Computing the biases of parity-check relations

Anne Canteaut, Maria Naya-Plasencia

Computing the biases of parity-check relations

Anne Canteaut
INRIA project-team SECRET
B.P. 105
78153 Le Chesnay Cedex, France
Email: Anne.Canteaut@inria.fr

Mara Naya-Plasencia
INRIA project-team SECRET
B.P. 105
78153 Le Chesnay Cedex, France
Email: Maria.Naya_Plasencia@inria.fr

***Abstract*—A divide-and-conquer cryptanalysis can often be mounted against some keystream generators composed of several (nonlinear) independent devices combined by a Boolean function. In particular, any parity-check relation derived from the periods of some constituent sequences usually leads to a distinguishing attack whose complexity is determined by the bias of the relation. However, estimating this bias is a difficult problem since the piling-up lemma cannot be used. Here, we give two exact expressions for this bias. Most notably, these expressions lead to a new algorithm for computing the bias of a parity-check relation, and they also provide some simple formulae for this bias in some particular cases which are commonly used in cryptography.**

I. Divide-and-conquer attacks against some stream ciphers

Parity-check relations are extensively used in cryptanalysis for building statistical distinguishers. For instance, they can be exploited in divide-and-conquer attacks against some stream ciphers which consist of several independent devices whose output sequences are combined by a nonlinear function. Here, we focus on such keystream generators as depicted on Figure 1. All the n constituent devices are updated independently from each other. The only assumption which will be used in the whole paper is that each sequence $\mathbf{x_i} = (x_i(t))_{t\geq 0}$ generated by the i-th device is periodic with least period T_i.

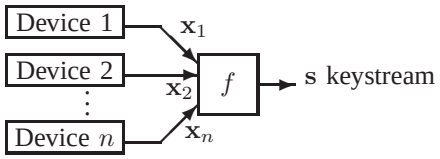

Fig. 1. Keystream generator composed of several independent devices combined by a Boolean function

The simplest case of a generator built according to the model depicted in Figure 1 is the combination generator, where all devices are LFSRs. However, our work is of greater interest in the case where the next-state functions of the constituent devices are nonlinear. The eSTREAM candidate Achterbahn and its variants [1], [2], designed by Gammel, Gttfert and Kniffler, follow this design principle: all these ciphers are actually composed of several nonlinear feedback shift registers (NLFSRs) with maximal periods. This design is very attractive since the use of independent devices enables to accommodate a large internal state with a small hardware footprint.

However, the main weakness of this design is obviously that it is inherently vulnerable to divide-and-conquer attacks. As originally pointed out by Siegenthaler [3], the cryptanalyst may actually mount an attack which depends on a small subset of the constituent devices only. This can be done if there exists a smaller generator which involves k constituent devices whose output is correlated to the keystream. This equivalently means that there exists a correlation between the output of the combining function and the output of a Boolean function depending on k variables. The smallest number k of devices that have to be considered together in the attack is then equal to $(t+1)$ where t is the correlation-immunity order (or resiliency order) of the combining function f. Recall that a Boolean function is said to be t-th order correlation-immune if its output distribution does not change when any t input variables are fixed. Moreover, a t-resilient function is a t-th order correlation-immune function which is balanced.

Now, we recall how parity-check relations can be used for mounting a divide-and-conquer attack against such a keystream generator. This technique has been introduced by Johansson, Meier and Muller [4] for cryptanalysing the first version of Achterbahn [1]. Then, it has been extensively exploited in several attacks against the following variants of the cipher [5], [6], [7], [8]. By analogy with coding theory, a parity-check relation for a binary sequence $\mathbf{x} = (x(t))_{t\geq 0}$ is a linear relation between some bits of $\mathbf{x}$ at different instants $(t+\tau)$ where τ varies in a fixed set and t takes any value:

$$\bigoplus_{\tau\in\mathcal{T}} x(t+\tau) = 0, \ \ \forall t \geq 0.$$

Then, the indexes τ corresponding to the nonzero coefficients of the characteristic polynomial of a linear recurring sequence provide a parity-check relation. A two-term parity-check relation,

$$x(t) \oplus x(t+\tau) = 0, \ \ \forall t \geq 0,$$

obviously corresponds to a period of the sequence. In the following, we only focus on parity-check relations between 2^s instants which are defined as follows.

Definition 1: Let $\mathbf{x}_1, \ldots, \mathbf{x}_n$ be n sequences and let f be a Boolean function of n variables. Then, for any set

$$\mathcal{T} = \Big\{ \sum_{i=1}^{s} c_i M_i, \ \ c_i \in \{0,1\} \Big\}$$

where $M_1, \ldots, M_s$ are some non-negative integers, $PC_{f,\mathcal{T}}$ is the binary sequence defined by

$$PC_{f,\mathcal{T}}(t) = \bigoplus_{\tau \in \mathcal{T}} f(x_1(t+\tau), \ldots, x_n(t+\tau)),\ \forall t \geq 0.$$

In the following, each M_i corresponds to a multiple of the least common multiple of the periods of some constituent sequences. Moreover, in order to simplify the notation, we will assume without loss of generality that the input variables are ordered in such a way that each integer M_i corresponds to a multiple of $\mathrm{lcm}(T_{\ell_i+1}, \ldots, T_{\ell_{i+1}})$ with $\ell_1 = 0$ and $\ell_{s+1} = k$. This notably implies that $\mathcal{T}$ involves the periods of the first k sequences, $\mathbf{x}_1$ ..., $\mathbf{x}_k$.

Proposition 2: Let $\mathbf{x}_1, \ldots, \mathbf{x}_n$ be n sequences with least periods $T_1, \ldots, T_n$ and

$$\mathcal{T} = \Big\{ \sum_{i=1}^{s} c_i M_i,\ \ c_i \in \{0,1\} \Big\}$$

where $M_i = q_i \mathrm{lcm}(T_{\ell_i+1}, \ldots, T_{\ell_{i+1}})$ with $q_i > 0$ and $\ell_1 = 0$ and $\ell_{s+1} = k$. Let g be any Boolean function of k variables of the form

$$g(x_1, \ldots, x_k) = \sum_{i=1}^{s} g_i(x_{\ell_i+1}, \ldots, x_{\ell_{i+1}})$$

where each g_i is any Boolean function of $(\ell_{i+1} - \ell_i)$ variables. Then, for all $t \geq 0$, we have

$$PC_{g,\mathcal{T}}(t) = \bigoplus_{\tau \in \mathcal{T}} g(x_1(t+\tau), \ldots, x_n(t+\tau)) = 0.$$

In the whole paper, we use the following notation.

Definition 3: Let f be a Boolean function of n variables. Then, the *bias* of f is

$$\mathcal{E}(f) = 2^{-n} \sum_{x \in \mathbf{F}_2^n} (-1)^{f(x)}.$$

This quantity is also called the imbalance of f (e.g. in [9], [10]) or the correlation between f and the all-zero function (e.g. in [11]).

The underlying principle of the attack presented by Johansson, Meier and Muller [4] consists in exhibiting a biased approximation g of the combining function f which involves k input variables, and a parity-check relation $PC_{g,\mathcal{T}} = 0$ for the sequence $g(\mathbf{x}_1, \ldots, \mathbf{x}_k)$. Then, the associated parity-check relation applied to $f(\mathbf{x}_1, \ldots, \mathbf{x}_n)$ does not vanish but it is biased in the sense that it is not uniformly distributed when the $(T_1 + \ldots + T_n)$ bits $x_1(0), \ldots, x_1(T_1 - 1), x_2(0), \ldots, x_2(T_2 - 1), \ldots, x_n(T_n - 1)$ are randomly chosen. The bias of $PC_{f,\mathcal{T}}$, denoted by $\mathcal{E}(PC_{f,\mathcal{T}})$ is then defined as the bias of a Boolean function with $(T_1 + \ldots + T_n)$ input variables corresponding to the concatenation of the first periods of the sequences. It follows that

$$\Pr[PC_{f,\mathcal{T}}(t) = 0] = \frac{1}{2}(1 + \mathcal{E}(PC_{f,\mathcal{T}}))$$

with $\mathcal{E}(PC_{f,\mathcal{T}}) > 0$. Then, computing

$$PC_{f,\mathcal{T}}(t) = \bigoplus_{\tau \in \mathcal{T}} s(t+\tau)$$

where $\mathbf{s}$ is the keystream for different values of $t \geq 0$ enables the attacker to distinguish the keystream from a random sequence. The complexity of this distinguishing attack depends on the bias ε of $PC_{f,\mathcal{T}}$. More precisely, the time complexity of the attack corresponds to $\varepsilon^{-2} 2^s$ where 2^s is the number of elements in $\mathcal{T}$ since the bias ε can be detected from at least ε^{-2} occurrences of the biased relation. The data complexity, *i.e.* the number of consecutive keystream bits required for the attack is then the maximal value which must be considered for $(t + \tau)$, *i.e.*

$$\varepsilon^{-2} + \max \mathcal{T}.$$

Many variants of this attack can be derived [5], [6], [7], [8]. However, determining the complexity of all these attacks requires an estimation of the bias of $PC_{f,\mathcal{T}}$. In several attacks [4], [5], [2], it was assumed that the piling-up lemma [12] holds, *i.e.*

$$\mathcal{E}(PC_{f,\mathcal{T}}) = [\mathcal{E}(f \oplus g)]^{2^s}.$$

But it clearly appears that this result does not apply since the terms $f(x_1(t+\tau), \ldots, x_n(t+\tau))$ for the different values of $\tau \in \mathcal{T}$ are not independent. Actually, Naya-Plasencia [6] and Hell and Johansson [7] have independently pointed out that the so-called *piling-up approximation* [10] is far from being valid in some cases.

For instance, the 11-variable Boolean function used in Achterbahn-80 is 6-resilient. An exhaustive search for the initial states of $\mathbf{x}_1$ and $\mathbf{x}_2$ and a decimation by T_7 enable the attacker to use parity-check relations for $f' = f + x_1 + x_2 + x_7$, which is 3-resilient. Then, the quadratic approximation

$$g = x_3 x_{10} + x_4 x_9 \text{ with } \mathcal{E}(f' \oplus g) = 2^{-5}$$

has been considered, corresponding to the set

$$\mathcal{T} = \{c_1 T_3 T_{10} + c_2 T_4 T_9,\ c_1, c_2 \in \{0,1\}\}.$$

It has been deduced that the bias of $PC_{f',\mathcal{T}}$ was $(2^{-5})^4 = 2^{-20}$, leading to an infeasible attack which exceeds the keystream length limitation [2]: the data complexity must be at least 2^{40} and must be multiplied by $T_7 = 2^{28}$. But, Naya-Plasencia in [6] used another approximation, namely

$$g = x_3 + x_{10} + x_4 + x_9 \text{ with } \mathcal{E}(f' \oplus g) = 2^{-3}.$$

This linear approximation leads to $\mathcal{E}(PC_{f,\mathcal{T}}) = 2^{-12}$ for the same set $\mathcal{T}$, and to a feasible attack with an overall data complexity close to 2^{52} (see [6] for a precise estimation of the complexity).

From this concrete example, it clearly appears that estimating the bias of $PC_{f,\mathcal{T}}$ may be a difficult problem. This issue has been raised in [6], [13] which have identified some cases where the piling-up approximation holds. However, since these equality cases are quite rare, a much more extensive study is needed in order to evaluate the resistance of such keystream generators to distinguishing attacks. In this paper, we first emphasize that, even if most attacks based on parity-check relations use an explicit correspondence between the set $\mathcal{T}$ and an approximation g of f depending on k variables,

the bias of $PC_{f,\mathcal{T}}$ does not depend on this approximation. Most notably, we show in the next section that the piling-up lemma applied to any approximation g compatible with $\mathcal{T}$ provides a lower bound on $\mathcal{E}(PC_{f,\mathcal{T}})$. Then, Section III gives two exact expressions for $\mathcal{E}(PC_{f,\mathcal{T}})$, one involving the biases of some restrictions of f, and the other one by means of its Walsh coefficients. These expressions lead to an algorithm for computing the bias of a parity-check relation with a much lower complexity than the usual approach, and they also provide some simple formulae for this bias in some particular cases which are commonly used in cryptography, especially when f is a plateaued function.

II. A lower bound on the bias of parity-check relations

However, we can prove that the piling-up approximation provides a lower bound on the bias of $PC_{f,\mathcal{T}}$.

Theorem 4: Let $\mathbf{x}_1, \ldots, \mathbf{x}_n$ be n sequences with least periods $T_1, \ldots, T_n$, f a Boolean function of n variables and $\mathbf{s} = f(\mathbf{x}_1, \ldots, \mathbf{x}_n)$. Let

$$\mathcal{T} = \{\sum_{i=1}^{s} c_i M_i, \;\; c_i \in \{0,1\}\}$$

where $M_i = q_i \mathrm{lcm}(T_{\ell_i+1}, \ldots, T_{\ell_{i+1}})$ with $q_i > 0$, $\ell_1 = 0$ and $\ell_{s+1} = k$. Then, for any Boolean function g of k variables of the form

$$g(x_1, \ldots, x_k) = \sum_{i=1}^{s} g_i(x_{\ell_i+1}, \ldots, x_{\ell_{i+1}}) \qquad (1)$$

where each g_i is a Boolean function of $(\ell_{i+1} - \ell_i)$ variables, we have

$$\mathcal{E}(PC_{f,\mathcal{T}}) \geq \left[\mathcal{E}(f \oplus g)\right]^{2^s}.$$

The keypoint in the previous theorem is that $\mathcal{E}(f \oplus g)$ provides a lower bound on the bias on the parity-check relation for any choice of the approximation g of the form (1). The linear approximation of f by the sum of the first k input variables is usually considered, but any linear approximation involving these variables can be chosen, as stated in the next corollary. In the following, for any $\alpha \in \mathbf{F}_2^n$, φ_α denotes the linear function of n variables: $x \mapsto \alpha \cdot x$, where $x \cdot y$ is the usual scalar product.

Corollary 5: With the notation of Theorem 4, we have

$$\mathcal{E}(PC_{f,\mathcal{T}}) \geq \max_{\alpha \in V_k} \left[\mathcal{E}(f \oplus \varphi_\alpha)\right]^{2^s}$$

where V_k is the subspace spanned by the first k basis vectors. It is worth noticing that this corollary leads to a lower bound on the bias of the parity check relation even if the functions f and $x \mapsto x_1 \oplus \ldots \oplus x_k$ are not correlated (*i.e.*, if the Walsh coefficient of f at point 1_k vanishes, where the first k coordinates of 1_k are 1 and the other $(n-k)$ are zero). This is the first known result in such a situation; the impossibility of deducing any estimation of the bias of the relation in such cases has been stressed in Example 1 in [13].

However, some other approximations g with a higher degree may lead to a better bound. But, since any Boolean function is completely determined by its Walsh transform, *i.e.* by the biases of all its linear approximations, it appears that $\mathcal{E}(PC_{f,\mathcal{T}})$ can be computed from the biases of the linear approximations of f only.

III. Exact formulae for the bias of the parity-check relation

In some situations, especially when the designer of a generator has to guarantee that the system resists distinguishing attacks, the previous lower bound on the bias of a parity-check relation is not sufficient, and its exact value must be computed. However, since a parity-check relation with 2^s terms involves $n2^s$ variables where n is the number of variables of f, computing its bias requires 2^{n2^s} evaluations of f, which is out of reach in many practical situations. For instance, Achterbahn-128 uses a combining function f of 13 variables, and the biases of parity-check relations with 8 terms (*i.e.* with $s = 3$) must be estimated; this requires 2^{104} operations. Here, we give two exact expressions of the bias of a parity-check relation, which can be computed with much fewer operations, *e.g.* with 2^{43} evaluations of f in the previous case. The first expression makes use of the biases of the restrictions of f when its first k inputs are fixed; the second one, which is related to a theorem due to Nyberg [11], is based on the Walsh coefficients of the combining function. A similar technique is also used in another context in [14].

A. Expression by means of the restrictions of f

Definition 6: Let f be a Boolean function of n variables and let V_k and V_{n-k} be two subspaces such that $V_k \times V_{n-k} = \mathbf{F}_2^n$ and $\dim(V_k) = k$. Then, the restriction of f to the affine subspace $a + V_{n-k}$, $a \in V_k$, denoted by $f_{a+V_{n-k}}$, is the Boolean function of $(n-k)$ variables defined by

$$f_{a+V_{n-k}} : x \in V_{n-k} \mapsto f(x + a).$$

Now, for computing the exact value of $\mathcal{E}(PC_{f,\mathcal{T}})$, we decompose $PC_{f,\mathcal{T}}$ according to the values of the first k variables in f since the other $(n-k)$ sequences $\mathbf{x}_i$, $k+1 \leq i \leq n$, are supposed to be such that $x_i(t+\tau)$ is statistically independent from $x_i(t)$ for any $\tau \in \mathcal{T}$. Amongst the $k2^s$ variables $x_i(t+\tau)$, $1 \leq i \leq k$ and $\tau \in \mathcal{T}$, we can easily see that each variable is repeated once. Indeed, for j such that $\ell_i < j \leq \ell_{i+1}$ we have $x_j(t+\tau) = x_j(t+\tau')$ if and only if $|\tau - \tau'| = M_i$.

It follows that the values of $x_j(t+\tau)$, $1 \leq j \leq k$ and $\tau \in \mathcal{T}$ are determined by a $k2^{s-1}$-bit word α. Let us split α into k words $(\alpha_1, \ldots, \alpha_k)$ of 2^{s-1} bits. We use the correspondence between the values of $\tau = \sum_{i=1}^{s} c_i M_i$ in $\mathcal{T}$ and the integers c, $0 \leq c \leq 2^s - 1$ defined by $c = \sum_{i=1}^{s} c_i 2^{i-1}$. Then, the value of the k-bit word $(x_1(t+\tau), \ldots, x_k(t+\tau))$ is equal to $\chi(c, \alpha) = (\chi_1(c, \alpha), \ldots, \chi_k(c, \alpha))$ where, for any j such that $\ell_i < j \leq \ell_{i+1}$, we have

$$\chi_j(c, \alpha) = \begin{cases} \chi_j(c - 2^i, \alpha) & \text{if } c_i \neq 0 \\ \alpha_{j, 2^i q + r} & \text{if } c = 2^{i+1}q + r, r < 2^i. \end{cases}$$

Clearly, if $c_i \neq 0$, we have that c and $c' = c - 2^i$ correspond to a pair (τ, τ') with $\tau - \tau' = M_i$. Since M_i is a period of $\mathbf{x}_j$, we deduce that $\chi_j(c, \alpha) = \chi_j(c', \alpha)$.

If $c_i = 0$, the corresponding value of $x_j(t+\tau)$ is statistically independent from the previous ones and must be defined by a bit of α which has not been used for smaller values of c. The number of bits of α_j which has been used for previous vectors $\chi_j(c', \alpha)$ for $c' < 2^{i+1}q$ is $2^i q$ since the set $\{0, \ldots, 2^{i+1}q - 1\}$ is composed of $2^i q$ pairs of the form $(c', c' + 2^i)$ with $c'_i = 0$. Moreover, all c' in $\{2^{i+1}q, \ldots, 2^{i+1}q + r - 1\}$ satisfy $c'_i = 0$ because $r < 2^i$. Therefore, exactly $(2^i q + r - 1)$ bits of α_j have been used for $\chi_j(c', \alpha)$, $c' < 2^{i+1}q + r$.

Example. Let us consider a set $\mathcal{T}$ composed of 2^3 elements which involve the periods of 4 sequences:

$$\mathcal{T} = \big\{c_1 T_1 T_2 + c_2 T_3 + c_3 T_4,\ c_1, c_2, c_3 \in \{0.1\}\big\}.$$

Then, the 4-bit words $\chi(c, \alpha)$, $0 \le c < 8$, are defined by the 16-bit word α as follows, where the bold elements correspond to those which have already been used for a smaller value of c:

$$\begin{array}{ll}
\chi(0,\alpha) = (\alpha_{00}\alpha_{10}\alpha_{20}\alpha_{30}) & \chi(4,\alpha) = (\alpha_{02}\alpha_{12}\alpha_{22}\alpha_{\mathbf{30}}) \\
\chi(1,\alpha) = (\alpha_{\mathbf{00}}\alpha_{\mathbf{10}}\alpha_{21}\alpha_{31}) & \chi(5,\alpha) = (\alpha_{\mathbf{02}}\alpha_{\mathbf{12}}\alpha_{23}\alpha_{\mathbf{31}}) \\
\chi(2,\alpha) = (\alpha_{01}\alpha_{11}\alpha_{\mathbf{20}}\alpha_{32}) & \chi(6,\alpha) = (\alpha_{03}\alpha_{13}\alpha_{\mathbf{22}}\alpha_{\mathbf{32}}) \\
\chi(3,\alpha) = (\alpha_{\mathbf{01}}\alpha_{\mathbf{11}}\alpha_{\mathbf{21}}a_{33}) & \chi(7,\alpha) = (\alpha_{\mathbf{03}}\alpha_{\mathbf{13}}\alpha_{\mathbf{23}}\alpha_{\mathbf{33}})
\end{array}$$

The definition of $\chi(c, \alpha)$ enables us to express the bias of $PC_{f,\mathcal{T}}$ by means of the biases of the restrictions of f to all cosets of the subspace V_{n-k} spanned by the last $(n-k)$ basis vectors.

Theorem 7: Let $\mathbf{x}_1, \ldots, \mathbf{x}_n$ be n sequences with least periods $T_1, \ldots, T_n$, f a Boolean function of n variables and $\mathbf{s} = f(\mathbf{x}_1, \ldots, \mathbf{x}_n)$. Let

$$\mathcal{T} = \{\sum_{i=1}^{s} c_i M_i,\ \ c_i \in \{0,1\}\}$$

where $M_i = q_i \mathrm{lcm}(T_{\ell_i+1}, \ldots, T_{\ell_{i+1}})$ with $q_i > 0$, $\ell_1 = 0$ and $\ell_{s+1} = k$. Assume that $\mathcal{T}$ does not contain any multiple of T_j, for any $k < j \le n$. Let V_{n-k} be the subspace spanned by the last $(n-k)$ basis vectors. Then, we have

$$\mathcal{E}(PC_{f,\mathcal{T}}) = \frac{1}{2^{k2^{s-1}}} \sum_{\alpha \in \mathbf{F}_2^{k2^{s-1}}} \prod_{c=0}^{2^s-1} \mathcal{E}(f_{\chi(c,\alpha)+V_{n-k}}).$$

Proof:

$$\begin{aligned}
\Pr[PC_{f,\mathcal{T}}(t) = 0] \ &= \ \frac{1}{2^{k2^{s-1}}} \sum_{\alpha \in \mathbf{F}_2^{k2^{s-1}}} \Pr[PC_{f,\mathcal{T}}(t) = 0 | \\
&\qquad (x_1(t+\tau), \ldots, x_k(t+\tau)) = \chi(c,\alpha)].
\end{aligned}$$

When the values of the first k input variables in every term of $PC_{f,\mathcal{T}}$ are fixed, the piling-up lemma can be applied since the remaining $(n-k)2^s$ variables are statistically independent. The reason is that τ is not a multiple of the period T_i, for any $k < i \le n$. Then, we deduce that the term corresponding to α in the previous sum equals

$$\frac{1}{2}\left[1 + \prod_{\tau \in \mathcal{T}} \mathcal{E}(f(x(t+\tau), y(t+\tau)) | x(t+\tau) = \chi(c,\alpha))\right] = \frac{1}{2}\left[1 + \prod_{c=0}^{2^s-1} \mathcal{E}(f_{\chi(c,\alpha)+V_{n-k}})\right].$$

We then deduce that

$$\Pr[PC_{f,\mathcal{T}}(t) = 0] = \frac{1}{2}\left[1 + \frac{1}{2^{k2^{s-1}}} \sum_{\alpha \in \mathbf{F}_2^{k2^{s-1}}} \prod_{c=0}^{2^s-1} \mathcal{E}(f_{\chi(c,\alpha)+V_{n-k}})\right].$$

■

This result provides an algorithm for computing the exact value of $\mathcal{E}(PC_{f,\mathcal{T}})$. The precomputation step consists in computing and storing in a table the 2^k values of $\mathcal{E}(f_{a+V_{n-k}}) = \frac{1}{2^k}\sum_{y \in Vn-k}(-1)^{f(a+y)}$, for all $a \in V_k$. This step requires 2^n evaluations of f. Then, computing the bias of the parity-check relation needs to compute, for all $\alpha \in \mathbf{F}_2^{k2^{s-1}}$, the product of 2^s precomputed values whose indexes are given by $\chi(c, \alpha)$, for $0 \le c < 2^s$. This requires $2^{k2^{s-1}} \times 2^s$ operations over integers. This leads to an overall complexity of $2^{k2^{s-1}+s} + 2^n$ which is much lower than the complexity of the trivial computation, 2^{n2^s} evaluations of f. For instance, the 13-variable function in Achterbahn-128 is 8-resilient. Estimating the bias of a parity-check relation involving 10 input variables with 8 terms (*i.e.* with $s = 3$) then requires 2^{43} operations.

B. Expression by means of the Walsh coefficients of f

A similar exact expression for the bias of $\mathcal{E}(PC_{f,\mathcal{T}})$ can be obtained from the Walsh coefficients of f, *i.e.* from all biases $\mathcal{E}(f + \varphi_a), a \in V_k$ where V_k is the subspace spanned by the first k basis vectors.

Theorem 8: Let $\mathbf{x}_1, \ldots, \mathbf{x}_n$ be n sequences with least periods $T_1, \ldots, T_n$, f a Boolean function of n variables and $\mathbf{s} = f(\mathbf{x}_1, \ldots, \mathbf{x}_n)$. Let

$$\mathcal{T} = \big\{\sum_{i=1}^{s} c_i M_i,\ \ c_i \in \{0,1\}\big\}$$

where $M_i = q_i \mathrm{lcm}(T_{\ell_i+1}, \ldots, T_{\ell_{i+1}})$ with $q_i > 0$, $\ell_1 = 0$ and $\ell_{s+1} = k$. Assume that $\mathcal{T}$ does not contain any multiple of T_j, for any $k < j \le n$. Then, we have

$$\mathcal{E}(PC_{f,\mathcal{T}}) = \sum_{\alpha \in \mathbf{F}_2^{k2^{s-1}}} \prod_{c=0}^{2^s-1} \mathcal{E}(f + \varphi_{\chi(c,\alpha)}).$$

This expression leads to an algorithm for computing the bias which is very similar to the one based on the biases of the restrictions of f. But, we need to precompute and to store the Walsh coefficients of f corresponding to all elements in V_k.

IV. Computing the bias in some particular cases

As a direct corollary of Theorem 8, we obtain the following theorem. It shows that equality holds in Corollary 5 when, amongst all linear functions depending on the k variables involved in $\mathcal{T}$, a single one corresponds to a biased approximation of f. With this theorem, we recover the value of the bias of a parity-check relation involving the periods of k input sequences when the resiliency order of f is equal to $(k-1)$. This particular case of our theorem corresponds to the case identified in [6], [13] where the piling-up approximation holds.

Theorem 9: With the notation of Theorem 8, suppose that there exists a single linear function φ_a with $a \in V_k$ such that $\mathcal{E}(f+\varphi_a) \neq 0$. Then, we have

$$\mathcal{E}(PC_{f,\mathcal{T}}) = \left[\mathcal{E}(f+\varphi_a)\right]^{2^s}.$$

In particular, if f is $(k-1)$-resilient, then

$$\mathcal{E}(PC_{f,\mathcal{T}}) = \left[\mathcal{E}(f+\varphi_{1_k})\right]^{2^s}.$$

where 1_k is the n-bit word whose first k coordinates are equal to 1 and the other ones are equal to 0.

For a t-resilient function, the bias of a parity-check relation involving any $(t+1)$ inputs is given by Theorem 9 but, as pointed out in [13], this result does not hold anymore when $\mathcal{T}$ involves $(t+2)$ sequences. However, this case can be treated when the function f is plateaued [15], *i.e.* when all values taken by its Walsh transform belong to $\{0, \pm W\}$ for some W. Note that both combining functions in Achterbahn-80 and in Achterbahn-128 are plateaued.

Theorem 10: With the notation and hypotheses of Theorem 8, suppose that f is $(k-2)$-resilient and plateaued, *i.e.* $\mathcal{E}(f+\varphi_a) \in \{0, \pm\varepsilon\}$ for all $a \in \mathbf{F}_2^n$. Let

$$\mathcal{A} = \{a \in V_k, \mathcal{E}(f+\varphi_a) \neq 0\}.$$

Then,

$$\mathcal{E}(PC_{f,\mathcal{T}}) \leq |A|^{2^{s-1}} \varepsilon^{2^s}.$$

Moreover, equality holds if and only if there exists i, $1 \leq i \leq s$, such that M_i is a period of all sequences $\mathbf{x}_j$ for all j in $\cup_{a\in\mathcal{A}} \mathrm{supp}(1_k \oplus a)$.

Acknowledgment

This work was supported in part by the French Agence Nationale de la Recherche under Contract ANR-06-SETI-013-RAPIDE.

References

[1] B. Gammel, R. Gttfert, and O. Kniffler, "The Achterbahn stream cipher," Submission to eSTREAM, 2005, http://www.ecrypt.eu.org/stream/.

[2] ——, "Achterbahn-128/80," Submission to eSTREAM, 2006, http://www.ecrypt.eu.org/stream/.

[3] T. Siegenthaler, "Decrypting a class of stream ciphers using ciphertext only," *IEEE Trans. Inform. Theory*, vol. C-34, no. 1, pp. 81–84, 1985.

[4] T. Johansson, W. Meier, and F. Muller, "Cryptanalysis of Achterbahn," in *Fast Software Encryption - FSE 2006*, ser. Lecture Notes in Computer Science, vol. 4047. Springer, 2006, pp. 1–14.

[5] M. Hell and T. Johansson, "Cryptanalysis of Achterbahn-Version 2," in *SAC 2006 - Selected Areas in Cryptography*, ser. Lecture Notes in Computer Science, vol. 4356. Springer, 2006, pp. 45–55.

[6] M. Naya-Plasencia, "Cryptanalysis of Achterbahn-128/80," in *Fast Software Encryption - FSE 2007*, ser. Lecture Notes in Computer Science, vol. 4593. Springer, 2007, pp. 73–86.

[7] M. Hell and T. Johansson, "Cryptanalysis of Achterbahn-128/80," *IET Information and Security*, vol. 1, no. 2, pp. 47–52, 2007.

[8] M. Naya-Plasencia, "Cryptanalysis of Achterbahn-128/80 with a new keystream limitation," in *WEWoRC 2007 - Second Western European Workshop in Research in Cryptology*, ser. Lecture Notes in Computer Science, vol. 4945. Springer, 2008, pp. 142–152.

[9] C. Harpes, G. Kramer, and J. L. Massey, "A generalization of linear cryptanalysis and the applicability of Matsui's piling-up lemma," in *EUROCRYPT'95*, ser. Lecture Notes in Computer Science, vol. 921. Springer-Verlag, 1995, pp. 24–38.

[10] Z. Kukorelly, *On the validity of certain hypotheses used in linear cryptanalysis*, ser. ETH Series in Information Processing. Konstanz: Hartung-Gorre Verlag, 1999, vol. 13.

[11] K. Nyberg, "Correlation theorems in cryptanalysis," *Discrete Applied Mathematics*, vol. 111, no. 1-2, pp. 177–188, 2001.

[12] M. Matsui, "Linear cryptanalysis method for DES cipher," in *EUROCRYPT'93*, ser. Lecture Notes in Computer Science, vol. 765. Springer-Verlag, 1994.

[13] R. Gttfert and B. Gammel, "On the frame length of Achterbahn-128/80," in *Proceedings of the 2007 IEEE Information Theory Workshop on Information Theory for Wireless Networks*. IEEE, 2007, pp. 1–5.

[14] Y. Lu and S. Vaudenay, "Faster correlation attack on Bluetooth keystream generator E0," in *Advances in Cryptology - CRYPTO 2004*, ser. Lecture Notes in Computer Science, vol. 3152. Springer-Verlag, 2004, pp. 407–425.

[15] Y. Zheng and X.-M. Zhang, "Plateaued functions," in *Information and Communication Security, ICICS'99*, ser. Lecture Notes in Computer Science, vol. 1726. Springer-Verlag, 1999, pp. 224–300.

[16] B. Gammel, R. Göttfert, and O. Kniffler, "An NLFSR-based stream cipher," in *ISCAS 2006 - International Symposium on Circuits and Systems*. IEEE, 2006.

[17] B. Gammel, R. Gttfert, and O. Kniffler, "Improved Boolean combining functions for Achterbahn," eSTREAM report 2005/072, 2005, http://www.ecrypt.eu.org/stream/papersdir/072.pdf.

[18] ——, "Status of Achterbahn and tweaks," in *Proceedings of SASC 2006 - Stream Ciphers Revisited*, 2006.