Int. J. Appl. Math. Comput. Sci., 2003, Vol. 13, No. 4, 567–575

THE SPECTRAL TEST OF THE BOOLEAN FUNCTION LINEARITY

PIOTR PORWIK*

* Institute of Informatics, Silesian University
ul. Będzińska 39, 41–200 Sosnowiec, Poland
e-mail: porwik@us.edu.pl

The paper discusses the problem of recognizing the Boolean function linearity. A spectral method of the analysis of Boolean functions using the Walsh transform is described. Linearity and nonlinearity play important roles in the design of digital circuits. The analysis of the distribution of spectral coefficients allows us to determine various combinatorial properties of Boolean functions, such as redundancy, monotonicity, self-duality, correcting capability, etc., which seems more difficult be performed by means of other methods. In particular, the basic synthesis method described in the paper allows us to compute the spectral coefficients in an iterative manner. The method can be easily used in investigations of large Boolean functions (of many variables), which seems very attractive for modern digital technologies. Experimental results demonstrate the efficiency of the approach.

Keywords: Walsh coefficients, coefficients distribution, Boolean functions, bent functions, linearity measure of a Boolean function

1. Introduction

Transformations between the Boolean and spectral domains have been extensively studied by several authors (Ahmed and Rao, 1975; Blahut, 1983; Harmuth, 1977; Hurst *et al.*, 1985; Karpovsky, 1976; Porwik and Falkowski, 1999). This research has been carried out because many problems of digital logic can be solved more efficiently in the spectral domain than in the Boolean one. Theoretically, the techniques based on the Walsh transform provide some nice properties such as Boolean function classification, disjoint decomposition, multiplexer and threshold logic synthesis, state assignment, testing and evaluation of logic complexity (Hurst *et al.*, 1985; Falkowski and Kannurao, 2000; Porwik and Falkowski, 1999). In practice, spectral methods are not always attractive because they involve the additional conversion from the Boolean to the spectral domain and unfortunately there are matrix-based methods which are inefficient for large Boolean functions (Clarke *et al.*, 1993). In some cases spectral methods can be effectively applied to solve mathematical and practical problems (Blahut, 1983). One of these problems is to check the linearity of Boolean functions by means of the Walsh-Hadamard spectral technique. This paper presents the method which allows us to investigate the linearity of Boolean functions directly on the basis of the Walsh coefficients. The presented method is characterized by low complexity and can be applied to all n variables of Boolean functions. Linearity or nonlinearity measures are a very important feature of a Boolean function. Nowadays, some investigations of the linearity (nonlinearity) of functions are applied in many areas, e.g. in cryptography, data encryption, ciphers, error control codes, projects of the so-called s-boxes, evaluation of the Reed-Muller form, etc.

2. Preliminaries

Let V_n be a vector space of n tuples of elements from $GF(2)$. For this space there is a natural one-to-one correspondence between any vectors in V_n and integers in $[0, \ldots, 2^n-1]$. This allows ordering the vectors according to their corresponding integer values. If f is a Boolean function from V_n, then it can be expressed as a unique polynomial in n co-ordinates $x_1, x_2, \ldots, x_n$. For this reason f will be identified as a unique multi-variable polynomial $f(x)$, where $x = (x_1, x_2, \ldots, x_n)$.

Definition 1. An n-variable Boolean function $f(x_1, x_2, \ldots, x_n)$ can be written as

$$\sum_{j=0}^{2^n-1} y_j x_1^{b_1} x_2^{b_2} \cdots x_n^{b_n},$$

where $b_1, b_2, \ldots, b_n \in \{0, 1\}$ and $b_1 b_2 \ldots b_n$ is an n-bit binary number represented by j, $x_i^{b_i=0} = \overline{x}_i$, $x_i^{b_i=1} = x_i$ for $i = 1, 2, \ldots, n$. Then $\mathbf{Y} = [y_0, y_1, \ldots, y_{2^n-1}]$, $y_j \in \{0, 1\}$ is the truth vector of f.

Example 1. The truth vector of the three-variable Boolean function $f(x_1, x_2, x_3) = \overline{x}_1\overline{x}_2\overline{x}_3 + \overline{x}_1 x_2 \overline{x}_3 + x_1\overline{x}_2\overline{x}_3 + x_1\overline{x}_2 x_3 + x_1 x_2 \overline{x}_3$ is $[1, 0, 1, 0, 1, 1, 1, 0]$. ♦

Definition 2. The linear combination of two Boolean functions $f,\ g : \{0,1\}^n \rightarrow \{0,1\}$ is defined as $(f \oplus g)(x) = f(x) \oplus g(x)$, where $\oplus$ denotes addition modulo 2.

Definition 3. The Hamming weight $w(c)$ of a vector c is equal to the number of non-zero components in the vector.

Definition 4. The Hamming distance $d(a, b)$ between two binary sequences a and b of length n is the number of the places in which they differ.

Example 2. Let f and g be two given functions on V_n. The Hamming distance $d(f, g)$ between them is defined as the Hamming weight of $f(x) \oplus g(x)$ and $d(f, g) = w(f(x) \oplus g(x))$. ♦

Definition 5. A map $f : v \rightarrow GF(2)$ is called the *bent function* if for any affine function $l : v \rightarrow GF(2)$ we have $w(f \oplus l) = (2^n \pm 2^{n/2})/2$.

Lemma 1. *Let* $[b_0, b_1, \ldots, b_{2^n-1}]$, $b_i \in \{0,1\}$ *be the truth vector of a bent function and* $[c_0, c_1, \ldots, c_{2^n-1}]$, $c_i \in \{0,1\}$ *be the truth vector of a linear function. Then the vector* $[b_0 c_0, b_1 c_1, \ldots, b_{2^n-1} c_{2^n-1}]$ *also represents the truth vector of a bent function.*

The proof of a similar lemma can be found in (Adams and Tavares, 1990).

Definition 6. Let $\mathbf{Y} = [y_0, y_1, \ldots, y_{2^n-1}]$ be the truth vector of a given Boolean function f in the $\{0,1\}$ domain. We call $[(-1)^{y_0}, (-1)^{y_1}, \ldots, (-1)^{y_{2^n-1}}]$ the truth vector of a given function f in the $\{1,-1\}$ domain. In other words, we obtain a mapping $v : \{0,1\} \rightarrow \{1,-1\}$. Such a representation will be called the sequence of the function f.

3. Spectral Analysis

Spectral data are used in many applications in digital logic design. Some of them offer a possibility of function classification (Hurst *et al.*, 1985; Porwik, 2002), fault synthesis, signal processing (Porwik and Falkowski, 1999; Karpovsky, 1976; Sasao, 1993) and others. A Boolean function $f(x_1, x_2, \ldots, x_n)$ can be transformed from the domain $\{0,1\}$ into the spectral domain by the linear transformation $\mathbf{H} \cdot \mathbf{Y} = \mathbf{R}$, where $\mathbf{H}$ is a $2^n \times 2^n$ orthogonal transform matrix, $\mathbf{Y} = [y_0, y_2, \ldots, y_{2^n-1}]^T$ is the two-valued truth vector of $f(x_1, x_2, \ldots, x_n)$, and $\mathbf{R} = [r_0, r_1, \ldots, r_{2^n-1}]^T$ is the vector of spectral coefficients. One of several ways to interpret the meaning of each spectral coefficient is to view it as a measure of correlation between two functions (vectors) (Hurst *et al.*, 1985; Porwik, 2000a; Porwik, 2002). Hence the first function f is a Boolean function represented by the two-valued truth vector $\mathbf{Y}$ and the second function is one from the collection of the constituent functions of the transformation matrix $\mathbf{H}$. The type of the information that is obtained from spectral coefficients depends on the transformation matrix. In this paper, the well-known Hadamard matrices are used as transform matrices. In (Harmuth, 1977) it was observed that for some N, where $n = \log_2 N$, the Hadamard matrices include the discrete Walsh functions.

Definition 7. The Sylvester-Hadamard (the Walsh-Hadamard) matrix of order 2^n is generated by the following recursive formulae:

$$\mathbf{H}_0 = [1],\ \mathbf{H}_n = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \otimes \mathbf{H}_{n-1},\ n = 1, 2, \ldots \quad (1)$$

where $\otimes$ denotes the Kronecker product.

The square matrix (1) can be alternatively generated on the basis of the formulae:

$$\mathbf{H}_0 = [1], \quad \mathbf{H}_n = \begin{bmatrix} \mathbf{H}_{n-1} & \mathbf{H}_{n-1} \\ \mathbf{H}_{n-1} & -\mathbf{H}_{n-1} \end{bmatrix}. \quad (2)$$

Additionally we have $\mathbf{H}_n = \mathbf{H}_n^T$ and $\mathbf{H}_n \cdot \mathbf{H}_n^T = 2^n \cdot I_n$, where I_n is the identity matrix of order 2^n. Because $\mathbf{H}_n^{-1} = \frac{1}{2^n}\mathbf{H}_n^T$, the matrix $\mathbf{H}_n$ is orthogonal. The spectral coefficients calculated on the basis of the matrix (1) are the so-called Walsh coefficients. This transformation is known as the Walsh-Hadamard Transform (WHT).

Each row of the matrix $\mathbf{H}_n$ created in this way includes a discrete Walsh sequence $wal(w, t)$ (in other words, a discrete Walsh function). In this notation, $w = 1, \ldots, 2^n$ identifies the index of the Walsh function, and $t = 1, \ldots, 2^n$ stands for a discrete point of the function determination interval. The relationship between the Walsh coefficients and the variables of a Boolean function f can be described as follows:

Definition 8. Any Boolean function $f(x_1, x_2, \ldots, x_n)$ of n variables can be expressed by means of the Walsh-Hadamard coefficients as an arithmetical polynomial:

$$\begin{aligned} f(x_1, x_2, \ldots, x_n) = \frac{1}{2^n}\big[& r_0 + r_1 \cdot (-1)^{x_n} \\ & + r_2 \cdot (-1)^{x_{n-1}} \\ & + r_3 \cdot (-1)^{x_n \oplus x_{n-1}} \\ & + \cdots + r_{2^n-1} \cdot (-1)^{x_n \oplus x_{n-1} \cdots \oplus x_1} \big], \end{aligned}$$

where $\oplus$ stands for the modulo-2 addition, and $r_0, r_1, \ldots, r_{2^n-1} \in \mathbf{R}$ are spectral coefficients.

Each spectral coefficient $r_i \in \mathbf{R}$ is described by its order. The order is equal to the number of variables describing the linear function, which corresponds to a row in the matrix $\mathbf{H}_n$ for a given spectral coefficient. The r_i elements of the vector $\mathbf{R}$ are ordered according to a straight binary code of literals describing the minterms of the original truth vector $\mathbf{Y}$:

r_0 $C_n^0 = 1$ – zeroth-order coefficient,
r_i $C_n^1 = n$ – first-order coefficients, $i = 1, \ldots, n$,
r_{ij} C_n^2 – second-order coefficients, $ij = 12, 13, 1n, \ldots, (n-1)n$,
r_{ijk} C_n^3 – third-order coefficients, $ijk = 123, 124, \ldots, (n-2)(n-1)n$,
$\vdots$ $\vdots$
$r_{12\ldots n}$ $C_n^n = 1$ – the coefficient of order n.

In this notation r_{1234} is a spectral coefficient which has been calculated for a given Boolean function at point $x_1 = x_2 = x_3 = x_4 = 1$.

Property 1. *Let $\varphi(x)$ be a Boolean function and let its spectrum have the form $\mathbf{R} = [r_0, r_1, \ldots, r_{2^n-1}]$. Let $\overline{\varphi}(x) = 1 - \varphi(x)$ be the negation of $\varphi(x)$ and let its spectrum have the form $\overline{\mathbf{R}} = [\overline{r}_0, \overline{r}_1, \ldots, \overline{r}_l]$. Then $\overline{r}_0 = 2^n - r_0$ and $\overline{r}_i = -r_i$ for $i = 1, 2, \ldots, 2^n - 1$.*

4. Spectral Description of the Linearity of a Boolean Function

Definition 9. The Boolean function $f_k(x_1, x_2, \ldots, x_n)$ of n variables is called affine if it takes the form of a polynomial $f_k(x) = a_1 x_1 \oplus a_2 x_2 \oplus \cdots \oplus a_n x_n \oplus c$, where $a_j, c \in GF(2)$ and $k = c + \sum_{i=1}^{n} a_i 2^i$.

In particular, if $c = 0$ then f is called a linear function.

Some authors (Sasao , 1995) (very often in a broader sense than here) have additionally classified these functions with respect to the c coefficient. In this paper, regardless of the c value, all Boolean functions will be called linear. In linear Boolean functions each coefficient a_i corresponds to a unique ordering x_i. Hence the ordering set of all a_i corresponds to a unique ordering of a Boolean function.

Corollary 1. (Porwik, 2000b) *By the definition of the Walsh functions, for any affine Boolean function f_k we have the following:*
for $c = 0$:

$$\mathbf{Y}_k = f_k(x) = \frac{1}{2}\big(\mathbf{1} - wal(k,t)\big),$$

for $c = 1$:

$$\mathbf{Y}_k = f_k(x) = \frac{1}{2}\big(\mathbf{1} - ((-1) \cdot wal(k,t))\big).$$

From Corollary 1 it follows that any linear Boolean function can be generated immediately from Hadamard matrices (Porwik, 2000a; Porwik, 2000b):

$$\begin{aligned} &\text{for } c = 0 \text{ from } \mathbf{H}_n, \\ &\text{for } c = 1 \text{ from } \overline{\mathbf{H}}_n = -1 \cdot \mathbf{H}_n. \end{aligned} \tag{3}$$

The space V_n generates 2^{2^n} different Boolean functions and it includes 2^{n+1} affine functions (Porwik, 2000b). By means of the Walsh-Hadamard transform we can find only 2^n linear functions. Theorem 1 allows us to find all affine Boolean functions in V_n.

Let $\mathbf{R} = [r_0, r_1, \ldots, r_{2^n-1}]$ be a vector of spectral coefficients and let $\mathbf{R} = [0, 0, \ldots, 0] \Leftrightarrow f(x) = \mathbf{0}$ and $\mathbf{R} = [2^n, 0, \ldots, 0] \Leftrightarrow f(x) = \mathbf{1}$ be trivial Boolean functions.

Theorem 1. *Any affine Boolean function f (except for the two above-mentioned trivial functions) is characterized by the unique Walsh-Hadamard spectrum distribution*

$$r_x = \begin{cases} +2^{n-1} & \text{for } x = 0, \\ -2^{n-1} & \text{for } x = k/2 \quad \Leftrightarrow c = 0, \\ +2^{n-1} & \text{for } x = (k-1)/2 \ \Leftrightarrow c = 1, \\ 0 & \text{otherwise,} \end{cases} \tag{4}$$

where $k = c + \sum_{i=1}^{n} a_i 2^i$, $a_j, c \in GF(2)$ have the same meaning as in Definition 9 and $x = 0, 1, \ldots, 2^n - 1$.

Proof. Directly from the definition of Walsh functions it is known that they form a complete orthogonal system. From the mutual orthogonality the rows of the Hadamard matrix satisfy

$$\sum_{t=0}^{2^n-1} wal(i,t) \cdot wal(j,t) = \begin{cases} 2^n & \text{for } i = j, \\ 0 & \text{for } i \neq j. \end{cases} \tag{5}$$

For any Walsh function we have (Hurst *et al.*, 1985)

$$\sum_{t=0}^{2^n-1} wal(i,t) = \begin{cases} 2^n & \text{for } i = 0, \\ 0 & \text{for } i \neq 0. \end{cases} \tag{6}$$

Using (5), (6) and Corollary 1, we obtain (4). ■

Hence, in the proposed method, a linear Boolean function can be defined by means of Walsh functions (Corollary 1) or by means of spectral coefficients $r_\omega \in \mathbf{R}$ (Theorem 1). Thus, in order to decide whether or not a Boolean function is linear, it is only necessary to calculate its spectrum. If the spectrum contains only two non-zero values, then the function is affine and it has the polynomial form (cf. Definition 8).

Property 2. *A Boolean function of n variables is affine if and only if $r_0 = 2^{n-1}$ and the value of the n-th order spectral coefficient is $\pm 2^{n-1}$.*

Example 3. Table 1 includes the description of the given Boolean functions f_1 and f_2. It is necessary to check whether these functions are linear. From the analysis of spectral coefficients it follows that the spectrum includes only two non-zero coefficients: r_0 and r_7. ♦

Table 1. Boolean functions and their spectrum.

$x_1x_2x_3$	$x = \sum_{i=1}^{n} x_i 2^{n-i}$	$f_1(x)$	r_x^1	$f_2(x)$	r_x^2
000	0	1	4	0	4
001	1	0	0	1	0
010	2	0	0	1	0
011	3	1	0	0	0
100	4	0	0	1	0
101	5	1	0	0	0
110	6	1	0	0	0
111	7	0	4	1	-4

Hence, according to Theorem 1, the functions $f_1$and f_2 are affine.

From Table 1 it follows that $f_1(x) = \overline{f}_2(x)$, and those functions can be described by the Boolean formulae $f_1(x_1, x_2, x_3) = 1 \oplus x_1 \oplus x_2 \oplus x_3$ and $f_2(x_1, x_2, x_3) = x_1 \oplus x_2 \oplus x_3$. The obtained results are consistent with Properties 1 and 2. ♦

Unfortunately, the above matrix-based method is impractical for large n, but, as has been shown, Boolean functions have particular properties which allow us to modify them.

Proposition 1. *Let $\mathbf{Y} = [y_0, y_1, \ldots, y_{2^n-1}]$ be the two-valued truth vector of a function $f(x_1, x_2, \ldots, x_n)$. From the properties of Hadamard matrices it follows that all the Walsh-Hadamard spectral coefficients of a Boolean function can be calculated recursively from the equation*

$$\mathbf{H}_n \times [y_0, y_1, \ldots, y_{2^n-1}] = \mathbf{H}_n [y_0, y_1, \ldots, y_{2^n-1}]^T = \begin{bmatrix} A+B \\ A-B \end{bmatrix}, \quad (7)$$

where $A = \mathbf{H}_{n-1}\ [y_0, y_1, \ldots, y_{2^{n-1}-1}]^T$ and $B = \mathbf{H}_{n-1}\ [y_{2^{n-1}}, y_{2^n}, \ldots, y_{2^n-1}]^T$.

Formula (7) can be used to efficiently calculate the Wash-Hadamard spectrum, because instead of inconvenient large matrices $\mathbf{H}_n$ some much better small matrices can be used. The described formula can be easily implemented in parallel computations as well. The parallel algorithms significantly accelerate the time of computations. In these cases the matrices $\mathbf{H}_i$ can be first determined by a look-up table. Additionally, by means of (7), it is easy to check whether or not a Boolean function is linear. In these instances each part of the spectrum calculated by means of (7) must fulfil the conditions of Theorem 1. Additionally, for those functions we have $r_i^A = |r_i^B|$, where r^A and r^B denote spectral coefficients of part A and B, respectively.

Example 4. Let $\mathbf{Y} = [01101001]^T$ be the truth vector of a given Boolean function. Split $\mathbf{Y}$ into four parts. Then

$$\mathbf{H}_1 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

and on the basis of (7) we have

$$\begin{array}{l} \mathbf{H}_1[01]^T = [1\ -1]^T \\ \mathbf{H}_1[10]^T = [1\quad 1]^T \\ \mathbf{H}_1[10]^T = [1\quad 1]^T \\ \mathbf{H}_1[01]^T = [1\ -1]^T \end{array} = \begin{array}{c} \nearrow \\ \\ \searrow \\ \\ \end{array} \begin{array}{l} A = \begin{bmatrix} +2 \\ 0 \\ 0 \\ -2 \end{bmatrix} \\ \\ B = \begin{bmatrix} +2 \\ 0 \\ 0 \\ +2 \end{bmatrix} \end{array} \begin{array}{c} \searrow \\ \\ \nearrow \\ \\ \end{array} = \begin{bmatrix} +4 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -4 \end{bmatrix}.$$

According to Theorem 1, the analysed function is linear. ♦

The main limiting factor while using spectral methods in the processing of switching functions is their calculation complexity in spite of the existence of fast FFT-like algorithms. For example, the total number of arithmetic operations required to determine all Walsh-Hadamard coefficients is $n2^n$ for Boolean functions of n variables. Therefore the complexity is $O(n2^n)$. Such complexity is attainable when fast transforms are applied, where only addition and subtraction are used as arithmetic operations. It is known that FFT-like algorithms are executed in n steps. In each step 2^n arithmetic operations are realized. Additionally, in order to store the calculated spectrum, 2^n memory locations are required (Ahmed and Rao, 1975; Clarke *et al.*, 1993; Harmuth, 1977; Karpovsky, 1976). In the presented method, addition and subtraction operations are also applied. From (7) it follows that the complexity of the new method is the same as in the FFT algorithm. If it is necessary to check whether a Boolean function is linear, then the complexity of such calculations is only $O(2^{n+1})$, because only the first two steps of the algorithm are needed. It is so because after the second step, parts of spectra are known and each of them must describe the spectrum of a linear function. Note that the new method of calculating spectra,

even though it has the complexity of FFT-like algorithms, is very efficient. Unlike other methods, it is not necessary to generate $\mathbf{H}_n$ matrices. The formation of matrices $\mathbf{H}_n$ is very time-consuming, especially for large n. Instead of this, additions and subtractions are solely applied.

Hence, for testing the linearity of Boolean functions it is necessary to reserve only 4 memory cells (instead of 2^n) for any n. Each such cell stores one spectral coefficient. Four spectral coefficients determine the so-called subset spectrum. Hence, according to Theorem 1, on the basis of a subset spectrum it can be checked whether a Boolean function is linear. In that case, after the first partition of the spectrum, the continuation of calculations is needless. In this way, the method exploits the property that the calculation of a subset of k Walsh-Hadamard coefficients may be interpreted as a windowing operation over the Hadamard matrix with a $k \times 2^n$ window in the multiplication of the Hadamard matrix and the truth vector of the function f.

Linearity and nonlinearity play important roles in cryptography, transmission of information, correction errors, etc. The main component of a stream cipher is a generator which produces a sequence of pseudo-random bits from a random seed. These random bits are added modulo 2 to bits in a plain text and, consequently, a ciphertext is sent to a receiver. The security of a block cipher depends on the properties of the so-called s-boxes. An $n \times m$ s-box is a mapping $B : \{0,1\}^n \to \{0,1\}^m$. B can be represented as $B(x) = [f_{m-1}(x), f_{m-2}(x), \ldots, f_0(x)]$, where f_i are fixed Boolean functions $f_i : (0,1)^n \to (0,1)$ for any i. The functions f_i are the columns of the s-box. Finally, B can be represented by a $2^n \times m$ binary matrix with the entry (i,j) being bit j of row i. In these boxes bent functions are applied because an important property of bent functions is that they have the highest possible nonlinearity (Mister and Adams, 1996; Seberry and Zhang, 1994).

Theorem 2. *Any bent function $f(x_1, \ldots, x_n)$ has the Walsh-Hadamard spectrum*

$$r_x = \begin{cases} (2^n - 2^{n/2})/2 \ \text{ or } \ (2^n + 2^{n/2})/2 & \text{for } x = 0, \\ \pm(2^{n/2})/2 & \text{for } x \neq 0 \end{cases} \tag{8}$$

for $x = 0, 1, \ldots, 2^n - 1$.

Proof. In case $x = 0$, the value of the coefficient r_0 follows immediately from Definition 5. Additionally, from the properties of Walsh functions it follows that $wal(0,t) = 1$ for any $t = 0, 1, \ldots, 2^n - 1$. Thus the value of r_0 is the number of cases when $f(x_1, \ldots, x_n) = 1$. In accordance with Definition 5, two such cases may occur for the function f and for the function $\overline{f}$. Hence there are two different values for the coefficient r_0. If for f the number of cases when $f(x_1, \ldots, x_n) = 1$ is equal to $a = (2^n + 2^{n/2})/2$, then for $\overline{f}$ this number is equal to $2^n - a = (2^n - 2^{n/2})/2$. Searching values for $r_{x \neq 0}$ can be considered similarly for the functions f and $\overline{f}$. Only one case can be shown for the function f. The proof for the second case is identical. The Hadamard matrix includes discrete Walsh functions. As has been shown in the paper, the Walsh basis is a generator of linear Boolean functions. On the other hand, we know (Porwik, 2000a) that spectral coefficients r_x can be calculated by means of the formula $r_x = 2^{n-1} - d(f, wal(x,t))$. Finally, from the fact that $d(f, wal(x,t) = w(f \oplus wal(x,t))$ and taking into account Definition 5, it follows that $r_{x \neq 0} = 2^{n-1} - (2^n + 2^{n/2})/2 = -(2^{n/2})/2$. ■

Theorem 2 and Definition 5 imply that for fixed n two bent functions can be extracted. When the bent function f is known, the second function can be found by the formula $\overline{f} = 1 \oplus f$. This operation is very simple owing to Property 1.

It can be observed that the function f on V_n attains the upper bound nonlinearities if and only if it is bent (Seberry and Zhang, 1994).

Example 5. Table 2 includes the description of two bent functions and presents the spectrum of each of them. ♦

Table 2. Boolean functions and their spectrum.

$x_1x_2x_3x_4$	$x = \sum_{i=1}^{n} x_i 2^{n-i}$	$f_1(x)$	r_x^1	$f_2(x)$	r_x^2
0000	0	0	6	1	10
0001	1	0	−2	1	2
0010	2	0	−2	1	2
0011	3	0	−2	1	2
0100	4	0	−2	1	2
0101	5	1	2	0	−2
0110	6	0	−2	1	2
0111	7	1	2	0	−2
1000	8	0	−2	1	2
1001	9	0	−2	1	2
1010	10	1	2	0	−2
1011	11	1	2	0	−2
1100	12	0	−2	1	2
1101	13	1	2	0	−2
1110	14	1	2	0	−2
1111	15	0	−2	1	2

Measuring nonlinearity, it is generally necessary to record the Boolean function result for each possible combination of the input variables. Unfortunately, the measuring of large functions rapidly becomes impossible. So,

we can measure nonlinearity in substitutional tables and small block constructions.

5. Nonlinearity (Linearity) Spectral Measure

Nonlinearity is a crucial criterion for cryptographic functions. That measure can be treated as a security system factor. If a system is described by linear equations, then it will be easily breakable by various attacks. The nonlinearity measure of a Boolean function $f : \{0,1\}^n \rightarrow \{0,1\}$ is defined as $N(f) = \min\{(d(f,\varphi_i),\ i = 1,2,\ldots,2^{n+1}\} = \min\{w(f \oplus \varphi_i),\ i = 1,2,\ldots,2^{n+1}\}$, where $\phi = \{\varphi_1, \varphi_2, \ldots, \varphi_{2^{n+1}}\}$ is a set of affine functions on V_n. The formula for nonlinearity thus described is very inconvenient in practice, because in order to calculate the Hamming distance between a given f and affine functions from the space V_n, we have to execute 2^{2n+1} operations of comparisons. On the basis of the above, this important problem can be defined as a more convenient one in the spectral domain. In that case $n2^n$ operations can be performed in the proposed spectral method (Porwik, 2000a).

In reference to the set of affine functions from ϕ, the measure of linearity or nonlinearity for any Boolean function can be calculated on the basis of the set of linear functions obtained from the Hadamard matrices $\mathbf{H}_n$ or $\overline{\mathbf{H}}_n$, respectively. Higher numerical values of spectral coefficients indicate the greater linearity of a function. By finding the largest value we can find the closest linear function.

It is obvious that using the theory described in the paper, we can immediately construct two spectral measures: the lowest nonlinearity $N_L(f)$ and the greatest nonlinearity $N_G(f)$ of a given function f with reference to the linear Boolean functions defined by matrices $\mathbf{H}_n$ or $\overline{\mathbf{H}}_n$, respectively. These measures can be defined formally by means of spectral coefficients.

Definition 10. The nonlinearity of a Boolean function f can be determined using

$$N_L(f) = 2^{n-1} - \frac{1}{2}\Big\{ \max \mathbf{S} \in \{2^n - 2r_0, -2r_1, \ldots, -2r_{2^n-1}\}\Big\},$$

$$N_G(f) = 2^{n-1} - \frac{1}{2}\Big\{ \min \mathbf{S} \in \{2r_0 - 2^n, +2r_1, \ldots, +2r_{2^n-1}\}\Big\},$$

where $\mathbf{R} = [r_0, r_1, \ldots, r_{2^n-1}]$ is the vector of the spectral coefficients of the function f.

If we calculate the coefficients by means of the Walsh-Hadamard transform, we simultaneously get the coefficients for all affine functions from the set ϕ .

Example 6. Let $\mathbf{Y} = [0,1,1,0,0,0,0,1]$ be the truth vector of the function $f(x_1, x_2, x_3)$. For matrices $\mathbf{H}_n$ and $\overline{\mathbf{H}}_n$ we obtain the spectra $\mathbf{R}_{H_n} = [3,-1,-1,-1,1,1,1,-3]$ and $\mathbf{R}_{\overline{H}_n} = -\mathbf{R}_{H_n} = [-3,1,1,1,-1,-1,-1,3]$, respectively.

Table 3 shows all linear functions $f_0, \ldots, f_{15}$ which were generated immediately on the basis of Definition 8. The spectra of the functions f_0, f_2, f_4,..., f_{14} were calculated using $\mathbf{H}_n$. The spectra of the functions f_1, f_3, f_5,..., f_{15} can be calculated by means of $\overline{\mathbf{H}}_n$, but these coefficients were computed more effectively using the spectrum for the first of the eight functions.

Hence, in accordance with Definition 10, we can easily characterize (by one spectrum calculation) both the nonlinearity measures $N_L(f) = 1$ and $N_G(f) = 7$. From Table 3 is follows that the same results can be obtained using the Hamming distance.

Table 3. Spectral coefficients and the Hamming distance of Boolean function f in reference to functions from set ϕ.

	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7
Spectrum $\mathbf{R}$	$+3$	-3	-1	$+1$	-1	$+1$	-1	$+1$
Spectrum $\mathbf{S}$	$+2$	-2	$+2$	-2	$+2$	-2	$+2$	-2
$d(\mathbf{Y}, f_k)$	3	5	3	5	3	5	3	5

f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}
$+1$	-1	$+1$	-1	$+1$	-1	-3	$+3$
-2	$+2$	-2	$+2$	-2	$+2$	$+6$	-6
5	3	5	3	5	3	1	7

Note. Index k for functions f_k has the same meaning as in Theorem 1. The functions f_k are arranged according to Definition 9.

From Table 3 we can conclude that the closest linear function to f is $f_{14} = x_1 \oplus x_2 \oplus x_3$. The most distant function from f is the affine function $f_{15} = 1 \oplus x_1 \oplus x_2 \oplus x_3$.

Similar results are obtained for the Boolean functions described in Example 5. As is said above, both the functions $f_1$and f_2 are bent. For these functions we can compute measures $N_L(f_1) = N_L(f_2) = 6$ and $N_G(f_1) = N_G(f_2) = 10$. ♦

The bent functions can be generated using several methods (Adams and Tavares, 1990; Mister and Adams, 1996; Seberry and Zhang, 1994). In this paper, on the

basis of the proposed definitions and theorems, a new method of generating such functions has been presented but generally, problems of building bent functions are not described.

Theorem 3. (Seberry and Zhang, 1994) *Let τ_n denote the number of bent functions which can be represented by truth vectors of length 2^n. Then $\tau_n \geq (2^n-2)\tau_{n-2}^2$.*

Since $\tau_2 = 8$, directly from Theorem 3 we can obtain $\tau_4 = (2^4-2)8^2 = 896$, $\tau_6 = (2^6-2)\tau_4 = 62 \cdot 896^2 = 49,774,592$ different bent functions.

In (Seberry and Zhang, 1994) it was shown that using two known bent functions which have truth vectors of length 2^{2k-2} one can construct 2^k bent functions which have truth vectors of length 2^{2k}, $k = 1, 2, \ldots$. The bent functions described by the authors were generated on the basis of the so-called bent matrices proposed and non-degenerated linear transformations. Unfortunately, matrices and transformations mentioned above must be found first. That task can be solved more easily for any number of pairs of bent functions.

Proposition 2. *Let B_{n-2} with n even be the set of bent functions $f : \{0,1\}^{n-2} \to \{0,1\}$ and f_a, $f_b \in B_{n-2}$. Then the function f_c defined by the formula*

$$f_c(x_1, x_2, \ldots, x_n) = \begin{cases} f_a(x_1, x_2, \ldots, x_{n-2}) & \text{if } x_{n-1}=0, \ x_n=0, \\ f_a(x_1, x_2, \ldots, x_{n-2}) & \text{if } x_{n-1}=0, \ x_n=1, \\ f_b(x_1, x_2, \ldots, x_{n-2}) & \text{if } x_{n-1}=1, \ x_n=0, \\ f_b(x_1, x_2, \ldots, x_{n-2}) \oplus \mathbf{1} & \text{if } x_{n-1}=1, \ x_n=1 \end{cases}$$

is bent.

Example 7. Let $\mathbf{Y}_a = [0,1,1,1]$ and $\mathbf{Y}_b = [0,1,0,0]$ be the truth vectors of the bent functions f_a and f_b, respectively. According to Proposition 2, we have obtained the new vector $\mathbf{Y}_{c=}[0,0,0,1,1,1,1,0,0,1,1,0,1,1,1,0,0,1]$ of the function f_c. The function f_c has the spectrum $\mathbf{R}_{f_c} = [10,-2,2,2,-2,-2,-2,2,-2,2,-2,-2,-2,-2,-2,2]$, and thus on the basis of Theorem 2 we conclude that the function f_c is bent. ♦

On the other hand, from Definition 1 and Corollary 1 it follows directly that bent functions can be generated also differently.

Proposition 3. *Let v_n^i ($\overline{v}_n^i$) be the i-th row of $\mathbf{H}_n$ (resp. $\overline{\mathbf{H}}_n$). Let a be any bent sequence of length 2^n, and $n = 2, 4, 6, \ldots$. Then simple vector multiplication $v_n^i \times a$ (resp. $\overline{v}_n^i \times a$) generates a bent sequence.*

Proof. The proof results immediately from Lemma 1 and Corollary 1. ■

Let $v_n^1, v_n^2, \ldots, v_n^n$ be a concatenation of the rows of $\mathbf{H}_n$. Then the new vector $[v_n^1, v_n^2, \ldots, v_n^n]$ is a bent sequence. This construction was proposed in (Adams and Tavares, 1990), but the authors did not notice that any permutation of rows v_n^j also gives a bent sequence. Hence immediately from $\mathbf{H}_n$ (resp. $\overline{\mathbf{H}}_n$) we obtain $(2^n)!$ new, different bent sequences of length $2^n \times 2^n = 4^n$. By taking into account Proposition 3, each from $(2^n)!$ sequences can be multiplied by rows of $\mathbf{H}_{2^n}$ (resp. $\overline{\mathbf{H}}_{2^n}$). This solution allows us to obtain $(2^n)! \times 2^{(2n)}$ bent sequences from each matrix.

6. Experimental Results

All experiments were performed by means of a PC running Linux. The computer was equipped with AMD Duron (Morgan) 1.2 GHz CPU and 128 MB main memory. All times are given in CPU microseconds.

In the first experiment for different methods of the spectrum calculation the time of computations was determined. As is known (Ahmed and Rao, 1975; Karpovsky, 1976; Porwik, 2002), Walsh-Hadamard spectral coefficients can be calculated on the basis of the recursive formula (2), by means of (7) or by a non-recursive method (Ahmed and Rao, 1975). In Table 4 the time of Walsh-Hadamard spectra calculations is presented.

As can be easily seen, the proposed method is more efficient because it gives significantly better results. Our technique also allows us to compute coefficients when recursive and nonrecursive approaches fail. This situation occurs for Boolean functions with large n (> 10) when the capacity of the RAM memory is insufficient. The presented method consumes substantially less memory than other methods.

In the second experiment, both the $N_L(f)$ and $N_G(f)$ measures for a function of $n = 3$ variables were calculated. The obtained results are presented in Figs. 1 and 2. In that experiment eight functions with the nonlinearity measure $N_L(f) = 0$ and eight functions with the nonlinearity measure $N_G(f) = 8$ were found. Both groups are affine. The first group is of type $x_1 \oplus \ldots$. The second group is of type $1 \oplus x_1 \oplus \ldots$.

7. Conclusions

Nowadays, many systems of automatic design are oriented towards detection of the linear part of a Boolean function. The proposed spectral method of investigation allows us to obtain fast information about the linearity of the analyzed function. The proposed method can be easily implemented and has low complexity. The basic concept of the spectral identification of linear Boolean functions was also explained.

Table 4. Experimental results: Runtime [μs]

n	recursive	non-recursive	our method
1	0.0612	0.0571	0.0218
2	0.2339	0.3783	0.0293
3	0.8136	3.8663	0.0653
4	3.7490	15.6660	0.1852
5	122.7700	73.4100	0.5527
6	1417.0600	306.2400	1.2778
7	6432.8000	1304.8000	2.8400
8	26697.0000	5326.0000	6.4140
9	142452.0000	22662.0000	14.2640
10	604650.0000	92300.0000	31.6900
11			69.1800
12			151.8800
13			330.1667
14			769.8333
15			3044.5000
16			13584.0000
17			33404.0000
18			67533.3333
19			145900.0000
20			310950.0000
21			693900.0000
22			1568800.0000
23			3313200.0000
24			6978181.8182
25			14298333.3333

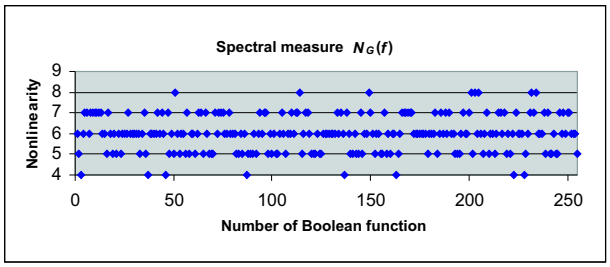

Fig. 1. Spectral measure for $N_L(f)$.

In the paper, the relationship between Hadamard matrices and linear Boolean functions and bent functions has been discussed. A simple method to determine the linearity of Boolean functions directly from their spectra was also shown. It was demonstrated how to generate bent functions using only Hadamard matrices. It was shown how to quantify the linearity and nonlinearity of Boolean functions by using Walsh-Hadamard spectral coefficients

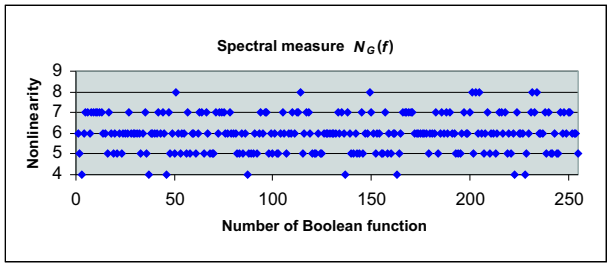

Fig. 2. Spectral measure for $N_G(f)$.

in complement and non-complement forms a basic set of linear functions. The traditional analysis methods (e.g. the Hamming distance calculation) are not effective in applications, because we must find the first set of all Boolean functions for given n.

Many applications of transforms like the Walsh-Hadamard transform were previously impossible to use because of memory constraints. Now it is possible. The new theorems, propositions and equations also show a new way which allows to find efficiently the spectral coefficients for Boolean functions and to find the bent functions.

Using a straightforward implementation, the complexity of these transformations rapidly increases with a number of variables of Boolean functions. Therefore the measuring of large functions rapidly becomes impossible. As has been proven above, calculations of spectra can be easily performed by means of the sum of some spectral sub-vectors. When it is necessary to check whether a Boolean function is linear, the test of linearity can be carried out on the basis of one part of the spectra. In these cases the computation complexity is $O(2^{n+1})$. Immediately from the spectra of a Boolean function f, the closest and the most distant linear function to f can be found.

References

Adams C.M. and Tavares S.E. (1990): *Generating and counting binary bent sequences*. — IEEE Trans. Inf. Th., Vol. IT–36, No. 5, pp. 1170–1173.

Ahmed N. and Rao K.R. (1975): *Orthogonal Transforms for Digital Signal Processing*. — Berlin: Springer.

Blahut R.E. (1983): *Theory and Practice in Error Control Codes*. — London: Addison-Wesley.

Clarke E.M., McMillan K.L., Zhao X. and Fujita M. (1993): *Spectral transformation for extremely large Boolean functions*. — Proc. IFIP WG 10.5 Workshop *Applications of the Reed-Muller Expansion in Circuit Design*, Hamburg, Germany, pp. 86–90.

Falkowski B.J. and Kannurao S. (2000): *Spectral theory of disjunctive decomposition for balanced functions*. — Proc. 13th Int. Conf. *VLSI Design*, Calcutta, India, pp. 100–105.

Harmuth H.F. (1977): *Sequency Theory. Foundations and Applications*. — New York: Academic Press.

Hurst S.L., Miller D.M. and Muzio J.C. (1985): *Spectral Techniques in Digital Logic*. — London: Academic Press.

Karpovsky M.G. (1976): *Finite Orthogonal Series in the Design of Design of Digital Devices*. — New York: Wiley.

Mister S. and Adams C. (1996): *Practical S-box design*. — Workshop *Selected Areas in Cryptography, SAC'96*, Queen's University Kingston, Ontario, Canada, pp. 61–76.

Porwik P. (2000a): *Towards calculation of Boolean functions nonlinearity using Walsh transform*. — Arch. Theoret. Appl. Comp. Sci. Polish Acad. Sci., Fasc. No. 1, Vol. 12, pp. 51–64.

Porwik P. (2000b): *Spectral modelling of digital systems with specified features*. — Sci. Works of the Silesian University No. 1898, Katowice (in Polish).

Porwik P. (2002): *Efficient calculation of the Reed-Muller forms by means of the Walsh spectrum*. — Int. J. Appl. Math. Comp. Sci., Vol. 12, No. 4, pp. 571–579.

Porwik P. and Falkowski B.J. (1999): *Informatics properties of the Walsh transform*. — Proc. 2nd Int. Conf. *Information Communications and Signal Processing, ICISC'99*, Singapore, paper 2B2.4, pp .1–5.

Sasao T. (1993): *Logic Synthesis and Optimization*. — Dordrecht: Kluwer.

Sasao T. (1995): *Representation of logic functions using EXOR operators*. — Proc. Workshop *Applications of the Reed-Muller Expansion in Circuit Design*, Makuhari, Japan, pp. 308–313.

Seberry J. and Zhang X.M. (1994): *Construction of bent function from two known bent functions*. — Australasian J. Comb., Vol. 9, pp. 21–34.

Received: 8 January 2003
Revised: 24 April 2003