

On a Resynchronization Weakness in a Class of Combiners with Memory

Yuri Borissov¹, Svetla Nikova² *, Bart Preneel², and Joos Vandewalle²

¹ Institute of Mathematics and Informatics,
Bulgarian Academy of Sciences,
8 G.Bonchev, 1113 Sofia, Bulgaria
`yborisov@moi.math.bas.bg`

² Department Electrical Engineering, ESAT/COSIC,
Katholieke Universiteit Leuven, Kasteelpark Arenberg 10,
B-3001 Heverlee-Leuven, Belgium

`svetla.nikova`, `bart.preneel`, `joos.vandewalle@esat.kuleuven.ac.be`

Abstract. In some applications for synchronous stream ciphers frequent resynchronization or resynchronization upon request may be necessary. We describe a weakness in a class of combiners with one-bit memory which makes them vulnerable in such applications requesting resynchronization. A correlation attack based on *chi-square* criterion, which in some aspects complements the attack studied by Daemen et. al., is presented.

Stream ciphers are an important class of encryption algorithms. They encrypt individual characters of a plaintext one at a time, using encryption transformation, which varies with time. We consider a weakness of synchronous stream ciphers with one-bit memory. Our investigations justify the designers intuitive understanding that in conservative design of such combiners should not be used linear memory functions, despite their easy implementation.

1 Introduction

Stream ciphers form an important class of symmetric-key encryption schemes. Their essential property is that the encryption transformation changes for each plaintext symbol. In situations where transmission errors are highly probable, stream ciphers are advantageous because they avoid error propagation. The success of the stream ciphers comes also from their easy implementation: they need only few logic gates in VLSI

* The author was partially supported by NATO research fellowship and Concerted Research Action GOA-MEFISTO-666 of the Flemish Government

circuitry. Therefore they are appropriate to embedded systems, like satellites, mobile communications (e.g. A5/1 for GSM [2]), or to systems which maintenance is either impossible or very difficult. Feedback shift registers, in particular linear feedback shift registers (LFSRs), are the basic building blocks in most stream ciphers.

LFSRs in binary key stream generators are commonly combined by nonlinear memoryless function for destroying their inherent linearity. However it is shown by Siegenthaler in [23] and [24] that such structures are vulnerable to divide-and-conquer correlation attacks based on bitwise correlation between the key stream sequence and a subset of the LFSR sequences. In [23] the relevant concept of correlation immunity (CI) of Boolean functions is introduced and the existence of trade-off between CI and linear complexity (LC) of the key stream sequence in such combiners is pointed out.

To avoid this tradeoff some researchers introduce and investigate cryptographic arrangements in which the usage of memory is permitted [7, 14, 21, 22]. The notion of CI is extended to combiners with memory and Rueppel has shown that maximum-order CI is achievable, regardless of the LC, by using only one-bit memory [21, 22]. Correlation properties of combiners with one-bit memory are further investigated in [14] and as a consequence it is shown the existence of such combiners for which each output bit is statistically independent of all the input sequences. For these combiners it is proved also that the sum of 2 successive output bits is memoryless function of 2 successive input states and this can be regarded as a weakness in view of [23] and [24]. For more general treatment on this subject see [7].

Recently some known-plaintext attacks on combiners with memory and specifically on so-called summation generator [21] have been proposed. The attack developed by Dawson and Clark in [6] is divide-and-conquer; it requires a key stream sequence length slightly larger than the sum of the input LFSR lengths but consists in exhaustive search over possible initial states of all the LFSRs except for the longest one. Another known-plaintext attack based on so-called 2-adic complexity measure is introduced in [11]. That attack requires on average a key stream sequence length proportional to the sum of the LFSR periods and its computational complexity is roughly quadratic in this length. Fast correlation attacks can be also carried out on these generators; Golic et al. [8] give detailed analysis of them.

In this paper we consider a class of binary combiners with one-bit memory having a specific memory function. We show that these combiners

have another serious weakness. Namely using them in some applications requesting frequent resynchronization or resynchronization upon request, enables a slightly more complex attack than the attack described in [24]. For a similar treatment of wide classes of ciphers (so-called synchronous stream ciphers) one can consult [5].

2 Background

Recall that LFSR of length R which produces a sequence with minimal period of maximal possible value ($\pi_{min} = 2^R - 1$) starting from an arbitrary nonzero initial state is called maximum-length and the produced sequence is called pseudonoise or m-sequence. It is well-known that a R -stage LFSR is maximal-length register if, and only if, a polynomial associated to it is primitive of degree R . Let a_t and $b_t, t = 0, 1, \dots$ are two pseudonoise sequences produced by two different nonzero initial states of one and the same maximal-length LFSR. Then obviously there exists an integer $k > 0$ such that: $b_t = a_{t+k}$ for all t . Because of this property throughout the paper, we shall refer to any such sequence as a translate of some fixed one. Pseudonoise sequences possess several properties which make the corresponding maximal-length LFSRs useful as building blocks in many applications such as synchronization, radio-location, modulation, spread-spectrum, cryptography etc. For precise definitions and proofs we refer to [9], [13].

Consider a binary combiner with one-bit memory described by the following output and memory functions:

$$\begin{aligned} z(t) &= f(x_1(t), x_2(t), \dots, x_s(t)) + \sigma(t-1) \\ \sigma(t) &= \sum_{l=1}^s x_l(t) + \sigma(t-1), \quad t = 1, 2, \dots \\ \sigma(0) &= const \end{aligned}$$

where f is an arbitrary Boolean function of s variables, $x_l(t)_{l=1}^s$ are the outputs of s maximum-length LFSRs at time-clock t , $\sigma(t)$ is the value of the memory bit at this time-clock and all additions are in $\mathbf{GF}(2)$ (that is, modulo 2).

Using the results in [14] it is easy to prove that this combiner achieves maximum-order CI imposing no restrictions upon the function f which determines the linear complexity of the produced key stream sequence.

It is common practice for stream cipher system based on LFSRs that the secret key (also called primary key) specifies the initial states of the registers. If the system works in an environment requiring resynchronization, one elegant (and cheap) solution is to specify the new initial states by using also publicly distributed vectors (so-called message keys). It should be noted that the message key has to be changed from message to message in order to prevent the re-use of key sequences. In some applications the resynchronization is performed essentially by XORing the message key and the contents of the registers loaded with the primary key [1]. It can be also advantageous to perform a fixed number of "blank" iterations during which pseudorandom digits are produced but not used. One in addition recent example for such type of resync is that performed in A5/1 [2]. In this paper we assume that one of these methods is used.

As the next-state transformation and the resynchronization mechanism of considered system are linear, there are possibilities for attacks, despite presumable nonlinearity of output function, as it is shown in [5]. However the attacks described in [5] are deterministic while here we point out the possibility of a statistical approach.

3 A Ciphertext-Only Resync Correlation Attack

Before describing the attack we shall give two propositions.

Proposition 1. *Let f be a Boolean function of s variables. For l , $1 \leq l \leq s$, denote by C_l^1 the set $\{\mathbf{c} \in \{0, 1\}^s : \mathbf{c}_l = 1\}$ and let ω_l be binary s -tuple having 1 only in its l -th component. Then the coefficients $G_{\mathbf{c}}(\omega_l)$ of the Walsh transform of the Boolean functions $g_{\mathbf{c}}(\mathbf{x}) = f(\mathbf{x}) + f(\mathbf{x} \oplus \mathbf{c})$, $\mathbf{c} \in C_l^1$, are all equal to 0.*

Proof. The statement of the proposition follows from the balancedness of the function $g_{\mathbf{c}}(\mathbf{x}) + x_l$ since the following simple property holds: $g_{\mathbf{c}}(\mathbf{x}) = g_{\mathbf{c}}(\mathbf{x} \oplus \mathbf{0})$ for all $\mathbf{x} \in \{0, 1\}^s$. \square

In other words this proposition says that the random variables corresponding to each one of the functions $g_{\mathbf{c}}$, $\mathbf{c} \in C_l^1$ are statistically independent of its l -th argument, when the arguments of the functions are considered as uniformly distributed binary r.v. (see [26]).

Proposition 2. *Let f be a Boolean function of s variables of degree s . For l , $1 \leq l \leq s$, denote by C_l^0 the set $\{\mathbf{c} \in \{0, 1\}^s \setminus \mathbf{0} : \mathbf{c}_l = 0\}$ and let ω_l be binary s -tuple having 1 only in its l -th component. Then all the coefficients $G_{\mathbf{c}}(\omega_l)$ of the Walsh transform of the Boolean functions $g_{\mathbf{c}}(\mathbf{x}) = f(\mathbf{x}) + f(\mathbf{x} \oplus \mathbf{c})$, $\mathbf{c} \in C_l^0$, are nonzero.*

Proof. We shall consider the case $\mathbf{c} = (00\dots 01)$ and $l < s$. It is easy to see that $g_{\mathbf{c}}(\mathbf{x})$ is of degree $s - 1$ and the only monomial of degree $s - 1$ in its algebraic normal form is $x_1x_2\dots x_{s-1}$. Let $V(j) = \{\omega \in \{0, 1\}^s : \omega_i = 0, i \neq j\}$. $V(j)$ is 1-dimensional vector space. By expressing the coefficients in the algebraic normal form of $g_{\mathbf{c}}(\mathbf{x})$ in terms of the Walsh transform (see [26]), we obtain that all expressions $\frac{1}{2} \sum_{\omega \in V(j)} G_{\mathbf{c}}(\omega) \pmod{2}$ are equal to 0, with the exception of the case $j = s$ when its value is 1. Since by the Proposition 1, $G_{\mathbf{c}}(\omega_s) = 0$, therefore $G_{\mathbf{c}}(\mathbf{0}) \neq 0$ and finally we conclude that $G_{\mathbf{c}}(\omega_l) \neq 0$. All other cases can be treated similarly. \square

Obviously $|C_l^0| = 2^{s-1} - 1$.

Now, let c_1 and c_2 are two encrypted messages with one and the same primary key K and different message keys M_1 and M_2 . Let us in addition introduce the following notations:

$c_1(t)$ and $c_2(t)$ denote the ciphertext bits at time t ;

$p_1(t)$ and $p_2(t)$ denote the plaintext bits at time t ;

$\mathbf{x}(t)$ denote the common unknown output of the LFSRs (considered as a vector) depending only on K ;

$\mathbf{R}_1(t)$ and $\mathbf{R}_2(t)$ denote the outputs of the LFSRs depending on the message keys M_1 and M_2 respectively;

$\sigma_1(t)$ and $\sigma_2(t)$ denote the values of the memory bits of two encryptions before time t ;

\oplus denotes bitwise XOR of binary vectors.

The attack is based on the following two simple observations:

F1. The overall transition transformation on the states of the LFSRs (through their feedback) is linear and therefore their output can be regarded as a bitwise XOR of the outputs of two identical sets of LFSRs – one initialized by the primary key and the other one by the message key, i.e.

$$\begin{aligned} c_1(t) &= p_1(t) + f(\mathbf{x}(t) \oplus \mathbf{R}_1(t)) + \sigma_1(t) \\ c_2(t) &= p_2(t) + f(\mathbf{x}(t) \oplus \mathbf{R}_2(t)) + \sigma_2(t) \end{aligned}$$

F2. Since the memory function is linear, at each time-clock the memory bit is an affine function of the initial contents of the LFSRs and therefore can be regarded as a sum of two bits – one depending only on the primary key and the other one depending only on the message key, i.e.

$$\begin{aligned} c'_1(t) &= p_1(t) + f(\mathbf{x}(t) \oplus \mathbf{R}_1(t)) + \sigma'(t) \\ c'_2(t) &= p_2(t) + f(\mathbf{x}(t) \oplus \mathbf{R}_2(t)) + \sigma'(t) \end{aligned} \tag{1}$$

where $\sigma'(t)$ is the value of the memory bit depending only on K .

In other words we can avoid the dependence of c_1 and c_2 on the memory bit, which is a function of the message keys only, and get “modified” ciphertexts bits $c'_1(t) = c_1(t) + \sigma'_1(t)$ and $c'_2(t) = c_2(t) + \sigma'_2(t)$, where $\sigma'_1(t)$ and $\sigma'_2(t)$ depend only on the message keys M_1 and M_2 respectively.

Hence XORing the equations (1) we get:

$$c'_1(t) + c'_2(t) = [p_1(t) + p_2(t)] + [f(\mathbf{x}(t) \oplus \mathbf{R}_1(t)) + f(\mathbf{x}(t) \oplus \mathbf{R}_2(t))].$$

If we introduce for simplicity the following notations:

$$\begin{aligned} s(t) &= c'_1(t) + c'_2(t) \\ p(t) &= p_1(t) + p_2(t) \\ \mathbf{y}(t) &= \mathbf{x}(t) \oplus \mathbf{R}_1(t) \\ \mathbf{C}(t) &= \mathbf{R}_1(t) \oplus \mathbf{R}_2(t) \end{aligned}$$

finally we obtain:

$$s(t) = p(t) + f(\mathbf{y}(t)) + f(\mathbf{y}(t) \oplus \mathbf{C}(t)) \quad t = 1, 2, \dots, L \quad (2)$$

where $L = \min\{\text{length}(c_1), \text{length}(c_2)\}$.

Let us make two remarks at this point:

1. $p(t)$ can be assumed to be the output of Binary Memoryless Source (BMS) with $Pr(p(t) = 0) > \frac{1}{2}$.

2. Consider the subset $T_{\mathbf{c}}$ of time-clocks for which $\mathbf{C}(t)$ takes on some fixed value \mathbf{c} . It is clear that the function $g_{\mathbf{c}}(\mathbf{y}) = f(\mathbf{y}) + f(\mathbf{y} \oplus \mathbf{c})$ plays the role of a memoryless combining function on $T_{\mathbf{c}}$. Note also that since the outputs of the LFSRs can be regarded as independent binary variables, each taking on the values 0 or 1 with probability $\frac{1}{2}$, then $Pr(\mathbf{C}(t) = \mathbf{c}) = \frac{1}{2^s}$ and the expected number of elements of $T_{\mathbf{c}}$ is $\frac{L}{2^s}$.

Let $N_{\mathbf{c}} = |T_{\mathbf{c}}|$ and $n_{\mathbf{c}}$ be the number of coincidences between $s(t)$ and a translate $x_l(t)$ of the m -sequence produced by the l -th LFSR. It is shown in [24] that if $x_l(t)$ is not the actually used translate (because of its independence of $s(t)$ and because of its statistics) the random variable $\eta_{\mathbf{c}} = \frac{2n_{\mathbf{c}} - N_{\mathbf{c}}}{\sqrt{N_{\mathbf{c}}}}$ for large $N_{\mathbf{c}}$ (say > 100), can be assumed to be normally distributed with parameters 0 and 1, due to the Central Limit theorem. Therefore the random variable

$$\chi = \sum_{\mathbf{c} \in C_l^0} \eta_{\mathbf{c}}^2 \quad (3)$$

will be χ^2 -distributed with $2^{s-1} - 1$ degrees of freedom when all $N_{\mathbf{c}}$ are large.

On the other hand the same statistics, calculated for the actually used translate, will not fit the above distribution if at least one of the functions $g_{\mathbf{c}}(\mathbf{y})$ is not statistically independent of its l -th argument (indeed by Proposition 2, if the degree of f is maximal possible, this is true for all $\mathbf{c} \in C_l^0$).

Let us study this case in more details. Denote by $p_{\mathbf{c}}$ the correlation probability between $s(t)$ and the actually used translate produced by the l -th LFSR for time-clocks in $T_{\mathbf{c}}$ and let $q_{\mathbf{c}} = 1 - p_{\mathbf{c}}$. Since $n_{\mathbf{c}}$ is binomially distributed with mean value $\mathbf{E}n_{\mathbf{c}} = p_{\mathbf{c}}N_{\mathbf{c}}$ and variance $\mathbf{D}n_{\mathbf{c}} = p_{\mathbf{c}}q_{\mathbf{c}}N_{\mathbf{c}}$, then the random variable $\zeta_{\mathbf{c}} = \frac{n_{\mathbf{c}} - p_{\mathbf{c}}N_{\mathbf{c}}}{\sqrt{p_{\mathbf{c}}q_{\mathbf{c}}N_{\mathbf{c}}}}$ for large $N_{\mathbf{c}}$ is normally distributed with parameters 0 and 1, again due to the Central Limit theorem. By straightforward computations we get:

$$\eta_{\mathbf{c}} = \zeta_{\mathbf{c}}\sqrt{4p_{\mathbf{c}}q_{\mathbf{c}}} + (2p_{\mathbf{c}} - 1)\sqrt{N_{\mathbf{c}}} \quad (4)$$

and therefore in this case $\eta_{\mathbf{c}}$ is normally distributed with mean value $\mathbf{E}\eta_{\mathbf{c}} = (2p_{\mathbf{c}} - 1)\sqrt{N_{\mathbf{c}}}$ and variance $\mathbf{D}\eta_{\mathbf{c}} = 4p_{\mathbf{c}}q_{\mathbf{c}}$. By using basic properties of the variance and of the normal distribution it is easy to find that:

$$\begin{aligned} \mathbf{E}\eta_{\mathbf{c}}^2 &= 4p_{\mathbf{c}}q_{\mathbf{c}} + (2p_{\mathbf{c}} - 1)^2N_{\mathbf{c}} \\ \mathbf{D}\eta_{\mathbf{c}}^2 &= 16p_{\mathbf{c}}q_{\mathbf{c}}[2p_{\mathbf{c}}q_{\mathbf{c}} + (2p_{\mathbf{c}} - 1)^2N_{\mathbf{c}}]. \end{aligned}$$

Finally, since the random variables $\eta_{\mathbf{c}}$, $\mathbf{c} \in C_l^0$ are independent, then for the mean value $\mathbf{E}\chi$ and the variance $\mathbf{D}\chi$ we get:

$$\begin{aligned} \mathbf{E}\chi &= \sum_{\mathbf{c} \in C_l^0} [4p_{\mathbf{c}}q_{\mathbf{c}} + (2p_{\mathbf{c}} - 1)^2N_{\mathbf{c}}] \quad (5) \\ &\approx 4 \sum_{\mathbf{c} \in C_l^0} p_{\mathbf{c}}q_{\mathbf{c}} + \frac{L}{2^s} \sum_{\mathbf{c} \in C_l^0} (2p_{\mathbf{c}} - 1)^2; \\ \mathbf{D}\chi &= \sum_{\mathbf{c} \in C_l^0} 16p_{\mathbf{c}}q_{\mathbf{c}}[2p_{\mathbf{c}}q_{\mathbf{c}} + (2p_{\mathbf{c}} - 1)^2N_{\mathbf{c}}] \\ &\approx 32 \sum_{\mathbf{c} \in C_l^0} p_{\mathbf{c}}^2q_{\mathbf{c}}^2 + \frac{L}{2^{s-4}} \sum_{\mathbf{c} \in C_l^0} p_{\mathbf{c}}q_{\mathbf{c}}(2p_{\mathbf{c}} - 1)^2. \end{aligned}$$

(For simplicity in the above considerations the index l was omitted).

Suppose the cryptanalyst searches for actually used initial state of the l -th LFSR, $1 \leq l \leq s$. Based on the above facts he/she can stick to the following statistical procedure.

Step 1. Choose a sufficiently low significance level α and determine a value χ_{α} such that the probability a χ^2 -distributed with $2^{s-1} - 1$ degrees

of freedom random variable takes values greater than χ_α , is equal to α [16, Ch5].

Step 2. For possible initial state, load the l -th LFSR with it, produce the corresponding translate $x_l(t)$ and count the number $n_{\mathbf{c}}$ of coincidences between $s(t)$ and $x_l(t)$, when $t \in T_{\mathbf{c}}$ for each $\mathbf{c} \in C_l^0$.

Step 3. Calculate the statistics (3) and if its value is greater than χ_α store the examined state. If there is no new possible state stop the procedure. Otherwise form the new possible initial state and go to Step 2.

The above procedure uses a version of what is known as K. Pearson's criterion of goodness-of-fit or χ^2 -criterion. Note that it is very important how the cryptanalyst will choose the significance level α because it gives the probability P_f of a "false alarm". He/she wishes to pick up the level α in such a way that both the probability P_f and the probability P_m for "missing the actually used translate" are negligible. However these requirements contradict each other and the cryptanalyst should be ready for a compromise. Suppose he/she has some estimates on correlation probabilities. Then by using Chebyshev's inequality, the cryptanalyst could estimate how much ciphertext L is needed such that the probability P_m is under a fixed value. Namely:

$$\begin{aligned} P_m &= \text{Prob}(\chi \leq \chi_\alpha) \\ &= \text{Prob}(\mathbf{E}\chi - \chi_\alpha \leq \mathbf{E}\chi - \chi) \\ &\leq \text{Prob}(\mathbf{E}\chi - \chi_\alpha \leq |\mathbf{E}\chi - \chi|) \\ &\leq \frac{\mathbf{D}\chi}{(\mathbf{E}\chi - \chi_\alpha)^2}. \end{aligned}$$

By (5) we can see that the last expression $\frac{\mathbf{D}\chi}{(\mathbf{E}\chi - \chi_\alpha)^2}$ decreases as $O(\frac{1}{L})$ when L increases. Of course, since P_f is always positive additional tests have to be performed, but this time on a considerably smaller set of plausible states.

The following straightforward proposition shows that the message keys M_1 and M_2 are in some sense symmetric. This property can be used to halve the computational efforts of the described procedure.

Proposition 3. *The statistics (3) obtains equal values on the pairs of nonzero initial states of the l -th LFSR having bitwise XOR equal to $M_{1,l} \oplus M_{2,l}$, where $M_{1,l}$ and $M_{2,l}$ are respectively the restrictions of the message keys M_1 and M_2 over that register.*

This attack can be successful even if f is unknown for the cryptanalyst provided that the length L , and the unknown correlation coefficients

(including the plaintext redundancy) are sufficiently large. It is feasible if the lengths of the registers are of moderate size. Let $K_l = 2^{R_l} - 1$, where R_l is the length of the l -th register. Then our attack makes $\sum_{l=1}^s K_l$ trials (ignoring additional tests efforts), which is considerably smaller than the average number of trials $(\prod_{l=1}^s K_l)/2$ in a brute force attack. However each trial requires $2^{s-1} - 1$ squarings and the same number of divisions which is computationally more complex than simple counting used in the attack from [24].

The described so far attack uses only first order correlations biases i.e. only the correlations between $s(t)$ and individual LFSR's output. In the similar way higher order correlation biases involving linear combinations of LFSR's outputs, could be also exploited.

The equation (2) reflects the situation when nonintersecting parts of a codeword of maximal-length (or simplex) code pass different binary symmetric channels generally with unequal transition probabilities. Thus, when f is public, there exist possibilities for correlation attacks by using well-known methods for decoding such codes (maximum-likelihood decoding, threshold decoding, iterative decoding, relaxation algorithms, Viterbi decoding etc.). For more details on such attacks see [3, 4, 10, 12, 15, 17–19, 24], etc...

Example 1. We shall compare the performance of our attack with the attack studied by Daemen et. al. in [5]. We consider the combiner with memory from Section 2 when $s = 5$ and the following 5 primitive feedback polynomials chosen from the list given in [20]:

$$\begin{aligned} &x^{16} + x^{12} + x^3 + x + 1 \\ &x^{15} + x + 1 \\ &x^{15} + x^{14} + 1 \\ &x^{14} + x^{10} + x^6 + x + 1 \\ &x^{13} + x^4 + x^3 + x + 1 \end{aligned}$$

The attack given in [5] is known-plaintext attack and it is possible if the number of resynchronizations is larger than 6 (5 for the LFSRs and plus one for the memory bit). The cryptanalyst needs to know at least 6 keysequence (or equivalently plaintext) bits for a number of $\lceil \frac{|K|}{6} \rceil$ time-clocks, where $|K|$ is the length of primary key K in bits. The expected workload is $2^6 \lceil \frac{|K|}{6} \rceil$ evaluations of the output function f and some additional linear algebra computations of order $O(|K|^3)$. It should be also noted that, if enough resynchronizations are allowed then it is possible to

convert the attack described in [5] from known-plaintext to ciphertext-only scenario by using majority-voting.

The attack proposed in this paper is ciphertext-only attack. To mount it we need only 2 resynchronizations. Of course, the required length depends on the probability $p_e = Pr(p(t) = 1)$. Fig.1 summarizes the results of our experiments with different probabilities p_e , lengths of ciphertext L and corresponding success rates for significance level $\alpha = 0.005$ ($\chi_\alpha = 32.8013$) and random nearly balanced combining functions of degree 5.

If we recapitulate, the first attack requires stronger assumptions, but less computational efforts than the second one.

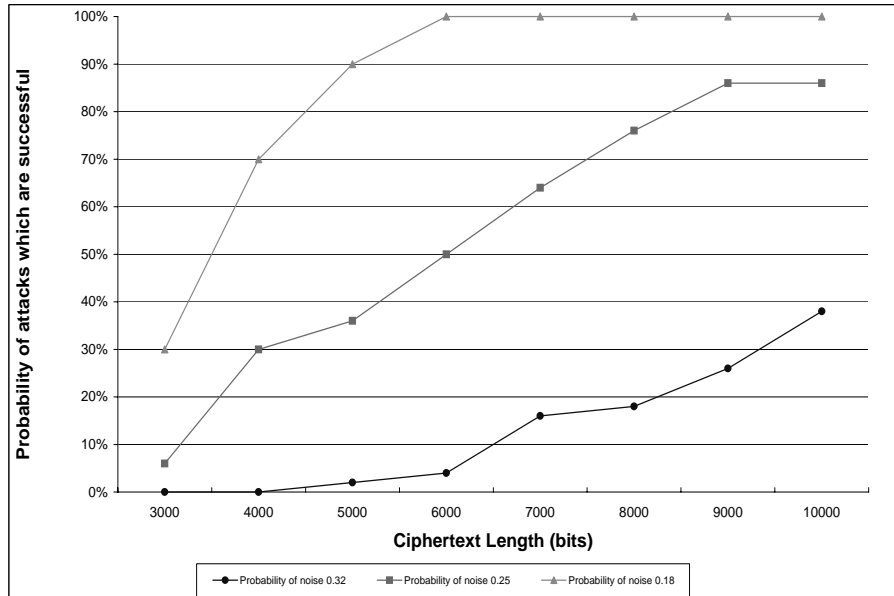


Fig. 1. Success rate versus ciphertext length for different p_e (probabilities of noise) with $\chi_\alpha = 32.8013$ (or $\alpha = 0.005$).

4 Conclusions

In this paper we point out a resynchronization weakness in a class of combiners with one-bit memory. We describe an attack which uses essentially

the linearity of the memory function of the considered combiners. Our investigations along with these from [5], justify the designers intuitive understanding that in conservative design of such combiners should not be used linear memory functions, despite their easy implementation and other good cryptographic properties.

The attack is slightly more complex than the one studied in [24] and requires ciphertext length of order $O(2^s)$, where s is the number of the involved LFSRs. It is successful when the combining function f is taken from large set of Boolean functions (and even if f is assumed unknown). At some favorable circumstances the described weakness can also allow more efficient fast correlation attacks.

Note also that one way to preclude attacks of this type, is to choose the combining function f such that all of the functions $g_{\mathbf{c}}(\mathbf{y}) = f(\mathbf{y}) + f(\mathbf{y} \oplus \mathbf{c})$ are high order correlation-immune and therefore f is obligatory of low degree.

References

1. H. Beker and F. Piper, *Cipher System: the protection of communications*, Northwood Publications, 1982.
2. A. Biryukov, A. Shamir, D. Wagner, *Real time cryptanalysis of A5/1 on a PC*, in Fast Software Encryption 2000, LNCS 1978, Springer-Verlag, pp.1-18.
3. A. Canteaut and M. Trabbia, *Improved correlation attacks using parity-check equations of weight 4 and 5*, Advances in Cryptology – Eurocrypt 2000, LNCS 1807, pp. 573-588.
4. V. Chepyshov, T. Johansson, B. Smeets, *A simple algorithm for fast correlation attacks on certain stream ciphers*, Fast Software Encryption 2000, LNCS 1978, Springer-Verlag, pp. 181-195.
5. J. Daemen, R. Govaerts and J. Vandewalle *Resynchronization weaknesses in synchronous stream ciphers*, Advances in Cryptology – Eurocrypt’93, LNCS 765, Springer-Verlag, Berlin, 1994, pp. 159-167.
6. E. Dawson and A. Clark, *Divide and conquer attacks on certain classes of stream ciphers*, Cryptologia, vol. 18(1), 1994, pp. 25-40.
7. J. D. Golic, *Correlation properties of a general binary combiner with memory*, J. Cryptology, vol. 9(2), 1996, pp. 111-126.
8. J. D. Golic, M. Salmasizadeh, E. Dawson, *Fast correlation attacks on the summation generator*, J.Cryptology vol., 2000, pp. 245-262.
9. S.W. Golomb, *Shift Register Sequences*, Holden-Day, Inc., 1967.
10. T. Johansson and F. Jonsson, *Improved fast correlation attacks on stream ciphers via convolutional codes*, Advanced in Cryptology - Eurocrypt’99, LNCS 1592, Springer-Verlag, 1999, pp. 347-362.
11. A. Klapper and M. Goresky, *Cryptanalysis based on 2-adic Rational Approximation*, Advances in Cryptology – Crypto 1995, LNCS 963, Springer-Verlag, 1995, pp. 262-273.
12. D. J. C. MacKay, *A Free energy minimization framework for inference problems in modulo 2 arithmetic*, Fast Software Encryption 1994, LNCS 1008, Springer-Verlag, pp. 179-195.

13. F.J. MacWilliams and N.J.A. Sloane, *The Theory of Error-Correcting Codes*, North-Holland Publishing Company, 1977.
14. W. Meier and O. Staffelbach, *Correlation properties of combiners with memory in stream ciphers*, *J. Cryptology*, vol. 5(1), 1992, pp. 67-86.
15. W. Meier and O. Staffelbach, *Fast correlation attacks on certain stream ciphers*, *J. Cryptology*, vol. 1(3), 1989, pp. 159-167.
16. A. Menezes, P. van Oorschot, S. Vanstone *Handbook of Applied Cryptography*, CRC Press, 1996.
17. M. J. Mihaljevic and J. D. Golic, *A method for convergence analysis of iterative probabilistic decoding*, *IEEE Trans. on Information Theory*, vol. 46(6), 2000, pp. 2206-2211.
18. M. J. Mihaljevic, M. P. C. Fossorier, H. Imai, *A low-complexity and high-performance algorithm for the fast correlation attack*, *Fast Software Encryption 2000*, LNCS 1978, Springer-Verlag, pp. 196-212.
19. W. T. Penzhorn, *Correlation attacks on stream ciphers: computing low-weight parity checks based on error-correcting codes*, *Fast Software Encryption 1996*, LNCS 1039, pp. 159-172.
20. W. W. Petersen, *Error-Correcting Codes*, John Wiley and Sons, Inc. 1961.
21. R. A. Rueppel, *Correlation immunity and the summation generator*, LNCS, vol. 218, 1986, pp. 260-272.
22. R. A. Rueppel, *Analysis and design of stream ciphers*, Springer-Verlag, Berlin, 1986.
23. T. Sighenthaler, *Correlation immunity of nonlinear combining functions for cryptographic applications*, *IEEE Trans. Inf. Theory*, vol. 30(6), 1984, pp. 776-780.
24. T. Sighenthaler, *Decrypting a class of stream ciphers using ciphertext only*, *IEEE Trans. Comput.*, vol. 34(1), 1985, pp. 2010-2017.
25. T. Sighenthaler, *Cryptanalists representation of nonlinearly filtered ML-sequences*, *Eurocrypt 1985*, LNCS 219, pp.103-110.
26. G. Z. Xiao, J. Massey *A Spectral characterization of correlation-immune combining functions* *IEEE Trans. Inf Theory*, vol. 34(3), 1988, pp. 569-571.