Tools in Analyzing Linear Approximation for Boolean Functions Related to FLIP

Subhamoy Maitra[1], Bimal Mandal[1], Thor Martinsen[2], Dibyendu Roy[3(✉)], and Pantelimon Stănică[2]

[1] Indian Statistical Institute, Kolkata, India
subho@isical.ac.in, bimalmandal90@gmail.com
[2] Naval Postgraduate School, Monterey, USA
{tmartins,pstanica}@nps.edu
[3] National Institute of Science Education and Research (HBNI), Bhubaneswar, India
roydibyendu.rd@gmail.com

Abstract. For cryptographic purposes, we generally study the characteristics of a Boolean function in n-variables with the inherent assumption that each of the n-bit inputs take the value 0 or 1, independently and randomly with probability 1/2. However, in the context of the FLIP stream cipher proposed by Méaux et al. (Eurocrypt 2016), this type of analysis warrants a different approach. To this end, Carlet et al. (IACR Trans. Symm. Crypto. 2018) recently presented a detailed analysis of Boolean functions with restricted inputs (mostly considering inputs with weight $\frac{n}{2}$) and provided certain bounds on linear approximation, which are related to restricted nonlinearity. The Boolean function used in the FLIP cipher reveals that it is actually a direct sum of several Boolean functions on a small number of inputs. Thus, with a different approach, we start a study in order to understand how the inputs to the composite function are distributed on the smaller functions. In this direction, we obtain several results that summarize the exact biases related to such Boolean functions. Finally, for the nonlinear filter function of FLIP, we obtain the lower bound on the restricted Walsh–Hadamard transform (i.e., upper bound on restricted nonlinearity). Our techniques provide a general theoretical framework to study such functions and better than previously published estimations of the biases, which is directly linked to the security parameters of the stream cipher.

Keywords: Bias · Boolean function · FLIP
Homomorphic encryption · Restricted domain · Stream cipher

1 Introduction

The search for practical solutions to efficient homomorphic encryption schemes, ushered in a new paradigm in stream cipher design, and received serious attention, recently. One important step in this direction has appeared in [1]. Shortly thereafter, the papers [5,6] started analyzing the constituent Boolean function(s)

D. Chakraborty and T. Iwata (Eds.): INDOCRYPT 2018, LNCS 11356, pp. 282–303, 2018.
https://doi.org/10.1007/978-3-030-05378-9_16

in the FLIP stream cipher. An initial version of this cipher was cryptanalyzed in [3]. In the FLIP stream cipher, the keystream bit is computed by using one nonlinear filter function, which takes input from a restricted domain. Recently, in [2,7], the properties of the Boolean functions in such a restricted domain [6] were studied in detail.

In this paper we consider a different approach. It is evident that for the implementation of efficient homomorphic encryption schemes, the underlying stream cipher must be simple. This requires Boolean functions with simple Algebraic Normal Form (ANF) having many linear and low degree terms connected by simple $\mathbb{F}_2$ addition. Further, the existing Boolean functions in the FLIP cipher require each variable to be part of only one subfunction in the ANF. Given such restrictions, it is evident that such functions will not have good cryptographic properties. Thus, one requires a large number of variables, and consequently, we want to get the required security with the least possible number of inputs. The study of such functions is much easier using the standard Walsh–Hadamard transform if we consider that the inputs appear independently and uniformly at random. However, this is not the case here, since only the inputs of a specific weight play a role. Quite involved mathematical techniques have been exploited in [2,7] to study such functions. The analysis of FLIP, as a consequence of these works, requires more attention, as specific numerical bounds on both sides are not available.

Let us now discuss the issue from a more technical viewpoint (we refer to the notations later in this section). Carlet et al. [2] observed that different properties of a Boolean function F defined over $\mathbb{F}_2^n$ degrade significantly when the inputs come from a restricted subset $E \subset \mathbb{F}_2^n$. In the case of the FLIP stream cipher, the inputs of the nonlinear filter function remain a 0/1 string of length n with weight $\frac{n}{2}$ for all rounds. So, the nonlinear filter function always takes input from a restricted subset $E \subset \mathbb{F}_2^n$. Based on this observation, Carlet et al. [2] studied several properties of a Boolean function in a restricted domain. Mesnager et al. [7] further analyzed Boolean functions on restricted domains and proposed a lower bound of the bias, although, the numerical computation of the upper bound of the bias is practically not possible by that technique. The papers [2,7] consider the properties of the complete function F (see Sect. 1.2), given that the input is of fixed weight.

In this paper, we concentrate on this issue and first notice that if $\mathbf{x} = \mathbf{x}_1||\mathbf{x}_2||\ldots||\mathbf{x}_n$ (concatenation) and $\mathbf{x} \in E_{n,k}$ (the definition of $E_{n,k}$ is provided in Sect. 1.1) then $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$ does not follow a uniform distribution. This observation motivates us to study the restricted Walsh–Hadamard transform by considering the exact probability distribution of $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$. In fact, it is worth mentioning now that if the input $\mathbf{x}_i$ of a Boolean function f_i does not follow a uniform distribution, then the original properties of f_i (assuming uniform distribution) changes significantly. Further, by considering the actual distribution of the input (rather than a uniform one), we expect to achieve a tighter bound for the bias given the nonlinear filter function used in the FLIP stream cipher. Naturally a tighter bound will provide much better approximation for the security parameters of the FLIP cipher.

Contribution and Organization. Our approach considers how the inputs to the composite function are distributed on the smaller functions. In this direction, we present some tools to start our analysis in Sect. 2. Then our main motivation is to obtain more accurate linear approximations of nonlinear Boolean functions when the inputs are restricted, which we discuss in Sect. 3. However, the formulae that we arrive at are quite complicated to be directly compared with equally complicated expressions of [2,7]. Thus, in this direction, numerical data will provide better understanding of these results, as we discuss in Sect. 4. For that we refer to the $n = 530$ variable Boolean function that has been considered in [2]. Straightforward analysis of the Walsh–Hadamard spectrum shows that when we consider that the inputs are uniform, such a function has maximum absolute Walsh–Hadamard transform value in $[2^{-79}, 2^{-78}]$. Thus, the bias to a linear function looks quite low. However, our analysis shows that when the inputs are taken of weight $\frac{n}{2} = 265$, then the restricted Walsh–Hadamard transform is much higher. The maximum absolute value is in $[2^{-18.49}, 2^{-13.59}]$. We obtain the upper bound by considering the idea of [2] and the lower bound is obtained from our detailed analysis in this paper. That is, our work complements the work of [2] to bound the maximum absolute restricted Walsh–Hadamard transform value of a function on large number of variables used in the FLIP stream cipher.

Before proceeding further let us present some background material.

1.1 Boolean Functions

Let $\mathbb{F}_2$ and $\mathbb{F}_2^n$ be the prime binary field, respectively, the extension field over $\mathbb{F}_2$ of degree n. Let $\mathbb{F}_2^n = \{\mathbf{x} = (x_1, x_2, \ldots, x_n) : \ x_i \in \mathbb{F}_2, \text{for all} \, 1 \leq i \leq n\}$ be the vector space over $\mathbb{F}_2$ of dimension n. We denote the concatenation of two (or more) binary strings $\mathbf{x}', \mathbf{x}''$ by $\mathbf{x}'||\mathbf{x}''$. The cardinality of a set S is denoted by $|S|$. Any function $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$ is said to be a Boolean function in n-variables, whose set is denoted by $\mathcal{B}_n$. These functions can be represented in a unique way (called the *Algebraic Normal Form* (ANF) of f) as

$$f(\mathbf{x}) = \sum_{\mathbf{a} \in \mathbb{F}_2^n} \mu_{\mathbf{a}} \left(\prod_{i=1}^{n} x_i^{a_i} \right), \text{for all} \, \mathbf{x} \in \mathbb{F}_2^n, \text{where} \, \mu_{\mathbf{a}} \in \mathbb{F}_2.$$

The *Hamming weight* of $\mathbf{x} \in \mathbb{F}_2^n$ is defined as $wt(\mathbf{x}) = \sum_{i=1}^{n} x_i$, where the sum is over $\mathbb{Z}$, the ring of integers. The *algebraic degree* of a Boolean function $f \in \mathcal{B}_n$ is defined as $\deg(f) = \max_{\mathbf{a} \in \mathbb{F}_2^n} \{wt(\mathbf{a}) : \ \mu_{\mathbf{a}} \neq 0\}$. Let $E_{n,i} = \{\mathbf{x} \in \mathbb{F}_2^n : \ wt(\mathbf{x}) = i\}$, for all $0 \leq i \leq n$. The *support* of $f \in \mathcal{B}_n$ is defined as $\text{supp}(f) = \{\mathbf{x} \in \mathbb{F}_2^n : \ f(\mathbf{x}) = 1\}$. A Boolean function is said to be *balanced* if the cardinality of its support set is $|\text{supp}(f)| = 2^{n-1}$. If the algebraic degree of a Boolean function $f \in \mathcal{B}_n$ is at most 1 then f is an affine function, and its set is $\mathcal{A}_n = \{l_{\mathbf{a},\varepsilon} : \ \mathbf{a} \in \mathbb{F}_2^n, \varepsilon \in \mathbb{F}_2\}$, where $l_{\mathbf{a},\varepsilon}(\mathbf{x}) = \mathbf{a} \cdot \mathbf{x} + \varepsilon$, for all $\mathbf{x} \in \mathbb{F}_2^n$. If $\varepsilon = 0$, then $l_{\mathbf{a},0}$ is a linear function. The *Hamming distance* between any $f, g \in \mathcal{B}_n$ is

defined by $d_H(f,g) = |\{\mathbf{x} \in \mathbb{F}_2^n : f(\mathbf{x}) \neq g(\mathbf{x})\}|$. The *correlation* between two Boolean functions $f, g \in \mathcal{B}_n$ is defined by

$$\mathsf{corr}(f,g) = \left| \frac{|\{\mathbf{x} : f(\mathbf{x}) = g(\mathbf{x})\}| - |\{\mathbf{x} : f(\mathbf{x}) \neq g(\mathbf{x})\}|}{2^n} \right|.$$

To measure the correlation between an n-variable Boolean function f and a linear function $l_{\mathbf{a},0}$, we use the *Walsh–Hadamard transform*, defined by

$$\mathcal{W}_f(\mathbf{a}) = \frac{1}{2^n} \sum_{\mathbf{x} \in \mathbb{F}_2^n} (-1)^{f(\mathbf{x}) + \mathbf{a} \cdot \mathbf{x}}.$$

We observe that the absolute value of the Walsh–Hadamard transform of $f \in \mathcal{B}_n$ at a fixed point $\mathbf{a} \in \mathbb{F}_2^n$ provides us the correlation between the Boolean function f and the linear function $l_{\mathbf{a},0}$, i.e., $\mathsf{corr}(f, l_{\mathbf{a},0}) = |\mathcal{W}_f(\mathbf{a})|$, for all $\mathbf{a} \in \mathbb{F}_2^n$. The multiset $[\mathcal{W}_f(\mathbf{a}) : \mathbf{a} \in \mathbb{F}_2^n]$, which is the *Walsh–Hadamard spectrum* of f, provides us the correlation between the Boolean function f and all possible linear functions. From the Parseval's identity for arbitrary $f \in \mathcal{B}_n$,

$$\sum_{\mathbf{a} \in \mathbb{F}_2^n} \mathcal{W}_f(\mathbf{a})^2 = 1,$$

we obtain $\max_{\mathbf{a} \in \mathbb{F}_2^n} |\mathcal{W}_f(\mathbf{a})| \geq \frac{1}{2^{n/2}}$.

A Boolean function $f \in \mathcal{B}_n$ (n even) is said to be bent if and only if the correlation between f and $\{l_{\mathbf{a},0} | \mathbf{a} \in \mathbb{F}_2^n\}$ is $\frac{1}{2^{n/2}}$, i.e., $\mathsf{corr}(f, l_{\mathbf{a},0}) = |\mathcal{W}_f(\mathbf{a})| = 2^{-\frac{n}{2}}$, for all $\mathbf{a} \in \mathbb{F}_2^n$.

Now if we assume that an n-variable Boolean function $f \in \mathcal{B}_n$ takes input from a restricted domain, then to calculate the correlation between f and a linear function $l_{\mathbf{a},0}$, we need to consider the inputs $\mathbf{x}$ only from a restricted domain. Here we assume that f takes inputs of weight k, i.e., $\mathbf{x} \in E_{n,k} := \{\mathbf{x} \in \mathbb{F}_2^n : wt(\mathbf{x}) = k\}$. Certainly, $|E_{n,k}| = \binom{n}{k}$. Under this assumption, the (restricted domain) correlation between the Boolean function f and a linear function $l_{\mathbf{a},0}$ is

$$\mathsf{corr}^{(k)}(f, l_{\mathbf{a},0}) = \left| \frac{|\{\mathbf{x} : f(\mathbf{x}) = l_{\mathbf{a},0}(\mathbf{x})\}| - |\{\mathbf{x} : f(\mathbf{x}) \neq l_{\mathbf{a},0}(\mathbf{x})\}|}{|E_{n,k}|} \right|.$$

Further, to calculate this correlation, we shall define the Walsh–Hadamard transform $\mathcal{W}_f^{(k)}(\mathbf{a})$ of a Boolean function f in a restricted domain $E_{n,k}$, $0 \leq k \leq n$, by

$$\mathcal{W}_f^{(k)}(\mathbf{a}) = \frac{1}{|E_{n,k}|} \sum_{\mathbf{x} \in E_{n,k}} (-1)^{f(\mathbf{x}) + \mathbf{a} \cdot \mathbf{x}}.$$

Here we define two more notations, which are used throughout the article.

Definition 1. *Let* $\mathbf{x} = (x_1, \ldots, x_n) \in E_{n,k}$ *and* $n = n_1 + n_2$, *and* $\mathbf{x}' = (x_1, \ldots, x_{n_1})$, $\mathbf{x}'' = (x_{n_1+1}, \ldots, x_n)$. *Then* $E_{n_1,i}^{n=n_1+n_2,k} = \{\mathbf{x}' \in \mathbb{F}_2^{n_1} \mid \mathbf{x} \in E_{n,k}, n = n_1 + n_2$ *and* $wt(\mathbf{x}') = i\}$ *and* $E_{n_2,j}^{n=n_1+n_2,k} = \{\mathbf{x}'' \in \mathbb{F}_2^{n_2} \mid \mathbf{x} \in E_{n,k}, n = n_1 + n_2$ *and* $wt(\mathbf{x}'') = j\}$.

Certainly, we can continue the splitting process, and if $n = n_1 + n_2 + n_3$ then $E_{n_1,i}^{n=n_1+n_2+n_3,k}$, $E_{n_2,j}^{n=n_1+n_2+n_3,k}$ and $E_{n_3,r}^{n=n_1+n_2+n_3,k}$ can be inferred from the above definition. More generally it can be extended to $n = n_1 + n_2 + \cdots + n_q$.

1.2 Design Specification of the FLIP Stream Cipher

In this section we describe the design specification of the FLIP family of stream ciphers (initially, presented in [5]). The main motivation behind this proposal was to construct a fully homomorphic encryption (FHE) scheme with the limited error growth using a symmetric key primitive. Since block ciphers are based on complicated round functions, it seems to be difficult to construct such a FHE. After this proposal, Duval et al. [3] came up with an attack on the FLIP ciphers. Shortly thereafter, Méaux et al. [6] modified the design specification of FLIP and proposed the final modified version of the FLIP stream cipher.

The FLIP cipher is based on three components: one register of length n, one pseudorandom number generator (PRNG), one nonlinear filter function F involving n-variables.

The cipher stores the secret key K of length n into the register and a PRNG is initialized with the initialization vector IV. In each clock, the PRNG generates a number which corresponds to a permutation. This pseudorandom permutation permutes the state bits of the register. Finally, the nonlinear filter function takes the current state as input to generate keystream bits. The pictorial description of the FLIP stream cipher is described in Fig. 1.

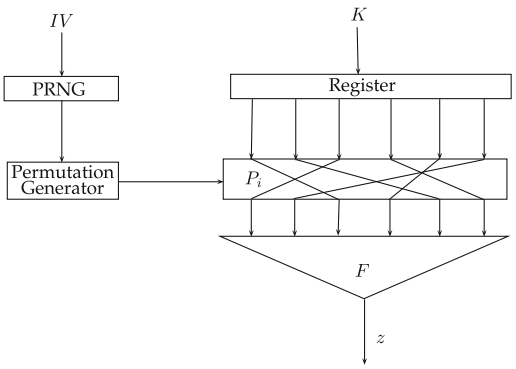


Fig. 1. Design specification of FLIP

The nonlinear filter function $F = f_1 + f_2 + f_3$ has three component functions: f_1 is a linear function, f_2 is a quadratic bent function and f_3 is a special type of triangular function. The ANFs of these functions are described below:

– **L-type function.** A Boolean function L_n in n-variables is said to be of L-type if it is of the following form $L_n(x_0, x_1, \ldots, x_{n-1}) = \sum_{i=0}^{n-1} x_i$.

- **Q-type function.** The algebraic normal form of the Q-type bent function Q_{2n} in $2n$-variables is $Q_{2n}(x_0, x_1, \ldots, x_{2n-1}) = \sum_{i=0}^{n-1} x_{2i}x_{2i+1}$.
- **T-type function.** For a positive integer n, the algebraic normal form of the n-th T-type triangular function T_n in $\frac{n(n+1)}{2}$ variables is $T_n(x_0, x_1, \ldots, x_{\frac{n(n+1)}{2}-1}) = \sum_{i=1}^{n} \prod_{j=0}^{i-1} x_{j+\sum_{\ell=0}^{i-1}\ell}$.

Thus, the nonlinear filter function F in n-variables is a direct sum of three Boolean functions f_1, f_2 and f_3 involving n_1, n_2 and n_3 variables (such that $n = n_1+n_2+n_3$), respectively, where the algebraic normal form of these functions are as follows:

- $f_1(x_0, x_1, \ldots, x_{n_1-1}) = L_{n_1}(x_0, x_1, \ldots, x_{n_1-1})$.
- $f_2(x_{n_1}, x_{n_1+1}, \ldots, x_{n_1+n_2-1}) = Q_{n_2}(x_{n_1}, x_{n_1+1}, \ldots, x_{n_1+n_2-1})$.
- $f_3(x_{n_1+n_2}, x_{n_1+n_2+1}, \ldots, x_{n_1+n_2+n_3-1})$ is the direct sum of r triangular function T_k, where each T_k involves independent variables.

The final algebraic normal form of the nonlinear filter function F is

$$F = L_{n_1} + Q_{n_2} + \sum_{i=1}^{r} T_k.$$

The function that we concentrate on in this paper is the one with the notation FLIP(42, 128, ${}^8\Delta^9$) as described in [6]. This means that $n_1 = 42, n_2 = 2 \cdot 64 = 128, n_3 = 8 \cdot (1 + 2 + \cdots + 9) = 360$. That is there are 42 terms in the linear functions (L-type), 64 many quadratic terms in the quadratic functions (Q-type) and further there are eight T-type functions each having terms of degree 1 to 9, i.e., each one having 45 many variables.

The designers of the FLIP stream ciphers suggested that the weight of the secret key of length n must be $\frac{n}{2}$. In each round, one pseudorandom permutation is applied on the register, which permutes the index of the secret key bits. The nonlinear filter function F takes the updated state of the register as input to produce an output bit. It is then clear that the weight of the state of the cipher in each round remains fixed (i.e., $\frac{n}{2}$). From the expression of the keystream we can formally write $F(S_n^t) = z_t$, where $wt(S_n^t) = \frac{n}{2}$.

At Crypto 2016, Duval et al. [3] proposed an attack on the old version of the FLIP stream cipher as introduced in [5]. The attack complexities for two instances of FLIP, namely, $n = 192$ ($n_1 = 47, n_2 = 40, n_3 = 105$) and $n = 400$ ($n_1 = 87, n_2 = 82, n_3 = 231$), are 2^{54}, respectively, 2^{68}. The previously described modified design has then been proposed by Méaux et al. [6] to counter this attack.

2 Tools for Our Analysis

Here we first review the existing techniques and then move forward to our new ideas. One may note that given the simple structure of the Boolean function in the FLIP cipher, it is not hard to study the nonlinearity under the framework of Walsh–Hadamard transform. One can easily verify that, in uniform domain $2^{-79} < \max_{\mathbf{a}\in\mathbb{F}_2^{530}} |\mathcal{W}_f(\mathbf{a})| < 2^{-78}$.

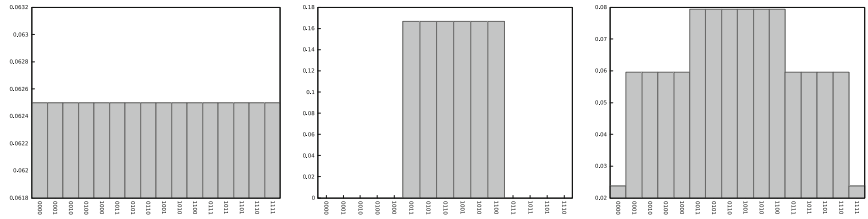
However, the scenario is completely different in restricted domain. We will actually see at the end of this paper, due to the restriction on inputs, this maximum absolute restricted Walsh–Hadamard spectrum value is indeed much higher, which is $\geq \frac{1}{2^{18.49}}$. Following the work of [2] it can be calculated that this is also less than $\frac{1}{2^{13.59}}$. Thus, the bound obtained by simple Walsh–Hadamard transform does not provide the actual picture and it is indeed much higher in FLIP stream cipher when restricted inputs are considered.

2.1 Our Idea: Frequency Distribution of Concatenated Sub-strings of a Fixed Weight Bit String

Recall that each element $\mathbf{x} = (x_1, x_2, \ldots, x_n) \in E_{n,k}$ is an n bit binary string with weight $wt(\mathbf{x}) = k$. In $\mathbf{x}$, if we consider the first n_1 components x_i's, then the weight distributions may not be uniform. This may affect different cryptographic properties of a Boolean function defined over the first n_1 number of variables of $\mathbf{x} \in E_{n,k}$. Carlet et al. [2] did not consider this issue, although they studied several properties of the complete function $F = f_1 + f_2 + f_3$, when the input is restricted to a set. As the nonlinear filter function in the FLIP stream cipher is $F = f_1 + f_2 + f_3$, we need to study the individual functions f_1, f_2 and f_3 by considering the weight distribution of inputs for each of these functions.

For $n = n_1 + n_2 + n_3$ and $\mathbf{x} \in \mathbb{F}_2^n$, we write $\mathbf{x} = \mathbf{x}'||\mathbf{x}''||\mathbf{x}'''$, where $\mathbf{x}' \in \mathbb{F}_2^{n_1}$, $\mathbf{x}'' \in \mathbb{F}_2^{n_2}$ and $\mathbf{x}''' \in \mathbb{F}_2^{n_3}$ with $Pr(\mathbf{x}) = \frac{1}{2^n}$, representing the probability of picking any element $\mathbf{x} \in \mathbb{F}_2^n$. The cardinality of $E_{n,k}$ is equal to $\binom{n}{k}$, $0 \leq k \leq n$, which follows the normal distribution (when n approaches to ∞). Also, if we consider the first n_1 bits of $\mathbb{F}_2^n$, then all elements belonging to $\mathbb{F}_2^{n_1}$ of whatever weight distribution will follow the normal distribution. However, fixing k, that is, by considering only one set $E_{n,k}$, the cardinality of $E_{n_1,i}^{n=n_1+n_2+n_3,k}$, $0 \leq i \leq n_1$, does not follow the normal distribution.

For example, let $n = 4$ and $n_1 = 2$. Then $E_{4,i}$, $0 \leq i \leq 4$, satisfy the normal distribution, as well as $E_{2,j}^{4=2+2,i}$, $0 \leq j \leq 2$, as $|E_{2,0}^{4=2+2,i}| = 4 = |E_{2,2}^{4=2+2,i}| = |E_{2,1}^{4=2+2,i}|/2$. ($|E_{2,1}^{4=2+2,i}| = 8$, as for each one weight element of length two there are four possibilities in the last two bits for $0 < i < 4$.) Let us consider only the set $E_{4,2}$ of cardinality 6 and, if we consider the first two bits then $|E_{2,0}^{4=2+2,2}| = 1$, $|E_{2,1}^{4=2+2,2}| = 4$ and $|E_{2,2}^{4=2+2,2}| = 1$, then $Pr(\mathbf{x}' = 00) = \frac{1}{6} = Pr(\mathbf{x}' = 11)$, $Pr(\mathbf{x}' = 01) = \frac{1}{3} = Pr(\mathbf{x}' = 10)$. The probability distribution is provided in Fig. 2.



(a) All weight uniform case (b) Weight 2 uniform case (c) For restricted case

Fig. 2. Probability distributions

Let P be any permutation of the set $\{x_1, x_2, \ldots, x_n\}$. Then $P(\mathbf{x}) \in E_{n,k}$, for all $\mathbf{x} \in E_{n,k}$. In the rest of the paper, we consider $\mathbf{x} = (x_1, x_2, \ldots, x_n) \in E_{n,\frac{n}{2}}$ and $\mathbf{x}' = (y_1, y_2, \ldots, y_{n_1}) \in \mathbb{F}_2^{n_1}$ where $y_i = x_i$, $1 \leq i \leq n_1$. We now calculate the frequency distributions of $\mathbf{x}'$'s with respect to their weights.

Case 1: Let $n_1 = \frac{n}{2}$. Then all possible elements $\mathbf{x}'$ of $\mathbb{F}_2^{n_1}$ exist. We observe that there are $\binom{\frac{n}{2}}{i}\binom{\frac{n}{2}}{\frac{n}{2}-i}$ elements $\mathbf{x}'$ such that $wt(\mathbf{x}') = i$, $0 \leq i \leq \frac{n}{2}$ and each bit pattern of the same weight will occur an equal number of times.
Case 2: Let $n_1 < \frac{n}{2}$. Again all possible elements $\mathbf{x}'$ of $\mathbb{F}_2^{n_1}$ exist. We observe that there are $\binom{n_1}{i}\binom{n-n_1}{\frac{n}{2}-i}$ elements $\mathbf{x}' \in \mathbb{F}_2^{n_1}$ such that $wt(\mathbf{x}') = i$; $0 \leq i \leq n_1$.
Case 3: Let $n_1 > \frac{n}{2}$. Now we find the number of possible $\mathbf{x}'$ with weight $wt(\mathbf{x}') = i$, $n_1 - \frac{n}{2} \leq i \leq \frac{n}{2}$. We observe that the cardinality $\left|\{\mathbf{x}' \in \mathbb{F}_2^{n_1} |\ wt(\mathbf{x}') = \frac{n}{2}\}\right| = \binom{n_1}{\frac{n}{2}}$, where every such element occurs exactly once. In general, for each

Table 1. Frequency distribution of $\mathbb{F}_2^{n_1}$ for $n_1 = 2, 3$ and 4

x_2x_1	Frequency
00	4
01	6
10	6
11	4

$x_3x_2x_1$	Frequency
000	1
001	3
010	3
011	3
100	3
101	3
110	3
111	1

$x_4x_3x_2x_1$	frequency
0000	0
0001	1
0010	1
0011	2
0100	1
0101	2
0110	2
0111	1
1000	1
1001	2
1010	2
1011	1
1100	2
1101	1
1110	1
1111	0

i, $0 \le i \le n - n_1$, such that $wt(\mathbf{x}') = \frac{n}{2} - i$, then $\mathbf{x}'$ occurs $\binom{n_1}{\frac{n}{2}-i}\binom{n-n_1}{i}$ times $(0 \le i \le n - n_1)$.

For example, let $n = 6$ and $\mathbf{x} \in E_{6,3}$, where $|E_{6,3}| = 20$. In Table 1, we display the frequency of occurrence of each element in $\mathbb{F}_2^{n_1}$ for $n_1 = 2, 3$ and 4.

In the remainder of the paper, we consider $n_i < n$, $1 \le i \le 3$, and the weight of the input $\mathbf{x} \in \mathbb{F}_2^n$ is equal to $\frac{n}{2}$.

3 Biased Walsh–Hadamard Transform

We define the Walsh–Hadamard transform of a Boolean function when the input elements have different probabilities, not necessarily uniform. We shall call this transform, a biased Walsh–Hadamard transform of a Boolean function (see [4], for yet another definition), which is the same as the bias between a Boolean function and a linear function over a non-uniform domain. If the input to a Boolean function does not follow the uniform distribution, several properties of the function change significantly.

Let $p(\mathbf{a})$ be the probability of an input element $\mathbf{a} \in \mathbb{F}_2^n$ in $f \in \mathcal{B}_n$. Recall that $0 \le p(\mathbf{a}) \le 1$, for all $\mathbf{a} \in \mathbb{F}_2^n$, and $\sum_{\mathbf{a}\in\mathbb{F}_2^n} p(\mathbf{a}) = 1$. For any $f, g \in \mathcal{B}_n$, we let $\mathcal{S}(f,g) = \{\mathbf{x} \in \mathbb{F}_2^n : \ f(\mathbf{x}) \ne g(\mathbf{x})\}$ and $\bar{\mathcal{S}}(f,g) = \mathbb{F}_2^n \setminus \mathcal{S}(f,g) = \{\mathbf{x} \in \mathbb{F}_2^n : \ f(\mathbf{x}) = g(\mathbf{x})\}$.

The *biased Hamming distance*, $d_H^B(f,g)$, between two Boolean functions $f, g \in \mathcal{B}_n$, when the inputs are not uniformly distributed, is defined by $d_H^B(f,g) = \sum_{\mathbf{x}\in\mathcal{S}(f,g)} p(\mathbf{x})$. Further, the *biased Hamming distance* between two Boolean functions $f, g \in \mathcal{B}_n$ is

$$\begin{aligned} d_H^B(f,g) &= \frac{1}{2}\left\{\sum_{\mathbf{x}\in\bar{\mathcal{S}}(f,g)} p(\mathbf{x}) + \sum_{\mathbf{x}\in\mathcal{S}(f,g)} p(\mathbf{x})\right\} - \frac{1}{2}\left\{\sum_{\mathbf{x}\in\bar{\mathcal{S}}(f,g)} p(\mathbf{x}) - \sum_{\mathbf{x}\in\mathcal{S}(f,g)} p(\mathbf{x})\right\} \\ &= \frac{1}{2} - \frac{1}{2}\sum_{\mathbf{x}\in\mathbb{F}_2^n} p(\mathbf{x})(-1)^{f(\mathbf{x})+g(\mathbf{x})}. \end{aligned}$$

In particular, $d_H^B(f, l_{\mathbf{a},\varepsilon}) = \frac{1}{2} - \frac{(-1)^\varepsilon}{2}\sum_{\mathbf{x}\in\mathbb{F}_2^n} p(\mathbf{x})(-1)^{f(\mathbf{x})+\mathbf{a}\cdot\mathbf{x}} = \frac{1}{2} - \frac{(-1)^\varepsilon}{2}\mathcal{W}_f^B(\mathbf{a})$, where $\mathcal{W}_f^B(\mathbf{a}) = \sum_{\mathbf{x}\in\mathbb{F}_2^n} p(\mathbf{x})(-1)^{f(\mathbf{x})+\mathbf{a}\cdot\mathbf{x}}$ is the *biased Walsh–Hadamard transform* of $f \in \mathcal{B}_n$ at $\mathbf{a} \in \mathbb{F}_2^n$. The multiset $\left[\mathcal{W}_f^B(\mathbf{a}) : \ \mathbf{a} \in \mathbb{F}_2^n\right]$ is the *biased Walsh–Hadamard spectrum* of $f \in \mathcal{B}_n$. Note that $\mathcal{W}_{l_{\mathbf{a},0}}^B(\mathbf{a}) = 1$ and for any other point, the value of this biased Walsh–Hadamard transform may or may not be zero, which is not the case for the uniform domain.

Further, for the non-uniform case, we define the $\mathsf{corr}^B(f,g)$ between $f, g \in \mathcal{B}_n$, by

$$\mathsf{corr}^B(f,g) = \left| \sum_{\mathbf{x}\in\bar{S}(f,g)} p(\mathbf{x}) - \sum_{\mathbf{x}\in S(f,g)} p(\mathbf{x}) \right|.$$

Note that $\mathsf{corr}^B(f, l_{\mathbf{a},0}) = \left|\mathcal{W}_f^B(\mathbf{a})\right|$.

3.1 The Biased Walsh–Hadamard Transform of a Direct Sum of Boolean Functions

This section presents a convolution theorem in the biased domain and several bounds related to a direct sum of Boolean functions. Let $n = n_1 + n_2$, and $\mathbf{x} = \mathbf{x}'||\mathbf{x}'' \in \mathbb{F}_2^n$, where $\mathbf{x}' \in \mathbb{F}_2^{n_1}$ and $\mathbf{x}'' \in \mathbb{F}_2^{n_2}$. Then, $Pr[\mathbf{x}] = Pr[\mathbf{x}', \mathbf{x}''] = Pr[\mathbf{x}'/\mathbf{x}'']Pr[\mathbf{x}''] = Pr[\mathbf{x}''/\mathbf{x}']Pr[\mathbf{x}']$, for any $\mathbf{x} \in \mathbb{F}_2^n$. The biased Walsh–Hadamard transform of $f(\mathbf{x}) = f_1(\mathbf{x}') + f_2(\mathbf{x}'')$ at $\mathbf{a} = \mathbf{a}'||\mathbf{a}''$ is equal to

$$\begin{aligned}\mathcal{W}_f^B(\mathbf{a}) &= \sum_{\mathbf{x}\in\mathbb{F}_2^n} p(\mathbf{x})(-1)^{f(\mathbf{x})+\mathbf{a}\cdot\mathbf{x}} \\ &= \sum_{\mathbf{x}''\in\mathbb{F}_2^{n_2}} p(\mathbf{x}'')(-1)^{f_2(\mathbf{x}'')+\mathbf{a}''\cdot\mathbf{x}''} \sum_{\mathbf{x}'\in\mathbb{F}_2^{n_1}} p(\mathbf{x}'/\mathbf{x}'')(-1)^{f_1(\mathbf{x}')+\mathbf{a}'\cdot\mathbf{x}'},\end{aligned} \tag{1}$$

where $p(\mathbf{x}'/\mathbf{x}'') = Pr[\mathbf{x}'/\mathbf{x}'']$. From Eq. (1), it is clear that we are unable to directly calculate the biased Walsh–Hadamard transform of $f = f_1 + f_2$ even though we may know the biased Walsh–Hadamard transform of two component functions f_1 and f_2, as $Pr[\mathbf{x}'/\mathbf{x}''] \neq Pr[\mathbf{x}']$, in general.

Let now $f = f_1 + f_2$ on $\mathbb{F}_2^n$, where f_1, f_2 depend upon independent sets of variables. If the domain is uniform, to calculate the Walsh–Hadamard transform of f at any point $\mathbf{a} \in \mathbb{F}_2^n$, we only need to calculate the Walsh–Hadamard transform of the component functions f_1 and f_2 at the points $\mathbf{a}'$ and $\mathbf{a}''$, respectively. Thus, we only need two tables of sizes 1×2^{n_1} and 1×2^{n_2} corresponding to the Walsh–Hadamard values of f_1 and f_2, respectively. However, for the biased domain, we need more data to calculate the biased Walsh–Hadamard value at any point $\mathbf{a}$, as it can be seen from Theorem 1 and Corollary 1 (under the assumption that if $wt(\mathbf{x}) = wt(\mathbf{y})$, then $Pr[\mathbf{x}] = Pr[\mathbf{y}]$). To compute the biased Walsh–Hadamard transform values of f at any point, we need three probability tables P_1, P_2 and P_3 of sizes $1\times(n+1)$, $1\times(n_1+1)$ and $1\times(n_2+1)$ corresponding to the probabilities $\mathbf{x} \in \mathbb{F}_2^n$, $\mathbf{x}' \in \mathbb{F}_2^{n_1}$ and $\mathbf{x}'' \in \mathbb{F}_2^{n_2}$, respectively. We also need two tables T_{f_1} and T_{f_2} of sizes $2^{n_1} \times (n_1 + 1)$ and $2^{n_2} \times (n_2 + 1)$ (worst case) corresponding to the restricted biased Walsh–Hadamard values of f_1 and f_2, respectively. Certainly, the complexity increases when the size of the partition for n gets larger.

Further, we show a convolution theorem, which will depend on the Walsh–Hadamard transform over the inputs of fixed weight. We want to compute $\mathcal{W}_f^{B(k)}$ (defined as in Eq. (1) but summing for $\mathbf{x} \in E_{n,k}$), where $f = f_1 + f_2$ and $0 \leq k \leq n$. Here we use the fact that if $\mathbf{x}, \mathbf{y} \in \mathbb{F}_2^m$ with $wt(\mathbf{x}) = wt(\mathbf{y})$, then $p(\mathbf{x}) = p(\mathbf{y})$, and $p_{m,i} = Pr[\mathbf{x}]$, for all $\mathbf{x} \in E_{m,i}$, $0 \leq i \leq m$. From the definition

of the Walsh–Hadamard transform of a Boolean function in both uniform and non-uniform cases, we infer the following relation

$$\mathcal{W}_f^{B(k)}(\mathbf{a}) = p\binom{n}{k}\mathcal{W}_f^{(k)}(\mathbf{a}), \quad \forall\, \mathbf{a} \in \mathbb{F}_2^n, \tag{2}$$

where $0 \leq k \leq n$ and $p = Pr[\mathbf{x} : wt(\mathbf{x}) = k]$.

Theorem 1 (Restricted Domain Convolution). *Let $n = n_1 + n_2$ and $f = f_1 + f_2$, where $f_i \in \mathcal{B}_{n_i}$, $i \in \{1, 2\}$. Then, for any $\mathbf{a} = \mathbf{a}'||\mathbf{a}'' \in \mathbb{F}_2^n$ and $0 \leq k \leq n$,*

$$\begin{aligned}\mathcal{W}_f^{B(k)}(\mathbf{a}) &= p_{n,k}\sum_{i=0}^{k}\binom{n_1}{i}\binom{n_2}{k-i}\mathcal{W}_{f_1}^{(i)}(\mathbf{a}')\mathcal{W}_{f_2}^{(k-i)}(\mathbf{a}'')\\ &= \sum_{i=0}^{k}\frac{p_{n,k}}{q_{n_1,i}q_{n_2,k-i}}\mathcal{W}_{f_1}^{B(i)}(\mathbf{a}')\mathcal{W}_{f_2}^{B(k-i)}(\mathbf{a}''),\end{aligned}$$

where $q_{n_1,i} = \frac{\binom{n_2}{k-i}}{\binom{n}{k}}, q_{n_2,k-i} = \frac{\binom{n_1}{i}}{\binom{n}{k}}$.

Proof. For any $\mathbf{a} = \mathbf{a}'||\mathbf{a}'' \in \mathbb{F}_2^n$ and $0 \leq k \leq n$, we have

$$\begin{aligned}\mathcal{W}_f^{B(k)}(\mathbf{a}) &= \sum_{\mathbf{x}\in E_{n,k}} Pr[\mathbf{x}](-1)^{f(\mathbf{x})+\mathbf{a}\cdot\mathbf{x}} = p_{n,k}\sum_{\mathbf{x}\in E_{n,k}}(-1)^{f(\mathbf{x})+\mathbf{a}\cdot\mathbf{x}}\\ &= p_{n,k}\sum_{i=0}^{k}\sum_{\mathbf{x}'\in E_{n_1,i}^{n=n_1+n_2,k}}\sum_{\mathbf{x}''\in E_{n_2,k-i}^{n=n_1+n_2,k}}(-1)^{f_1(\mathbf{x}')+\mathbf{a}'\cdot\mathbf{x}'}(-1)^{f_2(\mathbf{x}'')+\mathbf{a}''\cdot\mathbf{x}''}\\ &= p_{n,k}\sum_{i=0}^{k}\sum_{\mathbf{x}'\in E_{n_1,i}^{n=n_1+n_2,k}}(-1)^{f_1(\mathbf{x}')+\mathbf{a}'\cdot\mathbf{x}'}\sum_{\mathbf{x}''\in E_{n_2,k-i}^{n=n_1+n_2,k}}(-1)^{f_2(\mathbf{x}'')+\mathbf{a}''\cdot\mathbf{x}''}\\ &= p_{n,k}\sum_{i=0}^{k}\binom{n_1}{i}\binom{n_2}{k-i}\mathcal{W}_{f_1}^{(i)}(\mathbf{a}')\mathcal{W}_{f_2}^{(k-i)}(\mathbf{a}'').\end{aligned}$$

We can also rewrite the above in terms of the biased Walsh–Hadamard transform by using Eq. (2), obtaining

$$\mathcal{W}_f^{B(k)}(\mathbf{a}) = \sum_{i=0}^{k}\frac{p_{n,k}}{q_{n_1,i}q_{n_2,k-i}}\mathcal{W}_{f_1}^{B(i)}(\mathbf{a}')\mathcal{W}_{f_2}^{B(k-i)}(\mathbf{a}''),$$

Hence we get both equalities in terms of the Walsh–Hadamard transform in the uniform and biased domains. □

From Theorem 1, we obtain the next corollary.

Corollary 1. *Let $n = n_1 + n_2$ and $f = f_1 + f_2$, where $f_i \in \mathcal{B}_{n_i}$, $i \in \{1, 2\}$. For any $\mathbf{a} = \mathbf{a}'||\mathbf{a}'' \in \mathbb{F}_2^n$,*

$$\mathcal{W}_f^B(\mathbf{a}) = \sum_{k=0}^{n} \mathcal{W}_f^{B(k)}(\mathbf{a}) = \sum_{k=0}^{n} p_{n,k} \sum_{i=0}^{k} \binom{n_1}{i}\binom{n_2}{k-i} \mathcal{W}_{f_1}^{(i)}(\mathbf{a}') \mathcal{W}_{f_2}^{(k-i)}(\mathbf{a}'')$$
$$= \sum_{k=0}^{n} \sum_{i=0}^{k} \frac{p_{n,k}}{q_{n_1,i} q_{n_2,k-i}} \mathcal{W}_{f_1}^{B(i)}(\mathbf{a}') \mathcal{W}_{f_2}^{B(k-i)}(\mathbf{a}'').$$

We observe that it is very difficult to compute the biased Walsh–Hadamard transform for a direct sum of two Boolean functions, in arbitrary (large) number of variables. So we have to find an appropriate bound for the biased Walsh–Hadamard transform of $f \in \mathcal{B}_n$, where $f = f_1 + f_2 \in \mathcal{B}_n$, with $f_i \in \mathcal{B}_{n_i}$, $i = 1, 2$.

In the following theorem, we show that the biased Walsh–Hadamard transform may help us obtain a better bound.

Theorem 2. *For all $0 \le k \le n$, the following inequality holds*

$$\sum_{i=0}^{k} p_{n,k} \max_{\mathbf{a}_1 \in \mathbb{F}_2^{n_1}} \left| \sum_{\mathbf{x}_1 \in E_{n_1,i}} (-1)^{f_1(\mathbf{x}_1) + \mathbf{a}_1 \cdot \mathbf{x}_1} \right| \max_{\mathbf{a}_2 \in \mathbb{F}_2^{n_2}} \left| \sum_{\mathbf{x}_2 \in E_{n_2,k-i}} (-1)^{f_1(\mathbf{x}_2) + \mathbf{a}_2 \cdot \mathbf{x}_2} \right|$$
$$\ge \sum_{i=0}^{k} \max_{\mathbf{a}_1 \in \mathbb{F}_2^{n_1}} \left| \mathcal{W}_{f_1}^{B(i)}(\mathbf{a}_1) \right| \max_{\mathbf{a}_2 \in \mathbb{F}_2^{n_2}} \left| \mathcal{W}_{f_2}^{B(k-i)}(\mathbf{a}_2) \right|.$$

Proof. Using Vandermonde's identity, $\binom{n}{k} = \sum_{i=0}^{k} \binom{n_1}{i}\binom{n_2}{k-i}$, or directly using Stirling's formula, we infer that $\frac{\binom{n}{k}}{\binom{n_1}{i}\binom{n_2}{k-i}} \ge 1$, for all $0 \le i \le k$. Further,

$$\sum_{i=0}^{k} p_{n,k} \max_{\mathbf{a}_1 \in \mathbb{F}_2^{n_1}} \left| \sum_{\mathbf{x}_1 \in E_{n_1,i}} (-1)^{f_1(\mathbf{x}_1) + \mathbf{a}_1 \cdot \mathbf{x}_1} \right| \max_{\mathbf{a}_2 \in \mathbb{F}_2^{n_2}} \left| \sum_{\mathbf{x}_2 \in E_{n_2,k-i}} (-1)^{f_1(\mathbf{x}_2) + \mathbf{a}_2 \cdot \mathbf{x}_2} \right|$$
$$= \sum_{i=0}^{k} \frac{p_{n,k}}{q_{n_1,i} q_{n_2,k-i}} \max_{\mathbf{a}_1 \in \mathbb{F}_2^{n_1}} \left| \mathcal{W}_{f_1}^{B(i)}(\mathbf{a}_1) \right| \max_{\mathbf{a}_2 \in \mathbb{F}_2^{n_2}} \left| \mathcal{W}_{f_2}^{B(k-i)}(\mathbf{a}_2) \right|$$
$$= \sum_{i=0}^{k} \frac{1}{p_{n,k}\binom{n_1}{i}\binom{n_2}{k-i}} \max_{\mathbf{a}_1 \in \mathbb{F}_2^{n_1}} \left| \mathcal{W}_{f_1}^{B(i)}(\mathbf{a}_1) \right| \max_{\mathbf{a}_2 \in \mathbb{F}_2^{n_2}} \left| \mathcal{W}_{f_2}^{B(k-i)}(\mathbf{a}_2) \right|$$
$$= \sum_{i=0}^{k} \frac{\binom{n}{k}}{\binom{n_1}{i}\binom{n_2}{k-i}} \max_{\mathbf{a}_1 \in \mathbb{F}_2^{n_1}} \left| \mathcal{W}_{f_1}^{B(i)}(\mathbf{a}_1) \right| \max_{\mathbf{a}_2 \in \mathbb{F}_2^{n_2}} \left| \mathcal{W}_{f_2}^{B(k-i)}(\mathbf{a}_2) \right|$$
$$\ge \sum_{i=0}^{k} \max_{\mathbf{a}_1 \in \mathbb{F}_2^{n_1}} \left| \mathcal{W}_{f_1}^{B(i)}(\mathbf{a}_1) \right| \max_{\mathbf{a}_2 \in \mathbb{F}_2^{n_2}} \left| \mathcal{W}_{f_2}^{B(k-i)}(\mathbf{a}_2) \right|,$$

and the result is shown. □

From the above inequality we can theoretically claim that the bound provided by Carlet et al. [2] is much higher than our bound

$$G = \sum_{i=0}^{k} \max_{\mathbf{a}_1 \in \mathbb{F}_2^{n_1}} |\mathcal{W}_{f_1}^{B(i)}(\mathbf{a}_1)| \max_{\mathbf{a}_2 \in \mathbb{F}_2^{n_2}} |\mathcal{W}_{f_2}^{B(k-i)}(\mathbf{a}_2)|.$$

It might be tempting to conjecture that G is smaller than $\max_{\mathbf{a}\in\mathbb{F}_2^n} |\mathcal{W}_f^{(k)}(\mathbf{a})|$. Experimentally, using some small functions, we found that in many cases $G \leq \max_{\mathbf{a}\in\mathbb{F}_2^n} |\mathcal{W}_f^{(k)}(\mathbf{a})|$, but we also observed that under some conditions this inequality will change its direction. Now we are interested to find such conditions for which $G \geq \max_{\mathbf{a}\in\mathbb{F}_2^n} |\mathcal{W}_f^{(k)}(\mathbf{a})|$. We start with the following lemma.

Lemma 1. *Let a_i be positive numbers and b_i be any integer numbers (positive or negative), where $i = 0, 1, \ldots, k$. If* $\left|\left|\sum_{i=0}^{k} a_i b_i\right| - \left|\sum_{i,j=0; i\neq j=0}^{k} a_i b_j\right|\right| \leq \left|\sum_{i=0}^{k} a_i b_i\right|$, *and the sums $\sum_{i=0}^{k} a_i b_i$, $\sum_{i,j=0;\ i\neq j}^{k} a_i b_j$ have opposite signs, then*

$$\left|\sum_{i=0}^{k} a_i b_i\right| \geq \left(\sum_{i=0}^{k} a_i\right) \left|\sum_{j=0}^{k} b_j\right|.$$

Proof. We start with the following simple observation $\left(\sum_{i=0}^{k} a_i\right)\left(\sum_{j=0}^{k} b_j\right) = \sum_{i=0}^{k} a_i b_i + \sum_{i,j=0;\ i\neq j}^{k} a_i b_j$. By our assumption, $\left|\left|\sum_{i=0}^{k} a_i b_i\right| - \left|\sum_{i,j=0;\ i\neq j}^{k} a_i b_j\right|\right| \leq \left|\sum_{i=0}^{k} a_i b_i\right|$, and $\sum_{i=0}^{k} a_i b_i$, $\sum_{i,j=0;\ i\neq j}^{k} a_i b_j$ have opposite signs, so,

$$\begin{aligned}\left|\sum_{i=0}^{k} a_i b_i\right| &\geq \left|\left|\sum_{i=0}^{k} a_i b_i\right| - \left|\sum_{i,j=0;\ i\neq j}^{k} a_i b_j\right|\right| = \left|\sum_{i=0}^{k} a_i b_i + \sum_{i,j=0;\ i\neq j}^{k} a_i b_j\right| \\ &= \left|\left(\sum_{i=0}^{k} a_i\right)\left(\sum_{j=0}^{k} b_j\right)\right| = \left(\sum_{i=0}^{k} a_i\right)\left|\sum_{j=0}^{k} b_j\right|,\end{aligned}$$

and the lemma is shown. □

With the help of the above lemma we will prove that $G \geq \max_{\mathbf{a}\in\mathbb{F}_2^n} |\mathcal{W}_f^{(k)}(\mathbf{a})|$ holds under some conditions.

Theorem 3. *Let* $f = f_1 + f_2 \in \mathcal{B}_n$, $f_i \in \mathcal{B}_{n_i}$, $i = 1, 2$, $A_i := q_{n_1,i} q_{n_2,k-i}$ *and* $B_i := \sum_{\mathbf{x}_1 \in E_{n_1,i}} (-1)^{f_1(\mathbf{x}_1)+\mathbf{a}_1\cdot\mathbf{x}_1} \sum_{\mathbf{x}_2 \in E_{n_2,k-i}} (-1)^{f_2(\mathbf{x}_2)+\mathbf{a}_2\cdot\mathbf{x}_2}$, *for all* $0 \le i \le k$ *(here* $q_{n_1,i} = \frac{\binom{n_2}{k-i}}{\binom{n}{k}}$, $q_{n_2,k-i} = \frac{\binom{n_1}{i}}{\binom{n}{k}}$*). Then*

$$\max_{\mathbf{a}\in\mathbb{F}_2^n} \left|\mathcal{W}_f^{(k)}(\mathbf{a})\right| \le \sum_{i=0}^{k} \max_{\mathbf{a}_1\in\mathbb{F}_2^{n_1}} \left|\mathcal{W}_{f_1}^{B(i)}(\mathbf{a}_1)\right| \max_{\mathbf{a}_2\in\mathbb{F}_2^{n_2}} \left|\mathcal{W}_{f_2}^{B(k-i)}(\mathbf{a}_2)\right|,$$

if $\left|\left|\sum_{i=0}^{k} A_i B_i\right| - \left|\sum_{i=0}^{k} A_i B_i - p_{n,k} \sum_{j=0}^{k} B_j\right|\right| \le \left|\sum_{i=0}^{k} A_i B_i\right|$, *where* $p_{n,k} = \frac{1}{\binom{n}{k}}$, *and, the expressions* $\sum_{i=0}^{k} A_i B_i$, $p_{n,k} \sum_{j=0}^{k} B_j - \sum_{i=0}^{k} A_i B_i$ *have opposite signs.*

Proof. We compute

$$\begin{aligned}
&\sum_{i=0}^{k} \max_{\mathbf{a}_1\in\mathbb{F}_2^{n_1}} \left|\mathcal{W}_{f_1}^{B(i)}(\mathbf{a}_1)\right| \max_{\mathbf{a}_2\in\mathbb{F}_2^{n_2}} \left|\mathcal{W}_{f_2}^{B(k-i)}(\mathbf{a}_2)\right| \ge \max_{\mathbf{a}_1||\mathbf{a}_2} \sum_{i=0}^{k} \left|\mathcal{W}_{f_1}^{B(i)}(\mathbf{a}_1)\right| \left|\mathcal{W}_{f_2}^{B(k-i)}(\mathbf{a}_2)\right| \\
&= \max_{\mathbf{a}_1,\mathbf{a}_2} \sum_{i=0}^{k} \left| q_{n_1,i} q_{n_2,k-i} \sum_{\mathbf{x}_1\in E_{n_1,i}} (-1)^{f_1(\mathbf{x}_1)+\mathbf{a}_1\cdot\mathbf{x}_1} \sum_{\mathbf{x}_2\in E_{n_2,k-i}} (-1)^{f_2(\mathbf{x}_2)+\mathbf{a}_2\cdot\mathbf{x}_2} \right| \\
&\ge \max_{\mathbf{a}_1,\mathbf{a}_2} \left| \sum_{i=0}^{k} q_{n_1,i} q_{n_2,k-i} \sum_{\mathbf{x}_1\in E_{n_1,i}} (-1)^{f_1(\mathbf{x}_1)+\mathbf{a}_1\cdot\mathbf{x}_1} \sum_{\mathbf{x}_2\in E_{n_2,k-i}} (-1)^{f_2(\mathbf{x}_2)+\mathbf{a}_2\cdot\mathbf{x}_2} \right| \\
&\ge \left(\sum_{i=0}^{k} q_{n_1,i} q_{n_2,k-i}\right) \max_{\mathbf{a}_1,\mathbf{a}_2} \left| \sum_{i=0}^{k} \sum_{\mathbf{x}_1\in E_{n_1,i}} (-1)^{f_1(\mathbf{x}_1)+\mathbf{a}_1\cdot\mathbf{x}_1} \sum_{\mathbf{x}_2\in E_{n_2,k-i}} (-1)^{f_2(\mathbf{x}_2)+\mathbf{a}_2\cdot\mathbf{x}_2} \right|.
\end{aligned}$$

By Lemma 1, the last inequality holds if $\sum_{i=0}^{k} A_i B_i$ and $\sum_{i,j=0;i\neq j}^{k} A_i B_j$ have opposite signs, and $\left|\left|\sum_{i=0}^{k} A_i B_i\right| - \left|\sum_{i,j=0;i\neq j}^{k} A_i B_j\right|\right| \le \left|\sum_{i=0}^{k} A_i B_i\right|$. We argue that this last condition is equivalent to $\left|\left|\sum_{i=0}^{k} A_i B_i\right| - \left|\sum_{i=0}^{k} A_i B_i - p_{n,k} \sum_{j=0}^{k} B_j\right|\right| \le \left|\sum_{i=0}^{k} A_i B_i\right|$, where $p_{n,k} = \frac{1}{\binom{n}{k}}$. That follows from the observation that for any

$j,\ 0 \le j \le k$, we have $\sum_{i=0;i\neq j}^{k} A_iB_j = \sum_{i=0}^{k} A_iB_j - A_jB_j = p_{n,k}B_j - A_jB_j$. Further, using Vandermonde's identity, $\sum_{i=0}^{k} q_{n_1,i}q_{n_2,k-i} = \sum_{i=0}^{k} \frac{\binom{n_2}{k-i}\binom{n_1}{i}}{\binom{n}{k}\binom{n}{k}} = \frac{1}{\binom{n}{k}}$, therefore,

$$\begin{aligned}
&\sum_{i=0}^{k} \max_{\mathbf{a}_1\in\mathbb{F}_2^{n_1}} \left|\mathcal{W}_{f_1}^{B(i)}(\mathbf{a}_1)\right| \cdot \max_{\mathbf{a}_2\in\mathbb{F}_2^{n_2}} \left|\mathcal{W}_{f_2}^{B(k-i)}(\mathbf{a}_2)\right| \\
&\ge \max_{\mathbf{a}_1,\mathbf{a}_2} \frac{1}{\binom{n}{k}} \left| \sum_{i=0}^{k} \sum_{\mathbf{x}_1\in E_{n_1,i}} (-1)^{f_1(\mathbf{x}_1)+\mathbf{a}_1\cdot\mathbf{x}_1} \sum_{\mathbf{x}_2\in E_{n_2,k-i}} (-1)^{f_2(\mathbf{x}_2)+\mathbf{a}_2\cdot\mathbf{x}_2} \right| \\
&= \max_{\mathbf{a}} \left|\mathcal{W}_f^{(k)}(\mathbf{a})\right|,
\end{aligned}$$

and the claim is shown. □

In our next result we show that under some conditions we could achieve the lower bound of $\max_{\mathbf{a}\in\mathbb{F}_2^n} |\mathcal{W}_f^{(k)}(\mathbf{a})|$ in terms of the biased Walsh–Hadamard transform.

Theorem 4. *Let* $0 \le i \le k$, $\mathbf{c}_i \in \mathbb{F}_2^{n_1}$, $\mathbf{d}_i \in \mathbb{F}_2^{n_2}$, $q_{n_1,i} = \frac{\binom{n_2}{k-i}}{\binom{n}{k}}$, $q_{n_2,k-i} = \frac{\binom{n_1}{i}}{\binom{n}{k}}$, *and*

$$\max_{\mathbf{a}_1\in\mathbb{F}_2^{n_1}} \left|\mathcal{W}_{f_1}^{B(i)}(\mathbf{a}_1)\right| = q_{n_1,i} \left| \sum_{\mathbf{x}_1\in E_{n_1,i}} (-1)^{f_1(\mathbf{x}_1)+\mathbf{c}_i\cdot\mathbf{x}_1} \right|,$$

$$\max_{\mathbf{a}_2\in\mathbb{F}_2^{n_2}} \left|\mathcal{W}_{f_2}^{B(k-i)}(\mathbf{a}_2)\right| = q_{n_2,k-i} \left| \sum_{\mathbf{x}_2\in E_{n_2,k-i}} (-1)^{f_2(\mathbf{x}_2)+\mathbf{d}_i\cdot\mathbf{x}_2} \right|.$$

If $\sum_{\mathbf{x}_1\in E_{n_1,i}} (-1)^{f_1(\mathbf{x}_1)+\mathbf{c}_i\cdot\mathbf{x}_1} \sum_{\mathbf{x}_2\in E_{n_2,k-i}} (-1)^{f_2(\mathbf{x}_2)+\mathbf{d}_i\cdot\mathbf{x}_2}$ *has constant sign, for all* $0 \le i \le k$, *then,*

$$\sum_{i=0}^{k} \max_{\mathbf{a}_1\in\mathbb{F}_2^{n_1}} \left|\mathcal{W}_{f_1}^{B(i)}(\mathbf{a}_1)\right| \max_{\mathbf{a}_2\in\mathbb{F}_2^{n_2}} \left|\mathcal{W}_{f_2}^{B(k-i)}(\mathbf{a}_2)\right| \le \max_{\mathbf{a}\in\mathbb{F}_2^n} \left|\mathcal{W}_f^{(k)}(\mathbf{a})\right|.$$

Proof. We compute

$$
\begin{aligned}
&\sum_{i=0}^{k} \max_{\mathbf{a}_1 \in \mathbb{F}_2^{n_1}} \left|\mathcal{W}_{f_1}^{B(i)}(\mathbf{a}_1)\right| \max_{\mathbf{a}_2 \in \mathbb{F}_2^{n_2}} \left|\mathcal{W}_{f_2}^{B(k-i)}(\mathbf{a}_2)\right| \\
&= \sum_{i=0}^{k} \left|\mathcal{W}_{f_1}^{B(i)}(\mathbf{c}_i)\right| \left|\mathcal{W}_{f_2}^{B(k-i)}(\mathbf{d}_i)\right| = \sum_{i=0}^{k} \left|\mathcal{W}_{f_1}^{B(i)}(\mathbf{c}_i)\mathcal{W}_{f_2}^{B(k-i)}(\mathbf{d}_i)\right| \\
&= \sum_{i=0}^{k} \left| q_{n_1,i} q_{n_2,k-i} \sum_{\mathbf{x}_1 \in E_{n_1,i}} (-1)^{f_1(\mathbf{x}_1)+\mathbf{c}_i \cdot \mathbf{x}_1} \sum_{\mathbf{x}_2 \in E_{n_2,k-i}} (-1)^{f_2(\mathbf{x}_2)+\mathbf{d}_i \cdot \mathbf{x}_2} \right| \\
&= \sum_{i=0}^{k} q_{n_1,i} q_{n_2,k-i} \left| \sum_{\mathbf{x}_1 \in E_{n_1,i}} (-1)^{f_1(\mathbf{x}_1)+\mathbf{c}_i \cdot \mathbf{x}_1} \sum_{\mathbf{x}_2 \in E_{n_2,k-i}} (-1)^{f_2(\mathbf{x}_2)+\mathbf{d}_i \cdot \mathbf{x}_2} \right| \qquad (3) \\
&\leq \sum_{i=0}^{k} q_{n_1,i} q_{n_2,k-i} \sum_{i=0}^{k} \left| \sum_{\mathbf{x}_1 \in E_{n_1,i}} (-1)^{f_1(\mathbf{x}_1)+\mathbf{c}_i \cdot \mathbf{x}_1} \sum_{\mathbf{x}_2 \in E_{n_2,k-i}} (-1)^{f_2(\mathbf{x}_2)+\mathbf{d}_i \cdot \mathbf{x}_2} \right| \\
&= \frac{1}{\binom{n}{k}} \sum_{i=0}^{k} \left| \sum_{\mathbf{x}_1 \in E_{n_1,i}} (-1)^{f_1(\mathbf{x}_1)+\mathbf{c}_i \cdot \mathbf{x}_1} \sum_{\mathbf{x}_2 \in E_{n_2,k-i}} (-1)^{f_2(\mathbf{x}_2)+\mathbf{d}_i \cdot \mathbf{x}_2} \right| \qquad (4) \\
&= \frac{1}{\binom{n}{k}} \left| \sum_{i=0}^{k} \sum_{\mathbf{x}_1 \in E_{n_1,i}} (-1)^{f_1(\mathbf{x}_1)+\mathbf{c}_i \cdot \mathbf{x}_1} \sum_{\mathbf{x}_2 \in E_{n_2,k-i}} (-1)^{f_2(\mathbf{x}_2)+\mathbf{d}_i \cdot \mathbf{x}_2} \right|, \\
&\leq \frac{1}{\binom{n}{k}} \max_{\mathbf{a}=\mathbf{b}_1||\mathbf{b}_2} \left| \sum_{i=0}^{k} \sum_{\mathbf{x}_1 \in E_{n_1,i}} (-1)^{f_1(\mathbf{x}_1)+\mathbf{b}_1 \cdot \mathbf{x}_1} \sum_{\mathbf{x}_2 \in E_{n_2,k-i}} (-1)^{f_2(\mathbf{x}_2)+\mathbf{b}_2 \cdot \mathbf{x}_2} \right| \\
&\leq \max_{\mathbf{a} \in \mathbb{F}_2^n} \left|\mathcal{W}_f^{(k)}(\mathbf{a})\right|,
\end{aligned}
$$

and the theorem is shown. □

4 More Accurate Calculations of Biases by Our Technique and Comparisons with Previous Work

In this section we compare our bound with the one proposed by Carlet et al. [2]. We first consider a small Boolean function, then we further do the same for the nonlinear filter function used in the FLIP stream cipher.

4.1 Comparison for a Small Boolean Function

Here, we consider a small Boolean function (of FLIP type), involving 12 variables to compare our result with the one obtained from Carlet et al.'s technique [2]. The function $f = f_1 + f_2 + f_3$ is a direct sum of three Boolean functions f_1, f_2 and f_3, where $f_1(x_0, x_1) = x_0 + x_1$ is linear, $f_2(x_0, \ldots, x_3) = x_0x_1 + x_2x_3$ is quadratic

bent, and $f_3(x_0, \ldots, x_5) = x_0 + x_1x_2 + x_3x_4x_5$ is triangular, respectively. Note that out of 12 variables of f, f_1 depends on the first 2 variables, f_2 depends on the next 4 variables and f_3 depends on the last 6 variables. As before, we let $E_{n,k} = \{\mathbf{x} \mid wt(\mathbf{x}) = k\}$. We here assume that f takes inputs from the set $E_{12,6}$.

To obtain a bound for the bias of f, we consider the two types of Walsh–Hadamard transforms. First, the classical Walsh–Hadamard transform $\mathcal{W}_f$, which is also used in the paper of Carlet et al. [2], and the second is our newly defined biased Walsh–Hadamard transform $\mathcal{W}_f^B$. We compute both types of Walsh–Hadamard transform values for f_1, f_2, f_3 for all possible weights of $\mathbf{x}_1$, $\mathbf{x}_2$, $\mathbf{x}_3$. Here the functions depend on the variables $\mathbf{x}_1$, $\mathbf{x}_2$, $\mathbf{x}_3$ and $\mathbf{x} = \mathbf{x}_1||\mathbf{x}_2||\mathbf{x}_3$. For each weight of $\mathbf{x}_1$, $\mathbf{x}_2$ and $\mathbf{x}_3$ we find the maximum absolute Walsh–Hadamard transform values of f_1, f_2, f_3. From these maximum absolute Walsh–Hadamard transform values we can compute the maximum absolute value of Walsh–Hadamard transform of $f = f_1 + f_2 + f_3$, when $wt(\mathbf{x}) = 6$ is fixed. We multiply those maximum absolute Walsh–Hadamard transform values (corresponding to weights) when $wt(\mathbf{x}_1) + wt(\mathbf{x}_2) + wt(\mathbf{x}_3) = 6$ and add them. Finally, the bias bound of the function f in the classical Walsh–Hadamard transform set up can be found by dividing the maximum absolute value by $\binom{12}{6}$. We provide the bias comparison for the classical and biased Walsh–Hadamard transforms with the original bias in Table 2.

Table 2. Correlation bound comparison

Original bias	≈ 0.264069
Carlet et al. [2]	≤ 0.772727
This paper	≥ 0.20857

The comparison of Table 2 clearly shows that our correlation bound is much tighter than Carlet et al. [2]. For better understanding, we refer to Appendix A.

4.2 Comparison for the Actual Nonlinear Filter Function of FLIP

Here we compare the bound for the bias of the nonlinear filter function of the FLIP stream cipher, by extending the ideas explained in Sect. 4.1. The nonlinear filter function of the $\text{FLIP}_{530}(42, 128, 360)$ stream cipher is a direct sum of a linear function of 42 variables, a quadratic bent function of 128 variables and a direct sum of 8 triangular functions each of 45 variables. Since in the triangular part there are 8 terms of degree 1, the final linear function is of 50 variables. Also, since there are 8 terms of degree 2 in the triangular part, the complete quadratic function will be of 144 variables.

As in the toy example, we compute the bias by using the classical Walsh–Hadamard transform and our biased Walsh–Hadamard transform. To compute the bias of the complete function (in classical and biased domain) we break the

function in the following form: 5 linear Boolean function involving 10 variables, 18 quadratic function involving 8 variables and 8 degree $3, 4, \ldots, 9$ terms.

By following the same process, as described in Sect. 4.1 we compute the bias bound value for the normal and biased domain for the Carlet et al.'s study [2] and our study. For Carlet et al.'s case we get the bias bound $G_c = \frac{1}{2^{13.59}}$ and for our biased case the bias bound is $G_o = \frac{1}{2^{18.49}}$. Now, this shows that the computed bias value for Carlet et al.'s study [2], that is, G_c will be an upper bound of the original bias (which can be found in Lemma 3 of [2]).

Table 3. Correlation comparison

Carlet et al. [2]	$\leq \frac{1}{2^{13.59}}$
This paper	$\geq \frac{1}{2^{18.49}}$

Next, we show that G_o is a lower bound of the original bias. To show this we use our Theorem 4. In our computation, the product of the probabilities ($q_{n_1,i}, q_{n_2,k-i}$ of Theorem 4) will be the product of the probabilities corresponding to the 5 linear Boolean function involving 10 variables, probabilities corresponding to 18 quadratic function involving 8 variables, and probabilities corresponding to the 8 degree $3, 4, \ldots, 9$ terms. We have observed that the maximum of this for all product terms is much smaller than $\frac{1}{\binom{530}{265}}$. So we replace all these products of probabilities by $\frac{1}{\binom{530}{265}}$ to get the inequality between Eqs. (3) and (4) from the proof of Theorem 4. Computationally, we found that all these functions $f_1 = x_0+x_1+x_2+x_3+x_4+x_5+x_6+x_7+x_8+x_9$, $f_2 = x_0x_1+x_2x_3+x_4x_5+x_6x_7$, $f_3 = x_0x_1x_2$, $f_4 = x_0x_1x_2x_3$, $f_5 = x_0x_1x_2x_3x_4$, $f_6 = x_0x_1x_2x_3x_4x_5$, $f_7 = x_0x_1x_2x_3x_4x_5x_6$, $f_8 = x_0x_1x_2x_3x_4x_5x_6x_7$, $f_9 = x_0x_1x_2x_3x_4x_5x_6x_7x_8$ satisfy the required condition of Theorem 4. More specifically, in the case of all these functions f_j, there exists at least one point $\mathbf{b}$ for which $\sum_{\mathbf{x}\in E_{n,i}}(-1)^{f_j(\mathbf{x})+\mathbf{b}\cdot\mathbf{x}}$ attains $\max_{\mathbf{a}}\left|\sum_{\mathbf{x}\in E_{n,i}}(-1)^{f_j(\mathbf{x})+\mathbf{a}\cdot\mathbf{x}}\right|$ for all weights i (See Appendix B). Thus, by Theorem 4 we infer that $G_o = \frac{1}{2^{18.49}}$ is the lower bound of the original bias of the nonlinear filter function of FLIP. Table 3 summarizes the comparison of the biases.

4.3 Computation Process

To compute the bias by using the normal and biased Walsh–Hadamard transforms, we split each component function into very small functions and the choices we made are simply dependent upon the power of the machine we ran the code on. We first divide the 50 variables linear function into 5 functions each involving 10 variables. We divide the second function (which is quadratic) in 144 variables into 18 Boolean functions involving 8 variables and similarly, for the other degree terms. We compute the Walsh–Hadamard transform for each component

function of the linear, bent, and the combination of the other degree terms, separately. We find the Walsh–Hadamard transform values (corresponding to each weight of the input) for all functions and save them in separate files. Now we need to combine all these Walsh–Hadamard transform values to calculate a bound of the correlation value. We do the following to compute that bound.

1. For the linear function involving 50 variables, we do the following: from the Walsh–Hadamard transform value corresponding to each weight of the 10 variable linear function we compute the maximum absolute Walsh–Hadamard transform value corresponding to each weight.
2. Now we compute the maximum Walsh–Hadamard transform values corresponding to each weight of the quadratic function involving 144 variables. We first compute the maximum absolute Walsh–Hadamard transform values corresponding to each weight of the quadratic function involving 8 variables. Then we evaluate the maximum absolute Walsh–Hadamard coefficients of the 16 variable function by using the data for the 8 variable function. By doing this, we go up to a quadratic function involving 128 variables. After that we merge 128 variables with a 16 variable function to obtain a similar type of data for 144 variables.
3. Further, we need to do a similar analysis for the combination of degree 3 to degree 9 of the 8 triangular functions, each involving 42 variables. We first compute the maximum absolute Walsh–Hadamard transform values corresponding to each weight of the 42 variable function by considering each monomial as a separate Boolean function. From the maximum absolute Walsh–Hadamard transform values corresponding to each weight of the 42 variable function, we compute the maximum absolute Walsh–Hadamard transform values corresponding to each weight of the 84 variable function. By following the same technique we can compute the maximum absolute Walsh–Hadamard transform values corresponding to each weight of the combination of the degree 3 to degree 9 of 8 triangular functions involving 336 variables.
4. Finally we combine all these absolute Walsh–Hadamard transform values to compute the bias of the complete function F involving 530 variables which takes input of weight 265, only.

Finally, let us summarize the theoretical formulae of our work as well as those provided in [2,7]. Carlet et al. [2] showed the lower bound of the bias in restricted domain is $\max_{\mathbf{a}\in\mathbb{F}_2^n} |\mathcal{W}_f^{(k)}(\mathbf{a})| \geq \frac{1}{\binom{n}{k}} \sqrt{\binom{n}{k} + \lambda}$ (for a parameter λ defined in [2, Prop. 8, p. 207]). Later, Mesnager et al. [7] improved the lower bound of the bias to $\max_{\mathbf{a}\in\mathbb{F}_2^n} |\mathcal{W}_f^{(k)}(\mathbf{a})| \geq \frac{1}{\binom{n}{k}} \sqrt{\binom{n}{k} + \lambda + \max\left(\theta, \frac{1}{\binom{n}{k}}\gamma - \lambda\right)}$ (where λ, γ, θ are defined in [7, Thm. 16]). These two bounds are not related to the direct sum of functions in restricted domain. In this paper we have shown that the bias of direct sum of two functions $f_1 + f_2$ in a restricted domain can be expressed in terms of biased

Walsh–Hadamard transform of f_1, f_2. The lower bound of the bias under some constraints is $\max_{\mathbf{a}\in\mathbb{F}_2^n} \left|\mathcal{W}^{(k)}_{f_1+f_2}(\mathbf{a})\right| \geq \sum_{i=0}^{k} \max_{\mathbf{a}_1\in\mathbb{F}_2^{n_1}} \left|\mathcal{W}^{B(i)}_{f_1}(\mathbf{a}_1)\right| \max_{\mathbf{a}_2\in\mathbb{F}_2^{n_2}} \left|\mathcal{W}^{B(k-i)}_{f_2}(\mathbf{a}_2)\right|$.

Carlet et al. [2] found an upper bound of the bias of a direct sum of two functions in a restricted domain. The expression of the bound is $\max_{\mathbf{a}\in\mathbb{F}_2^n} \left|\mathcal{W}^{(k)}_{f_1+f_2}(\mathbf{a})\right| \leq \frac{1}{\binom{n}{k}} \sum_{i=0}^{k} \left(\max_{\mathbf{a}\in\mathbb{F}_2^{n_1}} \left| \sum_{\mathbf{x}\in E_{n_1,i}} (-1)^{f_1(\mathbf{x})+\mathbf{a}.\mathbf{x}} \right| \max_{\mathbf{b}\in\mathbb{F}_2^{n_2}} \left| \sum_{\mathbf{y}\in E_{n_2,k-i}} (-1)^{f_2(\mathbf{y})+\mathbf{b}.\mathbf{y}} \right| \right)$. We note that the paper [7] of Mesnager et al. does not contain any result related to the direct sum of Boolean functions in a restricted domain. Here, we found (under some technical conditions) an upper bound of the bias of a direct sum of two functions $f_1 + f_2$ in a restricted domain in terms of the biased Walsh–Hadamard transform of f_1 and f_2, namely, $\max_{\mathbf{a}\in\mathbb{F}_2^n} |\mathcal{W}^{(k)}_{f_1+f_2}(\mathbf{a})| \leq \sum_{i=0}^{k} \max_{\mathbf{a}_1\in\mathbb{F}_2^{n_1}} \left|\mathcal{W}^{B(i)}_{f_1}(\mathbf{a}_1)\right| \max_{\mathbf{a}_2\in\mathbb{F}_2^{n_2}} \left|\mathcal{W}^{B(k-i)}_{f_2}(\mathbf{a}_2)\right|$.

5 Conclusion

In this paper we have proposed a *non-uniform* (*biased*) way to investigate the cryptographic properties of a Boolean function, when the inputs to the Boolean function do not follow a uniform distribution. To study this we first define the notion of correlation (biased Walsh–Hadamard transform) for a non-uniform domain, along with the necessary tools. Further, we show how this correlation is related with our newly defined biased Walsh–Hadamard transform, which is used to study several cryptographic properties of a Boolean function in a non-uniform domain. As the computation using our theoretical convolution theorem for the biased Walsh–Hadamard transform cannot be done in an efficient way for Boolean functions with a large number of variables, we use several inequalities for these coefficients. Consequently, we find a lower bound for the bias of the nonlinear filter function of the FLIP stream cipher by exploiting the biased Walsh–Hadamard transform, and compare that with previous work. Certainly, the properties when the domain of the Boolean function does not follow a uniform distribution is worthy of investigation. In this context, our results provide a more accurate calculation of biases related to Boolean functions. This is important in the security evaluation of the stream ciphers, in particular, the ones used in efficient homomorphic encryption schemes.

Acknowledgments. We would like to thank the anonymous reviewers of Indocrypt 2018 for their valuable suggestions and comments, which considerably improved the quality of our paper. The work of T.M. and P.S. started during an enjoyable visit to ISI-Kolkata in March 2018. They would like to thank the hosts and the institution for the excellent working conditions. T.M. also acknowledges support from the Omar Nelson Bradley foundation officer research fellowship in mathematics.

A Biases for 12-variable Function

In our example, the function $F = x_0 + x_1 + x_2x_3 + x_4x_5 + x_6 + x_7x_8 + x_9x_{10}x_{11}$ takes input from $E_{12,6}$. The bias of the function F in this restricted domain is ≈ 0.264069. It is worth noticing that in the uniform domain (i.e., the function takes input from $\mathbb{F}_2^{12}$ instead of $E_{12,6}$) the bias between the original function F and the linear function $l_1 = l_{\mathbf{a}_1,0} = x_0 + x_1 + x_6$ is high, as the monomial of the form x_ix_j or $x_ix_jx_k$ is always 0 unless all variables involved in the monomials are 1. It can be observed that, the bias between F and l_1 in the domain $\mathbb{F}_2^{12}$ and $E_{12,6}$ are $|\mathcal{W}_F(\mathbf{a}_1)| = 0.09375$ and $|\mathcal{W}_F^{(6)}(\mathbf{a}_1)| = 0.099567$, respectively.

The situation is different when the domain of the function F is $E_{12,6}$ (restricted domain). In this domain, the bias between the original function F and a linear function is highest for $l_2 = l_{\mathbf{a}_2,0} = x_0 + x_1 + x_2 + x_3 + x_4 + x_5 + x_6$ instead of $l_1 = x_0 + x_1 + x_6$. The bias between F and l_2 in restricted domain $E_{12,6}$ is $|\mathcal{W}_F^{(6)}(\mathbf{a}_2)| = 0.264069$, but the bias between F and l_1 in the restricted domain $E_{12,6}$ is $|\mathcal{W}_F^{(6)}(\mathbf{a}_1)| = 0.099567$. All the linear function for which the bias is high in the restricted domain $E_{12,6}$ are provided below:

1. $l_{\mathbf{a}_2,0} = l_2 = x_0+x_1+x_2+x_3+x_4+x_5+x_6$: $|\mathcal{W}_F^{(6)}(\mathbf{a}_2)| = 0.264069$, $|\mathcal{W}_F(\mathbf{a}_2)| = 0.09375$.
2. $l_{\mathbf{a}_3,0} = l_3 = x_0+x_1+x_2+x_3+x_6+x_7+x_8$: $|\mathcal{W}_F^{(6)}(\mathbf{a}_3)| = 0.264069$, $|\mathcal{W}_F(\mathbf{a}_3)| = 0.09375$.
3. $l_{\mathbf{a}_4,0} = l_4 = x_0+x_1+x_4+x_5+x_6+x_7+x_8$: $|\mathcal{W}_F^{(6)}(\mathbf{a}_4)| = 0.264069$, $|\mathcal{W}_F(\mathbf{a}_4)| = 0.09375$.
4. $l_{\mathbf{a}_5,0} = l_5 = x_2 + x_3 + x_9 + x_{10} + x_{11}$: $|\mathcal{W}_F^{(6)}(\mathbf{a}_5)| = 0.264069$, $|\mathcal{W}_F(\mathbf{a}_5)| = 0$.
5. $l_{\mathbf{a}_6,0} = l_6 = x_4 + x_5 + x_9 + x_{10} + x_{11}$: $|\mathcal{W}_F^{(6)}(\mathbf{a}_6)| = 0.264069$, $|\mathcal{W}_F(\mathbf{a}_6)| = 0$.
6. $l_{\mathbf{a}_7,0} = l_7 = x_7 + x_8 + x_9 + x_{10} + x_{11}$: $|\mathcal{W}_F^{(6)}(\mathbf{a}_7)| = 0.264069$, $|\mathcal{W}_F(\mathbf{a}_7)| = 0$.

B Existence of a Point b Referred to in Sect. 4.2

This appendix describes the existence of a point $\mathbf{b}$ for each function f_j at which $\sum_{\mathbf{x}\in E_{n,i}} (-1)^{f_j(\mathbf{x})+\mathbf{b}\cdot\mathbf{x}}$ attains $\max_{\mathbf{a}} \left| \sum_{\mathbf{x}\in E_{n,i}} (-1)^{f_j(\mathbf{x})+\mathbf{a}\cdot\mathbf{x}} \right|$ for all weight i.

1. First, let $f_1 = x_0 + x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 + x_8 + x_9$. The existence of a point $\mathbf{b}$ corresponding to each weight starting from weight zero to weight ten is given below (points are provided in integer form): $0, 1023, 0, 1023, 0, 1023, 0, 1023,\ 0, 1023, 0$.
2. For $f_2 = x_0x_1 + x_2x_3 + x_4x_5 + x_6x_7$, the existence of a point $\mathbf{b}$ corresponding to each weight starting from weight zero to weight eight is mentioned below (points are provided in integer form): $0, 0, 0, 63, 15, 3, 0, 255, 0$.
3. For $f_3 = x_0x_1x_2$, the existence of a point $\mathbf{b}$ corresponding to each weight starting from weight zero to weight three is provided below (points are provided in integer form): $0, 0, 0, 1$.

4. For $f_4 = x_0x_1x_2x_3$, the existence of a point $\mathbf{b}$ corresponding to each weight starting from weight zero to weight four is mentioned below (points are provided in integer form): $0, 0, 0, 0, 1$.
5. For $f_5 = x_0x_1x_2x_3x_4$, the existence of a point $\mathbf{b}$ corresponding to each weight starting from weight zero to weight five is given below (points are provided in integer form): $0, 0, 0, 0, 0, 1$.
6. For $f_6 = x_0x_1x_2x_3x_4x_5$, the existence of a point $\mathbf{b}$ corresponding to each weight starting from weight zero to weight six is provided below (points are provided in integer form): $0, 0, 0, 0, 0, 0, 1$.
7. For $f_7 = x_0x_1x_2x_3x_4x_5x_6$, the existence of a point $\mathbf{b}$ corresponding to each weight starting from weight zero to weight seven is mentioned below (points are provided in integer form): $0, 0, 0, 0, 0, 0, 0, 1$.
8. For $f_8 = x_0x_1x_2x_3x_4x_5x_6x_7$, the existence of a point $\mathbf{b}$ corresponding to each weight starting from weight zero to weight eight is given below (points are provided in integer form): $0, 0, 0, 0, 0, 0, 0, 0, 1$.
9. For $f_9 = x_0x_1x_2x_3x_4x_5x_6x_7x_8$, the existence of a point $\mathbf{b}$ corresponding to each weight starting from weight zero to weight nine is mentioned below (points are provided in integer form): $0, 0, 0, 0, 0, 0, 0, 0, 0, 1$.

References

1. Canteaut, A., et al.: Stream ciphers: a practical solution for efficient homomorphic-ciphertext compression. In: Peyrin, T. (ed.) FSE 2016. LNCS, vol. 9783, pp. 313–333. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-52993-5_16
2. Carlet, C., Méaux, P., Rotella, Y.: Boolean functions with restricted input and their robustness, application to the FLIP cipher. IACR Trans. Symmetric Cryptology **3**, 192–227 (2017). (presented at FSE 2018)
3. Duval, S., Lallemand, V., Rotella, Y.: Cryptanalysis of the FLIP family of stream ciphers. In: Robshaw, M., Katz, J. (eds.) CRYPTO 2016. LNCS, vol. 9814, pp. 457–475. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-53018-4_17
4. Gangopadhyay, S., Gangopadhyay, A.K., Pollatos, S., Stănică, P.: Cryptographic Boolean functions with biased inputs. Crypt. Commun. **9**(2), 301–314 (2017)
5. Méaux, P.: Symmetric Encryption Scheme adapted to Fully Homomorphic Encryption Scheme. In: Journées Codage et Cryptographie - JC2 2015–12éme édition des Journées Codage et Cryptographie du GT C2, 5 au 9 octobre 2015, La Londes-Maures, France (2015). http://imath.univ-tln.fr/C2/
6. Méaux, P., Journault, A., Standaert, F.-X., Carlet, C.: Towards stream ciphers for efficient FHE with low-noise ciphertexts. In: Fischlin, M., Coron, J.-S. (eds.) EUROCRYPT 2016. LNCS, vol. 9665, pp. 311–343. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-49890-3_13
7. Mesnager, S., Zhou, Z., Ding, C.: On the nonlinearity of Boolean functions with restricted input. Crypt. Commun. (2018). https://doi.org/10.1007/s12095-018-0293-6