Nonlinear Pseudorandom Sequences Based on 90/150 LHGCA*

Un-Sook Choi[1], Sung-Jin Cho[2,**], Han-Doo Kim[3], Yoon-Hee Hwang[4],
and SeokTae Kim[5]

[1] Department of Multimedia Engineering, Tongmyong University
Busan 626-847, Korea
choies@tu.ac.kr

[2] Division of Mathematical Sciences, Pukyong National University
Busan 608-737, Korea
sjcho@pknu.ac.kr

[3] School of Computer Aided Science, Institute of Basic Science
Inje University, Gimhae 621-749, Korea
mathkhd@inje.ac.kr

[4] Department of Information Security, Graduate School, Pukyong National University
Busan 608-737, Korea
yhhwang@pknu.ac.kr

[5] Division of Electronic, Computer and Telecommunication Engineering,
Pukyong National University, Busan 608-737, Korea
setakim@pknu.ac.kr

Abstract. In a stream cipher the plaintext digits are encrypted one at a time, and the transformation of successive digits varies during the encryption. LFSRs produce sequences having large periods and good statistical properties, and are readily analyzed using algebraic techniques. But the output sequences of LFSRs are also easily predictable, if we know proper successive output sequences in output sequences. In this paper, we give a new method which generates nonlinear sequences using maximum-length cellular automata.

1 Introduction

Pseudorandom sequences have many applications in cryptography and communication engineering. The inherent simplicity of LFSRs, the ease and efficiency of implementation, some good statistical properties of the LFSR sequences, and the algebraic theory underlying these devices turn them into natural candidates for use in the construction of pseudorandom generator, targeted to the implementation of efficient stream cipher cryptographic strength. On the other hand, some of the attractive properties listed above are also the reason for the failure

* This work was supported by the Korea Research Foundation Grant funded by the Korean Government(MOEHRD)(KRF-2006-331-D00458).

** Corresponding author.

H. Umeo et al. (Eds): ACRI 2008, LNCS 5191, pp. 471–477, 2008.

of many of these constructions. The shrinking generator which is one of clock-controlled generators, is well-known pseudorandom sequence generator, proposed by Coppersmith et al.[1]. It is a very simple generator with good cryptographic properties.

Sabater et al.[2] proposed the algorithm to convert the shrinking generator into a 90/150 group CA-based linear model which is simple and can be applied to shrinking generators in a range of practical interest. But they didn't consider that individual cells of CA(Cellular Automata) can generate sequences having the same characteristic polynomial[3] and CA can generate sequences having short period according to seed vectors, even if the period of 90/150 group CA and the period of sequence generated by shrinking generator are same.

In this paper, we propose a new method which generates nonlinear pseudorandom sequences using two maximum-length 90/150 LHGCA obtained by Cho et al.'s algorithm[4]. The generator which generates these sequences is possible to overcome spatial weak points of the interleaved sequence generator proposed by Gong[5]. Unlike the method proposed by Sabater et al.[2], the new sequence generator can generate nonlinear sequences whose cycle lengths are always same for a given initial state. The nonlinear pseudorandom sequence obtained by our method has a larger period and a higher linear complexity than the shrunken sequence generated by LFSRs.

2 Preliminaries

CA is an array of cells where each cell is in any one of the permissible states. At each discrete time steps the next state of particular cell is usually assumed to depend only on itself and on its two neighbors (three-neighborhood dependency) for a local neighborhood CA. The state of the ith cell at time $(t+1)$ can thus be denoted as:

$$q_i(t+1) = f(q_{i-1}(t), q_i(t), q_{i+1}(t))$$

where f represents the combinatorial logic and it is called next state function.

In this paper we deal with 90/150 linear hybrid group cellular automata (LHGCA).

Characterizations of linear CA based on matrix algebraic tool have been reported in [6]. The matrix algebraic tool employing minimal polynomial and characteristic polynomial of the state transition matrix of CA showed various interesting features of CA behaviour. The most effective application of linear group CA has been proposed in the field of pseudorandom pattern generation, since many researchers[7] showed that maximum-length CA whose all nonzero states lying in a single cycle produce high quality pseudorandom patterns. It has been established that the maximum-length cycle in the state transition diagram of 90/150 CA can be produced only if the characteristic polynomial is primitive([3], [4]). In this paper only one-dimensional maximum-length 90/150 LHGCA are considered. 90/150 LHGCA is completely specified by which cells use rule 90 and 150. A natural form for the specification is an n-tuple $< d_1, d_2, \cdots, d_n >$, called the *rule vector*, where

$$d_i = \begin{cases} 0, & \text{if cell } i \text{ uses rule 90} \\ 1, & \text{if cell } i \text{ uses rule 150} \end{cases}$$

The shrinking generator was introduced by Coppersmith et al.[1]. Nevertheless, due to its simplicity and provable properties, it is a promising candidate for high-speed encryption application. The shrinking generator is a well-known keystream generator composed of two LFSRs. A control LFSR R_1 is used to select a portion of output sequence of a second LFSR R_2. Therefore the keystream produced is a shrunken version of the output sequence of R_2.

According to [1], let R_1 and R_2 be maximum-length LFSRs whose characteristic polynomials are primitive, of lengths L_1 and L_2, respectively, and let $\{k_i\}$ be an output sequence of the shrinking generator formed by R_1 and R_2. If $gcd(L_1, L_2) = 1$, the period of $\{k_i\}$ is $(2^{L_2} - 1) \cdot 2^{L_1 - 1}$ and its linear complexity LC satisfies the following inequality $L_2 \cdot 2^{L_1 - 2} < LC \leq L_2 \cdot 2^{L_1 - 1}$.

3 Interleaved Sequences

The interleaved sequences were introduced by Gong[5]. Interleaved sequences are constructed by taking sequences and combining them under control of a shift sequence $\mathbf{e}$. Let $\mathbf{a} = \{a_i\}$ be a binary sequence. If $\mathbf{a}$ is a periodic sequence with period l, then we write $\mathbf{a}$ by $[a_0, a_1, \cdots, a_{l-1}]$. The left shift operator L on $\mathbf{a}$ is defined as $L(\mathbf{a}) = \{a_1, a_2, \cdots\}$, i.e. the left shift operator L when applied to a sequence will shift the sequence to the left by one position. For $L^i(\mathbf{a}) = \{a_i, a_{i+1}, a_{i+2}, \cdots\}$, i is said to be *phase shift* of $\mathbf{a}$. Two periodic sequences $\mathbf{a}$ and $\mathbf{b}$ are shift equivalent if there exists an integer k such that $a_i = b_{i+k}$ for all $i = 0, 1, 2, \cdots$. Let $\mathbf{u} = [u_0, u_1, u_2, \cdots, u_{st-1}]$ be a binary sequence with period st, where s and t are integers greater than 1. We can arrange the elements of the sequence $\mathbf{u}$ into an $s \times t$ array as follows:

$$A = \begin{pmatrix} u_0 & u_1 & \cdots & u_{t-1} \\ u_t & u_{t+1} & \cdots & u_{2t-1} \\ \vdots & \vdots & \ddots & \vdots \\ u_{(s-1)t} & u_{(s-1)t+1} & \cdots & u_{st-1} \end{pmatrix}$$

If each column vector of A is either phase shift of a binary sequence $\mathbf{a}$ of period s, or zero sequence, then $\mathbf{u}$ is called an (s, t) *interleaved sequence*. Let A_j be the jth column vector of A which is the matrix form of $\mathbf{u}$, then $A = (A_0, A_1, \cdots, A_{t-1})$. A_j is the transpose of either $L^{e_j}(\mathbf{a})$, or $(0, \cdots, 0)$, where e_j is the phase shift of $\mathbf{a}$. If $A_j = (0, \cdots, 0)^t$, then we denote $e_j = \infty$. $\mathbf{u}$ is called an (s, t) *interleaved sequence associated with* $(\mathbf{a}, \mathbf{e})$, and $\mathbf{e} = (e_0, e_1, \cdots, e_{t-1})$ is called a *shift sequence* of $\mathbf{u}$.

This generator has some troubles that it must be paralleled t LFSRs with period s to generate a (s, t) interleaved sequence. In this paper we propose the method which is possible to overcome spatial weak points of the interleaved sequence generator. This method employs a maximum-length 90/150 LHGCA whose characteristic polynomial is primitive. High quality pseudorandom sequences can be generated from the CA. This is due to the apparent random phase shift of the output bit sequences from its various stages that are cell positions. Each cell position generates pseudorandom sequences. Unlike LFSRs, the phase shift is generally different between stages of a CA. Schemes finding phase shifts of maximum-length 90/150 LHGCA were proposed in [3].

4 SI Sequence Based on 90/150 LHGCA

Sabater et al.[2] considered the linear model of shrinking generator described in [8] in terms of 90/150 LHGCA. They proposed a synthesis algorithm for the 90/150 LHGCA which is equivalent to any shrinking generator. This LHGCA is formed by concatenations of basic maximum-length 90/150 LHGCA and their mirror images, with one or two modification in each LHGCA component. The characteristic polynomial of the 90/150 LHGCA obtained by the algorithm is the same as the one of the original shrinking generator. Since the number of concatenations is 2^{L_1-1}(L_1 is the length of R_1) and the length of the basic primitive 90/150 LHGCA is L_2, the required length of the 90/150 LHGCA is given by $L = L_2 2^{L_1-1}$. For example, consider a shrinking generator with the following component LFSRs: a selector register R_1 with length $L_1 = 3$ and the second register R_2 with length $L_2 = 4$. Then the period of the shrunken sequence is $(2^4 - 1)2^{(3-1)} = 60$. In order to generate the same sequence as shrunken sequence obtained by the shrinking generator, it needs the 90/150 LHGCA whose characteristic polynomial is $p(x)^N$, where $p(x)$ is a primitive polynomial with degree 4 and $2^{3-2} < N \leq 2^{3-1}$. But all cycles of 90/150 LHGCA synthesized are not equal to the period of the 90/150 LHGCA though the period of the 90/150 LHGCA is equal to the period of the shrunken sequence generated by the shrinking generator. Table 1 shows the configuration and cycle structure of 90/150 LHGCA whose characteristic polynomials are (x^4+x^3+1),$(x^4+x^3+1)^2$ and $(x^4+x^3+1)^4$. In Table 1, $a(b)$ means that the number of cycles with length b is a.

Table 1. Configuration and cycle structure of the 90/150 LHGCA synthesized

characteristic polynomial	configuration	cycle structure
x^4+x^3+1	1101	1(1), 1(15)
$(x^4+x^3+1)^2$	11000011	1(1), 1(15), 8(30)
$(x^4+x^3+1)^4$	1100001001000011	1(1),1(15),8(30),1088(60)

The period of the sequence generated by 90/150 LHGCA **C** with some initial state whose characteristic polynomial is of the form $p(x)^N$ is not always equal to the period of **C**. It means that the 90/150 LHGCA which is equivalent to any shrinking generator is not secure. To overcome this problem, we present a method which generates a new nonlinear pseudorandom sequence. Each cell position of 90/150 maximum-length LHGCA generates a pseudorandom sequence. In addition the phase shift is generally different between stages of a CA. The new sequence generator compose of two 90/150 maximum-length LHGCAs: a selector 90/150 maximum-length LHGCA $\mathbf{C_1}$ that produces a sequence used to decimate the sequences generated by the other 90/150 maximum-length LHGCA $\mathbf{C_2}$.

Let T_1 (resp. T_2) be the state transition matrix for a given m-cell (resp. n-cell) maximum-length 90/150 LHGCA and let u_0 (resp. v_0) be the initial state of T_1 (resp. T_2). Then we obtain a $(2^m - 1) \times m$ (resp. $(2^n - 1) \times n$) matrix A (resp. B) consisting of m (resp. n) independent pseudorandom sequences generated by T_1 (resp. T_2) as its columns. Here $gcd(2^m - 1, 2^n - 1) = 1$. Define a $(2^n - 1)(2^m - 1) \times (n+1)$ matrix $S = (s_{ij})$ as follows:

$$S = \left(\begin{array}{c|c} B & A_j \\ \vdots & \vdots \\ B & A_j \end{array} \right),$$

where A_j is the jth column of A.

Let S_I^* be the $2^{m-1}(2^n - 1) \times (n+1)$ matrix obtained from S by discarding the ith row of S if $s_{i,m+1} = 0$. Let S_I be the $2^{m-1}(2^n - 1) \times n$ matrix obtained by deleting the $(n+1)$th column of S_I^*. Then S_I is the following matrix:

$$S_I = \begin{pmatrix} k_0 & k_1 & \cdots & k_{n-1} \\ k_n & k_{n+1} & \cdots & k_{2n-1} \\ \vdots & \vdots & \ddots & \vdots \\ k_{(2^{m-1}(2^n-1)-1)n} & k_{(2^{m-1}(2^n-1)-1)n+1} & \cdots & k_{2^{m-1}(2^n-1)n-1} \end{pmatrix}$$

Definition 4.1. Let $\mathbf{K} = [k_0, k_1, \cdots, k_{2^{m-1}(2^n-1)n-1}]$ be a sequence obtained by S_I with period $2^{m-1}(2^n - 1)n$. We call $\mathbf{K}$ a $(2^{m-1}(2^n - 1), n)$ *shrunken interleaved sequence* (SI sequence).

Example 4.2. Consider a SI sequence generator with the following two component maximum-length 90/150 LHGCA $\mathbf{C_1}, \mathbf{C_2}$:
1. Let $\mathbf{C_1}$ be the maximum-length 90/150 LHGCA with length $m = 2$, rule vector $< 01 >$ and initial state $(0, 1)$. Then

$$T_1 = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix} \quad \text{and} \quad A = \begin{pmatrix} 0 & 1 \\ 1 & 1 \\ 1 & 0 \end{pmatrix}$$

2. Let $\mathbf{C_2}$ be the maximum-length 90/150 LHGCA with length $n = 3$, rule vector $< 011 >$ and initial state $(0, 0, 1)$. Then

$$T_2 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix} \quad \text{and} \quad B^t = \begin{pmatrix} 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 1 \end{pmatrix}$$

where B^t is the transpose of B. Since $A_2^t = \begin{pmatrix} 1 & 1 & 0 \end{pmatrix}$, we obtain the following matrix S:

$$S^t = \begin{pmatrix} 0&0&1&0&1&1&1&0&0&1&0&1&1&1&0&0&1&0&1&1&1 \\ 0&1&0&1&1&1&0&0&1&0&1&1&1&0&0&1&0&1&1&1&0 \\ 1&1&0&0&1&0&1&1&1&0&0&1&0&1&1&1&0&0&1&0&1 \\ 1&1&\underline{0}&1&1&\underline{0}&1&1&\underline{0}&1&1&\underline{0}&1&1&\underline{0}&1&1&\underline{0}&1&1&\underline{0} \end{pmatrix}$$

Hence we obtain the $(14, 3)$ SI sequence as the following:

$$\mathbf{K} = [001011010111101001100010110101011100111110]$$

Theorem 4.3. Let $\mathbf{C_1}$ be a maximum-length 90/150 LHGCA with length m and let $\mathbf{C_2}$ be a maximum-length 90/150 LHGCA with the n degree minimal polynomial $f(x)$, where $gcd(2^m - 1, 2^n - 1) = 1$. Then
(1) The minimal polynomial $m(x)$ of the SI sequence is of the following form:

$$m(x) = [f^*(x^n)]^N,$$

where $2^{m-2} < N \leq 2^{m-1}$ and $f^*(x)$ is the reciprocal of $f(x)$.

(2) The linear complexity LC of the SI sequence satisfies the following

$$2^{m-2}n^2 < LC \leq 2^{m-1}n^2$$

5 Conclusion

In this paper, we proposed a new method which generates nonlinear pseudorandom sequences using two maximum-length 90/150 LHGCA. The generator which generates these sequences is possible to overcome spatial weak points of the interleaved sequence generator proposed by Gong. Unlike the method proposed by Sabater et al., the SI sequence generator can generate nonlinear sequences whose cycle lengths are always same for a given initial state. The SI sequence obtained by our method has a larger period and a higher linear complexity than the shrunken sequence generated by LFSRs.

References

1. Coppersmith, D., Krawczyk, H., Mansour, Y.: The shrinking generator. In: Stinson, D.R. (ed.) CRYPTO 1993. LNCS, vol. 773, pp. 22–39. Springer, Heidelberg (1994)
2. Sabater, A.F., Martinez, D.G.: Modelling nonlinear sequence generators in terms of linear cellular automata. Applied Mathematical Modelling 31, 226–235 (2007)
3. Cho, S.J., Choi, U.S., Hwang, Y.H., Pyo, Y.S., Kim, H.D., Kim, K.S., Heo, S.H.: Computing phase shifts of maximum-length 90/150 cellular automata sequences. In: Sloot, P.M.A., Chopard, B., Hoekstra, A.G. (eds.) ACRI 2004. LNCS, vol. 3305, pp. 31–39. Springer, Heidelberg (2004)

4. Cho, S.J., Choi, U.S., Kim, H.D., Hwang, Y.H., Kim, J.G., Heo, S.H.: New synthesis of one-dimensional 90/150 linear hybrid group cellular automata. IEEE Trans. Comput-Aided Des. Integr. Circuits Syst. 26-9, 1720–1724 (2007)
5. Gong, G.: Theory and applications of q-ary interleaved sequences. IEEE Transaction on Information Theory 41-2, 400–411 (1995)
6. Das, A.K., Chaudhuri, P.P.: Vector space theoretic analysis of additive cellular automata and its application for pseudo-exhaustive test pattern generation. IEEE Trans. Comput. 42, 340–352 (1993)
7. Tsalides, P., York, T.A., Thanailakis, A.: Pseudorandom number generators for systems based on linear cellular automata. IEE Proc(Part E) Computers Digital Techniques 138, 241–249 (1991)
8. Sarkar, P.: Computing Shifts in 90/150 cellular automata sequences. Finite Fields Their Appl. 42, 340–352 (2003)