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

Abstract. This paper presents a generalization of the fast correlation attack presented by Chepyshov, Johansson and Smeets, 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 have only a minor influence on the length of the running-key subsequence needed for the attack.

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. Most keystream generators are composed of linear feedback shift registers (LFSRs). Therefore, they are vulnerable to *correlation attacks* [9]. 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. Meier and Staffelbach [7] 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 the result of the transmission of the LFSR output subsequence through a particular channel. In practice, 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 [7, 1, 2, 8, 5, 4] consist in decoding the running-key subsequence relatively to the LFSR code.

Here, 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. Some linear relations between the running-key bits and the LFSR initial state are derived from this approximation. Therefore, the involved transmission channel is a binary symmetric channel with cross-over probability $\mathcal{NL}(f)/2^n$ where n is the number of variables of the filtering function and $\mathcal{NL}(f)$ is its nonlinearity. Very recently,

Jönsson and Johansson [6] observed that the required running-key length 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 is 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 [2]. The main modification is that the involved transmission channel is now a non-stationary 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. We do not investigate other cryptanalysis techniques like inversion attacks [3].

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, $P(X) = \sum_{i=0}^{L} \lambda_i X^i$. Then, the output $(u_t)_{t\geq 0}$ of the LFSR satisfies the following recursion:

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

where $(u_0, \ldots, u_{L-1})$ is the LFSR initial state. Let f be a *balanced* Boolean function of n variables and let $(\gamma_i)_{1\leq i\leq n}$ be a decreasing sequence of nonnegative integers. Then, the output of the filter generator $(s_t)_{t\geq 0}$ is given by

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

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 $f(x) = 1$. Therefore, the *Walsh spectrum* of f is the multiset $\{\mathcal{F}(f+\varphi_\alpha), \alpha \in \mathbf{F}_2^n\}$. Note that f is *balanced* if and only if $\mathcal{F}(f) = 0$. 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

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

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.

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 $t \geq 0$

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

Then, we choose c such that $(-1)^c$ is equal to the sign of $\mathcal{F}(f+\varphi_\alpha)$. We obtain this way a set of F linear relations between s_t and some stages of the LFSR.

3 A general fast correlation attack

Now, we use the technique proposed by Chepyshov, Johansson and Smeets [2] 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 [6] 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, $(u_0, \ldots, u_{L-1})$: $u_t = \sum_{i=0}^{L-1} \lambda_i^{(t)} u_i$, where the involved coefficients $(\lambda_i^{(t)})_{0 \leq i < L}$ are obtained by $\sum_{i=0}^{L-1} \lambda_i^{(t)} X^i = X^t \bmod P(X)$. Then, we deduce that, for any $\alpha \in \mathbf{F}_2^n \setminus \{0\}$ and for any $t \geq 0$,

$$\sum_{i=1}^{n} \alpha_i u_{t+\gamma_i} = \sum_{i=1}^{n} \alpha_i \sum_{j=0}^{L-1} \lambda_j^{(t+\gamma_i)} u_j = \sum_{j=0}^{L-1} u_j \left(\sum_{i=1}^{n} \alpha_i \lambda_j^{(t+\gamma_i)} \right) = \sum_{j=0}^{L-1} u_j q_j \ .$$

It clearly appears that the coefficients $(q_j)_{0 \leq j < L}$ correspond to

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

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). It was proposed in [2] 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. For the j-th set of d such columns of G, namely $(q_{\alpha_1,t_1}, \ldots, q_{\alpha_d,t_d})$, we have

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

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) that, for $u = (u_0, \ldots, u_{k-1})$,

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

Let M be the number of d-tuples $(q_{\alpha_1,t_1}, \ldots, q_{\alpha_d,t_d})$ satisfying (3). The $k \times M$ matrix G' whose columns correspond to all $(h_j)_{0 \leq j < M}$ is a generator matrix of a 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 $(u_0, \ldots, u_{k-1})G'$ through a non-stationary binary channel, since the cross-over probability varies with j. We here assume that the 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 by applying a ML-decoding algorithm. Now, we sum up the algorithm used for the attack.

Precomputation.

- *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 $Q_{\alpha,t}$ defined by (2) and store all L-bit vectors $q_{\alpha,t}$ corresponding to its coefficients.
- *Find all sets of d vectors $(q_{\alpha_1,t_1}, \ldots, q_{\alpha_d,t_d})$ 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.
  $(h_{0,j}, \ldots, h_{k-1,j}) \leftarrow \sum_{i=1}^{d} q_{\alpha_i,t_i}$.

Decoding step.

Return the vector $\widehat{u} \in \mathbf{F}_2^k$ which minimizes

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

4 Theoretical analysis

We want to determine the average number N of 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 $q_{\alpha,t}$, the average number of d-tuples $(q_{\alpha_1,t_1}, \ldots, q_{\alpha_d,t_d})$ whose sum vanishes on the last $(L-k)$ positions is roughly

$$M \simeq \frac{(NF)^d}{d!\, 2^{L-k}} \tag{5}$$

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 $k/M \leq C$ where C is the capacity of the transmission channel. In

the following, we assume that the M positions in $\mathcal{C}'$ are independent. Then, 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 $C = \sum_{\varepsilon \in \mathcal{E}} \mu_\varepsilon C(\frac{1}{2} - \varepsilon)$, where $C(p)$ is 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 that, for any $\varepsilon < 1/2$,

$$C\left(\frac{1}{2} - \varepsilon\right) = \frac{1}{\ln(2)} \sum_{i>0} \frac{2^{2i}}{(2i-1)2i}\, \varepsilon^{2i} \ . \tag{6}$$

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 $M_{w_1 w_2}$ equations derived from two vectors α_1 and α_2 such that $|\mathcal{F}(f + \varphi_{\alpha_1})| = w_1$ and $|\mathcal{F}(f + \varphi_{\alpha_2})| = w_2$. The corresponding proportions are

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

Thus, we derive from (4) that

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

For $i = 1$, Parseval's relation 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

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

where equality holds if and only if $\mathcal{L}(f) = \pm 2^n$, i.e., if f is an affine function. Therefore, the capacity of the transmission channel satisfies

$$\frac{1}{2\ln(2)F^2} \leq C < \frac{1}{\ln(2)F^2} \sum_{i>0} \frac{1}{(2i-1)2i} = \frac{1}{F^2}$$

when $\deg(f) > 1$. Using Relation (5) for $d = 2$, we deduce that the minimum number $N_{\min}$of known running-key bits required for the attack satisfies

$$\sqrt{2k}\, 2^{\frac{L-k}{2}} < N_{\min} \leq 2\sqrt{k\ln(2)}\, 2^{\frac{L-k}{2}} \ . \tag{7}$$

We obtain a similar 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 channel involved in the attack with parameter d satisfies*

$$\frac{1}{2\ln(2)F^d} \leq C < \frac{1}{F^d} \ , \tag{8}$$

assuming that the M positions in $\mathcal{C}'$ are independent. Under this assumption, the minimum number of bits of the running-key required by the attack satisfies

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

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 $N_{\min}$ is tight 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.

5 Computational complexity of the attack

In the precomputation part, we have to find all d-tuples $(q_{\alpha_1,t_1}, \ldots, q_{\alpha_d,t_d})$ whose sum vanishes on the last $(L-k)$ positions. Thus, the number of operations required by the precomputation is $T_p = (NF)^{d-1} \ / \ (d-1)!$. We may also obtain a better time-memory trade-off if we use an algorithm based on a "birthday technique" as suggested in [7, Section 5].

The decoding complexity is of order $M\ 2^k$. If the filtering function has a high nonlinearity, the capacity is roughly $C \simeq \frac{1}{2\ln(2)F^d}$. Since $M \simeq k/C$, we derive that the number of operations performed by the ML-decoding procedure is of order $T_d = 2\ln(2)k2^kF^d$. Thus, for fixed values of d and k, the running-times of both precomputation and decoding parts increase with the number of nonzero Walsh coefficients.

Now, we compare the performance of our attack with the attack proposed in [6]. Both attacks are obviously similar when the filtering function has a three-valued extended Walsh spectrum, i.e., when all nonzero Walsh coefficients of f are equal to $\pm\mathcal{L}(f)$. Let $N^{(JJ)}$ be the number of running-key bits required by the attack proposed in [6], which uses the extremal Walsh coefficients only. Then, we obtain

$$\frac{N^{(JJ)}}{N} = \frac{2^{2n}}{\mathcal{L}(f)^2 F_{\mathcal{L}(f)}} \geq 1 \ . \tag{9}$$

If we compare the running times of both attacks, we have

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

Then, our attack needs a smaller running-key subsequence than the attack proposed in [6], but its running-time is higher. Our attack provides a significant improvement especially when the proportion of extremal Walsh coefficients amongst all nonzero values is small. For example, for $f = x_1x_2x_3 + x_2x_3x_4 + x_2x_3x_5 + x_1 + x_2 + x_3 + \sum_{i=3}^{\frac{n-1}{2}} x_{2i}x_{2i+1}$, n odd, we have $\mathcal{L}(f) = 3 \cdot 2^{(n+1)/2}$ and $F_{\mathcal{L}(f)} = 2^{n-5}$. Then, we deduce from (9) that $N^{(JJ)}/N = 16/9$.

6 Simulation results

We present some simulation for a LFSR of length 40. We use $d = 2$ and $k = 20$. By applying (7) 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 $\gamma_1 = L$. All success rates presented below have been computed over 500 trials on a DEC-alpha workstation at 500 MHz.

	N	M	expected M	precomp. time	decoding time	success rate
(I)	$n = 5$, $f = x_1x_2x_3 + x_1x_2x_4 + x_1x_2x_5 + x_1x_4 + x_2x_5 + x_3 + x_4 + x_5$ $\mathcal{NL}(f) = 12$, $F = 16$, 1-resilient and $F_0 = 16$, $F_8 = 16$					
	7625	7163	7097	1 s	24 s	66.8 %
	7000	6011	5981	1 s	20 s	50.4 %
(II)	$n = 5$, $f = x_2x_3x_4x_5 + x_1x_2x_3 + x_2x_4 + x_3x_5 + x_4 + x_5$ $\mathcal{NL}(f) = 12$, $F = 28$ and $F_0 = 4$, $F_4 = 16$, $F_8 = 12$					
	7625	22,197	21,735	2 s	1.3 min	66.4 %
	7000	18,730	18,318	2 s	1.1 min	49.8 %
(III)	$n = 5$, $f = x_2x_3x_4x_5 + x_2x_3 + x_1$ $\mathcal{NL}(f) = 6$, $F = 16$ and $F_0 = 16$, $F_4 = 12$, $F_{12} = 3$, $F_{20} = 1$					
	7625	7049	7097	1 s	25 s	82.2 %
	7000	5893	5981	1 s	21 s	75.2 %
(IV)	$n = 5$, $f = x_1x_2x_3 + x_2x_3x_4 + x_2x_3x_5 + x_1 + x_2 + x_3$ $\mathcal{NL}(f) = 4$, $F = 8$ and $F_0 = 24$, $F_8 = 7$, $F_{24} = 1$					
	7625	1964	1774	1 s	7 s	78.2 %
	7000	1661	1495	1 s	6 s	51.8 %
(V)	$n = 6$, $f = x_1x_2x_3 + x_2x_3x_6 + x_1x_2 + x_3x_4 + x_5x_6 + x_4 + x_5$ $\mathcal{NL}(f) = 24$, $F = 40$ and $F_0 = 24$, $F_8 = 32$, $F_{16} = 8$					
	7625	45,006	44,358	4 s	2.6 min	67.3 %
	7000	38,031	37,384	4 s	2.2 min	52.8 %

	N	M	expected M	precomp. time	decoding time	success rate
(VI)	$n = 7,\ f$ $\mathcal{NL}(f) = 56$, $F = 64$, 2-resilient and $F_0 = 64$, $F_{16} = 64$					
	7625	114,846	113,556	8 s	6.5 min	66.8 %
	7000	96,750	95,703	7 s	5.5 min	48.8 %
(VII)	$n = 7,\ f = x_1x_2x_3 + x_2x_3x_4 + x_2x_3x_5 + x_1 + x_2 + x_3 + x_6x_7$ $\mathcal{NL}(f) = 40$, $F = 32$ and $F_0 = 96$, $F_{16} = 28$, $F_{48} = 4$					
	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 (7), then the success rate of the attack is around 65 %. 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 (Functions (III)-(IV)). The reason is that the upper bound in (7) uses an approximation for the capacity of the binary symmetric channel which is not accurate for small cross-over probabilities.

References

1. A. Canteaut and M. Trabbia. Improved fast correlation attacks using parity-check equations of weight 4 and 5. In *EUROCRYPT 2000*, LNCS 1807, pages 573–588. Springer-Verlag, 2000.
2. V. Chepyshov, T. Johansson, and B. Smeets. A simple algorithm for fast correlation attacks on stream ciphers. In *Fast Software Encryption 2000*, LNCS 1978. Springer-Verlag, 2000.
3. J.Dj Golic, A. Clark, and E. Dawson. Generalized inversion attack on nonlinear filter generators. *IEEE Trans. Computers*, 49(10):1100–1109, 2000.
4. T. Johansson and F. Jönsson. Improved fast correlation attack on stream ciphers via convolutional codes. In *EUROCRYPT'99*, LNCS 1592, pages 347–362. Springer-Verlag, 1999.
5. T. Johansson and F. Jönsson. Fast correlation attacks through reconstruction of linear polynomials. In *CRYPTO'00*, LNCS 1880, pages 300–315. Springer-Verlag, 2000.
6. F. Jönsson and T. Johansson. A fast correlation attack on LILI-128. *Information Processing Letters*, 81(3):127–132, February 2002.
7. W. Meier and O. Staffelbach. Fast correlation attack on certain stream ciphers. *J. Cryptology*, pages 159–176, 1989.
8. 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*, LNCS 1978. Springer-Verlag, 2000.
9. T. Siegenthaler. Decrypting a class of stream ciphers using ciphertext only. *IEEE Trans. Computers*, C-34(1):81–84, 1985.