A FAST CORRELATION ATTACK ON NONLINEARLY FEEDFORWARD FILTERED SHIFT–REGISTER SEQUENCES

Réjane Forré

Inst. for Communication Technology
Swiss Federal Institute of Technology
CH–8092 Zürich, Switzerland

ABSTRACT

An algorithm recently introduced by Meier and Staffelbach is modified to be applicable to stream–ciphers with running key generators (RKG) consisting of a single linear feedback shift–register (LFSR) with a (nonlinear) feedforward filter applied to it. It is shown that, under certain assumptions, this modified algorithm can be used by a cryptanalyst to determine an equivalent system –consisting of a couple of LFSR's together with a suitable combining function– which generates the same running key sequence. Finally, design criteria are given, which ensure that a RKG withstands the modified attack.

I. INTRODUCTION

A running key generator consisting of a maximum–length (ML) linear feedback shift–register and some nonlinear filtering function f is investigated (Fig. 1). Siegenthaler showed in [1] that the

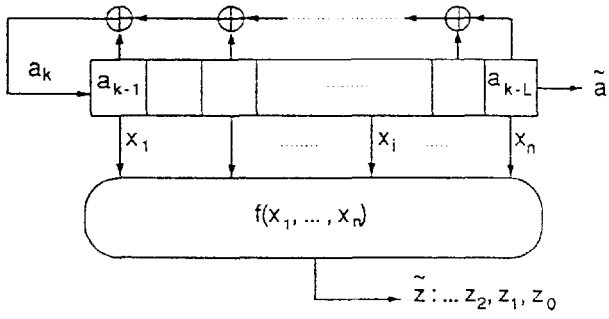

Figure 1: Structure of the investigated running key generator

output sequence $\tilde{z}$ and the ML–sequence $\tilde{a}$ of any RKG of the above type have a cross–correlation

J.J. Quisquater and J. Vandewalle (Eds.): Advances in Cryptology - EUROCRYPT '89, LNCS 434, pp. 586-595, 1990.

function (CCF) with a number of peaks (depending on the function f) whose magnitudes depend only on the Walsh–transform of f. He showed also by using a suitable set of linearly independent LFSR–sequences with initial states $S_1, S_2, \ldots, S_s$, which can be derived from these CCF–peaks, how it is possible to construct an equivalent system of the form as shown in Fig. 2, with s LFSR's with identical feedback connections and a nonlinear combining function g. However, the feasibility

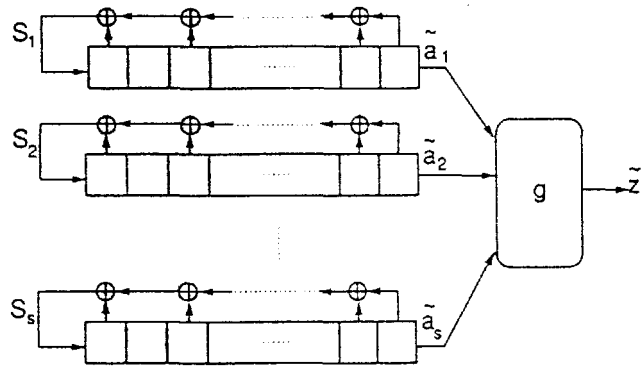

Figure 2: Cryptanalyst's equivalent system, with s identical LFSR's initially loaded with linearly independent states derived from the peaks of the CCF of $\tilde{a}$ and $\tilde{z}$

of this attack up to now was restricted to a RKG with a relatively short LFSR, because of the exponentially growing computational work needed to determine the peaks of the CCF of the sequences $\tilde{a}$ and $\tilde{z}$. In this paper, we show how to determine those peaks with a modified version of the correlation attack according to algorithm A as described by Meier and Staffelbach in [2,3], and therefore how to break a RKG of the above type with a long LFSR.

II. MODIFIED CORRELATION ATTACK

Meier and Staffelbach implicitely assumed in [2,3] a RGK built from a number s of LFSR's generating *cyclically different* (contrary to the situation as shown in Fig. 2) binary sequences $\tilde{a}_1, \tilde{a}_2, \ldots, \tilde{a}_s$ which are combined by some boolean function.

They considered the generated keystream $\tilde{z}$ as a noisy version of the sequence $\tilde{a}_i$, with the noise coming conceptually from the sequences $\tilde{a}_1, \tilde{a}_2, \ldots, \tilde{a}_j, \ldots, \tilde{a}_s$ for $j \neq i$. Their algorithms reconstruct from the sequence $\tilde{z}$ each of the sequences $\tilde{a}_i$ by using correlation properties between the sequences $\tilde{a}_i$ and $\tilde{z}$. The behaviour of their algorithms, however, is not clear for the case where the combined sequences are only cyclic shifts of each other. This is e.g. the case when the sequences are derived from the stages of a single LFSR as shown in Fig. 1 (or equivalently as shown in Fig. 2). Meier's and Staffelbach's algorithms may in this case not be able to converge to some defined result because there are instead of a single solution many (s) convergence points.

In this paper we investigate the corresponding problems and modify the algorithm to be applicable for RKG's as given in Fig. 1.

We first recall in this section the principles of the attack by Meier and Staffelbach.

Assume the cryptanalyst has observed N bits of a running key sequence $\tilde{z}$ known to be correlated to a ML–sequence $\tilde{a}$ produced by some MLLFSR of length k, having t feedback taps. The sequence $\tilde{z}$ may be viewed as a perturbation of $\tilde{a}$ by a binary asymmetric memoryless source (with $\mathrm{Prb}(0) = p_0 \neq 0.5$), and the purpose of the cryptanalyst is to reconstruct the LFSR–sequence $\tilde{a}$ from $\tilde{z}$. Every bit a_j of $\tilde{a}$ satisfies several linear relations (according to the basic feedback relation of the LFSR), each of them involving t other bits of $\tilde{a}$. The cryptanalyst checks how many of those

Table I: LFSR–initial states that yield ML–sequences highly correlated to the sequence $\tilde{z}$ of Fig. 3

Peak Nr.	Initial states	Correlation
1	101011	-69.84%
2	101010	68.25%
3	010100	68.25%
4	010000	-69.84%
5	000101	68.25%

relations hold for the corresponding bit z_j of $\tilde{z}$: the more they are, the higher is the probability for z_j to agree with a_j (if $p_0 > 0.5$), resp. to be the complement of a_j (if $p_0 < 0.5$). After having assigned to each bit z_j of $\tilde{z}$ a probability p_j of being equal to (resp. the complement of) a_j, the cryptanalyst selects the k bits of $\tilde{z}$ with the highest probabilities, uses these bits as a reference guess I_0 and computes the corresponding LFSR–initial state. Since some bits of the reference guess –usually with low probability– might be erroneous, the cryptanalyst sometimes has to test modifications of I_0 with Hamming–distances $1, 2, \ldots$ until he finds the correct initial state of the LFSR.

If this attack is applied to RKG's of the type of Fig. 1, there is no guarantee that it will succeed, since the high probable bits of the reference guess do not necessarily all correspond to the same CCF–peak, as the following example shows.

Example 1

The generator of Fig. 3 is investigated. The table I lists the initial states of the LFSR of Fig. 3

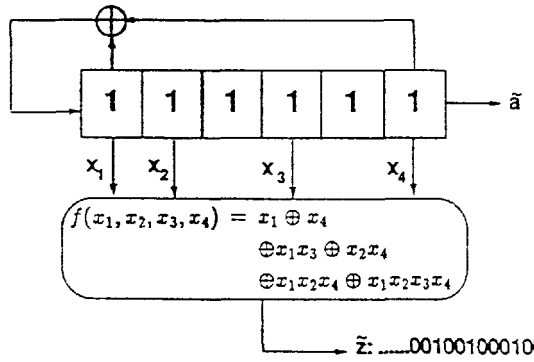

Figure 3: A running key generator with a MLLFSR of length 6 (feedback polynomial $x^6 + x + 1$)

that produce ML–sequences with a high correlation to $\tilde{z}$. A correlation of 70.00% in the second column of table I means that 70.00% of the bits of one period of $\tilde{z}$ coincide with the corresponding bits of the LFSR–sequence. The notation –70.00%, on the other hand, signifies that 70.00% of the bits of $\tilde{z}$ are the complements of the corresponding bits of the LFSR–sequence. Notice that the initial states and the correlations of Table I were computed without determining the full CCF of $\tilde{z}$ and $\tilde{a}$, by means of the Walsh–tranform of the function f [1].

We get a better insight into the situation by considering the table II. It contains the full period of 63 bits output by the generator of Fig. 3, the individual bit probabilities computed according

to [2,3], and the five LFSR–sequences yielded by the five initial states of table I, where the sequences with negative correlations (Nr. 1 and 4) have been complemented.

Table II: Sequence output by the generator of Fig. 3, corresponding bit probabilities and correlated LFSR–sequences.

i	0	1	2	3	4	5	6	7	8	9	10	11
$\mathrm{Prb}(z_i)$	0.493	0.621	0.493	0.621	0.887	0.621	0.366	0.887	0.734	0.366	0.887	0.887
z_i	0	0	1	0	0	0	1	0	0	1	0	0
$b_{1,i} = \overline{a_{1,i}}$	0	0	1	0	1	0	1	0	0	1	1	0
$b_{2,i} = a_{2,i}$	0	1	0	1	0	1	1	0	0	1	1	0
$b_{3,i} = a_{3,i}$	0	0	1	0	1	0	0	0	1	1	0	0
$b_{4,i} = \overline{a_{4,i}}$	1	1	1	1	0	1	1	1	1	1	0	0
$b_{5,i} = a_{5,i}$	1	0	1	0	0	0	1	1	0	0	0	0

i	12	13	14	15	16	17	18	19	20	21	22	23
$\mathrm{Prb}(z_i)$	0.621	0.984	0.823	0.887	0.957	0.974	0.887	0.621	0.887	0.734	0.957	0.887
z_i	0	0	0	0	0	0	0	0	1	0	1	1
$b_{1,i} = \overline{a_{1,i}}$	0	1	0	0	0	1	0	0	1	0	1	1
$b_{2,i} = a_{2,i}$	1	1	1	0	1	1	0	1	0	0	1	0
$b_{3,i} = a_{3,i}$	0	0	1	0	0	0	0	0	1	1	1	1
$b_{4,i} = \overline{a_{4,i}}$	0	0	0	0	1	0	1	0	1	0	0	1
$b_{5,i} = a_{5,i}$	1	0	0	0	0	0	1	1	1	1	1	1

i	24	25	26	27	28	29	30	31	32	33	34	35
$\mathrm{Prb}(z_i)$	0.929	0.957	0.823	0.957	0.957	0.493	0.734	0.734	0.887	0.621	0.734	0.366
z_i	0	1	0	1	0	0	0	1	0	0	1	0
$b_{1,i} = \overline{a_{1,i}}$	0	1	1	0	0	0	1	1	1	0	1	0
$b_{2,i} = a_{2,i}$	0	1	1	1	0	0	0	1	0	1	1	1
$b_{3,i} = a_{3,i}$	1	1	0	1	0	1	0	1	1	0	0	1
$b_{4,i} = \overline{a_{4,i}}$	1	0	0	1	0	0	0	1	0	0	1	0
$b_{5,i} = a_{5,i}$	0	1	0	1	0	1	1	0	0	1	1	0

i	36	37	38	39	40	41	42	43	44	45	46	47
$\mathrm{Prb}(z_i)$	0.929	0.957	0.887	0.734	0.621	0.621	0.957	0.734	0.994	0.734	0.887	0.929
z_i	1	0	0	1	1	0	0	0	0	1	1	0
$b_{1,i} = \overline{a_{1,i}}$	0	0	0	1	1	0	1	0	1	1	1	0
$b_{2,i} = a_{2,i}$	1	0	0	1	0	1	0	0	0	1	1	0
$b_{3,i} = a_{3,i}$	1	0	1	1	1	0	1	1	0	1	0	0
$b_{4,i} = \overline{a_{4,i}}$	1	1	0	1	1	0	0	0	1	1	1	0
$b_{5,i} = a_{5,i}$	1	1	1	0	1	1	0	1	0	0	1	0

i	48	49	50	51	52	53	54	55	56	57	58	59
$\mathrm{Prb}(z_i)$	0.991	0.991	0.887	0.957	0.957	0.887	0.991	0.957	0.984	0.887	0.929	0.823
z_i	0	0	0	1	0	0	0	0	0	1	1	1
$b_{1,i} = \overline{a_{1,i}}$	0	1	1	1	1	0	1	1	1	1	1	0
$b_{2,i} = a_{2,i}$	0	0	0	1	0	0	0	0	0	1	1	1
$b_{3,i} = a_{3,i}$	1	0	0	1	1	1	0	0	0	1	0	1
$b_{4,i} = \overline{a_{4,i}}$	1	0	0	0	0	1	1	0	1	0	1	1
$b_{5,i} = a_{5,i}$	0	1	1	1	0	0	0	1	0	1	1	1

i	60	61	62
$\mathrm{Prb}(z_i)$	0.734	0.493	0.493
z_i	1	0	0
$b_{1,i} = \overline{a_{1,i}}$	0	0	0
$b_{2,i} = a_{2,i}$	1	1	1
$b_{3,i} = a_{3,i}$	1	1	1
$b_{4,i} = \overline{a_{4,i}}$	1	0	0
$b_{5,i} = a_{5,i}$	1	0	0

The algorithm A of Meier and Staffelbach selects the most probable bits $(z_{13}, z_{44}, z_{48}, z_{49}, z_{54}, z_{56})$ as a reference guess I_0. Looking at the bits of the correlated LFSR–sequences at the same positions and introducing the notation $I_j = (b_{j,13}, b_{j,44}, b_{j,48}, b_{j,49}, b_{j,54}, b_{j,56})$, we see that I_0 has a Hamming–distance 1 of I_2, I_3 and I_5, but Hamming–distances 5 resp. 4 of I_1 resp. I_4.
Since the expected value of the number of errors in the reference guess is here one (computed according to [3]), modifications of I_0 with Hamming–distances up to one will be tried, as well as their complements (because we don't know in advance whether the sought initial state yields a positive or a negative correlation).
Thus, the states Nr. 1, or 2, or 3, or 5 can be discovered, but not the state Nr. 4. Note that the algorithm as described in [2,3] needs a special implementation (to be described later in this paper) to be able to find *all* of the above states (Nr. 1, 2, 3, 5). Note also that testing larger Hamming–distances –even if it seems at a first glance to be the best way to find state Nr. 4 in this small example– generally requires a very large amount of additional work in larger examples. Therefore, the algorithm needs another modification to deal with this situation (e.g. to eventually be able to find state Nr. 4). Instead of selecting the most probable bits as a reference guess, we propose to first select a set $\mathcal{S}$ of $M > 6$ high probable bits, and then randomly choose 6 bits out of this set. Several reference guesses leading to several initial states might be tested in that manner. Coming back to the above example, the set $\mathcal{S}$ of bits with probabilities ≥ 0.9 could be considered (21 bits). For $I_0 = (z_{13}, z_{17}, z_{27}, z_{28}, z_{42}, z_{49}) \subset \mathcal{S}$ we can check that $\mathrm{Hd}\,(I_0, I_4) = 0$, thus the initial state Nr. 4 will be detected. But if, for example, I_0 happens to be $(z_{13}, z_{16}, z_{37}, z_{49}, z_{52}, z_{58})$, we can check that $\mathrm{Hd}\,(I_0, I_2) = \mathrm{Hd}\,(I_0, I_3) = \mathrm{Hd}\,(I_0, I_4) = \mathrm{Hd}\,(I_0, I_5) = 2$ and $\mathrm{Hd}\,(I_0, I_1) = 3$ and neither of the five peaks will be discovered. Hereafter, we recapitulate the steps of the modified correlation attack.

1. Determine the average number m of linear relations per bit (according to [2,3]).

2. Determine, for each bit z_i of the observed running key sequence, the number of linear relations it fulfills, and compute the resulting probability p_i (again according to [2,3]).

3. *Select a set $\mathcal{S}$ consisting of M high probable bits z_i. The number M of bits in this set should be large enough to allow the selection of sufficiently many reference guesses I_0, but small enough to reduce the risk of enclosing erroneous bits.*

4. *Select randomly k bits in $\mathcal{S}$ (reference guess I_0) that form a non-singular linear system whose solution is the initial state of the LFSR leading to those particular bits at those particular positions.*

5. Test modifications of I_0 with Hamming–distances $0, 1, 2, \ldots, r$ by correlating the corresponding LFSR–sequences with the sequence $\tilde{z}$. Store the initial states that yield sufficiently high correlation values and go back to step 4, unless you have determined enough initial states.

6. Use a subset of linearly independent initial states (among those found in step 5) to construct an equivalent RKG according to the method described in [1].

In step 5, it is difficult to determine the upper limit r of the Hamming–distances to test. If we compute the expected number of erroneous bits in the reference guess I_0 according to [2,3], we obtain a value that sometimes lies far below the actual number of errors in the reference guess. This discordance ist due to the fact that, as already mentioned, the statistical model of [2,3] is not perfectly adequate for a running key generator of the type of Fig. 1. Instead of testing Hamming–distances up to an unknown upper bound, the cryptanalyst could just solve the linear system defined by the unmodified reference guess I_0, compute the LFSR–sequence associated to the obtained initial state and compare it with the running key sequence $\tilde{z}$. Then he keeps on selecting randomly new reference guesses until he has found enough initial states yielding high correlation values. We try now to answer the question, whether this alternative improves the efficiency of the modified attack or not.

Let M be the number of bits in the set $\mathcal{S}$, and assume that m bits ($0 \leq m \leq M$) in $\mathcal{S}$ are erroneous with respect to some correlated LFSR–sequence produced by a LFSR of length k. In order to find at least one correct reference guess in $\mathcal{S}$, the inequality

$$m \leq M - k \tag{1}$$

must hold. There are $\binom{M}{k}$ different reference guesses that can be chosen in $\mathcal{S}$, and $\binom{M-m}{k}$ of them are "correct", i.e. they contain no erroneous bit with respect to the correlated LFSR–sequence. If we neglect the fact that some of these "correct" reference guesses yield singular linear systems and are useless in computing the searched initial state, we obtain for the probability P_0 of selecting (randomly and uniformly) a correct reference guess

$$P_0 = \binom{M-m}{k} \cdot \binom{M}{k}^{-1} \tag{2}$$

$$= \frac{M-m}{M} \cdot \frac{M-m-1}{M-1} \cdot \ \ldots \ \cdot \frac{M-m-k+1}{M-k+1}. \tag{3}$$

The probability that among N randomly and uniformly selected reference guesses exactly one is correct, is given by

$$P(N) = P_0 \cdot (1-P_0)^{N-1}. \tag{4}$$

We are mainly interested in the expected number $E[N_1]$ of reference guesses to select in order to find a correct one:

$$E[N_1] = \sum_{n=1}^{n_{\max}} n \cdot P(n), \quad \text{where } n_{\max} = \binom{M}{k}, \tag{5}$$

$$= 1 \cdot P_0 + 2 \cdot P_0 \cdot (1-P_0) + 3 \cdot P_0 \cdot (1-P_0)^2 + \ldots + \binom{M}{k} \cdot P_0 \cdot (1-P_0)^{\binom{M}{k}-1}. \tag{6}$$

We now compute the expected number $E[N_2]$ of modifications of one randomly selected reference guess that are necessary to reconstruct the correct reference guess. We assume that the cryptanalyst begins by testing Hamming–distance 0, then 1, 2... and so on. The expected number m_0 of erroneous bits in a randomly selected reference guess is approximately given by (worst–case approximation)

$$m_0 = \left\lceil \frac{mk}{M} \right\rceil. \tag{7}$$

Therefore, the expected number $E[N_2]$ of modifications of a reference guess is given by

$$E[N_2] = \binom{k}{0} + \binom{k}{1} + \binom{k}{2} + \ldots + \binom{k}{m_0-1} + \frac{1}{2}\binom{k}{m_0}, \tag{8}$$

where we assumed that, on the average, half the possible tests with Hamming–distance m_0 have to be made for finding the correct reference guess. Fig. 1 shows a sample graph of the expected

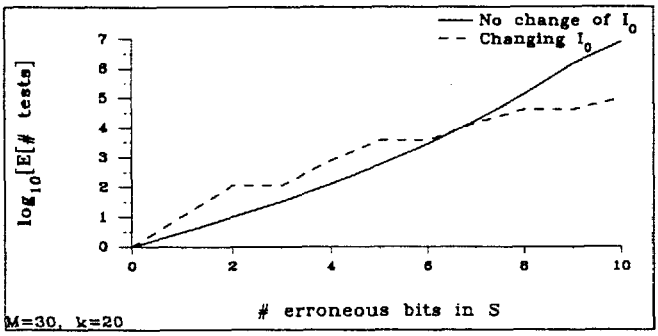

Figure 1: Expected number of tests to be made vs. the number of erroneous bits in the set $\mathcal{S}$, where $|\mathcal{S}| = M = 30$ and $k = 20$

number of tests to be made for both methods as a function of the number m of erroneous bits in the set $\mathcal{S}$. We observe that the first method (where reference guesses are picked up until a good one is found) is more efficient for the case where only few bits of $\mathcal{S}$ are erroneous. But when more bits in $\mathcal{S}$ are wrong, the method where Hamming–distances of some given reference guess are tested is to be preferred. This trend was confirmed by further sample curves. The two methods can of course be combined, for example in assigning a (small) value $r > 0$ to the maximal Hamming–distance to be tested and in picking up a new reference guess as soon as all possible tests have been made with the former one. This seems to be the most reasonable approach of the problem, since the cryptanalyst normally doesn't know the number m of erroneous bits in the set $\mathcal{S}$ but must necessarily choose a maximal Hamming–distance to test.

III. LIMITS OF THE ATTACK

In order to judge the feasibility of the modified attack, two kinds of computer experiments were carried out for concrete examples of running key generators. The first series of experiments consisted of the full execution of the attack, as it would be done by an enemy cryptanalyst who can observe a limited amount of running key bits and knows nothing but the single LFSR used in the RKG. These experiments showed that the success of the attack depends on following factors.

- *The number of feedback taps of the LFSR:* the more taps there are, the more bits are involved in each linear relation and the less reliable is the assignment of probabilities in step 2 [2,3].
- *The (absolute and relative) heights of the correlation peaks between the running-key sequence and the LFSR-sequence.* Higher peaks are much easier detected by the algorithm than lower ones. If the CCF has one or a few high peaks and some lower peaks, the last ones are not easily discovered by the algorithm. In this case, it might be necessary to test modifications of I_0 with quite large Hamming–distances.
- *The number of bits in the set $\mathcal{S}$ in step 3.* It must be large enough to allow the cryptanalyst to extract a sufficient number of linearly independent sets of k bits.

The second series of experiments is described hereafter and the obtained results are then discussed. For a given RKG with an arbitrary initial state, we first determine the initial states $S_1, S_2, \ldots, S_s$

yielding the sequences $\tilde{a}_1, \tilde{a}_2, \ldots, \tilde{a}_s$ which lead to high cross–correlation values with the running–key sequence. We continue by assigning probabilities to the observed bits of the running–key sequence (as in step 3 above) and by selecting a set $\mathcal{S}$ of M high probable bits (as in step 2). We then compare the values of the bits in this set to those of the corresponding bits of the correlated sequences. If m_i bits of $\mathcal{S}$ coincide with the corresponding bits of the i–th ML–sequence ($1 \leq i \leq s$), the expected number of erroneous bits in a reference guess of k bits uniformly randomly chosen in $\mathcal{S}$ can be computed as

$$\epsilon_i = (1 - \frac{m_i}{M}) \cdot k. \tag{9}$$

Notice that "erroneous" means here "does not coincide with the corresponding bit of the sequence $\tilde{a}_i$". Under the assumption that the statistical model of [2,3] is suitable for the investigated type of RKG, the following two facts are expected to be observed experimentally.

1. The average numbers of errors ϵ_i ($1 \leq i \leq s$), should grow with the number M of bits in the set $\mathcal{S}$, since including more bits in $\mathcal{S}$ implies including less reliable bits.

2. If the number M of bits in $\mathcal{S}$ gets very large, the number of errors ϵ_i should get closer and closer to the asymptotic value of $(1 - p_i) \cdot k$ (where p_i is the probability for any bit of the running key sequence to be equal to the corresponding bit of the i-th ML–sequence). This means that the bits of $\mathcal{S}$ could just as well have been chosen at random.

The first consideration was experimentally shown to hold more or less for RKG's with

- a small number of initial states $S_1, S_2, \ldots, S_s$ (the number s depends on properties of the feedforward function).
- cross–correlation peaks of sufficiently large amplitudes (70–75%),
- all the cross–correlation peaks having very similar amplitudes.

Fig. 2 shows the curves obtained for a RKG fulfilling the above conditions. We notice that

- The peaks Nr. 1 and Nr. 4 might be discovered by the attack if the set $\mathcal{S}$ contains for example 150 bits and if Hamming distances up to 17 are tested.

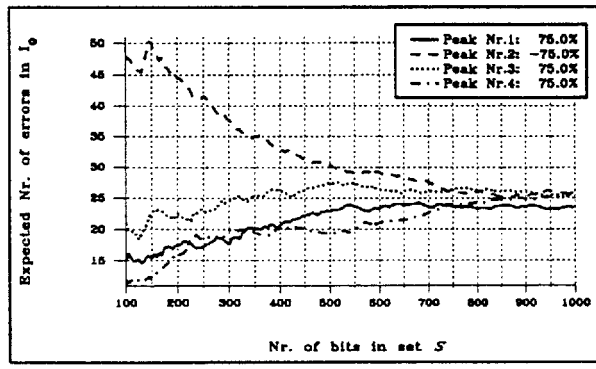

Figure 2: Expected number of errors in the reference guess vs. the number of bits in the set $\mathcal{S}$, measured for a RKG with an LFSR of length 100 having 4 feedback taps; the first 1000 bits of the running key sequence were observed.

- Hamming distances up to 23 will have to be tested in order to detect the peak Nr. 3.
- The peak Nr. 2 is not detectable by the attack. For reasonably small numbers of bits in $\mathcal{S}$, approximately half the bits of the reference guess are expected to be erroneous.
- As expected, taking more bits in $\mathcal{S}$ implies that the expected numbers of errors $\epsilon_1, \ldots, \epsilon_4$ tend to the asymptotic value of $(1 - 0.75) \cdot 100 = 25$.
- The average numbers of errors for the smallest set (of 100 bits) do not coincide with the theoretical value of $r = 14$ computed according to [2,3] (for $p_0 = 0.75$). This is due to the fact that the statistical model of [2,3] does not reproduce rigorously the situation where only cyclic shifts of *the same* LFSR–sequence are used.

Fig. 3 shows the error curves for a RKG with seven peaks of different amplitudes (one of 75%, one of 68.75% and five of 62.5%). Only the peak of 75% (lowest curve) is likely to be detected by the attack. In a way, this dominant peak "drowns" the effects of the lower peaks. For large sets $\mathcal{S}$, the curves can be checked to converge towards the asymptotic values of $(1 - 0.75) \cdot 100 = 25$, $(1 - 0.6875) \cdot 100 = 31.25$ and resp. $(1 - 0.625) \cdot 100 = 37.5$ errors.

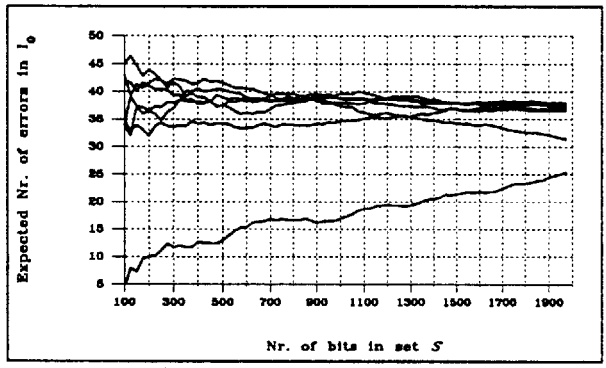

Figure 3: Expected number of errors in the reference guess vs. the number of bits in the set $\mathcal{S}$, measured for a RKG with an LFSR of length 100 having 4 feedback taps; the first 2000 bits of the running key sequence were observed.

The above observations lead to the formulation of design criteria for RKG's with a single LFSR that are to withstand the described modified correlation attack.

1. The feedforward nonlinear function f should be chosen so that the cross–correlation peaks between the running key sequence $\tilde{z}$ and the LFSR–sequence $\tilde{a}$ take values of much less than 75% (cf. Fig. 1).
2. It is more advisable to have many cross–correlation peaks than few, especially when the peaks are of similar amplitudes, since the computation of bit probabilities tend to be less reliable when the effects of many peaks are merged.
3. As pointed out in [2,3], the LFSR in use should have no less than 10 feedback taps.

Finally, we remark that the attack can be more efficiently executed if the cryptanalyst knows the exact structure of the RKG. In that case, it is easy for him to plot the curves of Fig. 2 or 3 (for an arbitrarily chosen initial state of the LFSR), and he can test in priority Hamming distances corresponding to the expected numbers of errors for a given size of the set $\mathcal{S}$. If he doesn't know the function f of Fig. 1, he has to systematically try out the Hamming distances $0, 1, \ldots$ up to some unknown upper bound. Indeed, we have seen that the expected number of errors r calculated according to [2,3] is not a reliable value for RKG's with a single LFSR.

Acknowledgements

The author is very grateful to Thomas Siegenthaler and Othmar Staffelbach for their helpful suggestions.

References

[1] Th. Siegenthaler, "Cryptanalysts Representation of Nonlinearly Filtered ML–Sequences", Advances in Cryptology, Eurocrypt'85, Springer–Verlag, pp. 103–110, 1986.

[2] W. Meier and O. Staffelbach, "Fast Correlation Attacks on Stream Ciphers", Advances in Cryptology, Eurocrypt'88, Springer–Verlag, pp. 301–314, 1988.

[3] W. Meier and O. Staffelbach, "Fast Correlation Attacks on Stream Ciphers", full paper, to appear in the Journal of Cryptology, Springer International.