Trivium vs. Trivium Toy

Antonio Castro Lechtaler[1,2], Marcelo Cipriano[1], Edith García[1], Julio César Liporace[1], Ariel Maiorano[1], Eduardo Malvacio[1]

[1]EST – Facultad de Ingeniería – Instituto Universitario del Ejército
[2]FCE - Universidad de Buenos Aires

{acastro, marcelocipriano}@iese.edu.ar
{edithxgarcia, jcliporace, maiorano, edumalvacio}@gmail.com

Abstract. We present the characteristic and minimal polynomials of the linear algorithms Trivium and Trivium Toy. We show the different cycles and minimum lengths obtained. The existence of initial states determining short cycles is verified. Finally, linear Trivium Toy is shown to be as cryptologically secure as the linear Trivium algorithm.

Keywords: Trivium, Trivium Toy, cycles, periods, weak initial states.

1 Introduction

1.1 Pseudo-random Sequences

Pseudo-random sequences with long cycles and high linear complexity are widely used in the fields of communications and cryptology.

Until recently, these sequences were generated with simple algorithms, using *Linear Feedback Shift Registers (LFSR)* – alone or combined – along with certain non-linear components.

Currently, their design has become more complex. Nonetheless, two cryptological properties should be carefully observed and controlled: the length of the cycle and the linear complexity. Generators with short periods, small cycles or low linear complexity are cryptanalized and then broken. Hence, algorithm design must verify that it achieves an acceptable limit value of the minimum period.

1.2 Trivium and Trivium Toy

The stream cipher Trivium – an e-STREAM finalist- has successfully endured every cryptological attack so far. [2, 6, 13]. However, its minimum period has not been determined neither by design nor by cryptanalysis, remaining as an open problem to this date.

2 Overview

2.1 Feedback Shift Register (FSR)

Let the polynomial $f(x)$:

$$f(x) = c_0x^0 + c_1x^1 + c_2x^2 + \dots\dots\dots + c_{n-1}x^{n-1} + x^n \quad c_i \in \{0,1\} \tag{1}$$

be an n^{th} degree characteristic polynomial over $GF(2)$.

A sequence $\boldsymbol{s} = \{s_i\}$ is a length n *LFSR* sequence generated by $f(x)$ if it satisfies the following linear recurrence relation:

$$s_{n+k} = \sum_{i=0}^{n-1} c_i s_{k+i} \quad (k=0;1;2;\dots) \tag{2}$$

Note that if the first n bits of $\boldsymbol{s}$ belong to the initial state, the register corresponds to a feedback polynomial (*feedback function*).

If, on the other hand, $\boldsymbol{s}$ begins with the fed bits, except for those in the initial state, the characteristic polynomial is considered a *feedforward function*.

The polynomial $f(x)$ can also be interpreted as a linear Boolean function

$$f\colon \{0,1\}^n \rightarrow \{0,1\} \tag{3}$$

$$f(x_0; x_1;..; x_{n-1}) = c_0x_0 + c_1x_1 + \dots + c_{n-1}x_{n-1} \tag{4}$$

If the resulting function is non-linear, it is considered a N*on-Linear Feedback Shift Register (NLFSR).*

$$(s_0; s_1; \dots; s_{n-1}) \qquad s_i \in \{0,1\} \tag{5}$$

where s_i is the initial state of the LFSR generating the sequence $\boldsymbol{s}$.

Given any polynomial $f(x)$ of degree n, the reciprocal polynomial $f^*(x)$ is defined as

$$f^*(x) = x^n f(x^{-1}) \tag{6}$$

2.2 Properties of m-sequences

If $f(x)$ is a *primitive polynomial*[1]*,* $\boldsymbol{s}$ is an *m-sequence,* thus $\boldsymbol{s}$ has a maximum cycle of 2^n-1; i.e., given any initial state (except when all values equal 0), all sequences belong to the same cycle.

If $f(x)$ is not primitive, different initial states generate cycles smaller than 2^n-1. [7]

A minimal polynomial of $\boldsymbol{s}$ is the polynomial of the smallest degree generating $\boldsymbol{s}$. If $m(x)$ is the minimal polynomial of $\boldsymbol{s}$, then $m(x)$ divides $f(x)$.

---

[1] A polynomial $f(x)$ over GF(2), irreducible of degree n, is *primitive* if the least positive integer m such that $f(x) \mid (x^m + 1)$ is $m = 2^n-1$.

The *linear complexity* of $\boldsymbol{s}$ *(LC(s))* is the degree of the minimal polynomial *m(x)*. In general, *m(x)* can be found using the *Berlekamp-Massey* algorithm, taking *2LC(s)* consecutive bits [10].

S(f(x)) is defined as the set of all binary sequences which satisfy the recurrence relation determined by *f(x)*.

The *order of f(x)* is defined as the least positive integer *e* such that $f(x) \mid x^e + 1$.

The period of a sequence $\boldsymbol{s}$ equals the order of its minimal polynomial. It is the least integer *p* such that $s_n = s_{n+p}$ for every positive *n*.

The array $(s_0; s_1; \ldots; s_{p-1})$ is the cycle of the sequence $\boldsymbol{s}$ and its size is equal to *p*.

2.3 Linear Trivium

The stream algorithm TRIVIUM was created by Christophe De Cannière and Bart Preneel. It was designed to generate at least 2^{64} bits, using an *80-bit* secret key and an initialization vector (IV) of also *80* bits [3].

It consists of three combined NLFSRs. The first register controls the second, the second controls the third, and the last one controls the first.

The core idea behind the design focuses on using the principles of block cipher design to create equivalent components in stream ciphers.

Three parts can be clearly identified in the design:

- A linear part originated by a *96-bit* sub-generator which consists of three linear feedforward and feedback registers.
- An interleave process *in threes* of the linear Trivium sub-generator [8].
- A non-linear part obtained from AND operations in the linear Trivium.

The output consists of three combined non-linear shift registers of lengths 93, 84, and 111 in which particular positions are selected to obtain a key bit stream. Whereas no efficient attack has successfully broken the generator, its period remains undetermined [11, 12].

A complete description is given by the following simple pseudo-code:

INPUT: $s_0, s_1, .., s_{287}$ initial state, integer n., s_i {0,1}.

OUTPUT: binary sequence $\{k_t\}$

1.Initialization.

1.1 $t_1 \leftarrow s_{65} \oplus s_{92}$
1.2 $t_2 \leftarrow s_{161} \oplus s_{176}$
1.3 $t_3 \leftarrow s_{242} \oplus s_{287}$
2.While (t<n) do the following:
2.1 $k_t \leftarrow t_1 \oplus t_2 \oplus t_3$
2.2 $t_1 \leftarrow t_1 \oplus s_{90} \otimes s_{91} \oplus s_{170}$
2.3 $t_2 \leftarrow t_2 \oplus s_{174} \otimes s_{175} \oplus s_{263}$
2.4 $t_3 \leftarrow t_3 \oplus s_{285} \otimes s_{286} \oplus s_{68}$
2.5 $(s_0; s_1; \ldots; s_{92}) \leftarrow (t_3; s_0; ..; s_{91})$
2.6 $(s_{93}; s_{94}; ..; s_{176}) \leftarrow (t_1; s_{93}; ..; s_{175})$

```
        2.7    (s177;s178;..;s287)←(t2;s177;..;s285)
    3.Return {kt}
```

Note that ⊕ is the XOR operation and ⊗ the AND operation.

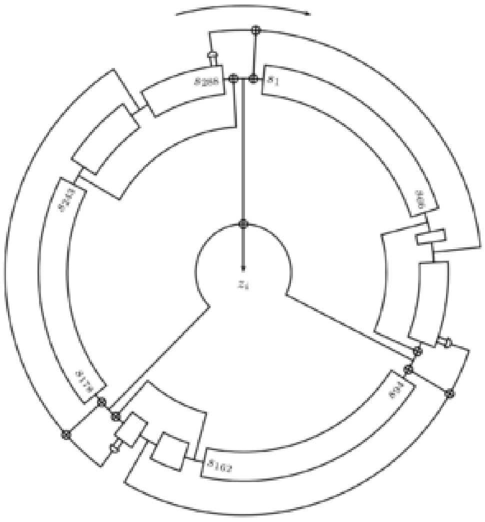

Fig.1: Original Trivium diagram

The linear Trivium algorithm follows the same procedure with the exception of the AND operations which are omitted. Terms $s_{90} \otimes s_{91}$, $s_{174} \otimes s_{175}$ and $s_{285} \otimes s_{286}$ are eliminated.

2.4 Linear Trivium Toy

In [1] we present a reduced model of the Trivium algorithm. The reduced model - decimated by 3 is based on previous work by Yun Tian et al, who developed an extended model of the TRIVIUM structure [14].

The model consists of three NLFSRs - X, Y, and Z - of lengths 31, 28 and 37 in the following states:

$$\begin{array}{rl} X(31): & X_0, X_1, \ldots\ldots\ldots\ldots, X_{30} \\ Y(28): & Y_0, Y_1, \ldots\ldots\ldots\ldots, Y_{27} \\ Z(37): & Z_0, Z_1, \ldots\ldots\ldots\ldots, Z_{36} \end{array} \quad (7)$$

Being the feedback of each register; i.e. the bit input in each:

$$\begin{array}{l} X_0: Z_{21} \oplus Z_{36} \oplus Z_{35} \otimes Z_{34} \oplus X_{22} \\ Y_0: X_{21} \oplus X_{30} \oplus X_{29} \otimes X_{28} \oplus Y_{25} \\ Z_0: Y_{22} \oplus Y_{27} \oplus Y_{26} \otimes Y_{25} \oplus Z_{28} \end{array} \quad (8)$$

and the key bit stream:

$$K_t: \qquad X_{21} \oplus X_{30} \oplus Y_{22} \oplus Y_{27} \oplus Z_{21} \oplus Z_{36} \tag{9}$$

In a stream cipher each plaintext bit is encrypted one at a time with the corresponding bit of the key bit stream, to give a bit of the ciphertext stream.

$$C_t = P_t \oplus K_t \tag{10}$$

where C_t is the cipher bit and P_t is the plaintext bit.

The linear Trivium Toy algorithm consists of the same equations shown in (8) omitting the AND operations. Terms $Z_{35} \otimes Z_{34}$, $X_{29} \otimes X_{28}$ *and* $Y_{26} \otimes Y_{25}$ are eliminated.

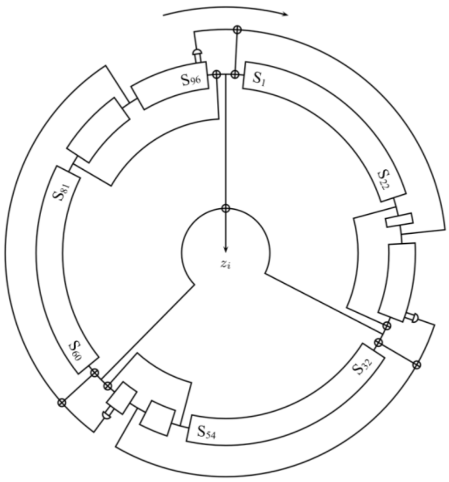

Fig.2: Trivium Toy diagram.

3 Characteristic Polynomial of the Linear Trivium Sub-generator and the Linear Trivium Toy

3.1 Feedforward and Feedback Functions of the Linear Trivium Sub-generator

The *feedforward* and *feedback* functions ($f_i(x)$ and $g_i(x)$ respectively) in their reciprocal form (6), define the Trivium sub-generator [2] and determine their characteristic polynomial $p(x)$:

$$p(x) = \prod_i f_i^*{}_{(x)} + \prod_i g_i^*{}_{(x)} \tag{11}$$

-

$$f_i^* \begin{cases} f_1^* = 1 + x^9 \\ f_2^* = 1 + x^5 \\ f_3^* = 1 + x^{15} \end{cases} \tag{12}$$

$$g_i^* \begin{cases} g_1^* = x^{31} + x^8 \\ g_2^* = x^{28} + x^2 \\ g_3^* = x^{37} + x^8 \end{cases} \tag{13}$$

$$\begin{aligned} p(x) = x^{96} + x^{73} + x^{70} + x^{67} + x^{47} + x^{44} + x^{41} + x^{29} + x^{24} + x^{20} \\ + x^{18} + x^{15} + x^{14} + x^9 + x^5 + 1 \end{aligned} \tag{14}$$

The polynomial is not irreducible, i.e., it can be expressed as a product of two polynomials such that:

$$p(x) = q(x) * r(x) \tag{15}$$

$$q(x)=(x+1)^3 \tag{16}$$

$$\begin{aligned} r(x) = x^{93} + x^{92} + x^{89} + x^{88} + x^{85} + x^{81} + x^{80} + x^{77} + x^{76} + x^{73} \\ + x^{72} + x^{70} + x^{68} + x^{67} + x^{44} + x^{43} + x^{41} + x^{39} \\ + x^{38} + x^{35} + x^{34} + x^{31} + x^{30} + x^{27} + x^{25} + x^{23} \\ + x^{20} + x^{19} + x^{17} + x^{14} + x^{13} + x^{12} + x^9 + x^8 \\ + x^6 + x^4 + x + 1 \end{aligned} \tag{17}$$

where *r(x)* is a primitive polynomial.

3.2 Feedforward and Feedback Functions of the Linear Trivium Toy

Due to [1], consider the X register. The feedforward of the Z_{21} position corresponds to x^{22} of f_3; and the feedback X_{22} corresponds to x^{23} of g_1.

For the Y register, the feed-forward of the X_{21} position corresponds to x^{22} of f_1; and the feedback is Y_{25}, corresponding to x^{26} of g_2.

For the Z register, the feed-forward of the Y_{22} position corresponds to x^{23} of f_2; and its feedback is Z_{28}, corresponding to x^{29} of g_3.

$$f^*_i(x) = \begin{cases} f^*_1(x) = (x^{-22} + x^{-31}) * (x^{31}) = (x^9 + 1) \\ f^*_2(x) = (x^{-23} + x^{-28}) * (x^{28}) = (x^5 + 1) \\ f^*_3(x) = (x^{-22} + x^{-37}) * (x^{37}) = (x^{15} + 1) \end{cases} \tag{18}$$

$$g^*_i(x) = \begin{cases} g^*_1(x) = (x^{-23}+1) * (x^{31}) = x^8 + x^{31} \\ g^*_2(x) = (x^{-26}+1) * (x^{28}) = x^2 + x^{28} \\ g^*_3(x) = (x^{-29}+1) * (x^{37}) = x^8 + x^{37} \end{cases} \qquad (19)$$

As explained above, the *feedforward* and *feedback* functions ($f_i(x)$ and $g_i(x)$ respectively) define the linear *Trivium Toy* and determine its characteristic polynomial $p(x)$.

The *characteristic polynomial* of the linear Trivium Toy is obtained by applying formulaes (6) and (11) to $f^*_i(x)$ and $g^*i(x)$. The resulting $p(x)$ is the same as the polynomial of the linear sub-generator of Trivium, as well as the one obtained in formula (14).

4 Calculating Sequences and Periods of the Linear Trivium Toy

4.1 Background

In order to establish the main results of this section, the following theorems must be considered [9]:

Theorem 1: Let f(x) = $\prod_i f_i^{b_i}$ where the $f_i(x)$ are distinct irreducible polynomials over $GF(2)$ and b_i are positive integers. Then:

$$S(f(x)) = S(f_1(x)^{b_1}) + S(f_2(x)^{b_2}) + \cdots + S(f_n(x)^{b_n}) \qquad (20)$$

Define $S(f_1(x)^{b_1}) + S(f_2(x)^{b_2}) + \cdots + S(f_n(x)^{b_n})$ to be the set of all sequences $\boldsymbol{s}_1+\boldsymbol{s}_2+\ldots+\boldsymbol{s}_n$ with $\boldsymbol{s}_i \in S(f_i(x)^{b(i)})$.

Theorem 2: for each i = 1; 2;…; n, let s_i be a linear recurring sequence in $GF(2)$ with a minimal polynomial $f_i(x) \in GF(2)[x]$ and a least period p_i.
If the polynomials $f_1(x), f_2(x),\ldots, f_n(x)$ are pair-wise relatively prime, then the least period of $\boldsymbol{s}_1+\boldsymbol{s}_2+\ldots+\boldsymbol{s}_n$ is equal to the least common multiple of $p_1; p_2; \ldots; p_n$.

Theorem 3: let $f(x) = (g(x))^b$ with $g(x) \in GF(2)[x]$ irreducible over $GF(2)$, $g(0) \neq 0$, degree $(g(x)) = k$, *order* $(g(x)) = e$, and b a positive integer. Let t be the smallest integer with $2^t \geq b$. Then, $S(f(x))$ contains the following numbers of sequences with least periods: one sequence with least period 1, 2^k-1 sequences with least period e, and for $b \geq 2$, $2^{2^j k} - 2^{2^{j-1}k}$ sequences with least period $e*2^j$ $(j=1;,.;t-1)$, and $2^{kb} - 2^{2^{t-1}k}$ sequences with least period $e*2^t$.

4.2 Linear Trivium Toy Sequences and Periods

Formula (15) shows that the characteristic polynomial *p(x)* of the Linear Trivium Toy is reducible. Thus, different initial states yield different Least Periods or cycles. Theorems 1 to 3 are applied to obtain the following values:

For $q(x) = (x+1)^3$ from (16), given that it is not primitive:

Number of Sequences	**Least Period**
2	1
2	2
4	4

Table 1: Number of sequences and least periods for *q(x)*.

For the primitive *r(x)* in (17), a null trivial sequence is obtained and the rest of all possible sequences of maximum length are shown in the following table:

Number of Sequences	**Least Period**
1	1
2^{93}-1	2^{93}-1

Table 2: Number of sequences and least periods for r*(x)*.

Thus, for the polynomial *p(x)*:

Number of Sequences	**Least Period**
2	1
2	2
4	4
2*(2^{93}-1)	2^{93}-1
2*(2^{93}-1)	2*(2^{93}-1)
4*(2^{93}-1)	4*(2^{93}-1)

Table 3: Number of sequences and least periods for *p(x)*.

It can be observed that there are 8 sequences with short periods (of length 1, 2 and 4 bits). Hence, these sequences have been generated by weak initial states.

5 Calculating Sequences and Periods of the Linear Trivium

5.1 Feedforward and Feedback Functions of Linear Trivium with interleave process

The feedforward and feedback functions defining the linear Trivium -i.e., the sub-generator and the interleave process- are:

$$f^*_i(x) = \begin{cases} f^*_1(x) = 1 + x^{27} \\ f^*_2(x) = 1 + x^{15} \\ f^*_3(x) = 1 + x^{45} \end{cases} \tag{21}$$

$$g^*_i(x) = \begin{cases} g^*_1(x) = x^{93} + x^{24} \\ g^*_2(x) = x^{84} + x^{6} \\ g^*_3(x) = x^{111} + x^{24} \end{cases} \tag{22}$$

Given that the characteristic polynomial of the linear Trivium takes the form in (11) but with the functions shown in (21) and (22), the characteristic polynomial *p(x)* is:

$$\begin{aligned} p(x) = x^{288} + x^{219} + x^{210} + x^{201} + x^{141} + x^{132} + x^{123} + x^{87} + x^{72} \\ + x^{60} + x^{54} + x^{45} + x^{42} + x^{27} + x^{15} + 1 \end{aligned} \tag{23}$$

The polynomial is not irreducible, that is, it can be expressed as the product of four polynomials such that:

$$p(x) = q(x) * s(x) * t(x) * u(x) \tag{24}$$

$$q(x) = (x+1)^3 \tag{25}$$

$$s(x) = (x^2 + x + 1)^3 \tag{26}$$

$$\begin{aligned} t(x) = x^{93} + x^{90} + x^{87} + x^{86} + x^{84} + x^{83} + x^{82} + x^{81} + x^{80} + x^{79} \\ + x^{78} + x^{77} + x^{74} + x^{72} + x^{71} + x^{70} + x^{67} + x^{65} \\ + x^{63} + x^{62} + x^{51} + x^{44} + x^{41} + x^{38} + x^{35} + x^{34} \\ + x^{31} + x^{29} + x^{27} + x^{25} + x^{24} + x^{21} + x^{19} + x^{17} \\ + x^{16} + x^{15} + x^{11} + x^{9} + +x^{8} + x^{6} + x^{5} + x + 1 \end{aligned} \tag{27}$$

$$\begin{aligned} u(x) = x^{186} + x^{180} + x^{179} + x^{175} + x^{174} + x^{173} + x^{172} + x^{167} + x^{166} \\ + x^{164} + x^{163} + x^{162} + x^{161} + x^{160} + x^{158} + x^{157} \\ + x^{152} + x^{151} + x^{150} + x^{149} + x^{148} + x^{147} + x^{144} \\ + x^{142} + x^{141} + x^{140} + x^{139} + x^{138} + x^{135} + x^{130} \\ + x^{129} + x^{127} + x^{124} + x^{121} + x^{120} + x^{116} + x^{115} \\ + x^{113} + x^{110} + x^{109} + x^{107} + x^{101} + x^{100} + x^{98} \\ + x^{96} + x^{95} + x^{91} + x^{90} + x^{88} + x^{86} + x^{80} + x^{78} \\ + x^{75} + x^{74} + x^{71} + x^{70} + x^{69} + x^{66} + x^{64} + x^{61} \\ + x^{58} + x^{53} + x^{52} + x^{50} + x^{48} + x^{46} + x^{45} + x^{44} \\ + x^{43} + x^{42} + x^{41} + x^{40} + x^{39} + x^{38} + x^{32} + x^{31} \\ + x^{29} + x^{26} + x^{22} + x^{21} + x^{19} + x^{17} + x^{16} + x^{9} \\ + x^{8} + x^{7} + x^{6} + x^{5} + x^{2} + x + 1 \end{aligned} \quad (28)$$

5.2 Linear Trivium Sequences and Periods

The characteristic polynomial *p(x)* yields different sequences and length cycles, depending on the initial states of the registers.

For $q(x) = (x+1)^3$, the same values of table 1 are obtained. For the polynomial $s(x) = (x^2 + x + 1)^3$ from (26), the following values are obtained:

Number of Sequences	**Least Period**
1	1
3	3
12	6
48	12

Table 4: Number of sequences and least periods for s*(x)*.

For *t(x)* is primitive:

Number of Sequences	**Least Period**
1	1
2^{93}-1	2^{93}-1

Table 5: Number of sequences and least periods for *t(x)*.

And, for *u(x)* irreducible but not primitive:

Number of Sequences	**Least Period**
1	1
2^{186} -1	$3*(2^{93}$-1)

Table 6: Number of sequences and least periods for u*(x)*.

-

The characteristic polynomial *p(x)* of the Linear Trivium obtained yields the values:

Number of Sequences	Least Period
2	1
2	2
6	3
4	4
54	6
444	12
2b	b
2b	2b
8a+6b+8ab	3b
4b	4b
56a+54b+56ab	6b
448a+636b+256ab	12b

Table 7: Number of sequences and least periods for *p(x)*.

Note: For clarity, values have been replaced with a = $(2^{186}-1)$ and b = $(2^{93}-1)$

Tables *3* and *7* show that the cycles of the linear Trivium Toy and the linear Trivium have the same order of magnitude, with a difference in the maximum length between them of a factor of *3*. In other words, the difference observed is linear and not exponential or of some other type, indicating that their recursion lengths or linear complexities are comparable.

In the case of the linear Trivium, note the existence of *512* short cycle sequences, among these *512*, *444* sequences producing cycles of size *12*. Thus, the existence of weak initial states can be verified.

6 Conclusion

This work shows a linear equivalence between the Linear Trivium and the Linear Trivium Toy generators. The complexity of both algorithms only differs in one linear factor and their minimum periods are both of the order of 2^{93}. In addition, the number of sequences in the Linear Trivium with short periods rises significantly in comparison to the Linear Toy, leading to a considerable increase of weak initial states.

7 Future Research

Further work shall explore AND operations in the generators, analyzing them as NLFSR [4, 5] or as the combination of linear filters (feedforward and feedback) with non-linear inputs. The authors of the stream cipher Trivium restricted their scope to linear expressions. Advancing their analysis to more complex forms seems a reasonable direction to pursue.

References

1. Castro Lechtaler, A.; Cipriano, M.; García, E.; Liporace, J.; Maiorano, A.; Malvacio, E. “Model Design for a Reduced Variant of a Trivium Type Stream Cipher” In XVIII Congreso Argentino de Ciencias de la Computación, ISBN. 978-987-23963-1-2. Pg. 1483-1491. Mar del Plata, Argentina, 2013.
2. De Canniére, C. and Preneel, B. “TRIVIUM A Stream Cipher Construction Inspired by Block Cipher Design Principles”. In Workshop on Stream Ciphers Revisited (SASC2006), 2006.
3. De Canniére, C. and Preneel, B. “TRIVIUM Specifications”. eSTREAM, ECRYPT Stream Cipher Project, Report. 2008.
4. Dubrova, E. “A List of Maximum-Period NLFSRs”**,** Cryptology ePrint Archive, Report 2012/166, March 2012, http://eprint.iacr.org/2012/166
5. Dubrova, E. “A Scalable Method for Constructing Galois NLFSRs with Period $2^n - 1$ using Cross-joint Pairs”. Technical Report 2011/632, Cryptology ePrint Archive, November 2011. http://eprint.iacr.org/2011/632.
6. eSTREAM: eSTREAM – The ECRYPT Stream Cipher Project: http://www.ecrypt.eu.org/stream/
7. Golomb. “Shift Register Sequences”. Aegean Park Press, 1982.
8. Gong, Guang. “Theory and Applications of q-ary Interleaved Sequences”. IEEE Transactions on Information Theory. Vol. 41 No. 2. March 1995.
9. Lidl R., Niederreiter H. “Introduction to Finite Fields and their Applications”. Cambridge University Press, 1986.
10. Massey, J.L. “Shift-Register Synthesis and BCH Decoding”. IEEE Transactions on Information Theory 15, 1969.
11. Maximov, A. and Biryukov, A. “Two Trivial Attacks on Trivium”, Selected Areas in Cryptography, Lecture Notes in Computer Science, Vol.4876, Springer, 2007.
12. McDonald, C. and Pieprzyk, C. “Attacking Bivium with MiniSat”, Cryptology ePrint Archive,Report 2007/040, 2007.
13. Raddum, H.“Cryptanalytic Results on Trivium”, eSTREAM, ECRYPT Stream Cipher Project, Report 2006/039, 2006.
14. Yun Tian, Gongliang Chen, Jianhua Li: “On the Design of Trivium”. IACR Cryptology ePrint Archive 2009.