{2n-3})|
= |2\widetilde{f_{1}^{2n-1}}(c^{2n-1}) + 4\widetilde{f_{1}^{2n-3}}(c^{2n-3})|
\leq |2\widetilde{f_{1}^{2n-1}}(c^{2n-1})| + 4|\widetilde{f_{1}^{2n-3}}(c^{2n-3})|
< \frac{1}{2}(2F_{2n}(0) + 4F_{2n-2}(0))(< \frac{1}{10}(2F_{2n+2}(0) + 4F_{2n}(0)))
= \frac{1}{2} F_{2n+2}(0)(= \frac{1}{10} F_{2n+4}(0)).
\end{align*}
\]
\[
\begin{align*}
|\widetilde{f_{3}^{2n+1}}(c^{2n+1})| &= |3\widetilde{f_{2}^{2n-1}}(c^{2n-1}) + \widetilde{f_{4}^{2n-1}}(c^{2n-1})|
= |2\widetilde{f_{2}^{2n-1}}(c^{2n-1}) + \widetilde{f_{4}^{2n-1}}(c^{2n-1}) + \widetilde{f_{2}^{2n-3}}(c^{2n-3})|
= |2\widetilde{f_{2}^{2n-1}}(c^{2n-1}) + 4\widetilde{f_{2}^{2n-3}}(c^{2n-3})|
\leq |2\widetilde{f_{2}^{2n-1}}(c^{2n-1})| + 4|\widetilde{f_{2}^{2n-3}}(c^{2n-3})|
< \frac{1}{4}(2F_{2n+2}(0) + 4F_{2n}(0))(< \frac{3}{40}(2F_{2n+4}(0) + 4F_{2n+2}(0)))
= \frac{1}{4} F_{2n+4}(0)(= \frac{3}{40} F_{2n+6}(0)).
\end{align*}
\]
\[
\begin{align*}
|\widetilde{f_{3}^{2n+1}}(c^{2n+1})| &= |3\widetilde{f_{4}^{2n-1}}(c^{2n-1}) - \widetilde{f_{3}^{2n-1}}(c^{2n-1})|
= |2\widetilde{f_{4}^{2n-1}}(c^{2n-1}) - \widetilde{f_{3}^{2n-1}}(c^{2n-1}) + \widetilde{f_{4}^{2n-3}}(c^{2n-3})|
= |2\widetilde{f_{4}^{2n-1}}(c^{2n-1}) - 4\widetilde{f_{4}^{2n-3}}(c^{2n-3})|
\leq |2\widetilde{f_{4}^{2n-1}}(c^{2n-1})| + 4|\widetilde{f_{4}^{2n-3}}(c^{2n-3})|
< \frac{1}{2}(2F_{2n}(0) + 4F_{2n-2}(0))
= \frac{1}{2} F_{2n+2}(0)(= \frac{1}{4} F_{2n+4}(0)).
\end{align*}
\]
\[
\begin{align*}
|\widetilde{f_{4}^{2n+1}}(c^{2n+1})| &= |2\widetilde{f_{2}^{2n-1}}(c^{2n-1}) - \widetilde{f_{4}^{2n-1}}(c^{2n-1})|
= |2\widetilde{f_{2}^{2n-1}}(c^{2n-1}) + \widetilde{f_{4}^{2n-1}}(c^{2n-1}) + \widetilde{f_{2}^{2n-3}}(c^{2n-3})|
= |2\widetilde{f_{2}^{2n-1}}(c^{2n-1}) + 4\widetilde{f_{2}^{2n-3}}(c^{2n-3})|
\leq |2\widetilde{f_{2}^{2n-1}}(c^{2n-1})| + 4|\widetilde{f_{2}^{2n-3}}(c^{2n-3})|
< \frac{1}{2}(2F_{2n}(0) + 4F_{2n-2}(0))
= \frac{1}{2} F_{2n+2}(0)(= \frac{1}{4} F_{2n+4}(0)).
\end{align*}
\]
A recursive formula for weights of Boolean rotation symmetric functions:


References

The work was supported by the National Natural Science Foundation of China(Grant No:11302188), Zhejiang Provincial Natural Science Foundation of China (Grant No. LY14F010015).

Theorem 8 For all \( c^{2n} = (c_1, c_2, \ldots, c_{2n}) \neq 0 \) and \( n \geq 6 \), we have

\[
|F_{2n}(c^{2n})| < F_{2n}(0).
\]

proof We factor \( F_{2n}(c^{2n}) \) into two sub-functions.

\[
|F_{2n}(c^{2n})| = |f_1^{2n-1}(c^{2n-1}) + f_2^{2n-1}(c^{2n-1})| = |f_1^{2n-1}(c^{2n-1})| + |f_2^{2n-1}(c^{2n-1})| < \frac{1}{2} F_{2n}(0) + \frac{1}{2} F_{2n}(0) = F_{2n}(0).
\]

Theorem 8 tells us that the nonlinearity of \( F_{2n}(x^{2n}) \) is the same as its weight.

4. Conclusion

This paper gives the recursive formula of weight about 2-values cubic Boolean functions with 2n variables, and proves that the weight of \( F_{2n} \) is the same as its nonlinearity. The recursive formula of weight about 2t-values(\( t=2,4,\cdots \)) cubic Boolean functions can be discussed and the relationship of weight between 2-value and 2t-value functions can also be studied.

Acknowledgements

The work was supported by the National Natural Science Foundation of China(Grant No:11302188), Zhejiang Provincial Natural Science Foundation of China (Grant No. LY14F010015).

References


http://dx.doi.org/10.1016/j.ins.2012.02.030


http://dx.doi.org/10.1016/j.dam.2008.06.022

**Copyrights**

Copyright for this article is retained by the author(s), with first publication rights granted to the journal. This is an open-access article distributed under the terms and conditions of the Creative Commons Attribution license (http://creativecommons.org/licenses/by/3.0/).