Improvement of the Edit Distance Attack to Clock-Controlled LFSR-Based Stream Ciphers*

Pino Caballero-Gil[1] and Amparo Fúster-Sabater[2]

[1] D.E.I.O.C. University of La Laguna. 38271 La Laguna, Tenerife, Spain
pcaballe@ull.es
[2] Institute of Applied Physics. C.S.I.C. Serrano 144, 28006 Madrid, Spain
amparo@iec.csic.es

Abstract. The main idea behind this paper is to improve a known plaintext divide-and-conquer attack that consists in guessing the initial state of a Linear Feedback Shift Register component of a keystream generator, and then trying to determine the other variables of the cipher based on the intercepted keystream. While the original attack requires the exhaustive search over the set of all possible initial states of the involved register, this work presents a new and simple heuristic optimization of such an approach that avoids the evaluation of an important number of initial states when launching a constrained edit distance attack on irregularly clocked shift registers.

1 Introduction

Stream ciphers have extensive applications in secure communications, e.g. wireless systems, due to different practical advantages such as easy implementation, high speed and good reliability. When designing a stream cipher, the main goal is to expand a short key into a long pseudorandom keystream in such a way that it should not be possible to reconstruct the short key from the keystream. In this work we focus on stream ciphers based on Linear Feedback Shift Registers ($LFSRs$), such as A5 for GSM [12] or the function E0 for Bluetooth [2]. Other examples of $LFSR$-based generators are LILI-II [3], Toyocrypt [5], Shrinking [4] and Alternating Step [7] generators. All these generators produce keystream sequence with high linear complexity, long period and good statistical properties, [10]. In particular, the two last generators were thoroughly analyzed in [14] through a correlation attack based on a decoding problem.

The main idea behind this paper is to improve a known plaintext divide-and-conquer attack that consists in guessing the initial state of an $LFSR$ component of the generator, trying to determine the other variables of the cipher based on the intercepted keystream, and then checking that the initial guess was consistent with the observed keystream sequence. Such an attack was first proposed in [8] by means of a theoretical model and a distance function known as Levenshtein or

* Research supported by the Spanish Ministry of Education and Science and the European FEDER Fund under Projects SEG2004-04352-C04-03 and SEG2004-02418.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 355–364, 2005.

edit distance. This distance was also used in [9] to attack a single $LFSR$-based generator. On the other hand, it has been proven [13] that when the length of the intercepted sequence is large enough, the number of candidate initial states is small. The attack considered here may be seen as an extension of the constrained edit distance attack to clock-controlled $LFSR$-based generators presented in [15]. Our main aim is to investigate whether the number of initial states to be analyzed can be reduced independently of the length of the intercepted sequence. In fact, this feature has already been pointed out in [6] as one of the most interesting problems in the cryptanalysis of stream ciphers today. So, while according to the original method, the attacker needs to traverse an entire search tree including all the possible $LFSR$ initial states, in this work we try to improve such an attack by simplifying the search tree in such a way that only the most efficient branches are retained. This new approach produces a significant improvement in the computing time of the original edit distance attack since it implies a dramatic reduction in the number of initial states that need to be evaluated.

This work is organized as follows. Section 2 introduces some definitions and basic concepts regarding the computation of constrained edit distances. In Section 3, some ideas for an efficient initial state selection method are introduced. Such a method allows us both to deduce a threshold value for the edit distance, and to discard beforehand an important number of initial states. Section 4 provides the full description of the improved algorithm, which takes advantage of the threshold value described in the previous Section. Finally, Section 5 contains simulation results while in Section 6 several conclusions are drawn.

2 Constrained Edit Distance Attack

The Levenshtein or edit distance may be defined as the minimum number of elementary operations (insertions, deletions and substitutions) required to transform one sequence X of length N into another sequence Y of length M. Some of the different applications of the edit distance are, for instance, file revision, spell correction, plagiarism detection, molecular biology, and speech recognition. The dynamic programming approach is a classical solution for computing the edit distance matrix, where the distances between longer and longer prefixes of the sequences are successively evaluated until the final result is achieved. When applying an edit distance attack to a stream cipher and depending on the generator design, some edit operations may be restricted. In this case, a so-called constrained edit distance may be necessary.

The specific theoretical model considered in this work for the attacked generator is described in Fig. 1. As usual, it is assumed that the $LFSR$ feedback polynomial is known. The use of this general model implies that the known plaintext attack is applicable not only to those generators that fit exactly the simplest version of such a model but also to all the sequences produced by more complex generators that also fulfills the description. In this latter case, it is understood that the attack will provide a simpler equivalent description of the original attacked generator.

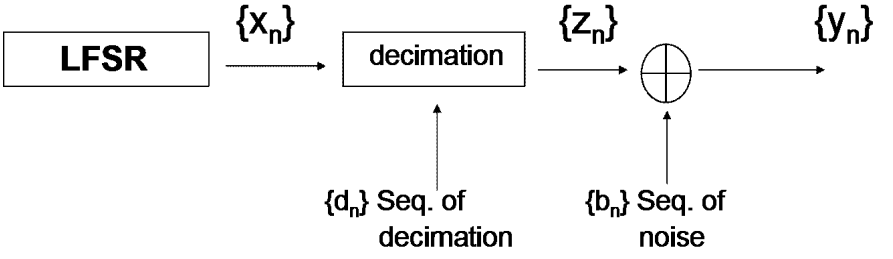

Fig. 1. Theoretical model

An essential step in edit distance attacks is the computation of the edit distance matrix $W = (w_{i,j}), i = 0, 1, \ldots, N - M, j = 0, 1, \ldots, M$ associated with each possible couple of sequences X and Y where Y represents the intercepted keystream sequence while X is each one of the $LFSR$ sequences produced by each one of the possible initial states. In the following, some of the parameters of such a matrix are described. Firstly, its dimension is $(N-M+1)(M+1)$. Secondly, the element $w_{N-M,M}$ represents the edit distance between the sequences X and Y. Lastly, each element of the matrix $w_{i,j}$ corresponds exactly to the edit distance between prefix subsequences $x_1, x_2, \ldots, x_{i+j}$ and $y_1, y_2, \ldots, y_j$.

In the constrained edit distance attack here analyzed only deletions and substitutions are allowed. Those two elementary operations may be seen as the result of an irregular decimation on the $LFSR$ sequence plus the addition of a noise sequence respectively. Furthermore, in this work it is assumed that the number of consecutive deletions is 1 (constrained edit distance). Under this hypothesis, the length of X may be estimated as $N \approx 3M/2$, which coincides with the mathematical expectation. The previous hypothesis implies that the computation of $2(N - M)(N - M + 1)$ elements of the matrix W corresponding to the two triangles: $\{w_{i,j} : i = 1, \ldots, N - M, j = 0, \ldots, i - 1\}$ and $\{w_{i,j} : i = 0, \ldots, N - M - 1, j = 2M - N + 1 + i, \ldots, M\}$ can be avoided. The remaining elements $w_{i,j}$ of the constrained edit distance matrix W may be computed recursively by columns according to the formulas in Equation (1).

$$w_{0,0} = 0$$
$$w_{i,j} = min\{w_{i,j-1} + P_s(x_{i+j}, y_j), w_{i-1,j-1} + P_d(x_{i+j}, y_j)\} \text{ where}$$
$$P_s(x_{i+j}, y_j) = \begin{cases} 0 \; if \; x_{i+j} = y_j \\ 1 \; if \; x_{i+j} \neq y_j \end{cases}$$

$$P_d(x_{i+j}, y_j) = \begin{cases} 1 \; if \; x_{i+j} = y_j \\ 2 \; if \; x_{i+j} \neq y_j \end{cases} \quad (1)$$

The elements of the matrix W may be seen as costs of optimal paths in an induced graph with as many vertices as elements in the matrix W. Moreover, the arcs have costs 0,1 or 2 depending on the coincidences between the corresponding bits of Y and X, (see equation (1)). In such an induced graph, the optimal paths between the source associated with the element $w_{0,0}$ and the sink

corresponding to the element $w_{N-M,M}$ give us the solution of the cryptanalytic attack by specifying both decimation and noise sequences $D = \{d_n\}$ and $B = \{b_n\}$, respectively.

Example : For an intercepted keystream sequence Y:1101011 of length M=7 and a candidate sequence X:1110110111 of length N=10, the constrained edit distance matrix is: $W = \begin{pmatrix} 0 & 0 & 0 & 1 & 2 & - & - & - \\ - & 1 & 1 & 1 & 1 & 2 & - & - \\ - & - & 3 & 3 & 2 & 2 & 2 & - \\ - & - & - & 5 & 5 & 4 & 3 & 3 \end{pmatrix}$. The graph induced by this matrix is shown in Fig. 2 where the twelve optimal paths are remarked in bold.

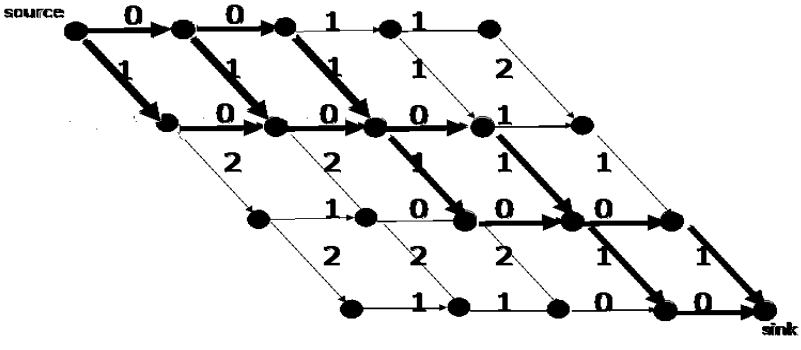

Fig. 2. Induced graph and optimal paths

From those optimal paths, the 12 possible solutions to the cryptanalysis corresponding in this case to decimation without noise are expressed in terms of decimation sequences D.

$$D = \{d_n\} : \begin{cases} 0010010010 : \ Solution1; \ 0010010100 : \ Solution2 \\ 0010100010 : \ Solution3; \ 0010100100 : \ Solution4 \\ 0100010010 : \ Solution5; \ 0100010100 : \ Solution6 \\ 0100100010 : \ Solution7; \ 0100100100 : \ Solution8 \\ 1000010010 : \ Solution9; \ 1000010100 : Solution10 \\ 1000100010 : Solution11; 1000100100 : Solution12 \end{cases}$$

3 Threshold Search

The main idea behind the method described in this section comes from the association between bits x_{i+j} of X and arcs of the graph induced by the matrix W. In particular, we consider cut sets between the source and the sink in the induced graph, which allow us to define two sets of conditions for the sequences X either to establish a threshold edit distance or to discard a set of initial states. In this way, once a subsequence of Y fulfills some of the previous conditions, the cost of the corresponding cut set can be guaranteed either to be minimum or

not to be minimum, respectively. This fact has direct consequences on the costs of the optimal paths, that is to say, on the edit distances.

The cut sets that we use in this work are defined as follows. Each cut set $C_{i+j}, 1 \leq i+j \leq N$ contains both the set of all the arcs corresponding to the vertex x_{i+j}, and all those arcs corresponding to bits x_w with $w > i+j$ whose output vertex is one of the output vertices of the former set. From these cut sets, we deduce several independent conditions on the sequence Y that may be used to guarantee both a decrease and an increase on the edit distances of different sequences X. In particular, the conditions obtained from the defined cut sets may be described by the formulas in Equation (2).

$$\begin{array}{l} \forall j : 1, 2, \ldots, \lfloor M/2 \rfloor; y_j = y_{j-1} = y_{j-2} = \cdots = y_{\lceil j/2 \rceil} \\ \forall j : \lfloor M/2 \rfloor + 1, \lfloor M/2 \rfloor + 2, \ldots, \lceil 3M/4 \rceil - 1; y_j = y_{j-1} = \cdots = y_{j-\lceil (M-2)/4 \rceil} \\ \forall j : \lceil 3M/4 \rceil, \lceil 3M/4 \rceil + 1, \ldots, M;\ y_j = y_{j-1} = y_{j-2} = \cdots = y_{2j-M} \end{array} \quad (2)$$

The checking procedure of these hypothesis takes polynomial time as it implies a simple verification of the lengths of the runs in Y. After having checked each hypothesis separately, the tools used to verify both sets of conditions on X are described in terms of a pattern and a counterpattern, which are made out of independent bits of X according to the formulas in Equation (3).

$$\begin{array}{l} \forall j : 1, 2, \ldots, \lfloor M/2 \rfloor; \text{ if } y_j = y_{j-1} = y_{j-2} = \cdots = y_{\lceil j/2 \rceil} \text{ then} \\ \quad \begin{cases} x_j = x_{j+1} = y_j & X - Pattern \\ x_j = x_{j+1} \neq y_j & X - Counterpattern \end{cases} \\ \forall j : \lfloor M/2 \rfloor + 1, \ldots, \lceil 3M/4 \rceil - 1; \text{ if } y_j = y_{j-1} = \cdots = y_{j-\lceil (M-2)/4 \rceil} \text{ then} \\ \quad \begin{cases} x_{2j-\lceil M/2 \rceil} = x_{2j-\lceil M/2 \rceil+1} = x_{2j-\lceil M/2 \rceil+2} = y_j & X - Pattern \\ x_{2j-\lceil M/2 \rceil} = x_{2j-\lceil M/2 \rceil+1} = x_{2j-\lceil M/2 \rceil+2} \neq y_j & X - Counterpattern \end{cases} \\ \forall j : \lceil 3M/4 \rceil, \lceil 3M/4 \rceil + 1, \ldots, M; \text{ if } y_j = y_{j-1} = y_{j-2} = \cdots = y_{2j-M} \text{ then} \\ \quad \begin{cases} x_{2j-\lceil M/2 \rceil} = x_{2j-\lceil M/2 \rceil+1} = x_{2j-\lceil M/2 \rceil+2} = y_j & X - Pattern \\ x_{2j-\lceil M/2 \rceil} = x_{2j-\lceil M/2 \rceil+1} = x_{2j-\lceil M/2 \rceil+2} \neq y_j & X - Counterpattern \end{cases} \end{array} \quad (3)$$

The X-pattern allows to discover initial states producing sequences X with a low edit distance. Furthermore, the X-pattern provides a good quality threshold for the method that will be described in the following Section. On the other hand, sequences X fulfilling the X-counterpattern lead to high edit distance values and consequently may be directly discarded.

Example : Given a sequence Y of length M=7 and a candidate sequence X of length N=10, we may define the cut sets shown in Fig. 3. The 6 independent hypothesis on runs in Y that are deduced from those cut sets are $y_2 = y_1$, $y_3 = y_2$, $y_4 = y_3 = y_2$, $y_5 = y_4 = y_3$, $y_6 = y_5 = y_4$, and $y_6 = y_5$. Consequently, since the example Y:1101011 only fulfills the first hypothesis, the second and third bits are fixed in both the X-pattern and the X-counterpattern. If the length of the $LFSR$ equals 9 and its feedback polynomial is $1 + x + x^3 + x^4 + x^9$, then only 16 initial states (out of the $2^9 = 512$ possible) will satisfy the X-pattern:111.....11 and may be considered as the most promising initial states. Consequently, they need to be fully evaluated in order to deduce a threshold on the edit distance. On the other hand, the counterpattern of X is defined by: 000.....00. In a similar

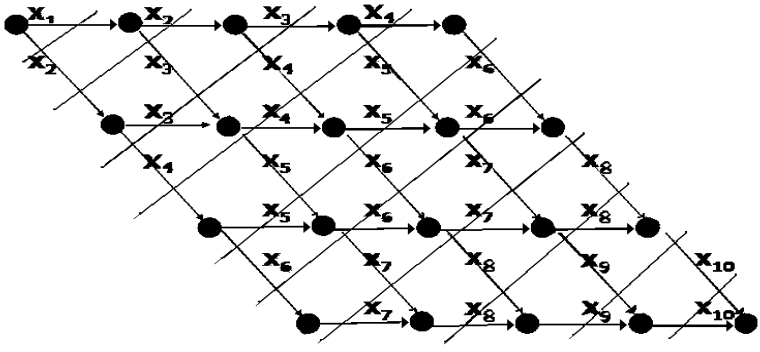

Fig. 3. Cut sets

way as before, only 16 initial states (out of the $2^9 = 512$ possible) will satisfy the X-counterpattern. They must be rejected as they are the less promising initial states. Indeed, in this example, it may be checked that there are 10 sequences X fulfilling the X-pattern that are solutions to the cryptanalysis. In fact, their edit distance is 3 that is precisely the minimum edit distance for this example. Moreover, it might be also verified that all the 16 initial states fulfilling the X-counterpattern lead to edit distances greater than 3.

4 Improved Attack

The threshold obtained through the X-pattern as well as the discarded initial states deduced from the X-counterpattern are items of the improved attack that is described in this Section. The algorithm here developed makes use of a new concept, the so-called *bad column*, which leads to a considerable saving in the computation of the edit distance matrices. A *bad column* with respect to a threshold T may be defined as a column j_0 of the edit distance matrix W such that each one of their elements fulfills the Equation (4):

$$w_{i,j_0} > T - (N - M - i), \forall i \tag{4}$$

Once an edit distance threshold has been obtained, we may use such a threshold to stop the computation of any matrix W as soon as a *bad column* has been detected. This is due to the knowledge that the edit distance corresponding to the candidate initial state will be greater than the threshold. In this simple way, two new improvements on the original attack may be achieved. On the one hand, as yet mentioned, the computation of any matrix may be stopped as soon as a *bad column* is obtained. On the other hand and thanks to the association between bits x_{i+j} and arcs of the graph, we may define a new counterpattern on the initial states of the $LFSR$, the so-called IS-counterpattern. This new concept will allow us to discard the set of initial states fulfilling such an IS-counterpattern when

an early *bad column* has been detected. This is so because once a *bad column* has been obtained, it is possible to discard directly all the initial states whose first bits coincide with those used within the computation of the *bad column*. Note that in order to take full advantage of *bad columns*, it is convenient to have some efficient way of obtaining soon a good threshold. That is exactly the effect of the X-pattern described in the previous section.

We are now ready to describe the improved edit distance attack algorithm

Algorithm
Input: The intercepted keystream sequence Y of length M, and the feedback polynomial of the $LFSR$ of length L.
Output: The initial states of the $LFSR$ producing sequences X of length $3M/2$ whose constrained edit distance is minimum, and the corresponding decimation and noise sequences D and B, respectively.

1. Verification of the hypothesis on Y described in Equation (2) .
2. Definition of the X-pattern and X-counterpattern according to Equation (3).
3. Rejection of all initial states that produce sequences X fulfilling the X-counterpattern.
4. For each initial state that produces a sequence X fulfilling the X-pattern:
   (a) Computation of the edit distance matrix according to Equation (1).
   (b) Updating of the threshold T.
5. For each initial state producing a sequence X that does not fulfill the X-pattern and that has not been previously discarded:
   (a) Computation of the edit distance matrix stopping when detection of *bad columns* according to threshold T and Equation (4).
   (b) Definition of the IS-counterpattern.
   (c) Rejection of all initial states producing sequences X fulfilling the IS-counterpattern.
6. For each initial state producing a sequence X with minimum edit distance:
   (a) Recovery of the optimal paths from the graph induced by the edit distance matrix.
   (b) Translation from each optimal path into a couple of decimation and noise sequences (D,B).

Example :
Given a sequence Y:1101011 of length M=7, and the feedback polynomial: $1+x+x^3+x^4+x^9$ of the $LFSR$ of length L=9. Since the unique hypothesis fulfilled by Y is: $y_2 = y_1$, the X-pattern:111.....11, and the X-counterpattern:000.....00. So, we have the consequent rejection of the 16 initial states producing sequences of the form 000.....00. Also, for each one of the 16 initial states generating sequences of the form 111.....11 the edit distance matrix is computed, and from this computation the threshold T=3 is obtained. For the remaining 480 initial states, we start computing the edit distance matrix, and stop as soon as a *bad column* for the threshold T=3 is detected. In particular, we have to start to evaluate:

- 1 initial state in order to discard $2^7 = 128$ initial states, including those ones discarded by the X-counterpattern, corresponding to matrices W containing the *bad column* j_0=1.
- 3 initial states in order to discard 96 initial states due to the fact that the column j_0=2 of W is a *bad column*.
- 5 initial states which allow us to discard 40 initial states due to the fact that the column j_0=3 of W is a *bad column*.
- 12 initial states which allows us to discard 48 initial states (including 2 that fulfill the pattern) due to the fact that the column j_0=4 of W is a *bad column*.
- 24 initial states in order to discard 48 initial states (including 3 states fulfilling the pattern) due to the fact that the column j_0=5 of W is a *bad column*.
- 42 initial states where the column j_0=6 of W is a *bad column*.
- 39 initial states where the column j_0=7 of W is a *bad column*.

Regarding solutions, in this example we find exactly 48 initial states producing sequences X with edit distance equal to 3. In fact, 10 initial states were directly detected in Step 4, while the remaining 38 solution states turned up as a result of the last step. For each one of the 48 initial states producing a sequence X with minimum edit distance, we have to recover the optimal paths from the graph in order to translate them into decimation and noise sequences.

5 Simulation Results

The next table shows some results for experimental implementations of the algorithm. At column denoted Pol., the positive exponents of the feedback polynomial of the $LFSR$ are represented. The columns denoted Seq.count.pat. display the number of sequences that fulfill the X-counterpattern (X-pattern). The column marked with Sol.pat. gives the number of initial states producing sequences X that are solutions. Thres. and Dist. are the columns where the obtained threshold and the minimum edit distance are shown. Finally, %Sav reflects a lower bound on the percentage of saving in the computing time and memory of the proposed algorithm compared with the original constrained edit distance attack.

From these randomly generated examples, we may deduce a general classification of inputs into several cases. The best ones correspond to patterns which directly identify solutions. Contrarily, bad cases are those in which the pattern is not fulfilled by any initial state. Such cases are generally associated with long runs at the beginning and at the end of the sequences Y. Finally, the medium cases are those for which, despite the non existence of solutions fulfilling the pattern, a good threshold is obtained. Such cases allow a good percentage of saving in computing thanks to the detection of many early *bad columns*.

N	M	L	Pol.	2^L	Seq.count.pat.	Sol.pat.	Thres.	Dist.	%Sav.
20	15	7	1,7	128	0	0	-	5	28.3
30	20	9	1,3,4,9	512	1	0	11	10	56.3
33	22	7	1,7	128	2	1	12	12	30.7
45	30	7	1,7	128	0	0	-	16	20.8
75	50	7	1,7	128	8	0	27	29	24
150	100	9	1,3,4,9	512	32	0	57	55	36.3
300	200	13	1,3,4,13	8192	128	0	115	114	25.9
450	300	17	3,17	131072	512	0	174	173	20.6
750	500	14	1,3,5,14	16384	1024	1	291	291	28.33

These empirical results show that in most cases the improvement in time complexity of the attack is greater than 25%. Furthermore, it is clear that the worst outcomes appear when the initial results in steps 1 to 4 are not adequate as there are not initial states fulfilling the pattern or the counterpattern. Since the hypothesis on Y are independent, the groups of bits in the X-pattern and in the X-counterpattern are also independent. Consequently, the conditions may be considered separately defining in this way a relaxed X-pattern and X-counterpattern which may lead to sequences that fulfill them. In addition, empirical results have shown that intercepted sequence Y with short runs at the beginning and at the end cause a greater improvement in the time complexity of the attack. Thus, another way to avoid a bad behavior of the original algorithm is by choosing subsequences from the intercepted sequence Y that have no too long runs at the beginning and at the end, and applying the algorithm to each one of these subsequences.

6 Conclusions

In this work a new algorithm based on two different and independent ways to improve a known constrained edit distance attack on clock-controlled $LFSR$-based generators has been proposed. The described algorithm avoids the exhaustive search over all the initial states of the involved $LFSRs$. The most remarkable aspect of this work is that the general ideas that have been proposed may be applied to attack any clock-controlled LFSR-based stream cipher.

References

1. Anderson, R.J.: A Faster Attack on Certain Ciphers, Electronics Letters, Vol. 29 No. 15, July (1993) 1322-1323.
2. Bluetooth, *Specifications of the Bluetooth system,* Version 1.1, February 2001, available at http://www.bluetooth.com/
3. Clark, A. *et al.*: The LILI-II Keystream Generator. Proc. ACISP 2002, Lecture Notes in Computer Science Vol.2384 Springer-Verlag (2002) 25-39.

4. Coppersmith, D., Krawczyk, H., Mansour, H.: The Shrinking Generator, Proc. Crypto'93, Lecture Notes in Computer Science Vol.773 Springer-Verlag (1994) 22-39.
5. CRYPTREC project- cryptographic evaluation for Japanese Electronic Government, www.ipa.go.jp/security/enc/CRYPTREC/index-e.html
6. Golic, J.D.: Recent Advances in Stream Cipher Cryptanalysis. Publication de l'Institut Mathematique Tome 64 (78) (1998) 183-204.
7. Golic, J.D., Menicocci, R.: Correlation Analysis of the Alternating Step Generator. Design Codes and Cryptography 31 (1) (2004) 51-74.
8. Golic, J.D., Mihaljevic, M.: A Generalized Correlation Attack on a Class of Stream Ciphers Based on the Levenshtein Distance, Journal of Cryptology, Vol. 3, No. 3 (1991) 201-212.
9. Golic, J.D., Petrovic, S.: A Generalized Correlation Attack with a Probabilistic Constrained Edit Distance, Proc. Eurocrypt'92, Lecture Notes in Computer Science Vol.658 Springer-Verlag (1993) 472-476.
10. Gollmann, D., Chambers, W.C.: Clock-Controlled Shift Registers: A Review, IEEE Transactions on Selected Areas in Communications SAC-7 May (1989) 525-533.
11. Golomb,S.W.: Shift Register-Sequences, Aegean Park Press, Laguna Hill, 1982.
12. GSM, *Global Systems for Mobile Communications,* available at http://cryptome.org/gsm-a512.htm
13. Jiang, S., Gong, G.: On Edit Distance Attack to Alternating Step Generator, Technical Report Corr2002-28, University of Waterloo (2002).
14. Johansson, T.: Reduced Complexity Correlation Attacks on Two Clock-Controlled Generators, Proc. Asiacrypt'98, Lecture Notes in Computer Science Vol.1514 Springer-Verlag (1998) 342-356.
15. Petrovic, S., Fúster, A.: Clock Control Sequence Reconstruction in the Ciphertext Only Attack Scenario, Proc. ICICS 2004, Lecture Notes in Computer Science Vol.3269 Springer-Verlag (2004) 427-439.