A new framework for keystream generators against Correlation, Linear and Distinguishing Attacks

Ganesh Yellapu
Central Research Laboratory,
Bharat Electronics Limited,
Bangalore, India-560013.
Email: ganeshyellapu@bel.co.in

Abstract

Designing a keystream generator which utilizes Linear Feedback Shift Registers (LFSRs) against correlation, linear attacks is a highly challenging task. In this paper, a new framework for keystream generators is proposed. It is comprised of a set of Linear Feedback Shift Registers (LFSRs), a Multiplicative Congruential Generator (MCG), a vector linear function and, a Boolean function which outputs the keystream. The framework is more generally discussed against correlation attacks, linear attacks and distinguishing (linear) attacks. It is shown that such attacks which are applicable to LFSR based keystream generators are not possible on the proposed framework.

Key words: Correlation attack, combination generators, distinguishing attack, lfsr, linear attack, multiplicative congruential generator.

1 Introduction

A binary additive stream cipher is a synchronous stream cipher in which the keystream, the plaintext and the ciphertext are sequences of binary digits. The output of the keystream generator $z_1, z_2, \ldots$ is xored to the plaintext bit sequence $m_1, m_2, \ldots$ to produce the ciphertext bit sequence $c_1, c_2, \ldots$.

The goal of a stream cipher is to imitate the *one-time pad* [1]. Equivalently, the stream cipher must *efficiently* produce random-looking like sequence that is indistinguishable from a true-random sequence.

A general assumption in cryptanalysis of stream ciphers is Kerckhoff's principle which defines that *the adversary knows everything about the cipher*

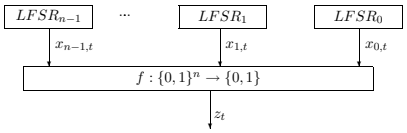

Figure 1: Nonlinear Combination Generator

except the secret information (called key K). From cryptanalysis point of view, a good stream cipher should be resistant against a known-plaintext attack. In a known-plaintext attack, adversary/cryptanalyst is given a plaintext and corresponding ciphertext, and the job is to determine the key K. For a synchronous stream cipher, this is equivalent to the problem of finding the key K that produced the given keystream $z_1, z_2, \ldots, z_n$.

Linear Feedback Shift Registers (LFSR) are the most frequently used building blocks of stream ciphers and the secret key K forms the initial states of LFSRs. Desired properties of an output sequence of an LFSR based keystream generator are large period (to imitate the one-time pad), large linear complexity (to resist Berlekamp-Massey attack [1]) and good statistical properties (to imitate true-random binary sequence). These properties are necessary but not sufficient.

In literature, nonlinear combination generator (simple combiner or combiner without memory) [1] is a framework for LFSR based keystream generators where several maximum-length LFSRs are combined by a nonlinear boolean function. The generator is shown in Fig. 1. The keystream generated by this generator possesses the above three desired properties. But this generator is vulnerable to an important cryptanalytic technique known as correlation attack [1], [2], [3], [4], [5]. A correlation attack may be successful, if there are found linear relations that hold with non-negligible probabilities, between single output bit and a subset of state bits of the LFSR's involved [6]. But a well-known fact for any boolean function is *its output is always correlated to at least one linear function of its inputs.* This helps a cryptanalyst to always find *such* linear relations. Hence the attack always exists against the framework irrespective of the function used.

To overcome this correlation attack, combiners with single bit memory can be used. For such combiner, the output bit is correlated to none of the linear functions of input bits. However, in this case sum of the two successive output bits is shown to be correlated to at least one linear function of the input sequences [7]. Combiner with M-bit memory is shown in Fig. 2. It employs two functions, an output function f and a memory update function δ. For a combiner with M-bit memory, in [7], it is shown that the existence

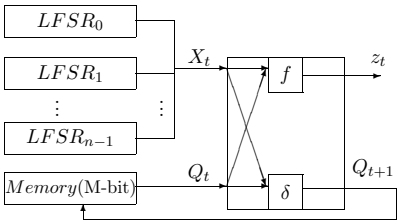

Figure 2: Combiner with M-bit Memory

of a linear function of at most $M+1$ successive outputs that is correlated to a linear function of at most $M+1$ successive inputs. Linear attacks [8] exploit *such* correlations [6], [7], [9]. It shows that for any pair of functions (f,δ), *such* correlations always exist and hence linear attacks are always possible against the framework of combiners with memory.

A distinguishing attack is a cryptanalytic technique in which the adversary tries to determine whether a given sequence is produced by a known cipher or if it appears to be a random sequence. In some cases a distinguishing attack can be used to create a key recovery attack. An overview of distinguishing attacks against stream ciphers, in particular against the non-linear combination generator, can be found in [10], [11], [12].

The above description shows that, using a single & simple boolean function, how difficult it is to prevent or avoid cryptanalytic techniques which exploit correlations. For this reason, complex output functions and/or building blocks are used in the design of keystream generators. Some designers rather choose more than one primitive to build a stream cipher; SSC2 [13] is one such stream cipher. In this paper, following a similar approach, a new model is proposed for keystream generator. The rest of the paper is organized as follows: In section 2, the new framework is presented and described. In Section 3, the model is discussed against correlation attacks, linear attacks and distinguishing (linear) attacks and Section 4 concludes the paper.

2 Proposed Model

The proposed model is shown in Fig. 3. It consists of k LFSRs, a multiplicative congruential Generator (MCG), a vector linear function F and a Boolean function f which outputs the keystream.

A linear congruential generator (LCG) can produce a sequence $\{Q_t\}_{t\geq 0}$

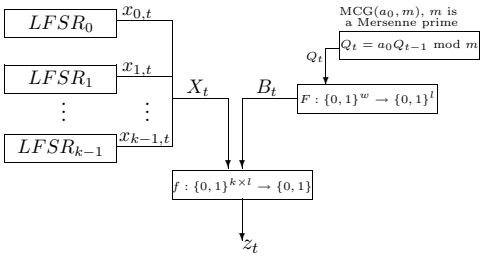

Figure 3: Proposed model for keystream generator

of *uniform random numbers* recursively by

$$Q_{t+1} = a_0 Q_t + c_0 \ (mod\, m) \ \ \forall t \geq 0 \tag{1}$$

where the variables a_0, c_0, m are constants: a_0 is the multiplier, c_0 is the increment, m is the modulus and Q_0 is the initial value or seed. When $c_0 = 0$, the generator is called multiplicative congruential generator (MCG).

Although these congruential generators are efficient and has good statistical properties, they are not cryptographically secure. It is proved that with a sufficiently long run of the pseudo random sequence - one can recover the seed in time polynomial in the bit-size of m and this is also the case even if one outputs only the most significant bits of each Q_t [14].

However, as it is said in [15], use of a linear congruential generator in a cryptographic algorithm does not mean that the algorithm is breakable, since it is possible none of the bits of the random numbers used by the algorithm are ever made public.

Regardless of the various cryptanalytic techniques against these congruential generators, the proposed framework, in addition to the LFSRs, uses a multiplicative congruential generator (MCG) as another primitive by ensuring that *the keystream bits are statistically independent of the output bits of the MCG* so that the keystream bits will never reveal any information about the output bits of MCG.

For any non-zero initial value Q_0 $(< m)$, the output sequence $\{Q_t\}_{t \geq 0}$ of a multiplicative congruential generator attains its maximal period $m - 1$ when m is a prime and a_0 is a primitive root modulo m [16]. In particular, if the modulus m is a Mersenne prime (i.e., $m = 2^w - 1$ is a prime for some integer $w \geq 2$), in one cycle, the periodic sequence $\{Q_t\}_{t=0}^{m-2}$ visits each non-zero w-bit number exactly once (except m). The description of the proposed framework for keystream generators is below.

Let L_i, $P_i(x)$, $x_{i,t}$ denote length, primitive connection polynomial and

the output bit at time t for i^{th} LFSR respectively for all $i \in \{0, 1, \cdots, k-1\}$. For each $t \geq 0$, output bits of all LFSRs are denoted by the vector $X_t = (x_{0,t}, x_{1,t}, \ldots, x_{k-1,t})$. For the MCG, let the modulus m be a w-bit Mersenne prime i.e., $m = 2^w - 1$ is a prime for some integer $w \geq 2$ and the multiplier a_0 be a primitive root modulo m. Then the output sequence $\{Q_t\}_{t \geq 0}$ with non-zero initial value Q_0 obtains full period $m-1$ and each sequence $\{q_{t,j}\}_{t=0}^{m-2}$ for $j \in \{0, 1, \ldots, w-1\}$ is balanced where $Q_t = (q_{t,w-1}, q_{t,w-2}, \ldots, q_{t,0})$ for all $t \geq 0$. If the primitive polynomials for LFSRs, the multiplier & the Mersenne prime for the modulus of the MCG are properly chosen then large period can be obtained for the keystream sequence.

The output Q_t from the MCG is passed to a vector-valued function F which outputs an l-bit vector $B_t = (b_{t,l-1}, b_{t,l-2}, \cdots, b_{t,0})$. Assume all l component functions of F are linear boolean functions on w-variables denoted by $S_0 = (s_{0,w-1}, s_{0,w-2}, \ldots, s_{0,0})$, $S_1 = (s_{1,w-1}, s_{1,w-2}, \ldots, s_{1,0})$, $\ldots$, $S_{l-1} = (s_{l-1,w-1}, s_{l-1,w-2}, \ldots, s_{l-1,0})$[1]. Therefore, each $b_{t,j}$ is a linear combination of $q_{t,w-1}, q_{t,w-2}, \ldots, q_{t,0}$. The keystream bit z_t is obtained by applying the output boolean function f on (X_t, B_t). Also, assume for each $j \in \{0, 1, \ldots, l-1\}$, weight of S_j is at least two (ensures that no bit $q_{t,j}$ is directly passed to the output function instead xor-sum of at least two $q_{t,j}$s are passed) and weights of $S_0, S_1, \ldots, S_{l-1}$ are all distinct (ensures that not only all component functions of F are distinct linear functions but also any non-zero linear combination of $l'(1 \leq l' \leq l)$ component functions of F must involve at least l' distinct variables). Further, it is assumed that $k > l \geq 1$ and the output boolean function $f : \{0,1\}^{k+l} \rightarrow \{0,1\}$ (with $k > l \geq 1$) is balanced and correlation immune of order k. Of course, f must have other cryptographic properties such as good nonlinearity, algebraic degree et al.

In any LFSR based keystream generator, such as nonlinear combination generator, avoiding correlations between keystream bits and the input bits of the output function is a highly challenging task and it is impossible for the nonlinear combination generator. Further, it is a well-known fact that various types of correlation attacks exploit such correlations to recover states of the LFSRs involved. To avoid such correlations between keystream bits and the input bits (which are output from LFSRs) of the output function, in the proposed framework, the keystream bit is carefully defined from the well-chosen output function f. The central idea in defining the keystream bit z_t is that *for each z_t, the output function must involve output bits of all*

[1]Given any n-bit vector $S = (s_{n-1}, s_{n-2}, \ldots, s_0)$, a linear function $l_s : \{0,1\}^n \rightarrow \{0,1\}$ is defined as $l_s(x_{n-1}, \ldots, x_0) = s_{n-1}x_{n-1} \oplus s_{n-2}x_{n-2} \oplus \cdots \oplus s_0 x_0)$ for all $x = (x_{n-1}, x_{n-2}, \ldots, x_0) \in \{0,1\}^n$.

LFSRs and one or more independent binary uniform random variables. For this reason,

1. the output of MCG is filtered using distinct linear boolean functions to produce binary independent uniform random variables[2].[3]

2. the output function f (with $k > l \geq 1$) is assumed to be a resilient function of order k so that any linear approximation (that use single or multiple keystream bits) must involve one or more independent uniform random variables along with the output of at least one LFSR.

The following assumptions are made about the proposed framework:

1. for each $j \in \{0, 1, \ldots, w-1\}$, as the sequence $\{q_{t,j}\}_{t=0}^{m-2}$ is balanced (number of zeros and number of ones are same), $Pr[q_{t,j} = 0] = \frac{1}{2} = Pr[q_{t,j} = 1]$.

2. the k-bit random variables X_{t_1}, X_{t_2}, $\ldots$, X_{t_M} all are independent, where $t_1, t_2, \ldots t_M$ are not necessarily be consecutive.

3. the w-bit random variables Q_{t_1}, Q_{t_2},$\ldots Q_{t_M}$ all are independent, where $t_1, t_2, \ldots t_M$ are not necessarily be consecutive.

4. the l-bit random variables $B_{t_1}, B_{t_2}, \ldots, B_{t_M}$ all are independent, where $B_t = F(Q_t)$ for all t, and $t_1, t_2, \ldots t_M$ are not necessarily be consecutive.

5. for a given $B_t = (b_{t,l-1}, b_{t,l-2}, \ldots, b_{t,0})$, $b_{t,l-1}, b_{t,l-2}, \ldots, b_{t,0}$ all are independent binary uniform random variables because they are output from different linear functions on the same input Q_t.

6. from the 5^{th} assumption, given any l-bit vector $D = (d_{l-1}, \ldots, d_1, d_0)$ and any B_t, from piling-up principle, the variable defined by the dot product $b_t = D \cdot B_t = d_{l-1}b_{l-1,t} \oplus \cdots \oplus d_0 b_{0,t}$ is a binary uniform random variable.

As the output function f is resilient of order k, no linear function of the output bits of LFSRs is correlated to any keystream bit. Further, $k > l$ implies that no linear function of $B_t = (b_{t,l-1}, b_{t,l-2}, \cdots, b_{t,0})$ is also correlated to any keystream bit. Hence, the keystream bit z_t is statistically

[2]Correlation between two distinct linear functions is zero.

[3]The output of MCG is linearly filtered because if it is nonlinearly filtered, linear approximations are possible on the nonlinear filter F of MCG.

independent of both the output bits LFSRs and linearly filtered bits from the output of MCG. It ensures that, despite the amount of keystream bits available, either the LFSRs alone or MCG alone can not be crypt-analyzed.

3 Proposed model against Correlation, Linear and Linear Distinguishing Attacks

In this section, the proposed model is discussed against correlation, linear and distinguishing (linear) attacks. Against LFSR based keystream generators, in the literature standard frameworks are available to perform these attacks and all of them exploit linear relations between one or more keystream bits and the corresponding inputs of the output function (that come from LFSRs) that hold with non-negligible probability. For the proposed generator, it is shown that such linear relations are always true with probability exactly half and it is achieved by the way the keystream bit is defined.

3.1 Correlation Attacks

As the output function f is resilient of order k, the keystream bit z_t can not be linearly approximated by any of at most k inputs of f. In particular, z_t can not be approximated by any linear combination of outputs of LFSRs i.e., for any non-zero $C = (c_{k-1}, \cdots, c_1, c_0) \in \{0,1\}^k$,

$$Pr[c_{k-1}x_{k-1,t} \oplus \cdots \oplus c_0 x_{0,t} = z_t] = \frac{1}{2} \tag{2}$$

and hence the keystream bit is statistically independent of the output bits of all LFSRs. It ensures that the proposed framework avoids any correlation (fast) attack.

As f is resilient of order k, any linear approximation to z_t must involve at least $k+1$ input variables of f. Consider a vector $A = (C, D) \in \{0,1\}^{k+l}$ with hamming weight $u + v > k, u > 0, v > 0$, where $C = (c_{k-1}, \ldots, c_1, c_0)$, $D = (d_{l-1}, \ldots, d_1, d_0)$. Further assume that weight of C is u so that weight of D becomes v. Then,

$$Pr[c_{k-1}x_{k-1,t} \oplus \cdots \oplus c_0 x_{0,t} \oplus d_{l-1}b_{l-1,t} \oplus \cdots \oplus d_0 b_{0,t} = z_t] = \frac{1}{2} + \epsilon, \epsilon > 0 \tag{3}$$

As weight D is v (> 0), in (3), at least one $b_{j,t}$ is always exist.

Let $\sigma_t = c_{k-1}x_{k-1,t} \oplus \cdots \oplus c_0 x_{0,t}$ and $b_t = d_{l-1}b_{l-1,t} \oplus \cdots \oplus d_0 b_{0,t}$. Then σ_t can be generated by using a single LFSR, denote it as LFSR_π,

with the connection polynomial $\pi(x) = \pi_0 + \pi_1 x + \cdots + \pi_{L_\pi} x^{L_\pi}$ of degree $L_\pi = \sum_{i=0}^{k-1} c_i L_i$ which is the lcm of the connection polynomials of LFSRs for which $c_i = 1$. Hence[4],

$$Pr[\sigma_t \oplus b_t = z_t] = \frac{1}{2} + \epsilon, \epsilon > 0 \tag{4}$$

As $B_t, B_{t+1}, B_{t+2}, \ldots$ are independent, $b_t, b_{t+1}, b_{t+2}, \ldots$ are independent uniform random variables. Hence given any $t_1, t_2, \ldots t_N$, need not be consecutive, the set of linear approximations

$$\begin{aligned} Pr[\sigma_{t_1} \oplus b_{t_1} = z_{t_1} \quad] &= \tfrac{1}{2} + \epsilon, \\ Pr[\sigma_{t_2} \oplus b_{t_2} = z_{t_2} \quad] &= \tfrac{1}{2} + \epsilon, \\ &\vdots \\ Pr[\sigma_{t_N} \oplus b_{t_N} = z_{t_N} \quad] &= \tfrac{1}{2} + \epsilon, \end{aligned} \tag{5}$$

must involve N independent uniform random variables viz., $b_{t_1}, b_{t_2}, \ldots, b_{t_N}$.

A cryptanalyst may try to exploit (4) to recover a state of the LFSR_π by performing a correlation attack. However, as $b_{t_1}, b_{t_2}, \ldots, b_{t_N}$ are independent uniform random variables, the framework defeats any correlation (fast) attack (because he must decode the bits $b_{t_1}, b_{t_2}, \ldots, b_{t_N}$).

3.2 Linear Attacks

In linear attacks, a cryptanalyst tries to find a linear function of some successive keystream bits that is correlated to a linear function of some successive (linear) inputs of the function f. However, for the proposed generator, given any M keystream bits $z_{t_1}, z_{t_2}, \ldots, z_{t_M}$ (not necessarily be consecutive) and corresponding inputs $X_{t_1}, X_{t_2}, \ldots X_{t_M}$, it is shown that xor-sum of the keystream bits is not correlated to the xor-sum of corresponding inputs.

As f is resilient of order k, for any non-zero $C_t = (c_{t,k-1}, c_{t,k-2}, \ldots, c_{t,0}) \in \{0,1\}^k$, $Pr[C_t \cdot X_t = z_t] = \frac{1}{2}$. Define a random variable $Y_t = C_t \cdot X_t \oplus z_t$. Then Y_t is a uniform random variable. As it is assumed that the variables $X_{t_1}, X_{t_2}, \ldots, X_{t_M}$ are independent (and hence $Y_{t_1}, Y_{t_2}, \ldots, Y_{t_M}$ are also), from piling-up principle,

$$Pr[Y_{t_1} \oplus Y_{t_2} \oplus \cdots \oplus Y_{t_M} = 0] = \frac{1}{2} \tag{6}$$

[4]In correlation (fast) attacks, in general, the term b_t does not exist in (4)

$$i.e., Pr[C_{t_1} \cdot X_{t_1} \oplus C_{t_2} \cdot X_{t_2} \oplus \cdots \oplus C_{t_M} \cdot X_{t_M} = z_{t_1} \oplus z_{t_2} \oplus \cdots \oplus z_{t_M}] = \frac{1}{2} \quad (7)$$

It shows that the proposed framework avoids linear attacks which exploit linear relations between keystream bits and corresponding linear inputs of the output function f which are true with probability different from half.

3.3 Linear Distinguishing Attacks

In linear distinguishing attacks against LFSR based generators, a cryptanalyst tries to find a linear approximation to the nonlinear output function (which is always possible for any nonlinear boolean function). He/she also tries to find a linear combination of the linear process that vanishes (one such linear relation always exists for any LFSR sequence). Finally, the cryptanalyst applies the linear combination to the ciphers output, and tries to find traces of the distinguishing property. In particular, a cryptanalyst looks for some linear combination of keystream bits which vanishes with non-negligible probability (and in this process, the LFSR's connection polynomial helps the cryptanalyst in obtaining such a linear combination). Nevertheless for the proposed framework, it is shown that *any such* linear combination of keystream bits vanishes with probability exactly half and hence the framework avoids standard linear distinguishing attacks available for LFSR based generators.

Consider a linear approximation to the output function f defined by the vector $A = (C, D) \in \{0,1\}^{k+l}$ with hamming weight $u + v > k, u > 0, v > 0$ and weight of $C = (c_{k-1}, \cdots, c_1, c_0)$ is u and weight of $D = (d_{l-1}, \cdots, d_1, d_0)$ is v. Then, the output sequence $\{\sigma_t\}$ of the LFSR, LFSR_π, with connection polynomial $\pi(x) = \pi_0 + \pi_1 x + \cdots + \pi_{L_\pi} x^{L_\pi}$ satisfies the recurrence relation

$$\bigoplus_{j=0}^{L_\pi} \pi_j \sigma_{t-j} = 0 \ \ \forall t \geq L_\pi \quad (8)$$

and from (4), $\sigma_t \oplus b_t \oplus e_t = z_t$ where e_t is a random (noise) variable with $Pr[e_t = 0] \neq \frac{1}{2} \neq Pr[e_t = 1]$. Therefore,

$$\bigoplus_{j=0}^{L_\pi} \pi_j z_{t-j} = \bigoplus_{j=0}^{L_\pi} \pi_j b_{t-j} \oplus \bigoplus_{j=0}^{L_\pi} \pi_j e_{t-j} \quad (9)$$

Let the noise variables e_j be independent and $Y_1 = \bigoplus_{j=0}^{L_\pi} \pi_j b_{t-j}$, $Y_2 = \bigoplus_{j=0}^{L_\pi} \pi_j e_{t-j}$. It is apparent that Y_1, Y_2 are independent and $Pr[Y_1 = 0] = \frac{1}{2}$

(because b_js are independent), $Pr[Y_2 = 0] \neq \frac{1}{2}$. Hence, from piling-up principle, $Pr[\bigoplus_{j=0}^{L_\pi} \pi_j z_{t-j} = 0] = \frac{1}{2}$. This is true not only for the polynomial $\pi(x)$ but also for any connection polynomial of the LFSR that generates the sequence $\{\sigma_t\}$. Hence, any linear combination of keystream bits that is defined by the LFSR's (involved in the attack) connection polynomial vanishes with probability exactly half [5] and therefore the framework avoids any linear distinguishing attack that exist for LFSR based keystream generators .

Finally, as the keystream bit z_t is statistically independent of $B_t = (b_{t,l-1}, b_{t,l-2}, \cdots, b_{t,0})$, it will not reveal any information about $b_{t,j}$s. In particular, z_t will never reveal any information about $Q_t = (q_{t,w-1}, q_{t,w-2}, \cdots, q_{t,0})$. Similarly, as above, it can be shown that

$$Pr[D_{t_1} \cdot B_{t_1} \oplus D_{t_2} \cdot B_{t_2} \oplus \cdots \oplus D_{t_M} \cdot B_{t_M} = z_{t_1} \oplus z_{t_2} \oplus \cdots \oplus z_{t_M}] = \frac{1}{2} \quad (10)$$

where $D_{t_1}, D_{t_2}, \ldots, D_{t_M}$ are non-zero l-bit vectors and $t_1, t_2, \ldots, t_M$ are need not be consecutive.

As the equations (4), (7),(9) are not exploitable to utilize the existing frameworks of correlation, linear and linear distinguishing attacks, it is concluded that the proposed framework avoids all of these attacks.

4 Conclusion

A new framework for keystream generators is proposed. In addition to the conventional LFSRs, the framework also uses another primitive viz., Multiplicative Congruential Generator. The framework is described and the keystream bit is carefully defined from the output function which is resilient of order k (where k is the number of LFSRs in the framework) so that any linear relation between a single keystream bit and outputs of one or more LFSRs which holds with non-negligible probability must involve one or more independent uniform random variables. It makes the proposed generator to avoid any correlation (fast) attack which tries to recover states of the LFSRs involved in the attack.

Against linear attacks, it is shown that the xor-sum of the keystream bits (not necessarily be consecutive) is not correlated to the xor-sum of the corresponding linear inputs of the output function. Against linear distinguishing

[5] In linear distinguishing attacks, in general, the term $\bigoplus_{j=0}^{L_\pi} \pi_j b_{t-j}$ does not exist in (9).

attacks, it is shown that any xor-combination of the keystream bits that is defined by the LFSR involved in the distinguishing attack vanishes with probability exactly half. As a result, the standard frameworks of both linear and linear distinguishing attacks for LFSR based keystream generators are not applicable to the proposed.

It is also shown that the xor-sum of the keystream bits is not correlated to the xor-sum of the corresponding inputs (outputs from MCG) of the output function.

Despite the amount of keystream bits available, the proposed framework ensures that the keystream bit is statistically independent of the output bits of all LFSRs and the output of MCG so that crypt-analyzing either LFSRs alone or MCG alone is not possible.

References

[1] A. J. Menezes, S. A. Vanstone, and P. C. V. Oorschot, *Handbook of Applied Cryptography*. CRC Press, 1996.

[2] T. Johansson and F. Jönsson, "Improved fast correlation attack on stream ciphers via convolutional codes," in *Advances in Cryptology - EUROCRYPT'99*, ser. Lecture Notes in Computer Science, J. Stern, Ed. Springer-Verlag, 1999, vol. 1592, pp. 347–362.

[3] W. Meier and O. Staffelbach, "Fast correlation attacks on stream ciphers," in *Advances in Cryptology-EUROCRYPT'88*, ser. Lecture Notes in Computer Science, C. G. Günther, Ed. Springer-Verlag, 1988, vol. 330, pp. 301–314.

[4] V. V. Chepyzhov, T. Johansson, and B. J. M. Smeets, "A simple algorithm for fast correlation attacks on stream ciphers," in *FSE*, 2000, pp. 181–195.

[5] T. Johansson and F. Jönsson, "Theoretical analysis of a correlation attack based on convolutional codes," *IEEE Transactions on Information Theory*, vol. 48, pp. 2173–2181, Aug 2002.

[6] W. Meier, "Fast correlation attacks: Methods and countermeasures," in *Fast Software Encryption - 18th International Workshop, FSE 2011, Lyngby, Denmark, February 13-16, 2011, Revised Selected Papers*, 2011, pp. 55–67. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-21702-9_4

[7] J. D. Golic, "Correlation via linear sequential circuit approximation of combiners with memory," in *Proceedings of the 11th annual international conference on Theory and application of cryptographic techniques*, ser. EUROCRYPT'92. Berlin, Heidelberg: Springer-Verlag, 1993, pp. 113–123. [Online]. Available: http://dl.acm.org/citation.cfm?id=1754948.1754962

[8] J. Golic, "Linear cryptanalysis of stream ciphers," in *Fast Software Encryption*, ser. Lecture Notes in Computer Science, B. Preneel, Ed. Springer Berlin Heidelberg, 1995, vol. 1008, pp. 154–169. [Online]. Available: http://dx.doi.org/10.1007/3-540-60590-8_13

[9] F. Armknecht, M. Krause, and D. Stegemann, "Design principles for combiners with memory," in *Proceedings of the 6th international conference on Cryptology in India*, ser. INDOCRYPT'05. Berlin, Heidelberg: Springer-Verlag, 2005, pp. 104–117. [Online]. Available: http://dx.doi.org/10.1007/11596219_9

[10] M. Hell, T. Johansson, and L. Brynielsson, "An overview of distinguishing attacks on stream ciphers," *Cryptography and Communications*, vol. 1, no. 1, pp. 71–94, 2009. [Online]. Available: http://dx.doi.org/10.1007/s12095-008-0006-7

[11] D. Coppersmith, S. Halevi, and C. Jutla, "Cryptanalysis of stream ciphers with linear masking," Cryptology ePrint Archive, Report 2002/020, 2002, http://eprint.iacr.org/.

[12] H. Englund, "Some results on distinguishing attacks on stream ciphers," Ph.D. dissertation, Lund University, December 2007.

[13] M. Zhang, C. Carroll, and A. Chan, "The software-oriented stream cipher ssc2," in *Fast Software Encryption*, ser. Lecture Notes in Computer Science, G. Goos, J. Hartmanis, J. van Leeuwen, and B. Schneier, Eds. Springer Berlin Heidelberg, 2001, vol. 1978, pp. 31–48. [Online]. Available: http://dx.doi.org/10.1007/3-540-44706-7_3

[14] A. Bauer, D. Vergnaud, and J.-C. Zapalowicz, "Inferring sequences produced by nonlinear pseudorandom number generators using coppersmiths methods," in *Public Key Cryptography PKC 2012*, ser. Lecture Notes in Computer Science, M. Fischlin, J. Buchmann, and M. Manulis, Eds. Springer Berlin Heidelberg, 2012, vol. 7293, pp. 609–626. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-30057-8_36

[15] M. Bellare, S. Goldwasser, and D. Micciancio, "pseudo-random number generation within cryptographic algorithms: The dds case," in *Advances in Cryptology CRYPTO '97*, ser. Lecture Notes in Computer Science, J. Kaliski, BurtonS., Ed. Springer Berlin Heidelberg, 1997, vol. 1294, pp. 277–291. [Online]. Available: http://dx.doi.org/10.1007/BFb0052242

[16] D. E. Knuth, *The Art of Computer Programming Volume 2: Seminumerical Algorithms*, 2nd ed. Addison-Wesley, 1981.

[17] G. Banegas, "Attacks in stream ciphers: A survey," Cryptology ePrint Archive, Report 2014/677, 2014, http://eprint.iacr.org/.

[18] C. Bruwer, "Correlation attacks on stream ciphers using convolutional codes," Ph.D. dissertation, University of Pretoria, October 2004.

[19] A. Canteaut and M. Trabbi, "Improved fast correlation attacks using parity-check equations of weight 4 and 5," in *Eurocrypt 2000*, ser. Lecture Notes in Computer Science, B. Preneel, Ed. Springer-Verlag, 2000, vol. 1807, pp. 573–588.

[20] K. N. Srinivasan, S. Boztas, and A. Rao, "Improving correlation attacks on stream ciphers," in *Proceedings of the Information Security & Cryptology Conference*, Ankara, Turkey, December 2007, pp. 306–311.

[21] A. Braeken, "Cryptographic properties of boolean functions and s-boxes," Ph.D. dissertation, Katholieke Universiteit Leuven, 2006.