RESEARCH NOTE

The Linear Complexity of the Universal Logic Sequences

M. Modarres-Hashemi and M.R. Aref[1]

A universal logic sequence is a sequence which has good properties as a running key sequence. Although it has been inferred, based on the emperical results, that the linear complexity of a universal logic sequence is much near to its period, it has not been theoretically proved. In this paper, a proof is given based on the algebraic method.

INTRODUCTION

In a stream cipher system, a binary sequence $\tilde{Z}$, called a running key sequence, is added modulo two to binary plaintexts. It is clear that the properties of $\tilde{Z}$ are very important in the security of stream cipher systems against the cryptanalytic attacks. Large linear complexity is one of the most important desired properties of $\tilde{Z}$. Linear complexity of a sequence $\tilde{Z}$ is the length of the shortest LFSR (linear feedback shift register) which can be used to generate the sequence $\tilde{Z}$ [1,2]. There are several methods for obtaining the linear complexity bounds of output sequences of a running key generator. An effective and feasible approach is the algebric method introduced in [3,4]. This method will be used in our analysis. In the second section, a definition of a universal logic sequence [5] is given. In the third section, an upper bound for the linear complexity of universal logic sequence is proved [6] and it will be shown that the theoretical results agree with the empirical results. Finally, we conclude by providing some comments and remarks.

UNIVERSAL LOGIC SEQUENCES

Let $\tilde{a_1}, \ldots, \tilde{a_k}$ be binary, sequences generated by linear feedback shift registers LFSR(1), ..., LFSR(k) whose characteristic polynomials are primitives of degrees $L_1, L_2, \ldots, L_k$, respectively, where L_i's are pairwise relatively prime integers. Let n be an integer which is pairwise relatively prime to L_i for $i = 1, \ldots, k$ and $n \geq 2^{k+1}$. Let $\tilde{b}$ be a binary sequence generated by LFSR(n) of length n whose characteristic polynomial is primitive. Let $x_j = (a_{1j}, \ldots, a_{kj}, s_{j-1})$ be a binary $k+1$-tuple and α define a one to one mapping between binary $k+1$-tuples and a subset of $\{0, 1, \ldots, n-1\}$. Define the sequence $\tilde{S}$ by:

$$s_j = b_{j-\alpha(xj)}, \tag{1}$$

and the sequence $\tilde{Z}$ by:

$$z_j = \sum_{i=1}^{k} a_{ij} + s_{j-1}. \tag{2}$$

The sequence $\tilde{Z}$ is called a universal logic sequence [5]. In the other words, a multiplexer

[1] Department of Electrical Engineering, Isfahan University of Technology, Isfahan, I.R. Iran.

Scientia Iranica, Vol. 2, No. 1, © Sharif University of Technology, March 1995.

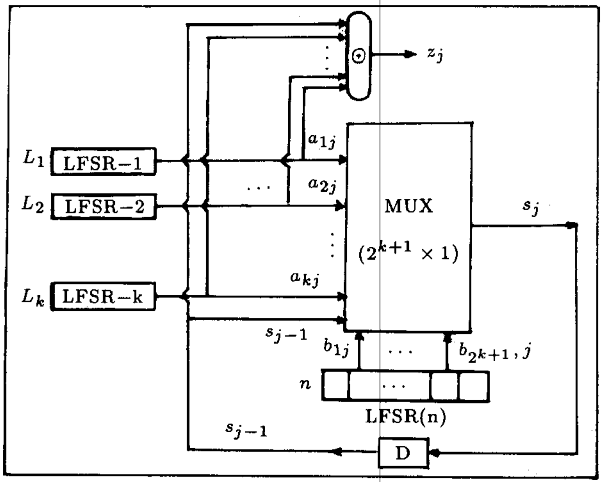

Figure 1. A universal logic system.

can have the role of α mapping and a universal logic sequence system which produces the universal logic sequences can be shown as in Figure 1.

The important properties of the universal logic sequences have been proved in [5]. It has been shown that the universal logic sequences have a period of length $T = (2^n - 1)(2^{L_1} - 1)\ldots(2^{L_k} - 1)$ provided, where necessary, that a few terms (at most $2^n - 1$ bits) at the start of some sequences are deleted. Also, it has been shown, based on the empirical results, that the linear complexity of a universal logic sequence is close to its period.

UPPER BOUND ON THE LINEAR COMPLEXITY

In this section, an upper bound on the linear complexity of the output sequences of the universal logic system is proved [6]. For simplicity, we assume that $k = 1$, $\alpha(00) = 0$, $\alpha(01) = 1$, $\alpha(10) = 2$ and $\alpha(11) = 3$. Figure 2 illustrates the universal logic system under these circumstances.

The system defines a finite state machine (FSM) whose output and next-state functions are given by the following equations:

$$z_j = a_j + s_{j-1} \,, \tag{3a}$$

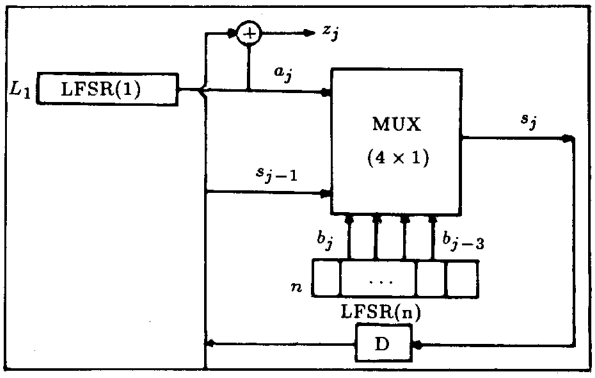

Figure 2. A universal logic system with $k = 1$.

$$s_j = (s_{j-1} + 1)(a_j + 1)b_j + s_{j-1}(a_j + 1)b_{j-1} + (s_{j-1} + 1)a_j b_{j-2} + s_{j-1}a_j b_{j-3} \,. \tag{3b}$$

An FSM is said to have finite input memory M, if M is the least integer such that the output digit at time j may be expressed as a function of the input variables at times $j - M, \ldots, j - 1, j$. Clearly, the FSM as described by Equation 3 has, in general, infinite input memory. If we assume that the input memory M is fixed at some value m, the feedback structure of the nonlinear combiner in Equation 3 can be converted into a feedforward structure of input memory m. Then, from the feedforward function f_m, it is possible to calculate the associated linear complexity $L^{(m)}$ of the output sequence.

Suppose we set $M = 0$, then Equation 3a implies that the output sequence $\tilde{Z}$ equals a_j, and, thus, has the linear complexity $L^{(0)} = L_1$.

If $M = 1$, then we have:

$$z_j = a_j + s_{j-1} \,, \tag{4a}$$

$$s_{j-1} = (a_j + 1)b_j \,, \tag{4b}$$

thus:

$$z_j = a_j + a_{j-1}b_{j-1} + b_{j-1} \,. \tag{5}$$

Let α in $GF(2^{L_1})$ and β in $GF(2^n)$ be roots of the primitive minimal polynomials $m_{\tilde{a}}(x)$ and $m_{\tilde{b}}(x)$, respectively. Then·$\tilde{a}^{-1}\tilde{b}^{-1}$ (related to $a_{j-1}b_{j-1}$) has an associated minimal polynomial whose root is $\alpha\beta$ in $GF(2^{L_1 n})$. Since

$ged(L_1, n) = 1$, then all of elements α, $\alpha\beta$, and β are distincts. Thus, the minimal polynomial of sequence $\tilde{Z}$ has L_1 roots in the form of α^{e1} with $w_2(e_1) = 1$, n roots in the form of β^{e2} with $w_2(e_2) = 1$, and $L_1 n$ roots in the form of $(\alpha\beta)^{e3}$ with $w_2(e_3) = 1$. Thus, the linear complexity of $\tilde{Z}$ is:

$$L^{(1)}(\tilde{z}) = L_1 + n + L_1 n \ . \tag{6}$$

For simplicity, we will assume the linear part of the nonlinear function f_m to be zero and we will redefine $L^{(0)} = 0$ and $L^{(1)} = L_1 n$.

With $M = 2$, Equation 3 results in:

$$\begin{aligned} z_j &= a_{j-1}b_{j-1} + a_{j-1}b_{j-1}a_{j-2}b_{j-2} \\ &+ a_{j-1}b_{j-2}b_{j-1} + a_{j-2}b_{j-2}b_{j-1} \\ &+ b_{j-1}b_{j-2} + a_{j-1}b_{j-2}a_{j-2} \ . \end{aligned} \tag{7}$$

Each part of z_j results in the roots of minimal polynomial of $\tilde{Z}$, which have the following forms:

$$\begin{cases} a_{j-1}b_{j-1} \rightarrow \alpha^{e1}\beta^{e2} \\ w_2(e_1) = 1, w_2(e_2) = 1 \ , \end{cases}$$

$$\begin{cases} a_{j-1}b_{j-1}a_{j-2}b_{j-2} \rightarrow \alpha^{e1}\beta^{e2} \\ w_2(e_1) \le 2, w_2(e_2) \le 2 \ , \end{cases}$$

$$\begin{cases} a_{j-1}b_{j-1}b_{j-2} \rightarrow \alpha^{e1}\beta^{e2} \\ w_2(e_1) = 1, w_2(e_2) \le 2 \ , \end{cases}$$

$$\begin{cases} a_{j-2}b_{j-2}b_{j-1} \rightarrow \alpha^{e1}\beta^{e2} \\ w_2(e_1) = 1, w_2(e_2) \le 2 \ , \end{cases}$$

$$\begin{cases} b_{j-1}b_{j-2} \rightarrow \beta^{e2} \\ w_2(e_2) \le 2 \ , \end{cases}$$

$$\begin{cases} a_{j-1}b_{j-2}a_{j-2} \rightarrow \alpha^{e1}\beta^{e2} \\ w_2(e_1) \le 2, w_2(e_2) = 1 \ . \end{cases}$$

Consequently, the minimal polynomial of $\tilde{Z}$ has roots in the form of $\alpha^{e1}\beta^{e2}$ in $GF(2^{L_1 n})$ with $w_2(e_1) \le 2$, $w_2(e_2) \le 2$ and roots in the form of β^{e2} in $GF(2^n)$ with $w_2(e_2) \le 2$. Then, the number of these roots will be:

$$\left[L_1 + \binom{L_1}{2}\right]\left[n + \binom{n}{2}\right] + n + \binom{n}{2} \ .$$

Thus:

$$L^{(2)}(\tilde{z}) \le \left[1 + L_1 + \binom{L_1}{2}\right]\left[n + \binom{n}{2}\right] \ . \tag{8}$$

Because of possible degeneracy in some of the roots, $L^{(2)}(\tilde{z})$ may be unequal to its upper bound. But, as it was proved in [3], the probability of any degeneracy happening goes to zero with increasing the lengths of shift registers.

In general, the set R_M of distinct elements in $GF(2^{L_1 n})$ which are possible roots of the minimal polynomial of $\tilde{Z}$, as produced by f_m, is given by:

$$\begin{aligned} R_M = \{\alpha^{e1}\beta^{e2} : & 0 \le w_2(e_1) \le M, \\ & 1 \le w_2(e_2) \le M\} \ . \end{aligned} \tag{9}$$

The number of these elements of R_M is $\sum\limits_{i=0}^{M}\binom{L_1}{i}\sum\limits_{j=1}^{M}\binom{n}{j}$.

Thus, we have:

$$L^{(M)}(\tilde{z}) \le \sum_{i=0}^{M}\binom{L_1}{i}\sum_{j=1}^{M}\binom{n}{j} \ . \tag{10}$$

The bound in Equation 10 increases with M and reaches its maximum value at $M = Max\{L_1, n\}$. Consequently,

$$L^{(M)}(\tilde{z}) \le 2^{L_1}(2^n - 1) \ , M \ge Max(L_1, n) \ . \tag{11}$$

In general, the finite memory of the system is much greater than $Max(L_1, n)$ and then we will have:

$$L(\tilde{z}) \le 2^{L_1}(2^n - 1) \ . \tag{12}$$

As described in the previous section, the period of output sequences of this system is $T = (2^{L_1}-1)(2^n-1)$, provided, where necessary, some bits (at most $2^n - 1$) at the start of some sequences are deleted. Thus, the intuitive upper bound of the linear complexity of the sequence is $T + 2^n - 1 = 2^{L_1}(2^n - 1)$. Therefore, the intuitive upper bound will be the real upper bound and since the probability of any degeneracy happening is nearly zero, then the upper bound in Equation 12 is reliable with near equality. The simulation results [5] confirm that the bound in Equation 12 is extremely tight. It is clear that, if k is increased, then the similar result can be given. In the other words, we have in general:

$$L(z) \leq (2^n - 1)(2^{L_1} - 1) \dots (2^{L_k} - 1) + 2^n - 1 \;, \tag{13}$$

with near equality.

CONCLUSION

Based on the given algebric proof in the third section, we concluded that the linear complexity of a universal logic sequence is close to its period provided that $(2^{L_1} - 1) \dots (2^{L_k} - 1) >> 1$. Otherwise, the linear complexity is near to $T + 2^n - 1$. This result had been already inferred and in this paper was proved. The algebraic method, used in this paper, can be employed in other systems with memory to reveal the linear complexity of their output sequences.

ACKNOWLEDGMENT

We are grateful to the Jihad-Daneshgahi Center of Isfahan University of Technology for its generous financial support.

REFERENCES

1. Beker, H. and Piper, F. *Cipher Systems: the Protection of Communication*, London (1982).

2. Massey, J.L. "Shift register synthesis and BCH decoding", *IEEE. Trans. Information Theory*, **15**(1), pp 122-127 (Jan. 1969).

3. Rueppel, R.A. *Analysis and Design of Stream Ciphers*, Springer-Verlag (1986).

4. Key, E.L. "An analysis of the structure and complexity of nonlinear binary sequence generators", *IEEE Trans. Information Theory*, **22**(11), pp 732-736 (Nov. 1976).

5. Dawson, E. and Goldburg, B. "Universal logic sequences", *Advances in Cryptology*, Auscrypt '90, Springer-Verlag, pp 426-432 (1990).

6. Modarres-Hashemi, M. "The design of stream cipher systems", Master's thesis, Department of Electrical Engineering, Isfahan University of Technology, Isfahan, I.R. Iran (1992).