On the Security of the Gollmann Cascades

Sang-Joon Park Sang-Jin Lee Seung-Cheol Goh

Electronics and Telecommunications Research Institute
161 Kajong-Dong, Yusong-Gu, Taejon, 305-350, Korea
e-mail: goh@dingo.etri.re.kr

Abstract. The purpose of this paper is to evaluate the security of the Gollmann m-sequence cascades of k stages. We give some theoretical results, which can be utilized to construct the transition matrix T_n of the conditional probabilities between the input and output strings of a stage. And then, we describe an attack algorithm for guessing the initial state of the first LFSR with desired reliability, using the transition matrix $S_n = T_n^{k-1}$ of the conditional probabilities between the input string of the second stage and the output of the final stage of the given k-stage cascade. We finally evaluate the security of the cascades against this attack. Menicocci recently conjectured that there do not exist the complete analysis of the Gollmann cascades of more than 4 stages and it is infeasible to attack the 10-stage cascades with LFSRs of degree 100. Our experimental results show that the 9-stage cascades with LFSRs of degree 100 are completely breakable and the 10-stage cascades may be insecure.

1 Introduction

The purpose of this paper is to evaluate the security of the Gollmann m-sequence cascades of k stages[1].

A Gollmann m-sequence cascade of k stages consists of a series of k Linear Feedback Shift Registers(LFSRs), with primitive feedback polynomials of same degree n. It produces pseudo-random binary sequences of period $(2^n-1)^k$, linear equivalence exceeds $n(2^n-1)^{k-1}$[1]. The first LFSR is regularly clocked, whereas all registers except the first are clock controlled by their predecessors. A binary input bit a_t clocks a LFSR if $a_t = 1$, and then is added to the output from the LFSR to give output c_t, which becomes the input of the next.

$$c_t = a_t + b(h_t) \bmod 2,\ h_t = h_{t-1} + a_t \bmod 2^n-1,\ t = 0, 1, 2, \cdots \qquad (1)$$

with the initial condition $h_{-1} = 0$, where $b(\cdot)$ is the sequence generated by the LFSR.

A weakness of the Gollmann cascades, called *lock-in effect*, was studied in [2]. But the attack by lock-in effect requires 10^{21} iterations to analysis the two-stage cascades with polynomials of degree 34. Menicocci proposed an efficient attack on the two-stage Gollmann m-sequence cascades, which utilized the correlations between the input and output strings of the final stage of cascades[4]. He also

D. Coppersmith (Ed.): Advances in Cryptology - CRYPTO '95, LNCS 963, pp. 148-156, 1995.

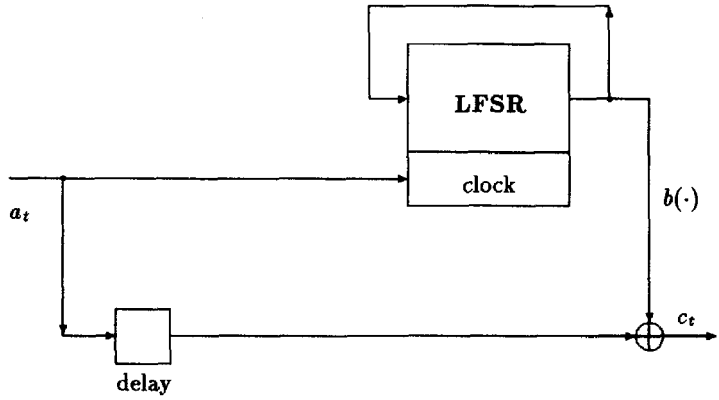

Fig 1: A stage of the m-sequence cascade; The input bit a_t clocks LFSR, and then is added to the output from the LFSR to give output c_t. The *delay* means the addition takes place after clock.

proposed an attack on the Gollmann cascade of k stages, and conjectured there do not exist algorithms for the complete analysis of the cascade of more than 4 stages, and it is infeasible to attack on the m-sequence cascade consists of more than 10 stages with primitive polynomials of degree greater than 100[5].

In this paper, we give some theoretical results, which can be utilized to construct the transition matrix T_n of the conditional probabilities between the input and output strings of a stage. And then, we describe an attack algorithm for guessing the initial state of the first LFSR with desired reliability, using the transition matrix $S_n = T_n^{k-1}$ of conditional probabilities between the input string of the second stage and the output of the final stage of the given k-stage cascade. We also evaluate the security of the cascades against this attack. We finally give experimental results on the cryptanalysis of the cascades with less than 10 stages. Our experimental results show that the 9-stage cascades with LFSRs of degree 100 are completely breakable and the 10-stage cascades may be insecure.

2 Constructions of Transition Matrices

Definition Let $w = x_0x_1 \cdots x_{n-1}$ be a n-bit word. Then the value of w, denoted by $f(w)$, is

$$f(w) = \sum_{k=0}^{n-1} x_k 2^{n-1-k} \tag{2}$$

Definition Let w and v be n-bit words such that $f(w) = i$, $f(v) = j$. And let $t^n[i,j]$ be the conditional probability that a given stage of the cascade produce v when w is applied to it. Then the $2^n \times 2^n$ matrix $T_n = (t^n[i,j])$, $0 \leq i < 2^n$,

$0 \le j < 2^n$, is called a transition matrix of the correlation probabilities between the input and output words of the stage.

Remark For a given cascade of k stages, $S^n = (T_n)^{k-1} = (s^n[i,j])$ is the transition matrix of the conditional probabilities, where $s^n[i,j]$ is the probability that the final stage generates $v = f^{-1}(j)$ when $w = f^{-1}(i)$ is applied to the second stage of the given cascade.

Example If $a_t = 0$, $c_t = b(S_t)$. The sequence $b(\cdot)$ is generated by LFSR, so we have $P(c_t = 0) = P(c_t = 1) = \frac{1}{2}$. If $a_t = a_{t+1} = 0$, then $c_{t+1} = b(S_{t+1}) = b(S_t) = c_t$, so we have $P(c_{t+1} = c_t = 0) = P(c_{t+1} = c_t = 1) = \frac{1}{2}$. Continuing this process, we will finally have

$$T_2 = \begin{pmatrix} \frac{1}{2} & 0 & 0 & \frac{1}{2} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \end{pmatrix}, \quad (T_2)^2 = T_2 \circ T_2 = \begin{pmatrix} \frac{3}{8} & \frac{1}{8} & \frac{1}{8} & \frac{3}{8} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \end{pmatrix}$$

Definition Let $S = \{t, t+1, \cdots, t+n-1\}$ be a set of n consecutive non-negative integers, and let $w = a_t a_{t+1} \cdots a_{t+n-1}$ be a n-bit word such that $f(w) = i$. Then, we may define a class P_i of subsets S_k by following procedure.

Procedure 1
Set $k = 1$ and $S_k = \{t\}$.
For all $l = t+1, \cdots, t+n-1$
    If $a_l = 0$, then set $S_k = S_k \cup \{l\}$.
    Otherwise, set $k = k+1$ and set $S_k = \{l\}$.

Property 1 Let $w = a_t a_{t+1} \cdots a_{t+n-1}$ be a n-bit word such that $f(w) = i$. And let m be the number of ones in $\{a_t, a_{t+1}, \cdots, a_{t+n-1}\}$. Then the number of subsets $|P_i|$ of the class P_i is

$$|P_i| = \begin{cases} m & \text{if } a_t = 1 \\ m+1 & \text{otherwise} \end{cases} \tag{3}$$

Proof. By definition of the class P_i, it is trivial. □

Property 2 Assume that a stage of a cascade produces $v = c_t c_{t+1} \cdots c_{t+n-1}$ when $w = a_t a_{t+1} \cdots a_{t+n-1}$ is applied. If $P_i = \{S_1, \cdots, S_m\}$ is the class determined by Procedure 1, then, for all l in S_k,

$$a_l + c_l \bmod 2 = \text{const} \tag{4}$$

Proof. From (1), it is trivial. □

Property 3 For given integers i and j, where $0 \le i < 2^n$, $0 \le j < 2^n$, let $w = a_t a_{t+1} \cdots a_{t+n-1}$ and $v = c_t c_{t+1} \cdots c_{t+n-1}$ be the binary representations of i and j respectively. And let $P_i = \{S_1, \cdots, S_m\}$ be the class determined by Procedure 1. Then $t^n[i,j] = 0$ if and only if there exists an integer $0 \le k \le m$ such that $a_l + c_l \ne a_{l'} + c_{l'} \bmod 2$ for some l, l' in S_k,

Proof. By Property 2, it can be easily proved. □

Property 4 If $t^n[i,j] \neq 0$ for some $0 \leq i < 2^n$, $0 \leq j < 2^n$, then $t^n[i,j] = \frac{1}{2^{|P_i|}}$.

Proof. Let $w = a_t a_{t+1} \cdots a_{t+n-1}$ be the binary representations of i. And let P_i be the class determined by Procedure 1. If a stage produces $v = c_t c_{t+1} \cdots c_{t+n-1}$ when w is applied, then, by Property 2, there exist constants $b_1, b_2, \cdots, b_{|P_i|}$, such that $c_l = a_l + b_k \bmod 2$ for all l in S_k, where S_k is in P_i. This means that, when w is applied, all outputs are uniquely determined by $b_1, b_2, \cdots, b_{P_i}$, so that if $t^n[i,j] \neq 0$, $t^n[i,j] = \frac{1}{2^{P_i}}$. □

Property 5 For a given integer $0 \leq i < 2^n$, there are exactly 2^{P_i} number of nonzero elements in $T_{n,i}$, where $T_{n,i}$ is the i-th row of the transition matrix T_n.

Proof. By Property 4, trivial. □

Property 6 If $0 \leq j < 2^{n-1}$, $t^n[i,j] = t^n[i, 2^n-1-j]$ for all $0 \leq i < 2^n$.

Proof. Let $w = a_t a_{t+1} \cdots a_{t+n-1}$ and $v = c_t c_{t+1} \cdots c_{t+n-1}$ be n-bit words such that $f(w) = i$, $f(v) = j$. If the value of $\bar{v} = d_t d_{t+1} \cdots d_{t+n-1}$ is 2^n-1-j, then we have, $d_l = c_l + 1 \bmod 2$ for all l, $l = t, \cdots, t+n-1$. Let $P_i = \{S_1, S_2, \cdots, S_{|P_i|}\}$ be the class determined by Procedure 1. If $t^n[i,j] = 0$, by Property 3, there is an integer k, $1 \leq k \leq |P_i|$, such that $a_l + c_l \neq a_{l'} + c_{l'} \bmod 2$ for some l, l' in S_k. Hence $a_l + d_l \neq a_{l'} + d_{l'} \bmod 2$, so $t^n[i, 2^n-1-j] = 0$. If $t^n[i,j] \neq 0$, by Property 3, we have, $a_l + d_l = a_l + c_l + 1 \bmod 2 = const$ for all l in S_k. Hence, by Property 4, $t^n[i,j] = t^n[i, 2^n-1-j] = \frac{1}{2^{|P_i|}}$. □

Property 7 For all $0 \leq i, j < 2^{n-1}$, $t^n[i,j] = t^n[2^{n-1}+i, 2^{n-1}+j]$.

Proof. We first note that $P_i = P_{2^{n-1}+i}$, where P_i and $P_{2^{n-1}+i}$ are the classes determined by Procedure 1. Let $w = a_t a_{t+1} \cdots a_{t+n-1}$, $v = c_t c_{t+1} \cdots c_{t+n-1}$. Then $w' = \bar{a}_t a_{t+1} \cdots a_{t+n-1}$ and $v' = \bar{c}_t c_{t+1} \cdots c_{t+n-1}$ are the binary representations of $2^{n-1}+i and 2^{n-1}+j$ respectively. Since $a_t + c_t = \bar{a}_t + \bar{c}_t$, we have $t^n[i,j] = t^n[2^{n-1}+i, 2^{n-1}+j]$. □

Similarly, we can prove:

Property 8 For all $0 \leq i, j < 2^{n-1}$, $t^n[2^{n-1}+i, j] = t^n[i, 2^{n-1}+j]$.

Property 9 In the case of $n \geq 3$, $t^n[i,j] = 0$ if $0 \leq i < 2^{n-2}$, $2^{n-2} \leq j < 2^{n-1}$.

Proof. Let $w = a_t a_{t+1} \cdots a_{t+n-1}$ and $v = c_t c_{t+1} \cdots c_{t+n-1}$ be n-bit words such that $f(w) = i$ and $(v) = j$. Then, we have, $a_t = a_{t+1} = c_t = 0$, $c_{t+1} = 1$. We note that $t, t+1$ belong to same set S_1 in the class P_i. But $a_t + c_t \neq a_{t+1} + c_{t+1} \bmod 2$ so $t^n[i,j] = 0$. □

Property 10 For all $0 \leq i, j < 2^{n-2}$, $t^n[i,j] = t^{n-1}[i,j]$.

Proof. Let P_i be the class of subsets of the set $S = \{t, t+1, \cdots, t+n-1\}$, and let P_i' be the class of subsets of the set $S = \{t+1, \cdots, t+n-1\}$ determined by Procedure 1, when i is applied. Note $|P_i| = |P_i'|$, $S_1' = S_1 - \{t\}$, $S_k' = S_k$, $k > 1$. Let $w = a_t a_{t+1} \cdots a_{t+n-1}$ and $v = c_t c_{t+1} \cdots c_{t+n-1}$ be the binary representations of i, j, respectively. If $t^n[i,j] \neq 0$, then $a_l + c_l \bmod 2 = const$ if l in S_k for all S_k in P_i. Hence $a_l + c_l \bmod 2 = const$ if l in S_k' for all S_k' in P_i', so that $t^{n-1}[i,j] \neq 0$.

Therefore, if $t^n[i,j] \neq 0$, then, by Property 4, $t^n[i,j] = t^{n-1}[i,j] = \frac{1}{2^{|P_i|}}$. In the case of $t^n[i,j] = 0$, we can prove similarly $t^n[i,j] = t^{n-1}[i,j] = 0$. □

Property 11 For all $2^{n-2} \leq i < 2^{n-1}$, $t^n[i,j] = \frac{1}{2}t^{n-1}[i,j]$.

Proof. Let P_i and P_i' be the classes defined in the proof of Property 10. In a similar way, we can easily prove that $t^n[i,j] = 0$ if and only if $t^{n-1}[i,j] = 0$. We note that, by Property 1, $|P_i| = |P_i'|+1$. Hence, by Property 4, if $t^n[i,j] \neq 0$, $t^n[i,j] = \frac{1}{2}t^{n-1}[i,j]$. □

Property 12 Let $w = a_t a_{t+1} \cdots a_{t+n-1}$ be the binary representation of i. Then $t^n[i,0] = 0$ if and only if there exist l, l' such that $t \leq l < l' \leq t+n-1$, $a_l = 1$ but $a_{l'} = 0$.

Proof. Assume that $a_l = 1$ but $a_{l'} = 0$ for some l, l' such that $t \leq l < l' \leq t+n-1$. Let $f(w) = i$. And let S_k be the subset of P_i which contains l'. Let us define $h = \max\{m \mid l \leq m < l', a_m = 1\}$. Then $h \in S_k$ and $a_h+0 \neq a_{l'}+0 \bmod 2$ so that $t^n[i,0] = 0$. Suppose that $t^n[i,0] = 0$. Then it is clear that $i \neq 0, 1$. Hence there exists at least one l such that $t \leq l < t+n-1$ and $a_l = 1$, say $p = \min\{l | t \leq l < t+n-1, a_l = 1\}$. In the case of $p = t$, if $a_l = 1$ for all $t+1 \leq l \leq t+n-1$, then each S_k, $k = 1, 2, \cdots, t+n-1$, has only one element so that $t^n[i,0] \neq 0$. So we may assume $p \geq t+1$ ie $a_t = 0$. If $a_l = 1$ for all l, $p \leq l \leq t+n-1$, then only S_0 may have more than one elements. Since all a_l in S_0 are 0 so $t^n[i,0] \neq 0$. □

From Property 12, we have

Property 13

$$t^n[i,0] = t^n[i,2^n-1] = \begin{cases} \frac{1}{2^{k+1}} & \text{if } i = 2^k-1 \text{ for some } k,\ 0 \leq k \leq n-1 \\ \frac{1}{2^n} & \text{if } i = 2^n-1 \\ 0 & \text{Otherwise} \end{cases}$$

Now we summarize Properties solved in this section to give Theorem 1.

Theorem 1. *For all $n \geq 3$, let $H_n = (h^n[i,j]), H_n' = (h'^n[i,j])$ be $2^{n-1} \times 2^{n-1}$ matrices of the probabilities such that $h'^n[i,j] = h^n[i, 2^{n-1}-1-j]$. Then*
1) the transition matrix $T_n = (t^n[i,j])$ can be represented as:

$$T_n = \begin{pmatrix} H_n & H_n' \\ H_n' & H_n \end{pmatrix} \tag{5}$$

2) the $2^{n-1} \times 2^{n-1}$ matrix $H_n = (h^n[i,j])$ can be represented as:

$$H_n = \begin{pmatrix} H_{n-1} & 0 \\ \frac{1}{2}H_{n-1}' & \frac{1}{2}H_{n-1} \end{pmatrix} \tag{6}$$

with the initial condition $H_1 = (\frac{1}{2})$.

3 The attack proposed

Our aim is to find all initial states of LFSRs of the given cascade, under the assumption that we know a sufficiently large number of consecutive bits of the output and all primitive polynomials.

For a given cascade of k stages, we define a matrix S^n by

$$S^n = (T_n)^{k-1} = (s^n[i,j]) \tag{7}$$

Then the component $s^n[i,j]$ is the probability that the final stage produces $v = f^{-1}(j)$ when $w = f^{-1}(i)$ is applied to the second stage of the cascade. This matrix S^n enables us to guess the initial state of the LFSR of the first stage with desired reliability.

Theorem 2. *For all i, $0 \le i < 2^n$, let w_i be the n-bit word such that $f(w_i) = i$, and A_i the event that w_i is applied to the second stage of the cascade. And let E_0 be the event that the cascade generates a run of n consecutive zeros and E_1 a run of n consecutive ones. If $E = E_0 \cup E_1$, then*

$$P(A_i|E) = s^n[i,0] \tag{8}$$

where $P(A_i|E)$ denotes the conditional probability of A_i occurs given that E has occurred.

Proof. By Bayes's theorem, we have,

$$P(A_i|E_0) = \{P(A_i)P(E_0|A_i)\}/\{\sum_{m=0}^{2^n-1} P(A_m)P(E_0|A_m)\}$$
$$= P(E_0|A_i)/\sum_{m=0}^{2^n-1} P(E_0|A_m) = s^n[i,0]/\sum_{m=0}^{2^n-1} s^n[m,0] = s^n[i,0]$$

Hence, $P(A_i|E) = \frac{1}{2}P(A_i|E_0) + \frac{1}{2}P(A_i|E_1) = \frac{1}{2}(s^n[i,0]+s^n[i,2^n-1]) = s^n[i,0]$.

Corollary 3. *Let $w = a_t a_{t+1} \cdots a_{t+n-1}$. Then, for all k, $0 \le k \le n-1$,*

$$P(a_{t+k} = 0|E) = \sum_i s^n[i,0] \tag{9}$$

where i ranges over integers such that $i \wedge 2^{n-k-1} = 0$. The notation $\wedge$ denotes the bitwise AND of two integers. Hereafter, for all $0 \le k \le n-1$, we define q_k by $q_k = 1-P(a_{t+k} = 0|E)$.

For a precise description of our attack, let m be the same degree of primitive polynomials and let p be the desired error rate. In order to find the initial state of LFSR in the first stage, we first scan the given output bits to find all runs of at least n consecutive zeros or ones. Then we scan each run for k such that $q_k \le p$, to set linear equations, which have the m unknowns for the initial state.

In this way, we can set a sufficiently large number, say l, of linear equations. Choose randomly m out of the l equations and solve the linear system. If it has a solution, say $\bar{x} = x_0, x_1, \cdots, x_{m-1}$, then examine how many equations hold when $\bar{x}$ is substituted. If it is the real initial state, the number of equations held is on average $l(1-p)$. This is repeated for each except final stage. And then, apply the algebraic technique[6] to the final stage to guess the initial state of its LFSR. Test, finally, whether the cascade produces the given output sequence. If not, repeat again.

How many trials of selections m from l equations is needed to find the initial state of the first stage. By a similar way in [3], we can estimate the probability q that a trial successes, so q^{-1} is the expected number of trials.

$$q = (1-\frac{pl}{l})(1-\frac{pl}{l-1})\cdots(1-\frac{pl}{l-m+1}) \geq (1-\frac{pl}{l-m+1})^m \qquad (10)$$

Example Consider a cascade of 2 stages with LFSRs of the same length 34. We are assumed to know 800 consecutive output bits, so that we can find, on average, 25 number of runs whose lengths are greater than or equal to 5. If we put $p = 2^{-4}$, we can set 50 linear equations so $q \approx 10^{-3}$. Hence about 1000 trials are expected to find the correct solution.

Algorithm: An attack on the Gollmann k-stage cascades

Step 1 Scan the given output bits, $a_h, a_{h+1}, a_{h+2}, \cdots\cdots$, to find all runs of at least n consecutive zeros or ones. This number n may be determined by the desired error rate p, the number of given output bits, and the degrees of LFSRs.

Step 2 Repeat the process of Steps 3 - 6 in order to guess the initial states of LFSRs of all but final stages with desired reliability $1-p$.

Step 3 Select a_t such that $P(a_t = 0|E) \geq 1-p$ to set a linear equation, which has the m unknowns for the initial state. In this way, we can set a sufficiently large number, say l, of linear equations.

Step 4 Select randomly m out of the l equations and solve the linear system.

Step 5 If it has no solution, goto Step 4.

Step 6 Examine how many equations hold when the solution is substituted. If the number is less than k, which is heuristically determined by $l(1-p)$, goto Step 4.

Step 7 Apply the algebraic technique to the kth stage using the sequence generated by the $k-1$th stage.

Step 8 Test whether the cascade produces the correct final output sequences. If success terminate, else goto Step 2.

4 Security evaluation

The security of the Gollmann cascades may be evaluated by the number of trials for obtaining the correct initial state of its first LFSR and the number of the final output bits for obtaining runs whose lengths are sufficiently large enough

to guarantee small error rate. From the view point of cryptanalysis, there may be trade-off between the number of trials and bits. When the length of runs is greater than 9, we could not calculate correctly the error rate and the number of trials due to the memory problem of computer. However, by some statistical observation, we expect that the runs of length 20 enable us to break completely the cascades of 10 stages with LFSRs of degree 100.

We have implemented the proposed attack in C on an Axil Hyundai workstation(80 MIPS), compatible with a SUN Sparc 10, in a UNIX environment. For the case of 2-stage cascades with primitive polynomials of degree 34, we have found all initial states of LFSRs within 2.8 CPU seconds on average.

Following table gives our experimental results. In those tables, Degree denotes the degree of primitive polynomials, Stage the numbers of stages, Bit the numbers of output bits, Run the lengths of runs, p desired error rates, Eqns the numbers of equations, Trial the numbers of trials, Theo the numbers theoretically calculated, Expe the numbers of experimental results, CPU the CPU seconds needed execution the algorithm.

Table: Experimental Results

Degree	Stage	Bit	Run	p	Eqns	Trial		CPU
						Theo	Expe	
34	2	800	5	0.0625	75	18	15	2.8
	3	1800	6	0.1269	82	423	119	4.5
	4	7000	8	0.1208	73	399	96	7.2
	5	35000	10	0.1145	110	146	241	49.4
44	2	1000	5	0.0625	95	46	23	8.7
	3	2300	6	0.1269	116	2033	626	17.4
	4	9000	8	0.1208	109	1576	436	28.4
	5	45000	10	0.1145	117	891	255	68.8
54	2	1200	5	0.0625	115	114	437	18.2
	3	3000	6	0.1269	127	16734	1843	56.7
	4	15000	8	0.0801	136	327	873	99.2
	5	60000	10	0.0801	121	412	271	155.3
100	3	20000	7	0.04699	304	350	896	123.0
	4	60000	9	0.05235	256	949	1892	265.1
	5	240000	11	0.05318	263	1006	3950	687.3
	6	500000	12	0.08440	242	99874	4282	721.0
	7	2000000	14	0.08960	230	267088	1741	828.0
	8	8000000	16	0.08562	262	88956	39147	2334.0
	9	30000000	18	0.07188	248	15340	51006	3322.0

5 Concluding Remarks

In this paper, we have evaluated the security of the Gollmann m-sequence cascades. We have described, in detail, an algorithm for guessing the initial state of

the first LFSR with desired reliability. We have also given experimental results on the complete analysis of 9-stage cascades with polynomials of degree 100. We have finally defined the security of cascades by the number of trials for obtaining the correct initial state of its first LFSR and the number of the final output bits obtaining runs whose lengths are sufficiently large enough to guarantee small error rate. Now we are trying the complete analysis of 10-stage cascades with LFSRs of degree 100. It is expected to be successful with 2^{26} output bits of the cascades.

References

1. Gollmann, D.: Pseudorandom Properties of Cascaded Connections of Clock Controlled Shift Registers, Advances in Cryptology-Eurocrypt'84, Lecture notes in Computer Science **209**, (1985) 93–98
2. Chambers, W. G., Gollmann, D.: Lock in Effect in Cascades of Clock Controlled Shift Register, Advances in Cryptology-Eurocrypt'88, Lecture notes in Computer Science **330**, (1989) 331–343
3. Meier, W., Staffelbach, O.: Correlation Properties of Combiners with Memory in Stream Ciphers, Journal of Cryptology, **5**, No. 1, (1992) 67–86
4. Menicocci, R.: Cryptanalysis of a Two-Stage Gollmann Cascade Generator, Proc. the 3rd Symposium of State and Progress of Research in Cryptography, Rome, (1993) 62–69
5. Menicocci, R.: Short Gollmann Cascade Generator May Be Insecure, presented at Fourth IMA Conf. on Cryptography and Coding, (1993)
6. Beker, H., Piper, F,: Cipher Systems: The Protection of Communications, Wiley-interscience, (1982)