On the Influence of the Filtering Function on the Performance of Fast Correlation Attacks on Filter Generators

Anne Canteaut and Eric Filiol

INRIA - Projet CODES
BP 105
78153 Le Chesnay Cedex, France
{Anne.Canteaut, Eric.Filiol}@inria.fr

Abstract. This paper presents a generalization of the fast correlation attack presented by Chepyshov, Johansson and Smeets at FSE 2000, for the particular case of filter generators. By considering not only the extremal Walsh coefficients of the filtering function but all the nonzero values in the Walsh spectrum, it is possible to significantly reduce the number of required running-key bits. Most notably, the properties of the filtering function, especially its number of variables, have only a minor influence on the length of the running-key subsequence needed for the attack.

Keywords: Stream ciphers, filter generators, fast correlation attacks.

1 Introduction

The running-key used in a stream cipher is produced by a pseudo-random generator whose initialization is the secret key shared by the users. Linear feedback shift registers (LFSRs) are basic components of most keystream generators because of their low implementation costs. Therefore, such generators are vulnerable to *correlation attacks* [19]. These techniques exploit the correlation that may appear between the observed output sequence (i.e., the running-key in a known plaintext attack) and the output of a constituent LFSR. The aim is to apply a divide-and-conquer attack in order to recover the initialization of this LFSR independently from the other unknown key bits. Meier and Staffelbach [15] formulated this attack as a decoding problem. Any subsequence of the LFSR output belongs to a binary linear code whose dimension is equal to the linear complexity of the LFSR. Any running-key subsequence can then be seen as a noisy version of the corresponding LFSR output subsequence through a particular transmission channel. In most practical situations the noise is produced by a Boolean function whose role is to break the linearity properties inherently attached to the LFSR. Thus, all techniques for fast correlation attacks [15, 5, 6, 16, 12, 11, 10] consist in decoding the running-key subsequence relatively to the LFSR code.

In this paper, we focus on fast correlation attacks against nonlinear filter generators. In such a device, the running-key is generated as a nonlinear function f of the stages of a single LFSR. A classical approach is then to consider an affine function whose distance to the filtering function is minimal (i.e., equals to the nonlinearity of f). Some linear relations between the running-key bits and the LFSR initial state are derived from this approximation. All these relations hold with probability $1 - \mathcal{NL}(f)/2^n$ where n is the number of variables of the filtering function and $\mathcal{NL}(f)$ is its nonlinearity. Therefore, the involved transmission channel is a binary symmetric channel with cross-over probability $\mathcal{NL}(f)/2^n$. Very recently, Jönsson and Johansson [13] observed that the length of running-key required for the attack can be reduced by using all affine functions at distance $\mathcal{NL}(f)$ from f. The underlying idea is that the number of available linear relations increase whereas the transmission channel and its cross-over probability are unchanged. It obviously appear that the attack becomes more powerful when the number of extremal Walsh coefficients of the filtering function increases. Here, we present a general attack which makes use of all nonzero Walsh coefficients of the filtering function. We get a larger number of linear relations, leading to a more efficient decoding when we use the technique presented in [6]. The main modification is that the involved transmission channel is now a non-stationary binary symmetric channel. However, we can derive a theoretical bound on the running-key length which guarantees a successful attack. Most notably, we show that the required running-key length is almost independent of the number of variables of the filtering function and of its nonlinearity. Both of these parameters only influence the running-time of the attack. Note that this paper does not investigate other cryptanalysis techniques (e.g. inversion attacks [1, 9, 14]).

The paper is organized as follows. Section 2 describes the filter generator family and it recalls some basic properties of Boolean functions. Section 3 presents the generalization of the fast correlation attack, with the complete algorithm. A theoretical analysis of this attack is provided in Section 4. In particular, it is proved that contrary to what could be forecast, the results do not depend on the number of variables of the filtering function. Section 5 focuses on the computational complexity of the attack and it gives some comparisons with the attack presented in [13]. Section 6 gives detailed simulation results which confirm the validity of the theoretical approach.

2 Definitions

The pseudo-random sequence $(s_t)_{t\geq 0}$ produced by a nonlinear filter generator corresponds to the output of a nonlinear Boolean function whose inputs are taken from some stages of a given LFSR. The LFSR is defined by its *characteristic polynomial* of degree L, $P(X) = \sum_{i=0}^{L} \lambda_i X^i$. Recall that the characteristic polynomial and the feedback polynomial are reciprocal polynomials [18]. Then, the output $(u_t)_{t\geq 0}$ of the LFSR satisfies the following recursion:

$$\forall t \geq L, \;\; u_t = \sum_{i=0}^{L-1} \lambda_i u_{t-L+i} \; ,$$

where $(u_0, \dots, u_{L-1})$ is the LFSR initial state. Let f be a *balanced* Boolean function of n variables, i.e., a balanced function from the set of n-bit words, $\mathbf{F}_2^n$, into $\mathbf{F}_2$. We consider a decreasing sequence of nonnegative integers $(\gamma_i)_{1\leq i \leq n}$. It is recommended that $(\gamma_1 - \gamma_n)$ be close to its maximum possible value $(L-1)$ [9]. Then, the output of the filter generator $(s_t)_{t\geq 0}$ is given by

$$\forall t \geq 0, \;\; s_t = f(u_{t+\gamma_1}, \dots, u_{t+\gamma_n}) \; .$$

Now, we focus on the Walsh spectrum of the filtering function f. In the following, for any $\alpha \in \mathbf{F}_2^n$, φ_α is the linear function of n variables: $x \mapsto \alpha \cdot x = \sum_{i=1}^n \alpha_i x_i$. For any Boolean function f of n variables, we denote by $\mathcal{F}(f)$ the following value related to the Walsh (or Fourier) transform of f:

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

where $wt(f)$ is the Hamming weight of f, i.e., the number of $x \in \mathbf{F}_2^n$ such that $f(x) = 1$. A function f is said to be *balanced* if $\mathcal{F}(f) = 0$.

Therefore, the *Walsh spectrum* of f is the multiset

$$\{\mathcal{F}(f + \varphi_\alpha), \alpha \in \mathbf{F}_2^n\} \; .$$

A major cryptographic parameter for a Boolean function is its nonlinearity. It is derived from the Walsh spectrum as follows:

Definition 1. *The* nonlinearity *of an n-variable Boolean function f is the Hamming distance between f and the set of affine functions. It is equal to*

$$2^{n-1} - \frac{1}{2}\mathcal{L}(f) \;\; \textit{with} \;\; \mathcal{L}(f) = \max_{\alpha \in \mathbf{F}_2^n} |\mathcal{F}(f + \varphi_\alpha)| \; .$$

Here, we are interested in all nonzero values in the Walsh spectrum and in the number of times they occur. We denote by $\mathcal{W}$ the set of all nonzero magnitudes appearing in the Walsh spectrum of f. For any integer w, $0 \leq w \leq 2^n$, we set

$$F_w = \#\{\alpha \in \mathbf{F}_2^n, \;\; |\mathcal{F}(f + \varphi_\alpha)| = w\} \; .$$

Moreover, we denote by F the number of nonzero Walsh coefficients, i.e., $F = 2^n - F_0$.

In the context of the previously described filter generator, any nonzero Walsh coefficient provides a linear approximation of the running key. For any $\alpha \in \mathbf{F}_2^n \setminus \{0\}$, for any $c \in \mathbf{F}_2$, we have for all $t \geq 0$

$$\begin{aligned} Pr[s_t \neq \sum_{i=1}^n \alpha_i u_{t+\gamma_i} + c] &= Pr[f(x) \neq \varphi_\alpha(x) + c] \\ &= \frac{1}{2} - \frac{(-1)^c}{2^{n+1}} \mathcal{F}(f + \varphi_\alpha) \; . \end{aligned} \tag{1}$$

Then, we choose the binary constant c such that $(-1)^c$ is equal to the sign of $\mathcal{F}(f+\varphi_\alpha)$. Therefore, we obtain this way a set of F linear relations between s_t and some stages of the LFSR. More precisely, for any $w \in \mathcal{W}$, we get F_w relations which hold with probability $1-p_w$ with

$$p_w = \frac{1}{2} - \frac{w}{2^{n+1}} \ .$$

Note that we do not consider the relation corresponding to $\alpha = 0$ because the filtering function is assumed to be balanced, i.e., $\mathcal{F}(f) = 0$.

3 A general fast correlation attack

Now, we use the technique proposed by Chepyshov, Johansson and Smeets [6] for fast correlation attacks. But, we exploit all approximations derived from the nonzero Walsh coefficients of the filtering function. A similar attack was presented in [13] on the stream cipher LILI-128 but it only exploits the $F_{\mathcal{L}(f)}$ relations corresponding to the extremal Walsh coefficients.

Any bit u_t of the LFSR output can be expressed as a linear combination of the initial bits, $(u_0, \dots, u_{L-1})$:

$$\forall t \geq 0, \ \ u_t = \sum_{i=0}^{L-1} \lambda_i^{(t)} u_i$$

where the involved coefficients $(\lambda_i^{(t)})_{0 \leq i < L}$ are obtained from the characteristic polynomial by

$$\forall t \geq 0, \ \ \sum_{i=0}^{L-1} \lambda_i^{(t)} X^i = X^t \bmod P(X) \ .$$

Then, we deduce that, for any $\alpha \in \mathbf{F}_2^n \setminus \{0\}$, we have

$$\begin{aligned}\forall t \geq 0, \ \ \sum_{i=1}^{n} \alpha_i u_{t+\gamma_i} &= \sum_{i=1}^{n} \alpha_i \sum_{j=0}^{L-1} \lambda_j^{(t+\gamma_i)} u_j \\ &= \sum_{j=0}^{L-1} u_j \left(\sum_{i=1}^{n} \alpha_i \lambda_j^{(t+\gamma_i)} \right) \\ &= \sum_{j=0}^{L-1} u_j q_j \ .\end{aligned}$$

It clearly appears that the coefficients $(q_j)_{0 \leq j < L}$ are obtained by

$$Q_{\alpha,t}(X) = \sum_{j=0}^{L-1} q_j X^j = \left(\sum_{i=1}^{n} \alpha_i X^{t+\gamma_i} \right) \bmod P(X) \ . \quad (2)$$

It follows that any sequence whose bits correspond to $\sum_{i=1}^{n} \alpha_i u_{t+\gamma_i}$ for some $\alpha \in \mathbf{F}_2^n \setminus \{0\}$ and for some $t \geq 0$ is a codeword of a linear binary code $\mathcal{C}$ of dimension L. Any column of a generator matrix G of $\mathcal{C}$ is a binary vector $q_{\alpha,t}$ corresponding to the coefficients of the polynomial $Q_{\alpha,t}$ defined by (2). Since the dimension of $\mathcal{C}$ is usually too large, a maximum-likelihood decoding cannot be directly applied. It was proposed in [6] to derive from $\mathcal{C}$ a new code $\mathcal{C}'$ having a lower dimension $k < L$, for which ML-decoding is feasible. Such a code $\mathcal{C}'$ is obtained by computing all linear combinations of d columns of the generator matrix G which vanish on the last $(L-k)$ positions. Parameter d does usually not exceed 4 or 5. For the j-th set of d such columns of G, namely $(q_{\alpha_1,t_1}, \dots, q_{\alpha_d,t_d})$, we have

$$\sum_{i=1}^{d} q_{\alpha_i,t_i} = (h_j, 0 \dots 0) \text{ with } h_j \in \mathbf{F}_2^k \ . \tag{3}$$

Let $z_j = \sum_{i=1}^{d} s_{t_i} + c$ where the binary constant c is such that $(-1)^c$ equals the sign of $\prod_{i=1}^{d} \mathcal{F}(f + \varphi_{\alpha_i})$. We derive from (1):

$$Pr[z_j \neq h_j \cdot u] = \frac{1}{2} - \varepsilon_j \text{ with } \varepsilon_j = 2^{d-1} \frac{\prod_{i=1}^{d} \mathcal{F}(f + \varphi_{\alpha_i})}{2^{(n+1)d}} \ , \tag{4}$$

where $u = (u_0, \dots, u_{k-1})$. The above value of ε_j is obtained by induction from the following result. Let X and Y be two independent binary random variables with $Pr[X = 1] = 1/2 - \varepsilon_X$ and $Pr[Y = 1] = 1/2 - \varepsilon_Y$. Then, we have

$$\begin{aligned} Pr[X + Y = 1] &= Pr[X = 0]Pr[Y = 1] + Pr[X = 1]Pr[Y = 0] \\ &= (\frac{1}{2} + \varepsilon_X)(\frac{1}{2} - \varepsilon_Y) + (\frac{1}{2} - \varepsilon_X)(\frac{1}{2} + \varepsilon_Y) \\ &= \frac{1}{2} - 2\varepsilon_X \varepsilon_Y \ . \end{aligned}$$

Now, we denote by M the number of d-tuples $(q_{\alpha_1,t_1}, \dots, q_{\alpha_d,t_d})$ satisfying (3). The $k \times M$ matrix G' whose columns correspond to all vectors $(h_j)_{0 \leq j < M}$ obtained by (3) is a generator matrix of a linear binary code $\mathcal{C}'$ of length M and dimension k. The M-bit sequence $(z_j)_{0 \leq j < M}$ can be seen as the result of the transmission of the codeword $(u_0, \dots, u_{k-1})G'$ through a non-stationary binary channel, since the cross-over probability varies with j as pointed out by Formula (4). Note that we here make the assumption that the involved channel is memoryless, i.e., that the M positions in $\mathcal{C}'$ are independent. The validity of this assumption will be discussed in the next sections. Now, we can recover the first k bits of the LFSR initialization, $(u_0, \dots, u_{k-1})$ by applying the following ML-decoding algorithm. For any $\widehat{u} \in \mathbf{F}_2^k$, we compute

$$\sum_{j=0}^{M-1} (\widehat{u} \cdot h_j + z_j)\varepsilon_j \tag{5}$$

and we choose for $(u_0, \dots, u_{k-1})$ the vector $\widehat{u}$ which minimizes the above quantity.

We now sum up the algorithm used for the fast correlation attack.

Precomputation.

- *For all* $\alpha \in \mathbf{F}_2^n \setminus \{0\}$*, compute* $\mathcal{F}(f + \varphi_\alpha)$*.*
- *For all* $\alpha \in \mathbf{F}_2^n$ *such that* $\mathcal{F}(f + \varphi_\alpha) \neq 0$

  For all t*,* $0 \leq t < N$*, compute*

$$Q_{\alpha,t}(X) = (\sum_{i=1}^{n} \alpha_i X^{\gamma_i}) X^t \bmod P(X)$$

  and store all L*-bit vectors* $q_{\alpha,t}$ *corresponding to the coefficients of* $Q_{\alpha,t}$*.*
- *Find all sets of* d *vectors* $(q_{\alpha_1,t_1}, \dots, q_{\alpha_d,t_d})$ *whose sum vanishes on the last* $(L - k)$ *positions. For the* j*-th such set:*

  $E_j \leftarrow \prod_{i=1}^{d} \mathcal{F}(f + \varphi_{\alpha_i})$

  $z_j \leftarrow \sum_{i=1}^{n} s_{t_i} + c$ *where* $(-1)^c$ *corresponds to the sign of* E_j*.*

  $(h_{0,j}, \dots, h_{k-1,j}) \leftarrow \sum_{i=1}^{d} q_{\alpha_i,t_i}$*.*

Decoding step.

For all $\widehat{u} \in \mathbf{F}_2^k$*, compute*

$$\sum_{j=0}^{M-1} (\widehat{u} \cdot h_j + z_j) E_j \ .$$

Return the vector $\widehat{u}$ *which minimizes this quantity.*

Note that the attack can be slightly improved by using the modification proposed in [12].

4 Theoretical analysis

Here, we want to determine the average number N of consecutive bits of the running-key $(s_t)_{t \geq 0}$ required by the attack. Since any $\alpha \in \mathbf{F}_2^n$ such that $\mathcal{F}(f + \varphi_\alpha) \neq 0$ provides N vectors $q_{\alpha,t}$, $0 \leq t < N$, the average number of d-tuples $(q_{\alpha_1,t_1}, \dots, q_{\alpha_d,t_d})$ whose sum vanishes on the last $(L - k)$ positions is roughly

$$M \simeq \frac{(NF)^d}{d!\, 2^{L-k}} \qquad (6)$$

where F is the number of nonzero Walsh coefficients. Thus the ML-decoding procedure for the obtained code of length M and dimension k succeeds as soon as M satisfies $k/M \leq C$ where C is the capacity of the transmission channel. In the following, we assume that the M positions in $\mathcal{C}'$ are independent (otherwise the transmission channel is not memoryless). It means that any bit of the running-key is involved in at most one position of $(z_j)_{j<M}$. This condition

notably holds when $M \ll N$. Under this assumption, the transmission channel is a non-stationary binary symmetric channel whose cross-over probability is given by $p = 1/2 - \varepsilon$, where ε varies in a set $\mathcal{E}$. If μ_ε is the proportion of transmitted bits for which the cross-over probability equals $1/2 - \varepsilon$, we have

$$C = \sum_{\varepsilon \in \mathcal{E}} \mu_\varepsilon C\left(\frac{1}{2} - \varepsilon\right) \tag{7}$$

where $C(p)$ denotes the capacity of the stationary binary symmetric channel with cross-over probability p, i.e., $C(p) = 1 + p\log_2(p) + (1-p)\log_2(1-p)$. We use the following expression for any $\varepsilon < 1/2$:

$$C\left(\frac{1}{2} - \varepsilon\right) = \frac{1}{\ln(2)} \sum_{i>0} \frac{2^{2i}}{(2i-1)2i}\, \varepsilon^{2i} \ . \tag{8}$$

We first compute the capacity of the channel involved in our attack when $d = 2$. The M obtained equations can be split as follows: for any $w_1, w_2 \in \mathcal{W}$, $w_1 \leq w_2$, we find $M_{w_1w_2}$ equations derived from two vectors α_1 and α_2 such that $|\mathcal{F}(f + \varphi_{\alpha_1})| = w_1$ and $|\mathcal{F}(f + \varphi_{\alpha_2})| = w_2$. Thus,

$$M_{w_1w_2} = \frac{N^2 F_{w_1} F_{w_2}}{2^{L-k}} \text{ for } w_1 < w_2$$

and

$$M_{w^2} = \frac{N^2 F_w^2}{2^{L-k+1}} \text{ for } w \in \mathcal{W} \ .$$

The corresponding proportions are then

$$\mu_{w_1w_2} = \frac{2F_{w_1}F_{w_2}}{F^2} \text{ if } w_1 < w_2 \text{ and } \mu_{w^2} = \frac{F_w^2}{F^2} \ .$$

Moreover, we deduce from (4) that, for any $w_1, w_2 \in \mathcal{W}$, we have

$$Pr[z_j \neq h_j \cdot u] = \frac{1}{2} - \frac{w_1 w_2}{2^{2n+1}}$$

for a proportion $\mu_{w_1w_2}$ of the values of j. Formula (7) applied with the above proportions leads to the following capacity

$$\begin{aligned} C &= \sum_{w_1 \leq w_2} \mu_{w_1w_2}\, C\left(\frac{1}{2} - \frac{w_1w_2}{2^{2n+1}}\right) \\ &= \sum_{w \in \mathcal{W}} \frac{F_w^2}{F^2}\, C\left(\frac{1}{2} - \frac{w^2}{2^{2n+1}}\right) + \sum_{w_1 < w_2} \frac{2F_{w_1}F_{w_2}}{F^2}\, C\left(\frac{1}{2} - \frac{w_1w_2}{2^{2n+1}}\right) \ . \end{aligned} \tag{9}$$

Expression (8) for the capacity of the binary symmetric channel leads to

$$\begin{aligned} C &= \frac{1}{\ln(2)F^2}\left[\sum_{w\in\mathcal{W}} F_w^2 \left(\sum_{i>0} \frac{1}{(2i-1)2i}\, \frac{w^{4i}}{2^{4ni}}\right) + 2\sum_{w_1<w_2} F_{w_1}F_{w_2}\left(\sum_{i>0} \frac{1}{(2i-1)2i}\, \frac{w_1^{2i}w_2^{2i}}{2^{4ni}}\right)\right] \\ &= \frac{1}{\ln(2)F^2} \sum_{i>0} \frac{1}{(2i-1)2i} \left(\frac{\sum_{w\in\mathcal{W}} F_w w^{2i}}{2^{2ni}}\right)^2 \ . \end{aligned}$$

Now, we have for any $i > 0$

$$\sum_{w \in \mathcal{W}} F_w w^{2i} = \sum_{\alpha \in \mathbf{F}_2^n} \mathcal{F}^{2i}(f + \varphi_\alpha) \ .$$

Note that we here use the fact that the filtering function is balanced. For $i = 1$, Parseval's relation which holds for any Boolean function leads to

$$\sum_{w \in \mathcal{W}} F_w w^2 = 2^{2n} \ .$$

Therefore, we deduce that

$$C \geq \frac{1}{2\ln(2)F^2} \ .$$

Moreover, for any $i \geq 2$, we have

$$\begin{aligned}\sum_{\alpha \in \mathbf{F}_2^n} \mathcal{F}^{2i}(f + \varphi_\alpha) &\leq \mathcal{L}(f)^{2(i-1)} \sum_{\alpha \in \mathbf{F}_2^n} \mathcal{F}^2(f + \varphi_\alpha) \\ &\leq 2^{2n} \mathcal{L}(f)^{2(i-1)} \ ,\end{aligned}$$

where $\mathcal{L}(f) = \max_{\alpha \in \mathbf{F}_2^n} |\mathcal{F}(f + \varphi_\alpha)|$. It follows that, for any $i \geq 2$,

$$\frac{1}{2^{2ni}} \sum_{w \in \mathcal{W}} F_w w^{2i} \leq \left(\frac{\mathcal{L}(f)}{2^n} \right)^{2(i-1)} \leq 1$$

where equality holds if and only if $\mathcal{L}(f) = \pm 2^n$, i.e., if and only if f is an affine function. It implies that, if $\deg(f) > 1$, we have

$$\begin{aligned}C &= \frac{1}{\ln(2)F^2} \sum_{i>0} \frac{1}{(2i-1)2i} \left(\frac{\sum_{w \in \mathcal{W}} F_w w^{2i}}{2^{2ni}} \right)^2 \\ &< \frac{1}{\ln(2)F^2} \sum_{i>0} \frac{1}{(2i-1)2i} = \frac{1}{F^2} \ .\end{aligned}$$

Therefore, the capacity of the transmission channel satisfies

$$\frac{1}{2\ln(2)F^2} \leq C < \frac{1}{F^2}$$

when $\deg(f) > 1$. Using Relation (6) for $d = 2$, we deduce that the minimum number $N_{\min}$of known running-key bits required for the attack satisfies

$$M = \frac{N_{\min}^2 F^2}{2^{L-k+1}} = \frac{k}{C} \ .$$

This implies that

$$\sqrt{2k}\, 2^{\frac{L-k}{2}} < N_{\min} \leq 2\sqrt{k \ln(2)}\, 2^{\frac{L-k}{2}} \ . \qquad (10)$$

Similarly, we obtain the following result for all values of parameter d in the attack.

Theorem 1. *For any balanced filtering function f such that $\deg(f) > 1$, the capacity of the non-stationary binary symmetric channel involved in the general fast correlation attack with parameter d satisfies*

$$\frac{1}{2\ln(2)F^d} \leq C < \frac{1}{F^d} \ . \tag{11}$$

Therefore, the minimum number of bits of the running-key required by the attack satisfies

$$(d!k)^{\frac{1}{d}}\ 2^{\frac{L-k}{d}} < N_{\min} \leq (2\ln(2)d!k)^{\frac{1}{d}}\ 2^{\frac{L-k}{d}} \ ,$$

assuming that the M positions in $\mathcal{C}'$ are independent.

Proof. For any d-tuple $(q_{\alpha_1,t_1},\dots,q_{\alpha_d,t_d})$, we denote by d_w, $w \in \mathcal{W}$ the number of α_i, $1 \leq i \leq d$ such that $|\mathcal{F}(f+\varphi_{\alpha_i})| = w$. Then, we have $\sum_{w\in\mathcal{W}} d_w = d$. For a given vector $(d_w)_{w\in\mathcal{W}}$, the number of corresponding d-tuples which vanish on the last $(L-k)$ positions is

$$M_{\prod w^{d_w}} = \frac{N^d \prod_{w\in\mathcal{W}} F_w^{d_w}}{2^{L-k}\prod_{w\in\mathcal{W}}(d_w)!}$$

and the corresponding proportion of such d-tuples is

$$\mu_{\prod w^{d_w}} = \frac{d!\ \prod_{w\in\mathcal{W}} F_w^{d_w}}{F^d \prod_{w\in\mathcal{W}}(d_w)!} \ .$$

Moreover, we get from (4) that any such d-tuple corresponds to the cross-over probability

$$\frac{1}{2} - 2^{d-1}\frac{\prod_{w\in\mathcal{W}} w^{d_w}}{2^{(n+1)d}} \ .$$

Thus, assuming that all positions in $\mathcal{C}'$ are independent, we obtain the following expression for the capacity of the transmission channel

$$C = \sum_{(d_w)} \mu_{\prod w^{d_w}}\ C\left(\frac{1}{2} - 2^{d-1}\frac{\prod_{w\in\mathcal{W}} w^{d_w}}{2^{(n+1)d}}\right)$$

where the summation takes place over all vectors $(d_w)_{w\in\mathcal{W}}$ of positive or zero integers such that $\sum_{w\in\mathcal{W}} d_w = d$. Then, Expression (8) and the multinomial identity lead to

$$\begin{aligned} C &= \frac{1}{\ln(2)F^d}\left[\sum_{(d_w)} \frac{d!}{\prod_{w\in\mathcal{W}} d_w!} \prod_{w\in\mathcal{W}} F_w^{d_w}\left(\sum_{i>0}\frac{1}{(2i-1)2i}\ \frac{\prod_{w\in\mathcal{W}} w^{2id_w}}{2^{2ndi}}\right)\right] \\ &= \frac{1}{\ln(2)F^d}\left[\sum_{i>0}\frac{1}{(2i-1)2i}\left(\frac{\sum_{w\in\mathcal{W}} F_w w^{2i}}{2^{2ni}}\right)^d\right] \end{aligned}$$

Now, we use that

$$\sum_{w\in\mathcal{W}} F_w w^2 = 2^{2n}$$

and that, for any $i \geq 2$ and any balanced Boolean function of degree at least 2,

$$\sum_{w\in\mathcal{W}} F_w w^{2i} < 2^{2ni} \ .$$

Therefore, we deduce that

$$\frac{1}{2\ln(2)F^d} \leq C < \frac{1}{F^d} \ .$$

Finally, we derive the minimum length required for the running-key by combining the previous inequalities and Formula (6).

Most notably, this result points out that the Walsh spectrum of the filtering function and its number of variables has only a minor influence on the length of the running-key required by the attack. Note that the upper bound on $N_{\min}$ given in Theorem 1 provides a good approximation of $N_{\min}$ in most practical situations since the nonlinearity of the filtering function is usually high.

But, it may happen that the M positions in $\mathcal{C}'$ are not independent. In that case, the transmission channel is not a memoryless channel anymore and the previous result on its capacity does not hold. However, simulations show that the attack still performs well and that the value of $N_{\min}$ given in Theorem 1 still provides a good approximation of the required running-key length (see Section 6).

5 Computational complexity of the attack

In this section, we focus on the computational complexity of the attack. In the precomputation part, we have to find all d-tuples $(q_{\alpha_1,t_1},\dots,q_{\alpha_d,t_d})$ whose sum vanishes on the last $(L-k)$ positions. We usually use the following technique: we store all $q_{\alpha,t}$ in a hash table indexed by their values on the last $(L-k)$ positions. In this case, the number of operations required by the precomputation corresponds to the number of $(d-1)$-tuples of vectors $q_{\alpha,t}$, i.e.,

$$T_p = \frac{(NF)^{d-1}}{(d-1)!} \ .$$

We may also obtain a better time-memory trade-off if we use an algorithm based on a "birthday technique" as suggested in [15, Section 5]. This consists in storing in a table the values of all linear combinations of d' vectors $q_{\alpha,t}$ where $d' < d$. The time complexity of the precomputation is now of order $\binom{NF}{d-d'}$ but the required memory is of order $\binom{NF}{d'}$.

In the decoding part of the attack, we need to compute Expression (5) for all $\widehat{u} \in \mathbf{F}_2^k$. Thus, the decoding complexity is of order $M\ 2^k$. Here, we suppose as in [13] that the filtering function has a high nonlinearity. Therefore, we have that the capacity of the transmission channel is roughly $C \simeq \frac{1}{2\ln(2)F^d}$. Since $M/k \simeq 1/C$, we derive that the number of operations performed by the ML-decoding procedure is of order

$$T_d = 2\ln(2)k2^kF^d\ .$$

For fixed values of d and k, the only influence of the filtering function on the computational complexity of the attack is that the running-times of both precomputation and decoding parts increase with the number of nonzero Walsh coefficients. Parseval's relation implies that

$$2^{2n} = \sum_{\alpha \in \mathbf{F}_2^n} \mathcal{F}^2(f + \varphi_\alpha) \leq F\mathcal{L}(f)^2\ .$$

It follows that the computation time increases with the number of variables and with the nonlinearity of the filtering function.

We can now compare the performance of our attack with the attack proposed in [13]. Both attacks are obviously similar when the filtering function has a three-valued extended Walsh spectrum [4], i.e., when all nonzero Walsh coefficients of f are equal to $\pm\mathcal{L}(f)$. When we use the extremal Walsh coefficients only as in [13], the attack requires the following number of bits of the running-key [13]:

$$N^{(JJ)} = \frac{1}{F_{\mathcal{L}(f)}}(2\ln(2)d!k)^{\frac{1}{d}}\ 2^{\frac{L-k}{d}}\ \left(\frac{2^{2n}}{\mathcal{L}(f)^2}\right)\ .$$

Therefore, we deduce that

$$\frac{N^{(JJ)}}{N} = \frac{2^{2n}}{\mathcal{L}(f)^2F_{\mathcal{L}(f)}} \geq 1\ ,$$

because Parseval's relation implies that

$$2^{2n} = \sum_{\alpha \in \mathbf{F}_2^n} \mathcal{F}^2(f + \varphi_\alpha) \geq F_{\mathcal{L}(f)}\mathcal{L}(f)^2\ .$$

The running-time of the decoding step in [13] is given by

$$T_d^{(JJ)} = 2\ln(2)k2^k\left(\frac{2^n}{\mathcal{L}(f)}\right)^{2d}\ .$$

Therefore, we have

$$\frac{T_d^{(JJ)}}{T_d} = \left(\frac{2^{2n}}{\mathcal{L}(f)^2F}\right)^d \leq 1\ .$$

We similarly obtain for the complexity of the precomputation step

$$\frac{T_p^{(JJ)}}{T_p} = \left(\frac{N^{(JJ)}F_{\mathcal{L}(f)}}{NF}\right)^{d-1} = \left(\frac{2^{2n}}{\mathcal{L}(f)^2 F}\right)^{d-1} \leq 1 \ .$$

Then, our attack needs a smaller subsequence of the running-key than the attack proposed in [13], but its running-time is higher. For the keystream generator LILI-128 [7], both attacks are very similar since most nonzero Walsh coefficients of the filtering function f are equal to $\pm\mathcal{L}(f)$. But, our attack provides a significant improvement especially when the proportion of extremal Walsh coefficients amongst all nonzero values is small. For example, let us consider the following filtering function of n variables, n odd

$$f(x_1, \dots, x_n) = x_1x_2x_3 + x_2x_3x_4 + x_2x_3x_5 + x_1 + x_2 + x_3 + \sum_{i=3}^{\frac{n-1}{2}} x_{2i}x_{2i+1} \ .$$

This function is derived from a 5-variable function described in [2] by adding a bent function. We have $\mathcal{L}(f) = 3 \cdot 2^{(n+1)/2}$ and $F_{\mathcal{L}(f)} = 2^{n-5}$. Then, we deduce that

$$\frac{N^{(JJ)}}{N} = \frac{2^{2n}}{\mathcal{L}(f)^2 F_{\mathcal{L}(f)}} = \frac{16}{9} \ .$$

For a LFSR of length 40, our attack with $d = 2$ and $k = 20$ requires the knowledge of $7,625$ bits of the running-key, whereas the attack presented in [13] needs $13,556$ bits.

6 Simulation results

We present some simulation results for a LFSR of length 40. We use the following parameters: $d = 2$ and $k = 20$. By applying Formula (10) with these values, we obtain that the minimum length of the running-key required for the attack satisfies

$$6476 < N_{\min} \leq 7625 \ ,$$

where the upper bound is tight when the filtering function has a high nonlinearity. Then, we try to recover the first 20 bits of the initialization of this generator for different balanced filtering functions of 5, 6 and 7 variables. We choose for γ a full positive difference set with $\gamma_1 = L$ as recommended in [9]. All success rates presented below have been computed over 500 trials. The running-times are given for a DEC workstation with an alpha EV6 processor at 500 MHz.

All considered filtering functions are balanced. Functions (I) to (VII) in the following table depend on 5 variables. Their Walsh spectra are given in [2]. The

7-variable function (IX) is a 2-resilient function with maximal nonlinearity. Its algebraic normal form is

$$\begin{aligned}
&x_1 + x_2 + x_1x_4 + x_3x_4 + x_5 + x_2x_5 + x_3x_5 + x_1x_4x_5 + x_2x_4x_5 + x_3x_4x_5 \\
&+x_6 + x_4x_6 + x_1x_4x_6 + x_2x_7 + x_1x_2x_7 + x_2x_3x_7 + x_1x_2x_3x_7 + x_4x_7 \\
&+x_1x_2x_4x_7 + x_3x_4x_7 + x_1x_3x_4x_7 + x_5x_7 + x_1x_5x_7 + x_1x_2x_5x_7 + x_1x_3x_5x_7 \\
&+x_1x_4x_5x_7 + x_2x_4x_5x_7 + x_3x_4x_5x_7 + x_6x_7 + x_2x_6x_7 + x_1x_2x_6x_7 + x_3x_6x_7 \\
&+x_1x_3x_6x_7 + x_1x_4x_6x_7 + x_2x_4x_6x_7 + x_3x_4x_6x_7 + x_5x_6x_7 + x_2x_5x_6x_7 \\
&+x_3x_5x_6x_7 \ .
\end{aligned}$$

This function was obtained by the technique described in [17].

	N	M	expected M	precomp. time	decoding time	success rate
(I)	$n = 5,\ \ f = x_1x_2x_3 + x_1x_4 + x_2x_5 + x_3$ $\mathcal{NL}(f) = 12$ and $F = 16$ $F_0 = 16,\ F_8 = 16$					
	7625	6995	7097	1 s	24 s	67.4 %
	7000	5929	5981	1 s	20 s	51.6 %
(II)	$n = 5,\ \ f = x_1x_2x_3 + x_1x_2x_4 + x_1x_2x_5 + x_1x_4 + x_2x_5 + x_3 + x_4 + x_5$ $\mathcal{NL}(f) = 12$ and $F = 16$, 1-resilient $F_0 = 16,\ F_8 = 16$					
	7625	7163	7097	1 s	24 s	66.8 %
	7000	6011	5981	1 s	20 s	50.4 %
(III)	$n = 5,\ \ f = x_2x_3x_4x_5 + x_1x_2x_3 + x_2x_4 + x_3x_5 + x_4 + x_5$ $\mathcal{NL}(f) = 12$ and $F = 28$ $F_0 = 4,\ F_4 = 16,\ F_8 = 12$					
	7625	$22,197$	$21,735$	2 s	1.3 min	66.4 %
	7000	$18,730$	$18,318$	2 s	1.1 min	49.8 %
(IV)	$n = 5,\ \ f = x_1x_2x_3 + x_1x_4x_5 + x_2x_3 + x_1$ $\mathcal{NL}(f) = 8$ and $F = 13$ $F_0 = 19,\ F_8 = 12,\ F_{16} = 1$					
	7625	5665	4687	1 s	20 s	55.5 %
	7000	4972	3949	1 s	17 s	46.2 %
(V)	$n = 5,\ \ f = x_2x_3x_4x_5 + x_2x_3 + x_1$ $\mathcal{NL}(f) = 6$ and $F = 16$ $F_0 = 16,\ F_4 = 12,\ F_{12} = 3,\ F_{20} = 1$					
	7625	7049	7097	1 s	25 s	82.2 %
	7000	5893	5981	1 s	21 s	75.2 %

	N	M	expected M	precomp. time	decoding time	success rate
(VI)	$n = 5,\ \ f = x_1x_2x_3x_5 + x_2x_3 + x_4$ $\mathcal{NL}(f) = 6$ and $F = 16$ $F_0 = 16$, $F_4 = 12$, $F_{12} = 3$, $F_{20} = 1$					
	7625	7041	7097	1 s	25 s	77.8 %
	7000	5939	5981	1 s	21 s	64.2 %
(VII)	$n = 5,\ \ f = x_1x_2x_3 + x_2x_3x_4 + x_2x_3x_5 + x_1 + x_2 + x_3$ $\mathcal{NL}(f) = 4$ and $F = 8$ $F_0 = 24$, $F_8 = 7$, $F_{24} = 1$					
	7625	1964	1774	1 s	7 s	78.2 %
	7000	1661	1495	1 s	6 s	51.8 %
(VIII)	$n = 6,\ \ f = x_1x_2x_3 + x_2x_3x_6 + x_1x_2 + x_3x_4 + x_5x_6 + x_4 + x_5$ $\mathcal{NL}(f) = 24$ and $F = 40$ $F_0 = 24$, $F_8 = 32$, $F_{16} = 8$					
	7625	$45,006$	$44,358$	4 s	2.6 min	67.3 %
	7000	$38,031$	$37,384$	4 s	2.2 min	52.8 %
(IX)	$n = 7,\ \ f$ $\mathcal{NL}(f) = 56$ and $F = 64$, 2-resilient $F_0 = 64$, $F_{16} = 64$					
	7625	$114,846$	$113,556$	8 s	6.5 min	66.8 %
	7000	$96,750$	$95,703$	7 s	5.5 min	48.8 %
(X)	$n = 7,\ \ f = x_1x_2x_3 + x_2x_3x_4 + x_2x_3x_5 + x_1 + x_2 + x_3 + x_6x_7$ $\mathcal{NL}(f) = 40$ and $F = 32$ $F_0 = 96$, $F_{16} = 28$, $F_{48} = 4$					
	7625	$28,526$	$28,389$	3 s	1.6 min	64.6 %
	7000	$23,954$	$23,926$	3 s	1.3 min	52.6 %

All results presented in the above table confirm the validity of the previous approach. First, we observe that the approximation of $N_{\min}$ derived from the assumption that the transmission channel is memoryless seems to be still accurate when the positions in $\mathcal{C}'$ are not independent.

Moreover, when the attacker knows N consecutive bits of the running-key, where N is given by the upper bound in Formula (10), then the success rate of the attack is around 65 %. It clearly appears that the required running-key length is almost independent of the number of variables of the filtering function. However, we observe that the success rate increases when the nonlinearity of the function is very small, especially for 5-variable functions (see Functions (V)-(VII)). The reason is that the upper bound in (10) uses the following approximation for the capacity of the binary symmetric channel

$$C\left(\frac{1}{2}-\varepsilon\right) \simeq \frac{2\varepsilon^2}{\ln(2)} \ , \tag{12}$$

which is not accurate for large values of ε. If we consider e.g. the 5-variable function (VII) with $\mathcal{NL}(f) = 4$, computing Formula (9) with the exact expressions of the capacities of all involved binary symmetric channels leads to

$$C = 0.1152 \ ,$$

instead of 0.1127 obtained by the upper bound in Formula (11). With this modified value for C, we now get a slightly lower value for N: $N = 7542$ instead of 7625. This minor influence of the nonlinearity of the filtering function tends to vanish for a higher number of variables. Let us consider two filtering functions, f of n variables and g of $(n+2i)$ variables, with "similar" Walsh spectra, e.g.

$$g(x_1, \ldots, x_{n+2i}) = f(x_1, \ldots, x_n) + h(x_{n+1}, \ldots, x_{n+2i})$$

where h is a bent function of $(2i)$ variables. Then, the Walsh spectrum of g can be easily deduced from the Walsh spectrum of f [3, 8]. Most notably, we have $\mathcal{L}(g) = 2^i \mathcal{L}(f)$. Therefore, the lowest value of the cross-over probability of the transmission channel when g is used is

$$\frac{1}{2} - \frac{\mathcal{L}(g)^2}{2^{2n+4i+1}} = \frac{1}{2} - \frac{\mathcal{L}(f)^2}{2^{2n+2i+1}} \ .$$

Thus, Approximation (12) of the capacity is more accurate for g than for f, especially when i is large. This can be observed by comparing the success rates obtained for the 5-variable function (VII) and for the 7-variable function (X) which have similar Walsh spectra.

7 Conclusions

This study points out that the performance of fast correlation attacks on any filter generator can be improved by using all nonzero Walsh coefficients of the filtering function. Our main result is that the running-key length which guarantees a successful attack does not depend on the filtering function, except for functions which are very close to an affine function. The only influence of the filtering function is that the computational complexity of the attack increases with the number of nonzero Walsh coefficients. Therefore, the choice of the Boolean function in the design of a filter generator should be mostly conditioned by other types of attacks as inversion attacks.

References

1. R. J. Anderson. Searching for the optimum correlation attack. In *Fast Software Encryption 1994*, number 1008 in Lecture Notes in Computer Science, pages 137–143. Springer-Verlag, 1995.

2. E.R. Berlekamp and L.R. Welch. Weight distributions of the cosets of the (32,6) Reed-Muller code. *IEEE Trans. Inform. Theory*, 18(1):203–207, 1972.
3. R.A. Brualdi, N. Cai, and V.S. Pless. Orphan structure of the first-order Reed-Muller codes. *Discr. Math.*, (102):239–247, 1992.
4. A. Canteaut, C. Carlet, P. Charpin, and C. Fontaine. Propagation characteristics and correlation-immunity of highly nonlinear Boolean functions. In *Advances in Cryptology - EUROCRYPT 2000*, number 1807 in Lecture Notes in Computer Science, pages 507–522. Springer-Verlag, 2000.
5. A. Canteaut and M. Trabbia. Improved fast correlation attacks using parity-check equations of weight 4 and 5. In *Advances in Cryptology - EUROCRYPT 2000*, number 1807 in Lecture Notes in Computer Science, pages 573–588. Springer-Verlag, 2000.
6. V. Chepyshov, T. Johansson, and B. Smeets. A simple algorithm for fast correlation attacks on stream ciphers. In *Fast Software Encryption 2000*, number 1978 in Lecture Notes in Computer Science. Springer-Verlag, 2000.
7. E. Dawson, A. Clark, J. Dj. Golić, W. Millan, L. Penna, and L. Simpson. The LILI-128 keystream generator. In *Proceedings of the first NESSIE Worksop*, 2000.
8. C. Fontaine. On some cosets of the First-Order Reed-Muller code with high minimum weight. *IEEE Trans. Inform. Theory*, 45(4):1237–1243, 1999.
9. J. Dj. Golić. On the security of nonlinear filter generators. In *Fast Software Encryption 1996*, number 1039 in Lecture Notes in Computer Science, pages 173–188. Springer-Verlag, 1996.
10. T. Johansson and F. Jönsson. Fast correlation attacks based on turbo code techniques. In *Advances in Cryptology - CRYPTO'99*, number 1666 in Lecture Notes in Computer Science, pages 181–197. Springer-Verlag, 1999.
11. T. Johansson and F. Jönsson. Improved fast correlation attack on stream ciphers via convolutional codes. In *Advances in Cryptology - EUROCRYPT'99*, number 1592 in Lecture Notes in Computer Science, pages 347–362. Springer-Verlag, 1999.
12. T. Johansson and F. Jönsson. Fast correlation attacks through reconstruction of linear polynomials. In *Advances in Cryptology - CRYPTO'00*, number 1880 in Lecture Notes in Computer Science, pages 300–315. Springer-Verlag, 2000.
13. F. Jönsson and T. Johansson. A fast correlation attack on LILI-128. *Information Processing Letters*, 81(3):127–132, February 2002.
14. S. Leveiller, J. Boutros, P. Guillot, and G. Zémor. Cryptanalysis of nonlinear filter generators with $\{0,1\}$-metric Viterbi decoding. In *Cryptography and Coding - 8th IMA International Conference*, number 2260 in Lecture Notes in Computer Science, pages 402–414. Springer-Verlag, 2001.
15. W. Meier and O. Staffelbach. Fast correlation attack on certain stream ciphers. *J. Cryptology*, pages 159–176, 1989.
16. M. J. Mihaljevic, M. P.C. Fossorier, and H. Imai. A low-complexity and high performance algorithm for the fast correlation attack. In *Fast Software Encryption 2000*, number 1978 in Lecture Notes in Computer Science. Springer-Verlag, 2000.
17. E. Pasalic, S. Maitra, T. Johansson, and P. Sarkar. Now constructions of resilient and correlation-immune functions achieving upper bound on nonlinearity. In *Proceedings of the International Worshop on Coding and Cryptography - WCC 2001*, 2001.
18. R.A. Rueppel. *Analysis and Design of stream ciphers*. Springer-Verlag, 1986.
19. T. Siegenthaler. Decrypting a class of stream ciphers using ciphertext only. *IEEE Trans. Computers*, C-34(1):81–84, 1985.