Linear Sequential Circuit Approximation of Grain and Trivium Stream Ciphers[*]

Shahram Khazaei[†‡] Mahdi M. Hasanzadeh[‡] Mohammad S. Kiaei[†]

[†] Electrical Engineering Department, Sharif University of Technology, Tehran, Iran
[‡] Zaeim Electronic Industries Company, P.O. BOX 14155-1434, Tehran, Iran
{khazaei, hasanzadeh}@zaeim.com, kiaeim@alum.sharif.edu

Abstract. Grain and Trivium are two hardware oriented synchronous stream ciphers proposed as the simplest candidates to the ECRYPT Stream Cipher Project, both dealing with 80-bit secret keys. In this paper we apply the linear sequential circuit approximation method to evaluate the strength of these stream ciphers against distinguishing attack. In this approximation method which was initially introduced by Golic in 1994, linear models are effectively determined for autonomous finite-state machines. We derive linear functions of consecutive key-stream bits which are held with correlation coefficient of about $2^{-63.7}$ and 2^{-126} for Grain and Trivium ciphers, respectively. Then using the concept of so-called generating function, we turn them into linear functions with correlation coefficient of 2^{-29} for Grain and 2^{-72} for Trivium. It shows that the Grain output sequence can be distinguished from a purely random sequence, using about 2^{58} bits of the output sequence with the same time complexity. However, our attempt fails to find a successful distinguisher for Trivium.

Keywords. Stream Cipher, Distinguishing Attack, Linear Sequential Circuit Approximation, Grain, Trivium, ECRYPT, Security Evaluation.

1 Introduction

Stream ciphers are widely used for fast encryption of sensitive data. Lots of old stream ciphers that have been formerly used can no longer be considered secure, because of their vulnerability to newly developed cryptanalysis techniques. In particular, the NESSIE project [9] did not select any of the proposed stream ciphers for its portfolio, as it was felt that none of the submissions was sufficiently strong. In order to create a portfolio of secure stream ciphers, the ECRYPT project [3] made a call for designs of new stream ciphers which led to submission of 34 proposals to the project by April 2005. Grain [6] and Trivium [1] are two of these proposals which were designed for hardware applications (constrained environments) and are structurally simpler than other ones. Both of them use 80-bit keys and public IV's.

One of the generic attacks on stream ciphers is distinguishing attack whose aim is to distinguish the output sequence of a given stream cipher from a purely random sequence, with small error probability, faster than exhaustive search of the key space. In this paper, we use the *linear sequential circuit approximation method* to evaluate the

[*] This paper is a combination of [11] and [12].

strength of these two stream ciphers against distinguishing attack. This approximation method was firstly introduced in [4, 5] as an effective method for the linear model determination based on linear sequential circuit approximation of autonomous finite-state machines.

Key-stream generators for stream cipher applications can generally be realized as autonomous finite-state machines whose initial state and possibly the structure depend on a secret key. Regarding this issue and utilizing the linear sequential circuit approximation method, we first derive a linear function of consecutive output bits for each of Grain and Trivium stream ciphers. These functions are held with correlation coefficient of about $2^{-63.7}$ and 2^{-126} for Grain and Trivium, respectively. Then using the generating function concept, we turn them into linear functions with correlation coefficient of about 2^{-29} for Grain and 2^{-72} for Trivium.

For Grain, a chi-square test could be applied to distinguish its output sequence from a purely random sequence. The required time and data complexity is $O(2^{58})$ for detecting this bias. A preprocessing phase for computing a trinomial multiple of a certain primitive polynomial with degree 80 is needed which can be performed using time and memory complexities of $O(2^{40})$. A key-recovery attack which requires 2^{43} computations and 2^{38} key-stream bits has also been mounted on Grain in [10].

However, for Trivium with the correlation coefficient of 2^{-72}, the time complexity for distinguishing its output sequence form a purely random sequence is $O(2^{144})$. It seems impossible to find a linear function of consecutive output bits with correlation coefficient greater than 2^{-40} to provide a successful distinguishing attack. A similar result has been mentioned in the Trivium specification [1] but not explained in details. However, the Trivium designers derived it in a slightly different way in [2] which was published after this work had been done. We decided to bring our results in this paper because of the straightforward and systematic application of linear sequential circuit approximation to both Grain and Trivium.

The paper is organized as follows. In Sections 2 a brief description of Grain and Trivium stream ciphers is given. The linear sequential circuit approximation method is shortly described in Section 3 and the results of applying this method to Grain and Trivium stream ciphers are presented in Sections 4 and 5 respectively. The paper is concluded in Section 6.

2 Outline of the Analyzed Ciphers

In this section we present a brief description of the key generator algorithms of Grain and Trivium which we are going to evaluate. We ignore their key set up process because the attack is independent of them.

2.1 Description of Grain

Grain [6] is a very simple hardware oriented synchronous stream cipher proposed as a candidate to the ECRYPT Stream Cipher Project [3]. Grain consists of an LFSR and an NFSR of length 80 and generates its key-stream from an 80-bit secret key and a 64-bit initial value (IV). The proposed design uses an 11-input Boolean function g as the feedback function of the NFSR, and a 5-input Boolean function h to filter the contents of five fixed cells of LFSR and NFSR. The output of the feedback function is masked with the output bit of the LFSR to update the NFSR and the output of the filter function is masked with the output bit from the NFSR to produce the key-stream z_t. The initial state of LFSR and NFSR denoted by $(s_0, s_1, \ldots, s_{79})$ and $(b_0, b_1, \ldots, b_{79})$ are determined through a certain key-IV setup procedure. A complete description of the cipher can be given by the following pseudo-code for producing N bits of the key-stream:

```
for $t = 1$ to $N$ do
    $t_s \leftarrow s_0 + s_{13} + s_{23} + s_{38} + s_{51} + s_{62}$
    $t_b \leftarrow s_0 + g(b_{63}, b_{60}, b_{52}, b_{45}, b_{37}, b_{33}, b_{28}, b_{21}, b_{15}, b_9, b_0)$
    $z_t \leftarrow b_0 + h(b_{63}, s_{64}, s_{46}, s_{25}, s_3)$
    $(s_0, s_1, \ldots, s_{79}) \leftarrow (s_1, s_2, \ldots, s_{79}, t_s)$
    $(b_0, b_1, \ldots, b_{79}) \leftarrow (b_1, b_2, \ldots, b_{79}, t_b)$
end for.
```

The g and h functions are as follows:

$$\begin{aligned} h(x_4, \ldots, x_0) = {} & x_1 + x_4 + x_0x_3 + x_2x_3 + x_3x_4 + x_0x_1x_2 \\ & + x_0x_2x_3 + x_0x_2x_4 + x_1x_2x_4 + x_2x_3x_4 \end{aligned} \tag{1}$$

$$\begin{aligned} g(x_{10}, \ldots, x_0) = {} & x_{10} + x_9 + x_8 + x_7 + x_6 + x_5 + x_4 + x_3 + x_2 + x_1 + x_0 + x_{10}x_9 \\ & + x_6x_5 + x_2x_1 + x_9x_8x_7 + x_5x_4x_3 + x_{10}x_7x_4x_1 + x_9x_8x_6x_5 \\ & + x_{10}x_9x_3x_2 + x_{10}x_9x_8x_7x_6 + x_5x_4x_3x_2x_1 + x_8x_7x_6x_5x_4x_3. \end{aligned} \tag{2}$$

The feedback polynomial of the LFSR is primitive and given by $1 + x^{18} + x^{29} + x^{42} + x^{57} + x^{67} + x^{80}$, in accordance with the first line of the pseudo-code and ensures that the period of the output sequence is at least 2^{80}-1. A graphical representation of the key-stream generation process of Grain can be found in Fig. 1.

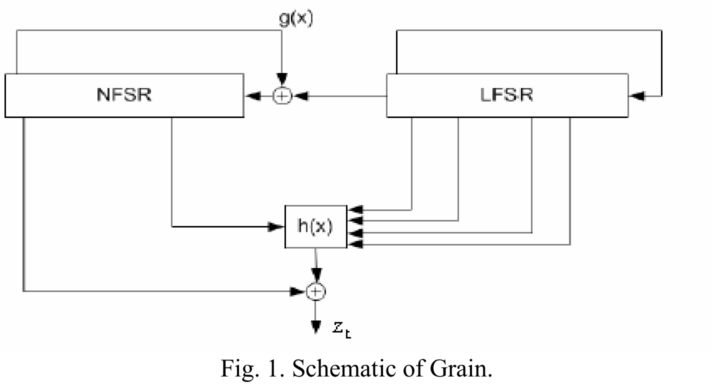

Fig. 1. Schematic of Grain.

2.2 Description of Trivium

Trivium [1] is another simple hardware oriented synchronous stream cipher proposed as a candidate to the ECRYPT Stream Cipher Project [3]. Trivium generates up to 2^{64} bits of key-stream from an 80-bit secret key and an 80-bit initial value (IV). The proposed design contains a 288-bit internal state denoted by $(s_1,\ldots, s_{288})$. The key-stream generation consists of an iterative process which extracts the values of 15 specific state bits and uses them both to update 3 bits of the state and to compute 1 bit of key-stream z_t. The state bits are then rotated and the process repeats itself until the requested $N \le 2^{64}$ bits of key-stream have been generated. A complete description is given by the following pseudo-code:

```
for $t = 1$ to $N$ do
    $t_1 \leftarrow s_{66} + s_{93}$
    $t_2 \leftarrow s_{162} + s_{177}$
    $t_3 \leftarrow s_{243} + s_{288}$
    $z_t \leftarrow t_1 + t_2 + t_3$
    $t_1 \leftarrow t_1 + s_{91} \,.\, s_{92} + s_{171}$
    $t_2 \leftarrow t_2 + s_{175} \,.\, s_{176} + s_{264}$
    $t_3 \leftarrow t_3 + s_{286} \,.\, s_{287} + s_{69}$
    $(s_1, s_2,\ldots, s_{93}) \leftarrow (t_3, s_1,\ldots, s_{92})$
    $(s_{94}, s_{95}, \ldots, s_{177}) \leftarrow (t_1, s_{94}, \ldots, s_{176})$
    $(s_{178}, s_{179}, \ldots, s_{288}) \leftarrow (t_2, \mathrm{s}_{178}, \ldots, s_{287})$
end for.
```

The most negative point about Trivium is the period of its output sequence which is not well conceived. A graphical representation of the key-stream generation process of Trivium is given in Fig. 2.

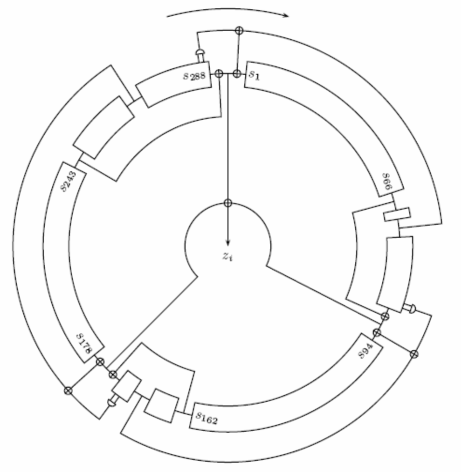

Fig. 2. Schematic of Trivium

3 Introduction to the Linear Sequential Circuit Approximation

Golic [4, 5] has shown that for a binary key-stream generator with M bits of memory whose initial state is chosen uniformly at random, there exists a linear function of at most $M + 1$ consecutive output bits which is an unbalanced function of the initial state variables. He also developed an effective method for the linear model determination based on linear sequential circuit approximation of autonomous finite-state machines. The linear function of consecutive output bits produces an unbalanced sequence to which one can apply the standard chi-square frequency statistical test. The test is successful if the length of the sequence is chosen to be inversely proportional to the square of the correlation coefficient. The *correlation coefficient*- also sometimes called *bias*- of the random variable $\boldsymbol{x}$ is defined as $\varepsilon = 1 - 2\Pr\{\boldsymbol{x} = 1\}$.

Key-stream generators for stream cipher applications can generally be realized as autonomous finite-state machines whose initial state and possibly the structure depend on a secret key. A binary autonomous finite-state machine is defined by

$$S_t = F(S_{t-1}), \quad t \geq 1 \tag{3}$$

$$z_t = f(S_t), \quad t \geq 1 \tag{4}$$

where F: GF(2)^M → GF(2)^M is the next-state vector Boolean function, f: GF(2)^M → GF(2) is the output Boolean function, $S_t = (s_{t,1}, s_{t,2}, \ldots, s_{t,M})^T$ is the state vector at time t, $S_0 = (s_{0,1}, s_{0,2}, \ldots, s_{0,M})^T$ is the initial state, and $\{z_t\}$ is the output key-stream sequence (the superscript ^{T} denotes the matrix transposition operation).

It can be shown that for the general finite-state-machine defined by (3) and (4) there exists a linear function of at most $M + 1$ consecutive output bits $L(z_t, z_{t+1}, \cdots, z_{t+M})$ which is an unbalanced function of the initial state variables [4, 5]. Moreover, its probability distribution is independent of time t if the next state function is balanced. This statement has been proposed as a Theorem in [4, 5], which is mentioned in the following.

Theorem 1. Let the next-state function of a binary autonomous finite state machine with M bits of memory be balanced. Then there exists a linear function L of at most $M + 1$ consecutive output bits $L(z_t, z_{t+1}, \cdots, z_{t+M})$ which is an unbalanced function of the initial state variables for each $t \geq 1$. Moreover, the correlation coefficient of $L(z_t, z_{t+1}, \cdots, z_{t+M})$ is the same for each t.

The linear function L of consecutive output bits produces an unbalanced sequence to which one can apply the standard chi-square frequency statistical test to make a distinguishing attack. If the correlation coefficient of L is equal to ε, we need approximately $1/\varepsilon^2$ bits of the output sequence to detect this bias [2]. If the key length is k, the distinguishing attack is effective if $\varepsilon > 2^{-k/2}$.

[2] This amount of of the output sequence does not provide reasonably negligible error probability for the distinguisher. The better choice would be $10/\varepsilon^2$ whose error probability is less than 10^{-3}. However, for convenience of dealing with powers of 2, we discard the coefficient 10.

Under the condition that the key merely controls the initial state, and therefore, next state function and output function are known, an efficient procedure has also been developed in [4,5] for finding unbalanced linear functions of the output which is based on the linear sequential circuit approximation approach. In this procedure, the output Boolean function and each of the Boolean functions in the next-state function of the key-stream generator are first decomposed into the sum of linear functions and an unbalanced Boolean function. Then, by virtue of the obtained linear approximations, the basic equations (3) and (4) are put into the following form

$$S_t = AS_{t-1} + \Delta(S_{t-1})\,,\ t \geq 1 \tag{5}$$

$$z_t = BS_t + \gamma(S_t)\,,\ t \geq 1 \tag{6}$$

where S_t is considered as an M-bit binary column vector, A and B are respectively $M{\times}M$ binary matrix and M-bit binary row vector, and γ and all components of $\Delta = (\delta_1, \cdots, \delta_M)^T$ are unbalanced Boolean functions called the noise functions.

Finally, considering the sequences $\{\gamma(S_t)\}$ and $\{\delta_j(S_{t-1})\}$, $1 \leq j \leq M$, as the input sequences to (5) and (6) it is shown that

$$\sum_{i=0}^{m} \varphi_i z_{t+i} = \sum_{i=0}^{m} \varphi_i \gamma(S_{t+i}) + \sum_{j=1}^{M} \sum_{i=0}^{m} c_{i,j} \delta_j (S_{t+i-1})\,, \tag{7}$$

where $\varphi(x) = \sum_{i=0}^{m} \varphi_i x^i$ ($m \leq M$) is the minimal polynomial of A and $c_{i,j}$ ($0 \leq i \leq m$, $1 \leq j \leq M$) is the j^{th} element of the M-bit row vector $\sum_{k=0}^{m-i} \varphi_{k+i} BA^k$.

Equation (7) is an unbalanced linear function of at most $M + 1$ consecutive output bits which is expressed as the sum of unbalanced functions of the initial state variables [4, 5]. In general, the sum of unbalanced Boolean functions can be balanced. However, it has been proved that if the functions are picked independently at random, then with high probability their sum is unbalanced with the correlation coefficient very close to the product of the individual correlation coefficients [4, 5]. We refer to (7) as *basic linear sequential circuit approximation* of autonomous finite-state machine defined by (3) and (4) corresponding to decompositions A and B.

Every linear function of a given sequence can be defined as a polynomial in the generating function domain. Let $\{a_t\}$ be an arbitrary binary sequence, and $\{b_t\}$ a linear function of $\{a_t\}$ defined by $b_t = \sum_{k=0}^{r} p_k a_{t+k}$. In generating function domain, the linear function $b_t = \sum_{k=0}^{r} p_k a_{t+k}$ is denoted by $b_t = p(D) a_t$ where $p(D) = \sum_{k=0}^{r} p_k D^k$. Moreover, in this domain, the relation (7) can be rewritten in the following way

$$\varphi(D)z_t = \varphi(D)\gamma(S_t) + \sum_{j=1}^{M} c_j(D)\delta_j(S_{t-1}), \qquad \textbf{(8)}$$

where $c_j(x) = \sum_{i=0}^{m} c_{i,j}x^i$.

For both Grain and Trivium stream ciphers, the next state function and the output function are independent of the secret key. Also, the balance condition of next state function is well satisfied for these two ciphers since their next sate functions are invertible. Thus, their linear sequential models can be investigated.

4 Linear Sequential Circuit Approximation of Grain

In this section, we derive the linear sequential circuit approximation of Grain stream cipher and show that this cipher is vulnerable to distinguishing attack.

4.1 Basic Linear Sequential Circuit Approximation

For Grain stream cipher we have M = 160. Let S_t be a 160-bit binary column vector which contains the state of LFSR and NFSR of Grain at time t, that is $(s_0, s_1,\ldots, s_{79}, b_0, b_1,\ldots, b_{79})^T$ in the pseudo-code introduced in Section 2.1. The function g is the only nonlinear part of the next-state function. The filter function h is also nonlinear. We utilize the linear approximation $L_{g,w}(x_{10},\cdots,x_0) = w_{10}x_{10} + \cdots + w_0x_0$ for the feedback function g and linear function $L_{h,v}(x_4,\cdots,x_0) = v_4x_4 + \cdots + v_0x_0$ for the filter function h. Using these decompositions of g and h functions, the linear approximations (5) and (6) for Grain can be written as follows

$$S_t = AS_{t\text{-}1} + H\delta_t,\ t \ge 1 \qquad \textbf{(9)}$$

$$z_t = BS_t + \gamma_t,\ t \ge 1. \qquad \textbf{(10)}$$

Here $H = [h_i]$ is a 160-bit binary column vector with all entries equal to zero except h_{160}, $\delta_t = \delta_{81}(S_{t-1})$ and $\gamma_t = \gamma(S_t)$ are respectively the scalar noise terms corresponding to the linear approximation $L_{g,w}$ of g and $L_{h,v}$ of h; and A and B are as follows

$$B = e_{80} + v_4e_{143} + v_3e_{64} + v_2e_{46} + v_1e_{25} + v_0e_3\ , \qquad (11)$$

$$A = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_{79} \\ e_0 + e_{13} + e_{23} + e_{38} + e_{51} + e_{62} \\ e_{81} \\ e_{82} \\ \vdots \\ e_{159} \\ w_g \end{bmatrix} \qquad \textbf{(12)}$$

where e_i ($0 \le i \le 159$) denotes the $(i+1)^{\text{th}}$ row of the 160×160 identity matrix and

$$\begin{aligned} w_g &= e_0 + w_{10}e_{143} + w_9 e_{140} + w_8 e_{132} + w_7 e_{125} + w_6 e_{117} \\ &\quad + w_5 e_{113} + w_4 e_{108} + w_3 e_{101} + w_2 e_{95} + w_1 e_{89} + w_0 e_{80}. \end{aligned} \qquad \textbf{(13)}$$

Using the general relation (7), the basic linear sequential circuit approximation of Grain corresponding to the decompositions A and B can be expressed by

$$u_t \overset{\Delta}{=} \sum_{i=0}^{m} \varphi_i z_{t+i} = \sum_{i=0}^{m} \varphi_i \gamma_{t+i} + \sum_{i=0}^{m} c_i \delta_{t+i} \,, \qquad \textbf{(14)}$$

or equivalently in the generating function domain by

$$u_t = \varphi(D) z_t = \varphi(D)\gamma_t + c(D)\delta_t \,, \qquad \textbf{(15)}$$

where $\varphi(x) = \sum_{i=0}^{m} \varphi_i x^i$ is the minimal polynomial of A and $c(x) = \sum_{i=0}^{m} c_i x^i$ whose coefficients are defined by

$$c_i \overset{\Delta}{=} \sum_{k=0}^{m-i} \varphi_{k+i} B A^k H \cdot \qquad \textbf{(16)}$$

Note that the coefficients φ_i ($0 \le i \le m$) just depend on the coefficients w_i ($0 \le i \le 10$) but the coefficients c_i ($0 \le i \le m$) depend on both the coefficients w_i ($0 \le i \le 10$) and v_j ($0 \le j \le 4$).

4.2 Correlation Coefficient Analysis

As it was explained in Section 3, the relation (15) produces an unbalanced sequence $u_t = \sum_{k=0}^{m} \varphi_k z_{t+k}$ if the errors of both linear approximations $L_{g,w}$ and $L_{h,v}$ of g and h have non-zero correlation coefficients.

The weight of a given polynomial $k(x)$, denoted by $hw(k)$, is defined as the number of its non-zero coefficients. Let $\varepsilon_{g,w}$ and $\varepsilon_{h,v}$ denote the correlation coefficients of δ_t and γ_t - the noise terms corresponding to the linear approximation $L_{g,w}$ of g and $L_{h,v}$ of h. Under the independence assumption of the noise terms in (15), the correlation coefficient of u_t denoted by $\varepsilon_{w,v}$ is equal to $\varepsilon_{w,v} = \varepsilon_{h,v}^{hw(\varphi)} \varepsilon_{g,w}^{hw(c)}$.

We carried out exhaustive search over all of the $2^{11} \times 2^5$ possible choices for w and v to find the one with the greatest correlation coefficient which resulted to the following choice for w and v,

$$w = [w_{10} \quad \cdots \quad w_0] = [0 \quad \cdots \quad 0 \quad 1] \tag{17}$$

$$v = [v_4 \quad \cdots \quad v_0] = [0 \quad 1 \quad 0 \quad 1 \quad 0] \tag{18}$$

in accordance with the linear approximations $L_{g,w}(x_{10}, \ldots, x_0) = x_0$ and $L_{h,v}(x_4, \ldots, x_0) = x_3 + x_1$ for g and h respectively. The correlation coefficient of noise terms corresponding to these linear approximations are $\varepsilon_{g,w} = 5/256$ and $\varepsilon_{h,v} = 1/4$. The corresponding $\varphi(x)$ and $c(x)$ are as follows

$$\varphi(x) = 1 + x^{13} + x^{23} + x^{38} + x^{51} + x^{62} + x^{93} + x^{103} + x^{118} + x^{131} + x^{142} + x^{160} \tag{19}$$

$$c(x) = x + x^{14} + x^{24} + x^{39} + x^{52} + x^{63} + x^{81}. \tag{20}$$

Since $hw(\varphi) = 12$ and $hw(c) = 7$, the corresponding correlation coefficient of u_t is equal to $\varepsilon_{w,v} = (1/4)^{12}(5/256)^7 \approx 2^{-63.7}$. The standard chi-square frequency statistical test can then be applied to $\{u_t\}$ to distinguish this sequence from a purely random binary sequence. The distinguishing attack is successful if the segment length is about $1/\varepsilon_{w,v}^2 \approx 2^{127.4}$. The computational complexity of processing this amount of keystream is $O(2^{127.4})$ which is beyond that of exhaustive key search $O(2^{80})$. In the next Section we explain how to achieve a sequence with correlation coefficient greater than 2^{-40}.

4.3 Linear Equation with Greater Correlation Coefficient

Given a linear equation of consecutive output bits of the form (15), linear equations with greater correlation coefficients may be found using the generating function concept. The clue is that if we have $b_t = p(D)a_t$, then for an arbitrary polynomial $k(x)$ we have $k(D)b_t = k(D)p(D)a_t$. Therefore, we must multiply both sides of (15) by an appropriate polynomial $k(D)$ to obtain

$$\begin{aligned} u_t^* \overset{\Delta}{=} k(D)u_t &= k(D)\varphi(D)z_t \\ &= k(D)\varphi(D)\gamma_t + k(D)c(D)\delta_t, \end{aligned} \tag{21}$$

such that the correlation coefficient of $\{u_t^*\}$ is greater than that of $\{u_t\}$. The less $hw(k\varphi)$ and $hw(kc)$ are, the greater the correlation coefficient of $\{u_t^*\}$ will be. In general, it is not easy to manage to keep both $hw(k\varphi)$ and $hw(kc)$ low. However, for the aforementioned values of w and v in (17) and (18), the corresponding polynomials $\varphi(x)$ and $c(x)$ in (19) and (20) have very special forms and can be factorized in the following way, which facilitates finding the desired $k(x)$.

$$\varphi(x) = (1 + x^{80})(1 + x^{13} + x^{23} + x^{38} + x^{51} + x^{62} + x^{80}) \qquad \textbf{(22)}$$

$$c(x) = x(1 + x^{13} + x^{23} + x^{38} + x^{51} + x^{62} + x^{80}) \qquad \textbf{(23)}$$

In order to find $k(x)$, suppose that $p^*(x) = 1 + x^b + x^t$, $(1 \le b < t)$, is a trinomial multiple of $p(x)$ where

$$p(x) = 1 + x^{13} + x^{23} + x^{38} + x^{51} + x^{62} + x^{80}. \qquad \textbf{(24)}$$

Then choosing $k(x) = p^*(x)/p(x)$ leads to

$$\begin{aligned} u_t^* \overset{\Delta}{=} k(D)u_t &= (D^{80}+1)p^*(D)z_t \\ &= (D^{80}+1)p^*(D)\gamma_t + Dp^*(D)\delta_t \cdot \end{aligned} \qquad \textbf{(25)}$$

If b = 80 then $hw((x^{80}+1)p^*(x)) = 4$, otherwise $hw((x^{80}+1)p^*(x)) = 6$. In the worst case, that is $b \neq 80$ which is more probable, the correlation coefficient of $\{u_t^*\}$ is equal to $\varepsilon_{w,v} = (1/4)^6(5/256)^3 \approx 2^{-29}$. Thus, the required output length and computational time complexity for distinguishing the Grain output sequence from a purely random sequence is about $1/\varepsilon^2 \approx 2^{58}$.

*Remark 1***.** The problem of finding a low weight multiple of a randomly chosen irreducible polynomial of degree n has been well considered in [7] and [8]. In short, a trinomial multiple of degree about $2^{n/2}$ can be found using $O(2^{n/2})$ time and space. Therefore, we expect that the required trinomial multiple $p^*(x)$ of the primitive polynomial $p(x)$ be found using time and memory complexities of $O(2^{40})$.

5 Linear Sequential Circuit Approximation of Trivium

In this section, we discuss the linear sequential circuit approximation of Trivium stream cipher and show that this approximation is not successful in distinguishing the Trivium output sequence from a purely random one.

For Trivium stream cipher we have M = 288. Let S_t be a 288-bit binary column vector which contains the state of Trivium at time t, that is $(s_1, s_2, \ldots, s_{288})^T$ in the pseudo-code introduced in Section 2.2. Since the output function and all components of the next-state function, except three of them, are linear for Trivium stream cipher, decomposition of these functions is performed easily. It is sufficient to consider the linear approximations of the 1st, 94th and 178th component of the next-state function given in the following

$$s_{t+1,1} = s_{t,243} + s_{t,288} + s_{t,286} \cdot s_{t,287} + s_{t,69} \qquad \textbf{(26)}$$

$$s_{t+1,94} = s_{t,66} + s_{t,93} + s_{t,91} \cdot s_{t,92} + s_{t,171} \qquad \textbf{(27)}$$

$$s_{t+1,178} = s_{t,162} + s_{t,177} + s_{t,175} \cdot s_{t,176} + s_{t,264} \, . \qquad \textbf{(28)}$$

The absolute value of the correlation coefficient of all four possible linear approximations of the Boolean function $x \cdot y$ is equal to ½. Replacing each of the quadratic terms of the above functions with one of the four possible linear approximations leads to $4^3 = 64$ different decompositions for the next state function. In this section we merely give the details of linear sequential circuit approximation for the decomposition which eliminates the quadratic terms in accordance with approximating the Boolean function $x \cdot y$ with constant zero function. The results of the remaining 63 decompositions are given in Section 5.4.

5.1 Basic Linear Sequential Circuit Approximation

Eliminating the quadratic terms from equations (26) to (28), the linear approximations (5) and (6) for Trivium can be written as follows

$$S_t = AS_{t\text{-}1} + H\Delta_t \, , \; t \ge 1 \qquad \textbf{(29)}$$

$$z_t = BS_t \, , \; t \ge 1 \, . \qquad \textbf{(30)}$$

Here $H = [h_{i,j}]$ is a 288×3 binary matrix whose all entries are zero, except $h_{1,1}$, $h_{94,2}$ and $h_{178,3}$, $\Delta_t = [\delta_{1,t} \quad \delta_{2,t} \quad \delta_{3,t}]^T = [\delta_1(S_{t-1}) \quad \delta_{94}(S_{t-1}) \quad \delta_{178}(S_{t-1})]^T$ is the 3-bit column noise vector corresponding to the 1st, 94th and 178th component of the next-state function, and A and B are as follows

$$B = e_{66} + e_{93} + e_{162} + e_{177} + e_{243} + e_{288} \, , \qquad (31)$$

$$A = \begin{bmatrix} e_{69} + e_{243} + e_{288} \\ e_1 \\ \vdots \\ e_{92} \\ e_{66} + e_{93} + e_{171} \\ e_{94} \\ \vdots \\ e_{176} \\ e_{162} + e_{177} + e_{264} \\ e_{178} \\ \vdots \\ e_{287} \end{bmatrix} \tag{32}$$

where e_i ($1 \leq i \leq 288$) denotes the i^{th} row of the 288×288 identity matrix.

Using the general relation (7), the basic linear sequential circuit approximation of Trivium corresponding to the decompositions A and B can be expressed by

$$u_t \overset{\Delta}{=} \sum_{i=0}^{m} \varphi_i z_{t+i} = \sum_{i=0}^{m} c_{1,i}\delta_{1,t+i} + \sum_{i=0}^{m} c_{2,i}\delta_{2,t+i} + \sum_{i=0}^{m} c_{3,i}\delta_{3,t+i} \ , \tag{33}$$

or equivalently in the generating function domain by

$$u_t = \varphi(D) z_t = c_1(D)\delta_{1,t} + c_2(D)\delta_{2,t} + c_3(D)\delta_{3,t} \ , \tag{34}$$

where $\varphi(x) = \sum_{i=0}^{m} \varphi_i x^i$ is the minimal polynomial of A and $c_j(x) = \sum_{i=0}^{m} c_{j,i} x^i$, $1 \leq j \leq 3$, whose coefficients are defined by

$$[c_{1,i} \quad c_{2,i} \quad c_{3,i}] \overset{\Delta}{=} \sum_{k=0}^{m-i} \varphi_{k+i} B A^k H \ \cdot \tag{35}$$

The polynomials $c_1(x), c_2(x), c_3(x)$ and $\varphi(x)$ are as follows

$$\begin{aligned} \varphi(x) = {} & 1 + x^6 + x^{12} + x^{15} + x^{18} + x^{21} + x^{24} + x^{30} + x^{36} + x^{45} + x^{51} + x^{54} + x^{57} \\ & + x^{63} + x^{69} + x^{72} + x^{75} + x^{78} + x^{81} + x^{84} + x^{90} + x^{96} + x^{102} + x^{108} + x^{114} \\ & + x^{120} + x^{123} + x^{126} + x^{129} + x^{135} + x^{201} + x^{207} + x^{210} + x^{213} + x^{216} + \\ & x^{222} + x^{228} + x^{234} + x^{240} + x^{246} + x^{252} + x^{258} + x^{264} + x^{270} + x^{276} + x^{282}, \end{aligned} \tag{36}$$

$$\begin{aligned} c_1(x) = {} & x + x^7 + x^{13} + x^{16} + x^{19} + x^{22} + x^{31} + x^{37} + x^{40} + x^{52} + x^{61} + x^{73} + x^{79} \\ & + x^{85} + x^{88} + x^{91} + x^{94} + x^{97} + x^{100} + x^{103} + x^{106} + x^{118} + x^{124} + x^{127} + \\ & x^{130} + x^{133} + x^{145} + x^{151} + x^{154} + x^{157} + x^{160} + x^{163} + x^{166} + x^{169} + x^{172} \\ & + x^{175} + x^{178} + x^{181} + x^{184} + x^{187} + x^{190} + x^{193} + x^{199} + x^{205} + x^{211} + \\ & x^{217}, \end{aligned} \tag{37}$$

$$\begin{aligned} c_2(x) = {} & x + x^7 + x^{13} + x^{16} + x^{19} + x^{22} + x^{40} + x^{43} + x^{61} + x^{64} + x^{67} + x^{85} + x^{88} \\ & + x^{91} + x^{97} + x^{103} + x^{118} + x^{124} + x^{130} + x^{133} + x^{151} + x^{154} + x^{157} + x^{160} \\ & + x^{163} + x^{166} + x^{169} + x^{172} + x^{175} + x^{178} + x^{181} + x^{184} + x^{187} + x^{190} + \end{aligned} \tag{38}$$

$$x^{193} + x^{196} + x^{199} + x^{202} + x^{208} + x^{214},$$

$$\begin{aligned} c_3(x) = {} & x + x^{16} + x^{22} + x^{31} + x^{34} + x^{37} + x^{40} + x^{52} + x^{58} + x^{61} + x^{64} + x^{67} + \\ & x^{70} + x^{79} + x^{88} + x^{94} + x^{109} + x^{112} + x^{115} + x^{118} + x^{121} + x^{124} + x^{127} + \\ & x^{133} + x^{139} + x^{154} + x^{157} + x^{160} + x^{163} + x^{166} + x^{169} + x^{172} + x^{175} + x^{181} \\ & + x^{187} + x^{193} + x^{199} + x^{205} + x^{211} + x^{217}. \end{aligned} \quad \textbf{(39)}$$

5.2 Correlation Coefficient Analysis

As it was explained in Section 3, the sum of unbalanced Boolean functions is also unbalanced with the correlation coefficient very close to the product of the individual correlation coefficients, provided that the functions are picked independently at random [4, 5]. All noise terms $\delta_{i,t}$ ($i = 1, 2, 3$ and $t \geq 1$) arises from the product of two (almost independent random) binary terms, and therefore have correlation coefficient equal to ½. While the noise terms $\delta_{i,t}$ and $\delta_{j,t'}$ can be considered (approximately) independent for $i \neq j$ and $t \neq t'$, the independence assumption is not satisfied for $\delta_{i,t}$ and $\delta_{i,t+1}$ ($i = 1, 2, 3$ and $t \geq 1$); because they are the product of two terms which one term is in common, see the equations (26) to (28).

However, all the blocks[3] in the polynomials c_1, c_2 and c_3 have length one and thus there is no concern about the independence of the sum of noise terms in (33), see (37) to (39). The total number of blocks in c_1, c_2 and c_3 is 126 which shows that the correlation coefficient of $\{u_t\}$ is $\varepsilon = 2^{-126}$.

Remark 2. If there were some blocks in c_1, c_2 and c_3 with length $n \geq 2$, we must have grouped the noise functions into suitable categories such that the required independence assumption is satisfied. In other words, we must have included the total effect of the noise terms corresponding to each run as one independent noise term. The some of n adjacent noise terms, that is $\delta_{i,t} + \delta_{i,t+1} + \cdots + \delta_{i,t+n-1}$ ($i = 1, 2, 3$ and $t \geq 1$), can be expressed by the Bent function $x_1x_2 + x_2x_3 + \cdots + x_nx_{n+1}$ which has correlation coefficient equal to $2^{-\lfloor (n+1)/2 \rfloor}$. Therefore, if we denote the total number of runs with length n ($n \geq 1$) in all the polynomials $c_1(x)$, $c_2(x)$ and $c_3(x)$ by k_n, the correlation coefficient of $\{u_t\}$ is $\varepsilon = \prod_{n \geq 1} 2^{-k_n \lfloor (n+1)/2 \rfloor}$ in general.

[3] A consecutive subsequence of ones in a sequence (or in its equivalent polynomial) which are followed immediately after and before by a zero (if there are such bits) is called a block. For example the sequence [1 0 1 1 0 0 111 0 1 0 0 11 0] (equivalent to the polynomial $1 + x^2 + x^3 + x^6 + x^7 + x^8 + x^{10} + x^{13} + x^{14}$) has two blocks of length one, two blocks of length two and one block of length three.

5.3 Linear Equation with Greater Correlation Coefficient

As it was explained in Section 4.3, linear equations with correlation coefficients greater than 2^{-126} may be found using the generating function concept. To this end, we must multiply both sides of (34) by an appropriate polynomial $k(D)$ to obtain

$$\begin{aligned} u_t^* \overset{\Delta}{=} k(D)u_t &= k(D)\varphi(D)z_t \\ &= k(D)c_1(D)\delta_{1,t} + k(D)c_2(D)\delta_{2,t} + k(D)c_3(D)\delta_{3,t}, \end{aligned} \qquad \textbf{(40)}$$

so that the correlation coefficient of $\{u_t^*\}$ is greater than that of $\{u_t\}$. Note that in computing the correlation coefficient of $\{u_t^*\}$, the Remark 2 must be taken into account. It seems too hard to make $\{u_t^*\}$ have greater correlation coefficient. We carried out thorough search over all polynomials $k(x)$ with non-zero constant term and degree up to 24. The maximum correlation coefficient, among all those polynomials, is achieved by the following two independent choices for $k(x)$ which is equal to 2^{-72}

$$k_1(x) = 1 + x^6 \qquad \textbf{(41)}$$

$$k_2(x) = (1+x)(1 + x^6). \qquad \textbf{(42)}$$

For $k_1(x)$, all of the polynomials $k_1(x)c_1(x)$, $k_1(x)c_2(x)$ and $k_1(x)c_3(x)$ have exactly just 24 runs of length one, while in case of $k_2(x)$ all of them have exactly just 24 runs of length two. According to the Remark 2, both of them are corresponding to correlation coefficient equal to 2^{-72}.

Looking into the polynomials $c_1(x)$, $c_2(x)$ and $c_3(x)$, it is obvious that they are all multiplications of some polynomials in x^3 and the polynomial x ($\varphi(x)$ is also a polynomial in x^3). One may think that linear functions with greater correlation coefficients could be found by considering $k(x)$ as a polynomial in x^3. We also carried out thorough search over all polynomials $k(x) = k'(x^3)$ which $k'(x)$ had non-zero constant term and degree up to 24. In this case, the maximum correlation coefficient among all those polynomials is again 2^{-72} achieved by $k'(x) = 1 + x^2$ which is in accordance with $k_1(x) = 1 + x^6$.

5.4 Results of Other Decompositions

As we discussed at the beginning of Section 5, there are 64 linear sequential circuit approximations for Trivium. In Sections 5.1 to 5.3 we presented the details of just one of them, i.e. the one which approximates the Boolean function $x \cdot y$ with the constant zero function. The other linear sequential models can easily be derived. The only important point which must be taken into account is the correlation coefficient of the categorized noise terms, i.e. $\delta_{i,t} + \delta_{i,t+1} + \cdots + \delta_{i,t+n-1}$, pointed out on Remark 2. It can be shown that for the linear approximation $\alpha x + \beta y$ of $x \cdot y$, the correlation coefficient of the categorized noise terms $\delta_{i,t} + \delta_{i,t+1} + \cdots + \delta_{i,t+n-1}$ is equal to $r_n s_n 2^{-\lfloor (n+1)/2 \rfloor}$ where

$$r_n = \begin{cases} (-1)^{\lfloor (n+2)/4 \rfloor} & \text{if } (\alpha,\beta)=(1,0) \\ 1 & \text{otherwise} \end{cases}$$
$$s_n = \begin{cases} 1 & \text{if } (n \neq 2 \bmod 4) \text{ or } (\alpha,\beta)=(0,0) \text{ or } (\alpha,\beta)=(1,1) \\ 0 & \text{otherwise} \end{cases} . \qquad \textbf{(43)}$$

In the special case $(\alpha,\beta)=(0,0)$, in accordance with Remark 2, we have $r_n = s_n = 1$ for all $n \geq 1$.

To summarize the results, having implemented linear sequential circuit approximation method for other decompositions, we could not find any linear relation with correlation coefficient greater than 2^{-72}, however, we found many relations with correlation coefficient of exactly 2^{-72}. The value 2^{-72} of the best correlation coefficient which we found shows that the time complexity for distinguishing the output sequence of Trivium from a truly random generator is $O(2^{144})$. It seems impossible to manage to find a linear function of consecutive output bits with correlation coefficient more than 2^{-40} in order to provide a successful distinguishing attack.

6 Conclusion

In this paper using the linear sequential circuit approximation method, we evaluated the strength of two candidates of ECRYPT Stream Cipher Project, Grain and Trivium, against distinguishing attack. We showed that on Grain, a distinguishing attack can be mounted which needs about 2^{58} bits of the key-stream and a preprocessing phase for computing a trinomial multiple of a certain primitive polynomial with degree 80 which can be performed using $O(2^{40})$ time and space. This result shows that the feedback functions of NFSR, the output filter function and maybe the feedback polynomial of LFSR have been poorly chosen for Grain. In [10], a key-recovery attack which requires 2^{43} computations and 2^{38} key-stream bits has also been mounted on Grain; moreover, some criteria for choosing Grain parameters have been introduced.

For Trivium, we extracted the linear sequential circuit approximation and derived a linear function of consecutive output bits which is held with correlation coefficient of about 2^{-72}. It seems very hard to find a linear function of consecutive output bits with correlation coefficient greater than 2^{-40} to have a successful distinguishing attack. In spite of the linearity of the output function and all of the components of the next-state function of Trivium- except three of them which also have very near distances from some linear functions- this general method fails. This arises from a novel view of stream cipher design [2] which we were unaware of till its publication and is worth to be mentioned here.

Typical design method of key-stream generators is based on providing a sufficient amount of period first, and then imposing additional requirements. Grain has been designed in this view. The new idea used in Trivium design is based on first keeping the largest correlations with linear functions below safe bounds. This method considers other important properties, such as a sufficiently long period afterwards. Refer to [2] for more details.

Acknowledgment. The authors would like to thank the anonymous referees of FSE'06 for their useful comments on this paper.

References

1. De Canniere C. and Preneel B.: Trivium Specifications. eSTREAM, ECRYPT Stream Cipher Project Report 2005/030 (2005) `http://www.ecrypt.eu.org/stream/`.
2. De Canniere C. and Preneel B.: Trivium A Stream Cipher Construction Inspired by Block Cipher Design Principles. eSTREAM, ECRYPT Stream Cipher Project Report 2006/021 (2006) `http://www.ecrypt.eu.org/stream/`.
3. eSTREAM, the ECRYPT Stream Cipher Project (2005). `http://www.ecrypt.eu.org/stream/`
4. Golic J. Dj.: Intrinsic statistical weakness of keystream generators. Advances in Cryptology - ASIACRYPT '94, Lecture Notes in Computer Science, vol. 917, pp. 91-103 (1995).
5. Golic J. Dj.: Linear models for keystream generators. IEEE Transaction on Computers, vol. 45 No. 1, pp. 41-49, Jan. (1996).
6. Hell M., Johansson T. and Meier W.: Grain - A Stream Cipher for Constrained Environments. eSTREAM, ECRYPT Stream Cipher Project Report 2005/010 (2005), `http://www.ecrypt.eu.org/stream/`
7. Penzhorn W.T., Kühn G.J.: Computation of Low-Weight Parity Checks for Correlation Attacks on Stream Ciphers. Cryptography and Coding, LNCS 1024, Springer, pp.74-83, (1995).
8. Wagner D.: A generalized birthday problem. Advances in Cryptology CRYPTO 2002, LNCS 2442, pp.288-304, Springer-Verlag (2002); extended abstract is available at: `http://www.cs.berkeley.edu/~daw/papers/genbday.html`
9. NESSIE: New European Schemes for Signature, Integrity and Encryption, `http://www.nessie.eu.org/nessie/`.
10. Berbain C., Gilbert G. and Maximov A.: Cryptanalysis of Grain. eSTREAM, ECRYPT Stream Cipher Project Report 2006/019 (2006) `http://www.ecrypt.eu.org/stream/`.
11. Khazaei S. and Hassanzadeh M.: Linear Sequential Circuit Approximation of the Trivium Stream Cipher. eSTREAM, ECRYPT Stream Cipher Project Report 2005/063 (2005) `http://www.ecrypt.eu.org/stream/`.
12. Khazaei S., Hassanzadeh M. and Kiaei M.: Distinguishing attack on Grain. eSTREAM, ECRYPT Stream Cipher Project Report 2005/071 (2005) `http://www.ecrypt.eu.org/stream/`.