Tensor Transform of Boolean Functions and Related Algebraic and Probabilistic Properties

Alexander Kholosha and Henk C.A. van Tilborg

Department of Mathematics and Computer Science
Technische Universiteit Eindhoven, P.O. Box 513,
5600 MB Eindhoven, The Netherlands
{A.Kholosha,H.C.A.v.Tilborg}@tue.nl

Abstract. We introduce a tensor transform for Boolean functions that covers the algebraic normal and Walsh transforms but which also allows for the definition of new, probabilistic and weight transforms, relating a function to its bias polynomial and to the weights of its subfunctions respectively. Our approach leads to easy proofs for some known results and to new properties of the aforecited transforms. Several new results about algebraic and correlation properties that depend on the weight transform of Boolean functions are proved. Finally, we present a new probabilistic characteristic of a Boolean function that is defined by its algebraic normal and probabilistic transforms over the reals.
Key words: cryptography, key-stream generator, Boolean function, tensor transform, Walsh transform, correlation, probabilistic properties.

1 Introduction

The two most common building blocks for key-stream generators are the nonlinear filter generator and the nonlinear combination generator [1]. They correspond respectively to a nonlinear transformation applied to several phases of the same linear feedback shift register (LFSR) or to the outputs of several independent LFSR's. The nonlinear transformation can be represented by a Boolean function and the security of the key-stream generators heavily relies on the specific qualities of this function. If the function is not chosen properly then the whole system is susceptible to different types of correlation [2] and linear [3] attacks.

It is currently generally accepted that secure Boolean function to be used in a key-stream generator must satisfy the following properties: balancedness, high nonlinearity, sufficiently high algebraic degree (this should hold for each individual variable), optimized with correlation properties. These conditions are necessary, although it is not clear if they are sufficient to resist all kinds of attacks. The algebraic degree of a Boolean function is the degree of its algebraic normal form (ANF), balancedness, nonlinearity, and correlation properties are defined by its Walsh transform [4]. Thus, the algebraic normal and Walsh transforms of a Boolean function define the most important cryptographic characteristics of the function. The objective of this paper is to generalize known transforms of

Boolean functions and develop new ones that would provide efficient means for analyzing security of these functions.

In Sect. 2 we describe the general basis for a tensor transform of Boolean functions. Special cases of this approach provide easy proofs for some known and new relations in the theory of algebraic normal and Walsh transforms. We also propose a new type of tensor transform, the probabilistic transform, giving an important insight in certain probabilistic properties of Boolean functions that are discussed in Sect. 4. Another new type of tensor transform that we propose, is the weight transform. It relates a Boolean function to the weights of its subfunctions. It is proved that coefficients of the ANF of a Boolean function depend on the values contained in its binary weight transform for the zero-valued vector.

In Sect. 3 we suggest that not correlation immune Boolean functions can be still cryptographically secure if only slight dependencies between input bits and the output are allowed. We show how correlation coefficients that provide an estimate for correlation dependencies of a Boolean function, can be obtained from its weight transform. It is proved that the number of fixed-order product terms in the ANF of a balanced Boolean function depends on its correlation coefficients. We also prove that highly resilient Boolean functions can not be approximated with a function nondegenerate on few variables.

A new probabilistic function of a Boolean function is introduced in Sect. 4. This function estimates the probabilistic distribution of bits at the output of a Boolean function if the distribution of the arguments, the function depends on, is known. Further, we suggest a characteristic for a balanced Boolean function that measures its ability to compensate a nonuniform distribution of the input. Resilient functions are proved to have good compensating qualities.

2 Tensor Transform of Boolean Functions

Let $M_n(P)$ denote the ring of n-dimensional square matrices over the field P. For a pair of matrices $A \in M_n(P)$ and $B \in M_m(P)$ let $A \otimes B$ denote the Kronecker product [5, p. 421] of these matrices and $A^{[k]}$ denote the kth Kronecker power of A. For any matrix $A \in M_{2^n}(P)$ by writing $A = (\boldsymbol{g}_0, \ldots, \boldsymbol{g}_{2^n-1})$ we mean that $\boldsymbol{g}_i$ $(i = 0, \ldots, 2^n - 1)$ is the ith column of A, entries in $\boldsymbol{g}_i$ are indexed lexicographically by the elements in $\{0, 1\}^n$, so

$$\boldsymbol{g}_i = \begin{pmatrix} g_i(0, \ldots, 0) \\ g_i(0, \ldots, 1) \\ \vdots \\ g_i(1, \ldots, 1) \end{pmatrix} .$$

Let α_i $(i = 0, \ldots, 2^n - 1)$ denote the n-bit binary expansion of i, so $\boldsymbol{g}_i = (g_i(\alpha_0), \ldots, g_i(\alpha_{2^n-1}))^T$, where the superscript T denotes transpose of a matrix.

Lemma 1. *Let* $A = (\boldsymbol{g}_0, \dots, \boldsymbol{g}_{2^n-1}) \in M_{2^n}(P)$ *and* $A' = (\boldsymbol{g}'_0, \dots, \boldsymbol{g}'_{2^{n-1}-1}) \in M_{2^{n-1}}(P)$*. Suppose that* $A = B \otimes A'$ *for some matrix* $B = \begin{pmatrix} b_{00} & b_{01} \\ b_{10} & b_{11} \end{pmatrix}$*. Then*

$$g_i(x_1, \dots, x_n) = \begin{cases} (b_{00}\overline{x_1} + b_{10}x_1)g'_{i'}(x_2, \dots, x_n), & \text{if} \quad \alpha_i = (0, \alpha_{i'}), \\ (b_{01}\overline{x_1} + b_{11}x_1)g'_{i'}(x_2, \dots, x_n), & \text{if} \quad \alpha_i = (1, \alpha_{i'}) \end{cases},$$

where $\alpha_{i'}$ *is the* $(n-1)$*-bit vector, binary expansion of* i'*.*

Proof. By the definition of the Kronecker product, $A = \begin{pmatrix} b_{00}A' & b_{01}A' \\ b_{10}A' & b_{11}A' \end{pmatrix}$. Thus,

$$g_i(0, x_2, \dots, x_n) = \begin{cases} b_{00}g'_{i'}(x_2, \dots, x_n), & \text{if} \quad \alpha_i = (0, \alpha_{i'}), \\ b_{01}g'_{i'}(x_2, \dots, x_n), & \text{if} \quad \alpha_i = (1, \alpha_{i'}) \end{cases} \quad \text{and}$$

$$g_i(1, x_2, \dots, x_n) = \begin{cases} b_{10}g'_{i'}(x_2, \dots, x_n), & \text{if} \quad \alpha_i = (0, \alpha_{i'}), \\ b_{11}g'_{i'}(x_2, \dots, x_n), & \text{if} \quad \alpha_i = (1, \alpha_{i'}) \end{cases}.$$

These equations combined together prove the claimed result. □

The following proposition easily follows from Lemma 1.

Proposition 1. *Let* $A = B_1 \otimes \ldots \otimes B_n$*, where* $B_j = \begin{pmatrix} b^{(j)}_{00} & b^{(j)}_{01} \\ b^{(j)}_{10} & b^{(j)}_{11} \end{pmatrix}$ *for* $j = 1, \dots, n$*, and* $A = (\boldsymbol{g}_0, \dots, \boldsymbol{g}_{2^n-1})$*. Then for any* $i \in \{0, \dots, 2^n - 1\}$

$$g_i(x_1, \dots, x_n) = \prod_{j=1}^{n} \left(\overline{\alpha_i^j} \left(b^{(j)}_{00}\overline{x_j} + b^{(j)}_{10}x_j \right) + \alpha_i^j \left(b^{(j)}_{01}\overline{x_j} + b^{(j)}_{11}x_j \right) \right),$$

where $\alpha_i = (\alpha_i^1, \dots, \alpha_i^n)$*.*

Let $A \in M_{2^n}(P)$ be an invertible matrix and $A = (\boldsymbol{g}_0, \dots, \boldsymbol{g}_{2^n-1})$. Further, let the function $f(x_1, \dots, x_n)$, mapping $\{0,1\}^n$ in P, be defined by its string of values $T^f = (f(\alpha_0), \dots, f(\alpha_{2^n-1}))^T \in P^{2^n}$ and let the function $F(x_1, \dots, x_n)$ be defined by the string $T^F = A^{-1}T^f = (F(\alpha_0), \dots, F(\alpha_{2^n-1}))^T \in P^{2^n}$. Vectors T^f and T^F are considered further as column-vectors. Then $T^f = AT^F$,

$$T^f = \sum_{i=0}^{2^n-1} \boldsymbol{g}_i F(\alpha_i) \quad \text{and} \quad f(x_1, \dots, x_n) = \sum_{i=0}^{2^n-1} g_i(x_1, \dots, x_n)F(\alpha_i) \qquad (1)$$

for any $(x_1, \dots, x_n) \in \{0,1\}^n$. Equations (1) represent the decomposition of function f in the basis vector set $(\boldsymbol{g}_0, \dots, \boldsymbol{g}_{2^n-1})$. Hereafter in this paper, by $f^{\beta_1,\dots,\beta_m}_{i_1,\dots,i_m}$ for any $1 \le i_1 < \ldots < i_m \le n$, we denote the subfunction of f obtained by fixing the variables $x_{i_1}, \dots, x_{i_m}$ with binary values $\beta_1, \dots, \beta_m$ respectively.

It is well known that if B_1 and B_2 are invertible matrices over P then the Kronecker product matrix $B_1 \otimes B_2$ is invertible too and $(B_1 \otimes B_2)^{-1} = B_1^{-1} \otimes B_2^{-1}$. In particular, if $B \in M_2(P)$ is an invertible matrix and $A = B^{[n]}$ then A is invertible too and $A^{-1} = (B^{-1})^{[n]}$.

Now we will demonstrate how Proposition 1 substantially facilitates proving of some important matrix identities for various representations of a function of Boolean variables. By convention, for a Boolean variable x we assume that $x^0 = \overline{x}$ and $x^1 = x$.

The Identity Transform. Let P be an arbitrary field and set $B = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ and $A = B^{[n]}$. Then, by Proposition 1,

$$g_i(x_1, \ldots, x_n) = \prod_{j=1}^{n} \left(\overline{\alpha_i^j}\,\overline{x_j} + \alpha_i^j x_j \right) = \prod_{j=1}^{n} x_j^{\alpha_i^j} = x_1^{\alpha_i^1} \cdot \ldots \cdot x_n^{\alpha_i^n}$$

and

$$\begin{aligned} f(x_1, \ldots, x_n) &\overset{(1)}{=} \sum_{i=0}^{2^n-1} g_i(x_1, \ldots, x_n) F(\alpha_i) = \\ &= \sum_{i=0}^{2^n-1} \left(x_1^{\alpha_i^1} \cdot \ldots \cdot x_n^{\alpha_i^n} \right) F(\alpha_i) = F(x_1, \ldots, x_n) \ . \end{aligned}$$

The Algebraic Normal Transform. Take $P = \mathrm{GF}(2)$ and set $B = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} = B^{-1}$ and $A = B^{[n]}$. Then, by Proposition 1,

$$g_i(x_1, \ldots, x_n) = \prod_{j=1}^{n} \left(\overline{\alpha_i^j} + \alpha_i^j x_j \right) = \prod_{j=1,\ldots,n:\, \alpha_i^j = 1} x_j \tag{2}$$

and

$$f(x_1, \ldots, x_n) \overset{(1)}{=} \sum_{i=0}^{2^n-1} g_i(x_1, \ldots, x_n) F(\alpha_i) = \sum_{i=0}^{2^n-1} \left(\prod_{j=1,\ldots,n:\, \alpha_i^j = 1} x_j \right) F(\alpha_i) \ .$$

One can easily recognize the ANF of function f on the right hand side of the last identity, where $F(\alpha_i)$ $(i = 0, \ldots, 2^n - 1)$ are the coefficients of the ANF polynomial. Let P^f denote the coefficient vector of the ANF polynomial for function f and denote also $R_2 = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}$, $R_{2^n} = R_2^{[n]}$. Then

$$T^f = R_{2^n} P^f \quad \text{and} \quad P^f = R_{2^n} T^f \ . \tag{3}$$

This transform of f is called the algebraic normal transform.

If R_{2^n} is considered as a matrix over the real number field $\mathbb{R}$ and the algebraic normal transform of f is implemented over $\mathbb{R}$ then T^F is equal to the coefficient vector of a real-valued, square-free (in variables) polynomial of n variables with integer coefficients that takes on the same values as function f on the points from $\mathrm{GF}(2)^n$. Let Π_f denote the coefficient vector of such a polynomial. In this case $R_2^{-1} = \begin{pmatrix} 1 & 0 \\ -1 & 1 \end{pmatrix}$, $R_{2^n}^{-1} = (R_2^{-1})^{[n]}$,

$$T^f = R_{2^n} \Pi^f \quad \text{and} \quad \Pi^f = R_{2^n}^{-1} T^f \ . \tag{4}$$

This real-valued polynomial gives an important insight in certain probabilistic properties of a Boolean function that will be discussed further in Sect. 4.

The Probabilistic Transform. Assume that $P = \mathbb{R}$ and set $B = \frac{1}{2}\begin{pmatrix} 2 & -1 \\ 2 & 1 \end{pmatrix}$ and $A = B^{[n]}$. Then $B^{-1} = \frac{1}{2}\begin{pmatrix} 1 & 1 \\ -2 & 2 \end{pmatrix}$ and, by Proposition 1,

$$\begin{aligned} g_i(x_1, \ldots, x_n) &= \prod_{j=1}^{n} \left(\overline{\alpha_i^j} + \frac{1}{2} \alpha_i^j (x_j - \overline{x_j}) \right) = \\ &= \prod_{j=1,\ldots,n:\, \alpha_i^j = 1} \frac{1}{2}(x_j - \overline{x_j}) \overset{(\circ)}{=} \prod_{j=1,\ldots,n:\, \alpha_i^j = 1} \delta_j \ , \end{aligned}$$

where $(\circ)$ is obtained by using $\overline{x_j} = 1 - x_j$ and introducing the new variable $\delta_j := x_j - 1/2$. Therefore,

$$f(x_1, \ldots, x_n) \overset{(1)}{=} \sum_{i=0}^{2^n-1} g_i(x_1, \ldots, x_n) F(\alpha_i) = \sum_{i=0}^{2^n-1} \left(\prod_{j=1,\ldots,n:\, \alpha_i^j = 1} \delta_j \right) F(\alpha_i) \ .$$

The right hand side of the last identity contains the real-valued, square-free polynomial of n variables $\delta_1, \ldots, \delta_n$ that for $\{\delta_1, \ldots, \delta_n\} \in \{-1/2, 1/2\}^n$ takes on the same values as function f on corresponding arguments $\{x_1, \ldots, x_n\}$ if identity $x_j = \delta_j + 1/2$ is assumed. Therefore, if $D_f(x_1, \ldots, x_n)$ denotes a polynomial obtained by the algebraic normal transform over the reals then the probabilistic transform gives coefficients for polynomial $D_f(1/2+\delta_1, \ldots, 1/2+\delta_n)$ that we will denote by Δ^f. Denote also $Q_2 = \frac{1}{2}\begin{pmatrix} 2 & -1 \\ 2 & 1 \end{pmatrix}$, $Q_{2^n} = Q_2^{[n]}$. Then $Q_2^{-1} = \frac{1}{2}\begin{pmatrix} 1 & 1 \\ -2 & 2 \end{pmatrix}$, $Q_{2^n}^{-1} = (Q_2^{-1})^{[n]}$,

$$T^f = Q_{2^n} \Delta^f \quad \text{and} \quad \Delta^f = Q_{2^n}^{-1} T^f \ . \tag{5}$$

We will call this transform of f the *probabilistic transform*. Applications of this transform will be discussed further in Sect. 4.

The Walsh Transform. According to [1, p. 118], the direct and inverse *Walsh transforms* of a real-valued function f over $\mathrm{GF}(2)^n$ are defined as

$$S_f(\alpha_i) = \sum_{\boldsymbol{x}=0}^{2^n-1} f(\boldsymbol{x})(-1)^{\langle \alpha_i, \boldsymbol{x} \rangle} \quad \text{and} \quad f(\boldsymbol{x}) = \frac{1}{2^n} \sum_{i=0}^{2^n-1} S_f(\alpha_i)(-1)^{\langle \alpha_i, \boldsymbol{x} \rangle} \ , \qquad (6)$$

where $\boldsymbol{x} = (x_1, \ldots, x_n)$ and $\langle \alpha_i, \boldsymbol{x} \rangle = \alpha_i^1 x_1 \oplus \ldots \oplus \alpha_i^n x_n$ is the standard inner product over GF(2). In the sum over $\boldsymbol{x}$ in (6) the summation index is considered as an integer in the range $0, \ldots, 2^n - 1$ but written in its binary expansion. The vector $S^f = (S_f(\alpha_0), \ldots, S_f(\alpha_{2^n-1}))$ is called the Walsh transform of function f.

Assume that $P = \mathbb{R}$ and set $B = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} = 2B^{-1}$ and $A = B^{[n]}$. Thus, A is a Hadamard matrix of order 2^n (see [5, p. 422]). Then, by Proposition 1,

$$g_i(x_1, \ldots, x_n) = \prod_{j=1}^{n} \left(\overline{\alpha_i^j} + \alpha_i^j (\overline{x_j} - x_j) \right) = \prod_{j=1,\ldots,n:\, \alpha_i^j = 1} (\overline{x_j} - x_j) = (-1)^{\langle \alpha_i, \boldsymbol{x} \rangle}$$

and

$$f(x_1, \ldots, x_n) \overset{(1)}{=} \sum_{i=0}^{2^n-1} g_i(x_1, \ldots, x_n) F(\alpha_i) = \sum_{i=0}^{2^n-1} F(\alpha_i)(-1)^{\langle \alpha_i, \boldsymbol{x} \rangle} \ .$$

In the latest identity one can recognize the inverse Walsh transform (6) but without the multiplicative coefficient. Therefore, in this case $F(\alpha_i) = 1/2^n S_f(\alpha_i)$, where $S_f(\alpha_i)$ is the Walsh transform of f evaluated in α_i. Let $H_2 = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$ and $H_{2^n} = H_2^{[n]}$. Then

$$T^f = \frac{1}{2^n} H_{2^n} S^f \quad \text{and} \quad S^f = H_{2^n} T^f \ . \qquad (7)$$

It is possible to generalize property (7) of the Walsh transform. Let us assume that function f is Boolean. From now on $wt(\omega)$ denotes the Hamming weight of a binary string ω and $wt(f)$ denotes the Hamming weight of a Boolean function f, i.e., the weight of T^f. Let r be an integer in the range $1 \leq r \leq n$ and let $i_1, \ldots, i_r$ be a set of indices with $1 \leq i_1 < \ldots < i_r \leq n$. Let $k_1, \ldots, k_{n-r}$ with $1 \leq k_1 < \ldots < k_{n-r} \leq n$ denote the indices complementing $i_1, \ldots, i_r$ with respect to $\{1, \ldots, n\}$. Let also the real-valued function $w(y_1, \ldots, y_r)$ of r Boolean variables be defined as follows

$$w(\alpha_j^1, \ldots, \alpha_j^r) = wt\left(f_{i_1,\ldots,i_r}^{\alpha_j^1,\ldots,\alpha_j^r}(x_{k_1}, \ldots, x_{k_{n-r}}) \right) = w_j$$

for $0 \leq j < 2^r$, where $(\alpha_j^1, \ldots, \alpha_j^r) = \alpha_j$ is the r-bit binary expansion of j. Then, by (7), $S^w = H_{2^r}(w_0, \ldots, w_{2^r-1})^T$. On the other hand,

$$S_w(\alpha_i) \overset{(6)}{=} \sum_{j=0}^{2^r-1} w(\alpha_j)(-1)^{\langle \alpha_j, \alpha_i \rangle} = \sum_{j=0}^{2^r-1} \sum_{t=0}^{2^{n-r}-1} f_{i_1,\ldots,i_r}^{\alpha_j^1,\ldots,\alpha_j^r}(\alpha_t)(-1)^{\langle \alpha_j, \alpha_i \rangle} =$$

$$= \sum_{k=0}^{2^n-1} f(\alpha_k)(-1)^{\langle \alpha_k, \theta_i \rangle} \overset{(6)}{=} S_f(\theta_i) \ ,$$

where θ_i is the n-bit vector whose coordinates at the index positions $i_1, \ldots, i_r$ are equal to $\alpha_i^1, \ldots, \alpha_i^r$ respectively (where $(\alpha_i^1, \ldots, \alpha_i^r) = \alpha_i$) and the remaining $(n-r)$ coordinates are set to zero. Thus,

$$H_{2^r}(w_0, \ldots, w_{2^r-1})^T = (S_f(\theta_0), \ldots, S_f(\theta_{2^r-1}))^T \ , \tag{8}$$

which is the generalization of [6, Proposition 3.1], while the proof here is less complicated. If r is set equal to n then $w_j = f(\alpha_j)$, $\theta_i = \alpha_i$ and (8) transforms into (7).

If function f is Boolean then in some cases it is more convenient to work with the real-valued counterpart (sign function) of f, defined as $\hat{f}(\boldsymbol{x}) = 1 - 2f(\boldsymbol{x})$, and to apply the Walsh transform to $\hat{f}$. Function $\hat{f}$ can be recovered by the inverse Walsh transform of $S^{\hat{f}}$. Further, since $f(\boldsymbol{x}) = 1/2 - 1/2\hat{f}(\boldsymbol{x})$, the original function f can be obtained from the Walsh transform $S^{\hat{f}}$ by the following inverse transform:

$$f(\boldsymbol{x}) = \frac{1}{2} - \frac{1}{2^{n+1}} \sum_{i=0}^{2^n-1} S_{\hat{f}}(\alpha_i)(-1)^{\langle \alpha_i, \boldsymbol{x} \rangle} \ .$$

The relationship between the Walsh transform of $f(\boldsymbol{x})$ and $\hat{f}(\boldsymbol{x})$ is given by [7, Lemma 1] as follows

$$S_{\hat{f}}(0) = 2^n - 2S_f(0) \quad \text{and} \quad S_{\hat{f}}(w) = -2S_f(w) \quad \text{for} \quad 0 < w < 2^n \ . \tag{9}$$

By these identities and (7),

$$T^f = \left(\frac{1}{2}, \ldots, \frac{1}{2}\right)^T - \frac{1}{2^{n+1}} H_{2^n} S^{\hat{f}} \quad \text{and} \quad S^{\hat{f}} = (2^n, 0, \ldots, 0)^T - 2H_{2^n} T^f \tag{10}$$

since $H_{2^n}\left(\frac{1}{2}, 0, \ldots, 0\right)^T = \left(\frac{1}{2}, \ldots, \frac{1}{2}\right)^T$. On the other hand, identities, similar to (7), hold:

$$T^{\hat{f}} = \frac{1}{2^n} H_{2^n} S^{\hat{f}} \quad \text{and} \quad S^{\hat{f}} = H_{2^n} T^{\hat{f}} \ .$$

Combining (3) with (7) or (10), we obtain the following identities relating the coefficient vector of the ANF polynomial of f with the Walsh transforms S^f and $S^{\hat{f}}$:

$$P^f = \frac{1}{2^n} R_{2^n} H_{2^n} S^f \pmod 2 = \frac{1}{2^n} \begin{pmatrix} 1 & 1 \\ 2 & 0 \end{pmatrix}^{[n]} S^f \pmod 2 \tag{11}$$

$$P^f = R_{2^n} \left(\left(\frac{1}{2}, \ldots, \frac{1}{2}\right)^T - \frac{1}{2^{n+1}} H_{2^n} S^{\hat{f}} \right) \pmod 2 \ ,$$

where all operations on the right hand side are performed in $\mathbb{R}$ and the final result is reduced modulo 2. Using Proposition 1 for $B = \begin{pmatrix} 1 & 1 \\ 2 & 0 \end{pmatrix}$ and $A = B^{[n]}$, we obtain

$$g_i(x_1, \ldots, x_n) = \prod_{j=1}^{n} \left(\overline{x_j} + 2\overline{\alpha_i^j} x_j \right) = \begin{cases} 2^{wt(x_1,\ldots,x_n)}, & \text{if } \alpha_i^j \le \overline{x_j} \ (j = 1, \ldots, n), \\ 0, & \text{otherwise} \end{cases} .$$

Therefore, by (11),

$$P_f(\omega) = \frac{1}{2^{n-wt(\omega)}} \sum_{i=0,\ldots,2^n-1:\, \alpha_i \preceq \overline{\omega}} S_f(\alpha_i) \pmod 2 \stackrel{(9)}{=} \tag{12}$$
$$= 2^{wt(\omega)-1} \left(1 - \frac{1}{2^n} \sum_{i=0,\ldots,2^n-1:\, \alpha_i \preceq \overline{\omega}} S_{\hat{f}}(\alpha_i) \right) \pmod 2 \ ,$$

where $\overline{\omega}$ denotes the bitwise completion to 1, $\preceq$ is the partial ordering on the Boolean lattice (defined as $\alpha \preceq \omega$ if and only if $\alpha^j \le \omega^j$ for $j = 1, \ldots, n$) and $P_f(\omega)$ is the ωth coefficient of the ANF of function f. The relation (12) between the ANF and Walsh coefficients can be found in [8, Proposition 3] but our proof seems easier.

Finally, if (5) is combined with (7) then the resulting identities relate the probabilistic transform of f with the Walsh transform S^f:

$$\Delta^f = \frac{1}{2^n} \begin{pmatrix} 1 & 0 \\ 0 & -2 \end{pmatrix}^{[n]} S^f \quad \text{and} \quad S^f = \begin{pmatrix} 2 & 0 \\ 0 & -1 \end{pmatrix}^{[n]} \Delta^f \ . \tag{13}$$

Since the matrix of the transform (13) is diagonal, coordinates of zero values in vectors Δ^f and S^f are the same. Now, using Proposition 1 for $B = \begin{pmatrix} 1 & 0 \\ 0 & -2 \end{pmatrix}$ and $A = B^{[n]}$, we obtain

$$g_i(x_1, \ldots, x_n) = \prod_{j=1}^{n} \left(\overline{\alpha_i^j}\,\overline{x_j} - 2\alpha_i^j x_j \right) = \begin{cases} (-2)^{wt(\alpha_i)}, & \text{if } x_j = \alpha_i^j \ (j = 1, \ldots, n), \\ 0, & \text{otherwise} \end{cases} .$$

Therefore, by (13),

$$\Delta_f(\omega) = \frac{1}{2^n} (-2)^{wt(\omega)} S_f(\omega) \stackrel{(9)}{=} \begin{cases} \frac{1}{2^n} (-2)^{wt(\omega)-1} S_{\hat{f}}(\omega), & \text{if } \ \omega \ne 0, \\ \frac{1}{2} - \frac{1}{2^{n+1}} S_{\hat{f}}(0), & \text{if } \ \omega = 0 \end{cases} , \tag{14}$$

where $\Delta_f(\omega)$ is the ωth coordinate of the probabilistic transform of function f.

The Weight Transform. Take $P = \mathbb{R}$ and set $B_0 = \begin{pmatrix} 0 & 1 \\ 1 & -1 \end{pmatrix}$, $B_1 = \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix}$ and $A = B_{\beta_1} \otimes \ldots \otimes B_{\beta_n}$ for some n-bit vector $\beta = (\beta_1, \ldots, \beta_n)$. Let also $A^{-1} =$

$B_{\beta_1}^{-1} \otimes \ldots \otimes B_{\beta_n}^{-1} = (\tilde{\boldsymbol{g}}_0, \ldots, \tilde{\boldsymbol{g}}_{2^n-1})$, where $B_0^{-1} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$ and $B_1^{-1} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$. By Proposition 1 and since $b_{11}^{(j)} = b_{12}^{(j)} = 1$ for any $j = 1, \ldots, n$,

$$\tilde{g}_i(x_1, \ldots, x_n) = \prod_{j=1}^{n} \left(\overline{x_j} + x_j \left(\overline{\alpha_i^j} b_{21}^{(j)} + \alpha_i^j b_{22}^{(j)} \right) \right) \stackrel{(*)}{=}$$

$$\stackrel{(*)}{=} \prod_{j=1}^{n} \left(\overline{x_j} + x_j \left(\overline{\alpha_i^j} \overline{\beta_j} + \alpha_i^j \beta_j \right) \right) = \prod_{j=1,\ldots,n:\, \alpha_i^j \neq \beta_j} \overline{x_j} \ . \quad (15)$$

Equality (*) holds because $b_{21}^{(j)} = \overline{\beta_j}$ and $b_{22}^{(j)} = \beta_j$. Thus, $\tilde{g}_i(x_1, \ldots, x_n)$ is equal to one if and only if the coordinates, where vectors α_i and β differ, correspond to the zero entries in vector $(x_1, \ldots, x_n)$.

Let us assume that function f is Boolean. Then

$$F(x_1, \ldots, x_n) = \sum_{i=0}^{2^n-1} \tilde{g}_i(x_1, \ldots, x_n) f(\alpha_i) =$$

$$= \sum_{i=0}^{2^n-1} \left(\prod_{j=1,\ldots,n:\, \alpha_i^j \neq \beta_j} \overline{x_j} \right) f(\alpha_i) = wt\left(f_{t_1,\ldots,t_k}^{\beta_{t_1},\ldots,\beta_{t_k}} \right) \ ,$$

where $k = wt(x_1, \ldots, x_n)$ and $t_1, \ldots, t_k$ are the coordinates of the nonzero entries in $(x_1, \ldots, x_n)$. Here it is assumed that if $\alpha_i = \beta$ then $\prod_{j=1,\ldots,n:\, \alpha_i^j \neq \beta_j} \overline{x_j} = 1$. Therefore, $wt\left(f_{1,\ldots,n}^{\beta_1,\ldots,\beta_n} \right) = f(\beta)$.

Let Θ_β^f denote the ordered 2^n-tuple, containing the weights of the subfunctions of f, obtained by fixing all possible subsets of variables with corresponding values from vector β. Thus,

$$\Theta_\beta^f = \left\{ wt\left(f_{i_1,\ldots,i_k}^{\beta_{i_1},\ldots,\beta_{i_k}} \right) \mid 1 \leq i_1 < \ldots < i_k \leq n;\ k \in \{0, \ldots, n\} \right\} \ .$$

Denote also $D_\beta = B_{\beta_1} \otimes \ldots \otimes B_{\beta_n}$. Then

$$T^f = D_\beta \, \Theta_\beta^f \quad \text{and} \quad \Theta_\beta^f = D_\beta^{-1} T^f \ . \quad (16)$$

We will call this transform of f the *weight transform.* In particular, if vector β consists of zeros only then $D_\beta = B_0^{[n]}$, and if it consists only of ones then $D_\beta = B_1^{[n]}$.

If we consider matrices B_0 and B_1 as matrices over the field GF(2) and perform all operations in (16) in this field then (16) will relate the string of values of function f with binary weights of its subfunctions.

Let us compare the basis vector set (15) of the weight transform when $\beta = (0, \ldots, 0)$ with the basis vector set (2) of the inverse algebraic normal transform. It is clear that they are directly related via a simple variable complementation.

Since $R_{2^n} = R_{2^n}^{-1}$, the basis vector sets of the algebraic normal transform and its inverse are equal. Therefore,

$$P_f(\alpha_i^1, \dots, \alpha_i^n) = \Theta_0^f(\overline{\alpha_i^1}, \dots, \overline{\alpha_i^n}) \pmod 2 \tag{17}$$

for any $i = 0, \dots, 2^n - 1$, where $(\alpha_i^1, \dots, \alpha_i^n) = \alpha_i$. This identity is easily accounted for by the well-known fact that a Boolean function has maximal algebraic degree if and only if it has an odd weight. Indeed, the right hand side of the identity contains the binary weight of the subfunction which maximal possible order term in the ANF is equal to $\prod_{j=1,\dots,n:\, \alpha_i^j=1} x_j$ and the coefficient for this term in the ANF of f is the value on the left hand side of the identity. To construct the subfunction, relevant variables of f are being fixed only with zero values, therefore, the term $\prod_{j=1,\dots,n:\, \alpha_i^j=1} x_j$ is either present in the ANFs of both f and the subfunction or is missing in both.

In Sect. 3 we define correlation coefficients of a function f that are estimated by means of the weight transform of f. This indicates to the importance of the weight transform for assessing cryptographic characteristics of Boolean functions.

It is important to note that the P^f, Π^f, Δ^f, S^f, $S^{\hat{f}}$ and Θ_β^f transforms of a function f can be represented by matrix equations (3), (4), (5), (7), (10) and (16), all based on the Kronecker product of appropriate elementary cells. This fact allows to use fast Fourier and Walsh transform algorithms [9, 10] for efficient estimation of these transforms and easy transition from one transform to another. Indeed, let a and b be arbitrary 2^n-dimensional vectors over P, such that $b = (B_1 \otimes \dots \otimes B_n)a$, where $B_j = \begin{pmatrix} b_{11}^{(j)} & b_{12}^{(j)} \\ b_{21}^{(j)} & b_{22}^{(j)} \end{pmatrix}$ $(j = 1, \dots, n)$ are arbitrary elementary cells over P. Then

$$b = \begin{pmatrix} b_{11}^{(1)} B' & b_{12}^{(1)} B' \\ b_{21}^{(1)} B' & b_{22}^{(1)} B' \end{pmatrix} a = \begin{pmatrix} b_{11}^{(1)} B' \overline{a} + b_{12}^{(1)} B' \underline{a} \\ b_{21}^{(1)} B' \overline{a} + b_{22}^{(1)} B' \underline{a} \end{pmatrix} , \tag{18}$$

where $B' = B_2 \otimes \dots \otimes B_n$ and $a = (\overline{a}, \underline{a})$ is the split of a into two halves. Thus, estimation of b requires 2^{n+1} arithmetic operations in P and two transforms of order $n-1$. It is easy to prove by induction that the total complexity of the nth-order transform is equivalent to $O(n2^n)$ arithmetic operations in P.

Also note that if a is the string of values of a function f, i.e., $a = T^f$, then $\overline{a}$ and $\underline{a}$ are strings of values of subfunctions f_1^0 and f_1^1 respectively. Thus, $B'\overline{a}$ and $B'\underline{a}$ are the adequate transforms of these subfunctions. Thus (18) provides a relation between the transform of a function f and transforms of its subfunctions f_1^0 and f_1^1.

3 Algebraic and Correlation Properties of Boolean Functions Related to the Weight Transform

The concept of correlation immunity relates to the statistical dependency between m-tuples of input bits and the output of a cryptographic transformation.

This idea is extremely important, especially for stream cipher design, where filter and combination generators with not correlation immune filtering and combining functions are susceptible to ciphertext-only attacks [2].

High order correlation immunity was first introduced in [11] where the well-known Siegenthaler's inequality has also been first proved in [11, Theorem 1]. According to this inequality, the sum of the algebraic degree and the order of correlation immunity for a Boolean function of n variables can not exceed n and $n-1$ if function is balanced. Therefore, high-order correlation immune functions necessarily have low algebraic degree and vice versa. In order to handle this situation one has either to find a trade-off between these two properties or somehow to weaken the requirement for a function to be correlation immune. From the practical point of view, those functions which correlation dependencies are low are as secure as correlation immune ones. In this case we need an estimate for correlation dependencies of a Boolean function and the following definition of correlation coefficients generalizes the basic concept of correlation immunity.

Further we assume that $X_1, \ldots, X_n$ are uniform, independent and identically distributed random binary variables and $f(\boldsymbol{x})$, where $\boldsymbol{x} = (x_1, \ldots, x_n) \in \mathrm{GF}(2)^n$, is a Boolean function of n variables that is not identical 0 or 1. If we denote $\boldsymbol{X} = (X_1, \ldots, X_n)$ then $f(\boldsymbol{X})$ would denote the binary random variable obtained by substituting variables of f with random values X_i.

Definition 1. *Let m and $i_1, \ldots, i_m$ be integers with $1 \leq m \leq n$ and $1 \leq i_1 < \ldots < i_m \leq n$. Then the set of 2^m conditional probabilities*

$$c_{i_1,\ldots,i_m}^{\beta_1,\ldots,\beta_m} = P(X_{i_1} = \beta_1, \ldots, X_{i_m} = \beta_m \mid f(\boldsymbol{X}) = 0) \ ,$$

evaluated for all possible values of the m-bit tuple $(\beta_1, \ldots, \beta_m)$ and ordered lexicographically along these values, is called a vector of mth-order correlation coefficients of f, evaluated for the input subset $(i_1, \ldots, i_m)$.

It is obvious that function f is mth-order correlation immune (as defined in [11]) if all its mth-order correlation coefficients are equal to $1/2^m$. On the other hand, for any Boolean function f,

$$\begin{aligned} P(X_{i_1} = \beta_1, \ldots, X_{i_m} = \beta_m \mid f(\boldsymbol{X}) = 0) &= \frac{2^{n-m} - wt\left(f_{i_1,\ldots,i_m}^{\beta_1,\ldots,\beta_m}\right)}{2^n - wt(f)} \\ P(X_{i_1} = \beta_1, \ldots, X_{i_m} = \beta_m \mid f(\boldsymbol{X}) = 1) &= \frac{wt\left(f_{i_1,\ldots,i_m}^{\beta_1,\ldots,\beta_m}\right)}{wt(f)} \ . \end{aligned} \tag{19}$$

Conditional probabilities in (19) are equal to $1/2^m$ if and only if $wt\left(f_{i_1,\ldots,i_m}^{\beta_1,\ldots,\beta_m}\right) = 2^{-m} wt(f)$. Therefore, the mth-order correlation immunity of f implies that the output of f and any m input variables, considered jointly, are statically independent. Correlation coefficients $c_{i_1,\ldots,i_m}^{\beta_1,\ldots,\beta_m}$ are easily estimated, making use of (19), if the weights of function f and of the subfunctions $f_{i_1,\ldots,i_m}^{\beta_1,\ldots,\beta_m}$ are known (see Sect. 2 about the weight transform). For instance, 1st order correlation coefficients satisfy the identity

$$c_i^\beta = \frac{1}{2^n - wt(f)} \left(2^{n-1} - wt\left(f_i^\beta\right)\right) \quad ,$$

and $wt\left(f_i^\beta\right)$ is equal to the number of n-bit vectors $(x_1, \ldots, x_n)$ in the support of f that have the ith coordinate x_i, equal to β. By the support of f we mean the subset of $\mathrm{GF}(2)^n$, where f is equal to 1.

Note 1. It is well known [4] that a Boolean function of n variables is mth-order correlation immune for $1 \le m \le n$, if and only if its Walsh transform is equal to zero for any nonzero vector with a Hamming weight not exceeding m. In particular, this property implies that for $m > 1$, any mth-order correlation immune function is also $(m-1)$st-order correlation immune.

Proposition 2. *A Boolean function f of n variables is mth-order correlation immune for $1 \le m \le n$, if and only if for every $k \in \{1, \ldots, m\}$ and any set of indices $i_1, \ldots, i_k$ with $1 \le i_1 < \ldots < i_k \le n$, there exists at least one k-bit tuple $(\beta_1, \ldots, \beta_k)$, such that correlation coefficient $c_{i_1,\ldots,i_k}^{\beta_1,\ldots,\beta_k}$ is equal to $1/2^k$.*

Proof. By Definition 1 and Note 1, it is obvious that the condition stated in the proposition is necessary for a function to be mth-order correlation immune. To show that this condition is sufficient, we apply induction on m.

Let $m = 1$ and assume that for any i with $1 \le i \le n$ there exists some β_i, such that the corresponding correlation coefficient $c_i^{\beta_i}$ is equal to $1/2$. Then, by (19), $wt(f_i^{\beta_i}) = wt(f)/2$. Therefore,

$$wt\left(f_i^{\beta_i \oplus 1}\right) = wt(f) - wt(f_i^{\beta_i}) = \frac{wt(f)}{2}$$

and

$$c_i^{\beta_i \oplus 1} = \frac{1}{2^n - wt(f)} \left(2^{n-1} - wt\left(f_i^{\beta_i \oplus 1}\right)\right) = \frac{1}{2} \quad .$$

Thus, function f is 1st-order correlation immune.

Now, supposing that the proposition is true for $m = l-1$, we prove it for $m = l$. Conditions imposed imply that for any set of indices $i_1, \ldots, i_l$ with $1 \le i_1 < \ldots < i_l \le n$, there exists an l-bit tuple $(\beta_1, \ldots, \beta_l)$, such that $c_{i_1,\ldots,i_l}^{\beta_1,\ldots,\beta_l}$ is equal to $1/2^l$. According to the induction hypothesis, the imposed conditions are sufficient for f to be $(l-1)$st-order correlation immune and thus, $c_{i_1,\ldots,i_{l-1}}^{\beta_1,\ldots,\beta_{l-1}} = 1/2^{l-1}$. Then, by (19), $wt\left(f_{i_1,\ldots,i_l}^{\beta_1,\ldots,\beta_l}\right) = wt(f)/2^l$ and $wt\left(f_{i_1,\ldots,i_{l-1}}^{\beta_1,\ldots,\beta_{l-1}}\right) = wt(f)/2^{l-1}$. Therefore,

$$wt\left(f_{i_1,\ldots,i_l}^{\beta_1,\ldots,\beta_{l-1},\beta_l \oplus 1}\right) = wt\left(f_{i_1,\ldots,i_{l-1}}^{\beta_1,\ldots,\beta_{l-1}}\right) - wt\left(f_{i_1,\ldots,i_l}^{\beta_1,\ldots,\beta_{l-1},\beta_l}\right) = \frac{wt(f)}{2^l}$$

and $c_{i_1,\ldots,i_l}^{\beta_1,\ldots,\beta_{l-1},\beta_l \oplus 1} = 1/2^l$. Any l-bit tuple can be obtained by consecutive inverting of required coordinates in the fixed tuple $(\beta_1, \ldots, \beta_l)$. This way it follows that for any m-bit tuple $(\gamma_1, \ldots, \gamma_l)$, the correlation coefficient $c_{i_1,\ldots,i_l}^{\gamma_1,\ldots,\gamma_l}$ is equal to $1/2^l$. Thus, function f is lth-order correlation immune. $\square$

A Boolean function f can not be considered cryptographically secure if there exists a function, having low algebraic degree or depending on small number of variables, that coincides with f on the larger half of the domain or, in other words, that *approximates* f. Balanced mth-order correlation immune functions are called m-resilient [6] and any balanced function is also called 0-resilient. The following proposition shows that an m-resilient Boolean function (if m is sufficiently large) does not have approximations nondegenerate on few variables.

Proposition 3. *Any balanced Boolean function f of n variables is m-resilient for $1 \leq m < n$, if and only if there are no approximations of f, depending on at most m variables.*

Proof. Suppose that function $f(x_1, \ldots, x_n)$ is m-resilient for some $1 \leq m < n$ and that there exists a function $g(x_{i_1}, \ldots, x_{i_m})$ approximating f. Then

$$
\begin{aligned}
P(f(\boldsymbol{X}) \neq g(X_{i_1}, \ldots, X_{i_m})) = \frac{wt(f(\boldsymbol{x}) \oplus g(x_{i_1}, \ldots, x_{i_m}))}{2^n} &= \\
= \frac{\sum_{(\beta_{i_1},\ldots,\beta_{i_m}) \in \mathrm{GF}(2)^m} wt\left(f_{i_1,\ldots,i_m}^{\beta_{i_1},\ldots,\beta_{i_m}} \oplus g(\beta_{i_1}, \ldots, \beta_{i_m})\right)}{2^n} &= \\
= \frac{2^m 2^{n-m-1}}{2^n} = \frac{1}{2} \ .
\end{aligned}
$$

Thus, $g(x_{i_1}, \ldots, x_{i_m})$ does not approximate f.

Suppose now that there are no approximations of f, depending on at most m variables. In particular, there are no linear approximations, depending on at most m variables, meaning that for any n-bit vector $\beta = (\beta_1, \ldots, \beta_n)$ such that $0 < wt(\beta) \leq m$, $P(f(\boldsymbol{X}) = (\beta_1 X_1 \oplus \ldots \oplus \beta_n X_n)) = 1/2$. On the other hand, for any nonzero β the following known [1, p. 121] identity holds

$$
P(f(\boldsymbol{X}) = (\beta_1 X_1 \oplus \ldots \oplus \beta_n X_n)) = \frac{1}{2} - \frac{S_f(\beta)}{2^n} \ , \tag{20}
$$

where $S_f(\beta)$ is the Walsh transform of f evaluated in β. Thus, $S_f(\beta) = 0$ and by Note 1, function f is mth-order correlation immune. $\square$

For any Boolean function f of n variables let $S_m(f)$ denote the number of subfunctions obtained by fixing m variables of f with zero values and having an even weight. Let also $D_{n-m}(f)$ denote the total number of $(n-m)$th-order product terms, contained in the ANF of f. The following proposition, that easily follows from (17), establishes a relation between the values of $S_m(f)$ and $D_{n-m}(f)$.

Proposition 4. *For any Boolean function f of n variables and any positive integer $m \leq n$,*

$$
S_m(f) + D_{n-m}(f) = \binom{n}{m} \ .
$$

Further, let $C_m(f)$ denote the number of mth-order correlation coefficient vectors of f for which coordinate $c^{0,\ldots,0}_{i_1,\ldots,i_m}$ is equal to $1/2^m$. From (19) it is clear that an nth-order correlation coefficient of a nonconstant Boolean function of n variables can not be equal to $1/2^n$ and thus, for such a function $C_n(f) = 0$.

Corollary 1. *For any balanced Boolean function f of n variables and any positive integer $m < n-1$,*

$$C_m(f) + D_{n-m}(f) \leq \binom{n}{m} \ .$$

Moreover, equality holds if function f is such that

$$\frac{2^{n-m-1}-1}{2^{n-1}} \leq c^{0,\ldots,0}_{i_1,\ldots,i_m} \leq \frac{2^{n-m-1}+1}{2^{n-1}} \tag{21}$$

for all index values $i_1,\ldots,i_m$ with $1 \leq i_1 < \ldots < i_m \leq n$.

Proof. By (19), coordinate $c^{0,\ldots,0}_{i_1,\ldots,i_m}$ of the mth-order correlation coefficient vector of f, evaluated for the input subset $(i_1,\ldots,i_m)$, is equal to $1/2^m$ if and only if $wt\left(f^{0,\ldots,0}_{i_1,\ldots,i_m}\right) = 2^{n-m-1}$, which is an even value for any $m < n-1$. Thus, $C_m(f) \leq S_m(f)$ and the claimed inequality directly follows from Proposition 4.

Suppose now that condition (21) holds for all index values $i_1,\ldots,i_m$ with $1 \leq i_1 < \ldots < i_m \leq n$. Then, by (19),

$$2^{n-m-1} - 1 \leq wt\left(f^{0,\ldots,0}_{i_1,\ldots,i_m}\right) \leq 2^{n-m-1} + 1 \ .$$

Suppose also that the ANF of function f does not contain the $(n-m)$th-order product term $\prod_{j=1,\ldots,n:\, j\notin\{i_1,\ldots,i_m\}} x_j$. Then, by (17), the subfunction $f^{0,\ldots,0}_{i_1,\ldots,i_m}$ has an even weight, equal to 2^{n-m-1}. Then, by (19),

$$c^{0,\ldots,0}_{i_1,\ldots,i_m} = \frac{1}{2^{n-1}}\left(2^{n-m} - wt\left(f^{0,\ldots,0}_{i_1,\ldots,i_m}\right)\right) = \frac{1}{2^m} \ .$$

So, if condition (21) holds then every missing $(n-m)$th-order product term in the ANF of f gives rise to a $1/2^m$ valued coordinate of the corresponding correlation vector and thus,

$$C_m(f) \geq \binom{n}{m} - D_{n-m}(f) \ .$$

Now the latest inequality combined with the one argued in the first part of the corollary produces the claimed equality. □

From Corollary 1 and Note 1 it easily follows, that for $m < n-1$ and any m-resilient Boolean function f of n variables, $D_{n-k}(f) = 0$ for all $k = 1,\ldots,m$ (since $C_k(f) = \binom{n}{k}$). The maximal order product term $x_1 \cdot \ldots \cdot x_n$ is missing in the ANF of f since function f has an even weight. Therefore, the algebraic degree of f does not exceed $(n-m-1)$. So, it can be concluded that if k is the attainable algebraic degree and m is the attainable degree of resiliency for a balanced Boolean function of n variables then $k + m \leq n-1$ (that is Siegenthaler's inequality for a balanced function).

4 Probabilistic Function of a Boolean Function

Let us consider the arrangement when n sequences of nonuniform, independent and identically distributed (i.i.d.) random binary variables are combined with a Boolean function to produce an output sequence hopefully having better algebraic and statistical properties relevant to a key-stream. In this section we show that an appropriately chosen combining function can compensate the nonuniform distribution of the inputs and generate the close-to-uniform output. Similar problems were considered in a recent paper [12] where maximized estimates for the bias of the distribution of the output bits were made. Our approach allows to obtain explicit polynomial expression for this bias.

Definition 2. *Let $f(x_1,\dots,x_n)$ be a Boolean function of n variables. Assume that $\boldsymbol{X}=(X_1,\dots,X_n)$ is an n-tuple consisting of i.i.d. random binary variables with $P(X_i=1)=p_i$ for $i=1,\dots,n$. Then function $F_f(p_1,\dots,p_n)=P(f(\boldsymbol{X})=1)$ is called the probabilistic function of f.*

From Definition 2 it follows that $F_f(p_1,\dots,p_n)=\sum_{\beta\colon f(\beta)=1}P(\boldsymbol{X}=\beta)$ and if $\beta=(\beta_1,\dots,\beta_n)$ then $P(\boldsymbol{X}=\beta)=\prod_{i=1}^n p_i^{\beta_i}(1-p_i)^{1-\beta_i}$. Thus, $F_f(p_1,\dots,p_n)$ is the polynomial of n variables $p_1,\dots,p_n$ with integer coefficients.

Further, let $D_f(x_1,\dots,x_n)$ denote the real-valued, square-free (in variables) polynomial of n variables with integer coefficients such that

$$D_f(x_1,\dots,x_n)=f(x_1,\dots,x_n) \quad \text{for any} \quad (x_1,\dots,x_n)\in \mathrm{GF}(2)^n \ . \qquad (22)$$

Let us write down the polynomial D_f in the canonical form

$$D_f(x_1,\dots,x_n)=\sum_{i=0}^{2^n-1} a_i\left(\prod_{j=1,\dots,n\colon \alpha_i^j=1} x_j\right) \ ,$$

where $\alpha_i=(\alpha_i^1,\dots,\alpha_i^n)$ is the n-bit binary expansion of i and $a_i\in\mathbb{Z}$. Then, since (22) holds, the integer coefficients a_i form the solution of the following system of linear equations

$$M(a_0,\dots,a_{2^n-1})^T=(f(0,\dots,0),\dots,f(1,\dots,1))^T \ ,$$

where $M=(m_{i,j})_{2^n\times 2^n}$ $(i,j=0,\dots,2^n-1)$ is a nondegenerate triangular $\{0,1\}$-matrix with $m_{i,j}=1$ if and only if the positions of ones in the n-bit binary expansion of j are a subset of those in the binary expansion of i (in particular, it is necessary that $j\le i$). Therefore, this system has a unique solution and that proves the *uniqueness* of the polynomial D_f. Moreover, the coefficient vector of D_f can be obtained by the algebraic normal transform of function f over $\mathbb{R}$ (see Sect. 2).

Identities $\overline{x}=1-x$, $x_1\wedge x_2=x_1x_2$ and $x_1\oplus x_2=x_1+x_2-2x_1x_2$ convert elementary Boolean operations into integer expressions. Thus, using these identities any formula representing $f(x_1, \ldots, x_n)$ in the basis $\{^-, \wedge, \oplus\}$ (for instance, the ANF) can be transformed into the real-valued polynomial of n variables with integer coefficients that satisfies (22). Moreover, if we assume that $x_i^2 \equiv x_i$ $(i = 1, \ldots, n)$ then the constructed polynomial is square-free and, therefore, by the uniqueness, is equal to D_f. That provides an alternative way for constructing polynomial D_f starting from a formula representing the Boolean function.

Proposition 5. *For any Boolean function $f(x_1, \ldots, x_n)$ and arbitrary values $p_1, \ldots, p_n$ with $0 \le p_i \le 1$ for all $i = 1, \ldots, n$*

$$F_f(p_1, \ldots, p_n) = D_f(p_1, \ldots, p_n) \ .$$

Proof. To prove this identity we apply induction on n.

Let $n = 1$. Then function f is one of the following four functions of a single variable

$$f_0 \equiv 0, \quad f_1 = x_1, \quad f_2 = \overline{x_1}, \quad f_3 \equiv 1 \ .$$

But

$$\begin{aligned}
&P(f_0 = 1) = 0 = D_{f_0} \\
&P(f_1 = 1) = P(X_1 = 1) = p_1 = D_{f_1}(p_1) \\
&P(f_2 = 1) = P(X_1 = 0) = 1 - p_1 = D_{f_2}(p_1) \\
&P(f_3 = 1) = 1 = D_{f_3} \ .
\end{aligned}$$

Now, supposing that the proposition is true for $n = l-1$, we prove it for $n = l$. It is easy to see that the following decomposition of function f into subfunctions holds:

$$f(x_1, \ldots, x_l) = \overline{x_1} f_1^0(x_2, \ldots, x_l) \oplus x_1 f_1^1(x_2, \ldots, x_l) \ .$$

According to the induction hypothesis, $F_{f_1^i}(p_2, \ldots, p_l) = D_{f_1^i}(p_2, \ldots, p_l)$ for $i = 0, 1$. On the other hand,

$$D_f(x_1, \ldots, x_l) = (1 - x_1) D_{f_1^0}(x_2, \ldots, x_l) + x_1 D_{f_1^1}(x_2, \ldots, x_l)$$

since $\overline{x_1} f_1^0(x_2, \ldots, x_l) x_1 f_1^1(x_2, \ldots, x_l) \equiv 0$ on $\mathrm{GF}(2)^n$. On the other hand, by the rule of total probability

$$F_f(p_1, \ldots, p_l) = (1 - p_1) F_{f_1^0}(p_2, \ldots, p_l) + p_1 F_{f_1^1}(p_2, \ldots, p_l) \ .$$

Thus, $F_f(p_1, \ldots, p_l) = D_f(p_1, \ldots, p_l)$ for any $p_1, \ldots, p_n$ with $0 \le p_i \le 1$ for all $i = 1, \ldots, n$. $\square$

Let w_i $(i = 0, \ldots, n)$ denote the number of vectors having the weight i in the support of a Boolean function f of n variables. Then vector $(w_0, \ldots, w_n)$ is called *the weight distribution* of function f.

Let us assume first that $p_1 = \ldots = p_n = p = 1/2 + \delta$, where $\delta \in (-1/2, 1/2)$ is the bias of the distribution of the random variable x_i ($i = 1, \ldots, n$). Then, since $\sum_{i=0}^{n} w_i = wt(f)$,

$$F_f(p) = \sum_{i=0}^{n} w_i p^i (1-p)^{n-i} = \sum_{i=0}^{n} w_i \left(\frac{1}{2} + \delta\right)^i \left(\frac{1}{2} - \delta\right)^{n-i} =$$
$$= d_1\delta + d_2\delta^2 + \ldots + d_n\delta^n + \frac{1}{2^n} wt(f) \ ,$$

where $d_1, \ldots, d_n$ are some real values. Let $\Delta_f(\delta) = F_f(1/2+\delta) - 1/2$ denote the bias of the distribution of the function f output. In particular, if function f is balanced then $\Delta_f(\delta) = d_1\delta + d_2\delta^2 + \ldots + d_n\delta^n$.

In case when the values of $p_1, \ldots, p_n$ are different let $p_i = 1/2 + \delta_i$ ($i = 1, \ldots, n$). The bias of the distribution of the function f output is defined in a similar way as the polynomial of n variables

$$\Delta_f(\delta_1, \ldots, \delta_n) = F_f\left(\frac{1}{2} + \delta_1, \ldots, \frac{1}{2} + \delta_n\right) - \frac{1}{2} \ . \tag{23}$$

If function f is balanced then the constant term of polynomial $\Delta_f(\delta_1, \ldots, \delta_n)$ is equal to

$$\Delta_f(0, \ldots, 0) = F_f\left(\frac{1}{2}, \ldots, \frac{1}{2}\right) - \frac{1}{2} = \frac{wt(f)}{2^n} - \frac{1}{2} = 0 \ .$$

And the other way around: if the constant term of polynomial $\Delta_f(\delta_1, \ldots, \delta_n)$ is equal to zero then function f is balanced. We will call polynomial $\Delta_f(\delta_1, \ldots, \delta_n)$ the *bias polynomial* of function f.

The coefficient vector of the bias polynomial is equal to the probabilistic transform of function f (see Sect. 2) except for the initial coordinate of Δ^f which has to be corrected by subtracting 1/2. On the other hand, combining (9) and (13), the coefficient vector can be expressed as $-\frac{1}{2^{n+1}}\begin{pmatrix} 1 & 0 \\ 0 & -2 \end{pmatrix}^{[n]} S^{\hat{f}}$. Coefficients of the bias polynomial can also be estimated using identities (14) that are equivalent to [12, Theorem 3.1].

Definition 3. *For $k \in \{1, \ldots, n\}$ a Boolean function f is called k-compensating if the bias polynomial of f does not contain product terms having degree lower than k.*

Note that any balanced Boolean function is 1-compensating. For the particular case when $p_1 = \ldots = p_n$, Definition 3 means that function f is k-compensating if it is balanced and $d_1 = \ldots = d_{k-1} = 0$. In other words, if the input of a k-compensating Boolean function is nonuniform with bias δ then the bias on its output is at most the size of order δ^k. The following proposition provides a method for constructing k-compensating functions.

Proposition 6. *Let* $f(x_1, \dots, x_n) = f_1(x_1, \dots, x_k) \oplus f_2(x_{k+1}, \dots, x_n)$*, where* $k \in \{1, \dots, n-1\}$*. Then*

$$F_f(p_1, \dots, p_n) - \frac{1}{2} = -2\left(F_{f_1}(p_1, \dots, p_k) - \frac{1}{2}\right)\left(F_{f_2}(p_{k+1}, \dots, p_n) - \frac{1}{2}\right) \ ,$$

i.e., $\Delta_f(\delta_1, \dots, \delta_n) = -2\Delta_{f_1}(\delta_1, \dots, \delta_k)\Delta_{f_2}(\delta_{k+1}, \dots, \delta_n)$.

Proof. Since $f_1 \oplus f_2 = f_1 + f_2 - 2f_1 f_2$,

$$\begin{aligned} D_f(x_1, \dots, x_n) = D_{f_1}(x_1, \dots, x_k) + D_{f_2}(x_{k+1}, \dots, x_n) - \\ - 2D_{f_1}(x_1, \dots, x_k)D_{f_2}(x_{k+1}, \dots, x_n) \ . \end{aligned}$$

Therefore, by Proposition 5,

$$\begin{aligned} F_f(p_1, \dots, p_n) = F_{f_1}(p_1, \dots, p_k) + F_{f_2}(p_{k+1}, \dots, p_n) - \\ - 2F_{f_1}(p_1, \dots, p_k)F_{f_2}(p_{k+1}, \dots, p_n) \ , \end{aligned}$$

which is equivalent to the statement of the proposition. □

The following corollary is obvious.

Corollary 2. *Let* $f(x_1, \dots, x_n)$ *be a Boolean function of* n *variables. If*

(i) $f(x_1, \dots, x_n) = f_1(x_1, \dots, x_k) \oplus f_2(x_{k+1}, \dots, x_n)$*, function* f_1 *is* k_1*-compensating and function* f_2 *is* k_2*-compensating then function* f *is* $(k_1 + k_2)$*-compensating;*

(ii) $f(x_1, \dots, x_n) = x_{i_1} \oplus \dots \oplus x_{i_k} \oplus a_0$ *then*

$$F_f(p_1, \dots, p_n) - \frac{1}{2} = (-1)^{a_0}(-2)^{k-1}\delta_{i_1} \cdot \dots \cdot \delta_{i_k} \ .$$

In other words, an affine function consisting of k *linear terms is* k*-compensating.*

The following proposition, that is similar to [12, Theorem 3.2], easily follows from (13) and Note 1. The proof provided further is not based on the previous results and is given to keep this section integral.

Proposition 7. *A Boolean function* $f(x_1, \dots, x_n)$ *is* k*-resilient if and only if it is* $(k+1)$*-compensating.*

Proof. For $k = 0$ the statement is obvious since a 0-resilient function is balanced by the definition and, therefore, it is 1-compensating. Further we assume that $k > 0$.

Let $P(X_i = 1) = p_i = 1/2 + \delta_i$ $(i = 1, \dots, n)$. By Definition 2,

$$\begin{aligned} &F_f(p_1, \dots, p_n) = P(f(\boldsymbol{X}) = 1) = \\ &= \sum_{(\beta_1, \dots, \beta_k) \in \mathrm{GF}(2)^k} p_1^{\beta_1} \cdot \dots \cdot p_k^{\beta_k} P(f(\beta_1, \dots, \beta_k, X_{k+1}, \dots, X_n) = 1) = \\ &= \sum_{(\beta_1, \dots, \beta_k) \in \mathrm{GF}(2)^k} p_1^{\beta_1} \cdot \dots \cdot p_k^{\beta_k} F_f(\beta_1, \dots, \beta_k, p_{k+1}, \dots, p_n) \ , \end{aligned} \tag{24}$$

where $p_i^{\beta_i} = \begin{cases} p_i, & \text{if} \quad \beta_i = 1, \\ 1 - p_i, & \text{if} \quad \beta_i = 0 \end{cases} \quad (i = 1, \ldots, n).$

Function $F_f(\beta_1, \ldots, \beta_k, p_{k+1}, \ldots, p_n)$ is a probabilistic function of subfunction $f^{\boldsymbol{\beta}} = f_{1,\ldots,k}^{\beta_1,\ldots,\beta_k}(x_{k+1}, \ldots, x_n)$, where $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_k)$, and

$$F_f(\beta_1, \ldots, \beta_k, p_{k+1}, \ldots, p_n) = \mathit{\Delta}_{f^{\boldsymbol{\beta}}}(\delta_{k+1}, \ldots, \delta_n) + \frac{1}{2} \ . \tag{25}$$

Therefore,

$$\begin{aligned} &F_f\left(\frac{1}{2} + \delta_1, \ldots, \frac{1}{2} + \delta_n\right) \overset{(24,25)}{=} \\ &= \frac{1}{2} \sum_{(\beta_1,\ldots,\beta_k)\in \mathrm{GF}(2)^k} \left(\frac{1}{2} + \delta_1\right)^{\beta_1} \cdot \ldots \cdot \left(\frac{1}{2} + \delta_k\right)^{\beta_k} + \\ &+ \sum_{(\beta_1,\ldots,\beta_k)\in \mathrm{GF}(2)^k} \left(\frac{1}{2} + \delta_1\right)^{\beta_1} \cdot \ldots \cdot \left(\frac{1}{2} + \delta_k\right)^{\beta_k} \mathit{\Delta}_{f^{\boldsymbol{\beta}}}(\delta_{k+1}, \ldots, \delta_n) = \\ &= \frac{1}{2} + \sum_{(\beta_1,\ldots,\beta_k)\in \mathrm{GF}(2)^k} \left(\frac{1}{2} + \delta_1\right)^{\beta_1} \cdot \ldots \cdot \left(\frac{1}{2} + \delta_k\right)^{\beta_k} \mathit{\Delta}_{f^{\boldsymbol{\beta}}}(\delta_{k+1}, \ldots, \delta_n) \ . \end{aligned} \tag{26}$$

Let us assume that function f is k-resilient. Then, by (19), its subfunction $f^{\boldsymbol{\beta}}$ is balanced for any $(\beta_1, \ldots, \beta_k) \in \mathrm{GF}(2)^k$ and for the probabilistic function of $f^{\boldsymbol{\beta}}$ holds $F_f(\beta_1, \ldots, \beta_k, 1/2, \ldots, 1/2) = 1/2$. Now, by (25), $\mathit{\Delta}_{f^{\boldsymbol{\beta}}}(0, \ldots, 0) = F_f(\beta_1, \ldots, \beta_k, 1/2, \ldots, 1/2) - 1/2 = 0$ and thus, the constant term of bias polynomial $\mathit{\Delta}_{f^{\boldsymbol{\beta}}}(\delta_{k+1}, \ldots, \delta_n)$ is equal to zero. If we look at function $F_f(1/2+\delta_1, \ldots, 1/2+\delta_n)$ as a polynomial of n variables then it is clear that all its product terms depend on at least one of the variables $\delta_{k+1}, \ldots, \delta_n$ and its constant term is equal to $1/2$.

Further, by (19), for a k-resilient function f any subfunction $f_{i_1,\ldots,i_k}^{\beta_{i_1},\ldots,\beta_{i_k}}$ with $1 \le i_1 < \ldots < i_k \le n$ and any $(\beta_{i_1}, \ldots, \beta_{i_k}) \in \mathrm{GF}(2)^k$ is balanced. In a similar way, it can be proved that all product terms in F_f depend on at least one of the variables contained in the subset $\{\delta_1, \ldots, \delta_n\} \setminus \{\delta_{i_1}, \ldots, \delta_{i_k}\}$. The minimal set containing representatives from all these subsets contains $k+1$ elements. Therefore, all product terms in F_f depend on at least $k+1$ variables. Thus, by (23), bias polynomial $\mathit{\Delta}_f(\delta_1, \ldots, \delta_n)$ does not contain product terms having degree lower than $k+1$.

Let us now assume that function f is $(k+1)$-compensating. Then, in particular, all product terms of polynomial $\mathit{\Delta}_f(\delta_1, \ldots, \delta_n)$ depend on at least one of the variables $\delta_{k+1}, \ldots, \delta_n$ and its constant term is equal to zero. Therefore,

$$\begin{aligned} &\mathit{\Delta}_f(\delta_1, \ldots, \delta_k, 0, \ldots, 0) = F_f\left(\frac{1}{2} + \delta_1, \ldots, \frac{1}{2} + \delta_k, \frac{1}{2}, \ldots, \frac{1}{2}\right) - \frac{1}{2} \overset{(26)}{=} \\ &= \sum_{(\beta_1,\ldots,\beta_k)\in \mathrm{GF}(2)^k} \left(\frac{1}{2} + \delta_1\right)^{\beta_1} \cdot \ldots \cdot \left(\frac{1}{2} + \delta_k\right)^{\beta_k} \mathit{\Delta}_{f^{\boldsymbol{\beta}}}(0, \ldots, 0) \equiv 0 \end{aligned}$$

and polynomial $\Delta_f(\delta_1, \ldots, \delta_k, 0, \ldots, 0)$ is identically equal to zero. It is easy to see that the coefficient of the multiple term $\delta_{i_1} \cdot \ldots \cdot \delta_{i_t}$ in the canonical form of this polynomial is equal to

$$\frac{1}{2^{k-t}} \sum_{(\beta_1,\ldots,\beta_k)\in \mathrm{GF}(2)^k} (-1)^{t-(\beta_{i_1}+\ldots+\beta_{i_t})} \Delta_{f^{\beta}}(0, \ldots, 0) = 0 \ ,$$

that should hold for any $0 \leq t \leq k$ and $0 \leq i_1 < \ldots < i_t \leq k$. Thus, we have got the system of 2^k linear equations of 2^k unknown $\Delta_{f^{\beta}}(0, \ldots, 0)$ with the matrix consisting of elements $m_{i,j} = (-1)^{wt(\alpha_i)-\langle \alpha_i, \alpha_j \rangle}$ for $i, j = 0, \ldots, 2^k - 1$, where α_i and α_j are k-bit binary expansions of i and j respectively. This matrix is a Hadamard matrix [5, p. 44] and it is nondegenerate. Therefore, this system has the unique zero solution and $\Delta_{f^{\beta}}(0, \ldots, 0) = 0$ for any $(\beta_1, \ldots, \beta_k) \in \mathrm{GF}(2)^k$. Thus, subfunctions $f^{\boldsymbol{\beta}}$ are balanced. In a similar way, it can be proved that any subfunction $f_{i_1,\ldots,i_k}^{\beta_{i_1},\ldots,\beta_{i_k}}$ with $1 \leq i_1 < \ldots < i_k \leq n$ and any $(\beta_{i_1}, \ldots, \beta_{i_k}) \in \mathrm{GF}(2)^k$ is balanced. Then, by (19), function f is k-resilient. $\square$

Therefore, highly resilient Boolean functions significantly increase the size of order for the bias of the distribution of the output bits compared to the bias of the input. On the other hand, due to Siegenthaler's inequality, Proposition 7 means that Boolean functions with high algebraic degree have poor compensating properties and vice versa. This fact underlines again the need for optimizing algebraic degree with correlation and compensating properties when constructing secure Boolean functions.

5 Conclusion

The classical algebraic normal and Walsh transforms appear to be a special case of the tensor transform that also allows for the definition of new transforms, in particular, probabilistic and weight transforms. The new transforms are cryptographically important since they relate a Boolean function directly to its bias polynomial and to the weights of its subfunctions. Easy proofs for some known and new properties of algebraic normal and Walsh transforms can be given basing on general properties of the tensor transform. A tensor transform is based on the Kronecker product of appropriate elementary cells. This fact allows to use fast Fourier and Walsh transform algorithms for efficient estimation of any tensor transform and easy transition from one transform to another.

The requirement for a cryptographically secure Boolean function to be correlation immune can be weakened without undermining security if only slight dependencies between input bits and the output are allowed. Correlation coefficients provide an estimate for correlation dependencies and can be obtained from the weight transform of a Boolean function. The number of $(n - m)$th-order product terms in the ANF of a Boolean function f is directly related to the number of subfunctions obtained by fixing m variables of f with zero values and having an even weight. By increasing resiliency order of a Boolean function, approximations of this function nondegenerate on few variables are eliminated.

The probabilistic function allows to estimate the probabilistic distribution of bits at the output of a Boolean function if the distribution of the arguments, the function depends on, is known. The probabilistic function is a polynomial which coefficients can be obtained by the algebraic normal transform of a Boolean function over $\mathbb{R}$. The newly introduced measure for a Boolean function to compensate a nonuniform distribution of its input bits was called the compensating degree. Compensating degree can be efficiently estimated by the probabilistic transform. Highly resilient Boolean functions significantly increase the size of order for the bias of the distribution of the output bits compared to the bias of the input. However, correlation and compensating properties need to be optimized with the algebraic degree when constructing secure Boolean functions.

References

1. Rueppel, R.A.: Analysis and Design of Stream Ciphers. Communications and Control Engineering Series. Springer-Verlag, Berlin (1986)
2. Siegenthaler, T.: Decrypting a class of stream ciphers using ciphertext only. IEEE Transactions on Computers **C-34** (1985) 81–85
3. Ding, C., Xiao, G.Z., Shan, W.: The Stability Theory of Stream Ciphers. Volume 561 of Lecture Notes in Computer Science. Springer-Verlag, Berlin (1991)
4. Xiao, G.Z., Massey, J.L.: A spectral characterization of correlation-immune combining functions. IEEE Transactions on Information Theory **34** (1988) 569–571
5. MacWilliams, F.J., Sloane, N.J.A.: The Theory of Error-Correcting Codes. Volume 16 of North-Holland Mathematical Library. North-Holland, Amsterdam (1996) Ninth impression.
6. Sarkar, P.: Spectral domain analysis of correlation immune and resilient Boolean functions. Cryptology ePrint Archive, Report 2000/049 (2000) http://eprint.iacr.org/2000/049/. Revised version published in [13].
7. Forré, R.: The strict avalanche criterion: Spectral properties of Boolean functions and an extended definition. In Goldwasser, S., ed.: Advances in Cryptology - Crypto '88. Volume 403 of Lecture Notes in Computer Science., Berlin, Springer-Verlag (1989) 450–468
8. Canteaut, A., Filiol, E.: Ciphertext only reconstruction of stream ciphers based on combination generators. In Schneier, B., ed.: Fast Software Encryption, 7th International Workshop, FSE 2000. Volume 1978 of Lecture Notes in Computer Science., Berlin, Springer-Verlag (2001) 165–180
9. Aho, A.V., Hopcroft, J.E., Ullman, J.D.: The Design and Analysis of Computer Algorithms. Addison-Wesley Series in Computer Science and Information Processing. Addison-Wesley, Amsterdam (1974)
10. Beauchamp, K.: Applications of Walsh and Related Functions (with an Introduction to Sequency Theory). Microelectronics and Signal Processing. Academic Press, London (1984)
11. Siegenthaler, T.: Correlation-immunity of nonlinear combining functions for cryptographic applications. IEEE Transactions on Information Theory **IT-30** (1984) 776–780
12. Miranovich, K.: Spectral analysis of Boolean functions under non-uniformity of arguments. Cryptology ePrint Archive, Report 2002/021 (2002) http://eprint.iacr.org/2002/021/.

13. Carlet, C., Sarkar, P.: Spectral domain analysis of correlation immune and resilient Boolean functions. Finite Fields and Their Applications **8** (2002) 120–130