The Filter-Combiner Model for Memoryless Synchronous Stream Ciphers

Palash Sarkar
Cryptology Research Centre
Applied Statistics Unit
Indian Statistical Institute
203, B.T. Road,
Kolkata 700035 India.
e-mail:palash@isical.ac.in

Abstract

We introduce a new model - the Filter-Combiner model - for memoryless synchronous stream ciphers. The new model combines the best features of the classical models for memoryless synchronous stream ciphers - the Nonlinear-Combiner model and the Nonlinear-Filter model. In particular, we show that the Filter-Combiner model provides key length optimal resistance to correlation attacks and eliminates LFSR based weaknesses of the NF model such as the Inversion attack and the Anderson leakage. Further, practical length sequences extracted from the Filter-Combiner model cannot be distinguished from true random sequences based on linear complexity test. We show how to realise the Filter-Combiner model using Boolean functions and cellular automata. In the process we point out an important security advantage of sequences obtained from cellular automata over sequences obtained from LFSRs.

Keywords : synchronous stream ciphers, linear feedback shift registers, cellular automata, nonlinear filter model, nonlinear combiner model, filter-combiner model.

1 Introduction

Stream ciphers are a basic cryptographic primitive. They are used widely for both defence communications and industrial applications. The underlying principle behind stream ciphers is the following. Let $m^{(t)}$, $t \geq 0$ be the sequence of message bits. Let $z^{(t)}$, $t \geq 0$ be a sequence of pseudorandom bits (also called the key sequence). Then $c^{(t)} = m^{(t)} \oplus z^{(t)}$, $t \geq 0$ is the sequence of cipher bits. Decryption is done by computing $c^{(t)} \oplus z^{(t)} = m^{(t)}$. The security of the system depends on the security of the pseudorandom bits $z^{(t)}$.

Stream ciphers are usually classified into two broad categories - synchronous and asynchronous stream ciphers. In synchronous stream ciphers the key bits do not depend on the message or cipher bits while in asynchronous stream ciphers the key bits depend on previous cipher and/or message bits. There are two classical models of memoryless synchoronous stream ciphers - the Nonlinear-Filter model and the Nonlinear-Combiner model. See [11, 13, 5] for more details on stream ciphers.

Both the standard models are built using Linear Feedback Shift Registers (LFSRs) and Boolean functions. In the Nonlinear-Combiner model exactly one bit sequence is extracted from one LFSR and all the bit sequences are combined using a Boolean function to generate the key sequence. In the Nonlinear-Filter model several bit sequences are generated from a single LFSR and these are then combined using a Boolean function to generate the key sequence.

Here we introduce the Filter-Combiner model for memoryless synchronous stream ciphers. This model is a combination of the Nonlinear-Filter and the Nonlinear-Combiner model. In the Filter-Combiner model there are several Linear Finite State Machines (LFSMs) each of which generate multiple bit sequences. These sequences are combined using a Boolean function to produce the key sequence. We show that the Filter-Combiner model has the following features.

1. Provides key length optimal resistance to correlation attack and hence overcomes the main disadvantage of the Nonlinear-Combiner model.

2. Eliminates Linear Feedback Shift Register (LFSR) based weaknesses of the Nonlinear-Filter model.

3. Practical sized key sequences extracted from the Filter-Combiner model cannot be distinguised from random strings based on linear complexity tests.

Thus the new model combines the best features of the previous two models. An important part in eliminating LFSR based weaknesses is the realisation of the LFSMs by Cellular Automata (CA). We identify the main problem of using LFSR in the Nonlinear-Filter model and show that this can be eliminated by using an important property of sequences obtained from CA. To the best of our knowledge, this is the first work to identify the important security advantage that can be obtained in replacing LFSR by CA.

We believe that as a consequence of our work, future models for practical stream ciphers will be based on the Filter-Combiner model rather than the Nonlinear-Filter or the Nonlinear-Combiner model.

2 Standard Models

$\mathbb{F}_2$ is the finite field of two elements and $\oplus$ denotes addition over $\mathbb{F}_2$ as well as the vector space $\mathbb{F}_2^l$ over $\mathbb{F}_2$. The common models of generating the key stream are built out of two kinds of primitives - linear finite state machines (LFSMs) and Boolean functions.

An l-bit LFSM $\mathcal{M}$ is a pair $(\mathbb{F}_2^l, M)$, where M is an $l \times l$ matrix. The internal state of $\mathcal{M}$ is described by an l-bit vector. The evolution of $\mathcal{M}$ over discrete time points $t \geq 0$ is described by a sequence of l-bit vectors $S^{(0)}, S^{(1)}, \ldots$, where $S^{(t+1)} = MS^{(t)}$. Thus only the vector $S^{(0)}$ (called the initial condition of $\mathcal{M}$) need to be specified for $\mathcal{M}$ to start operation. For $t \geq 0$, the vector $S^{(t)}$ will be written as $S^{(t)} = (s_1^{(t)}, \ldots, s_l^{(t)})$.

An n-variable Boolean function f is a map $f : \{0,1\}^n \rightarrow \{0,1\}$. The weight of a binary string s, denoted by $wt(s)$ is defined to be the number of ones in s.

2.1 Nonlinear-Filter (NF) Model

In this model one LFSM $\mathcal{M} = (\mathbb{F}_2^l, M)$ and one n-variable Boolean function $f(x_1, \ldots, x_n)$ are used. Let $S^{(0)}, S^{(1)}, \ldots$ be the sequence of n-bit vectors generated by $\mathcal{M}$. Then the key stream $z^{(t)}$ is obtained in the following manner.

$$z^{(t)} = f(s_{i_1}^{(t)}, \ldots, s_{i_n}^{(t)}), \qquad \text{for } t \geq 0, \tag{1}$$

where $S^{(t)} = (s_1^{(t)}, \ldots, s_l^{(t)})$, $i_1, \ldots, i_n \in \{1, \ldots, l\}$ and are distinct.

In this case, the secret key of the entire system is the initial condition $S^{(0)}$ of the LFSM giving rise to an l-bit secret key. We will call the nonlinear filter model the NF model.

2.2 Nonlinear-Combiner (NC) Model

In this model n LFSMs $\mathcal{M}_1 = (\mathbb{F}_2^{l_1}, M_1), \ldots, \mathcal{M}_n = (\mathbb{F}_2^{l_n}, M_n)$ and one n-variable Boolean function $f(x_1, \ldots, x_n)$ are used. Let $S_i^{(t)}$, $t \geq 0$ be the sequence of vectors generated by LFSM $\mathcal{M}_i$, $1 \leq i \leq n$. Further, let $S_i^{(t)} = (s_{i,1}^{(t)}, \ldots, s_{i,l_i}^{(t)})$. Then the key stream $z^{(t)}$ is generated in the following manner.

$$z^{(t)} = f(s_{1,1}^{(t)}, \ldots, s_{n,1}^{(t)}), \qquad \text{for } t \geq 0. \tag{2}$$

In this case the secret key of the entire system consists of the intial conditions $S_1^{(0)}, \ldots, S_n^{(0)}$ of all the LFSMs giving rise to an $(l_1 + \ldots + l_n)$-bit secret key. We will call the nonlinear combiner model the NC model.

3 Model Components

3.1 Linear Finite State Machines

Let $\mathcal{M} = (\mathbb{F}_2^l, M)$ be an LFSM generating the sequence of l-bit vectors $\mathcal{S} = S^{(0)}, S^{(1)}, \ldots$, where $S^{(t)} = (s_1^{(t)}, \ldots, s_l^{(t)})$. Let $p(x) = x^l \oplus a_{l-1}x^{l-1} \oplus \ldots \oplus a_1 x \oplus a_0$ be the characteristic polynomial for M. It is known that if $p(x)$ is primitive over $\mathbb{F}_2$, then the sequence $\mathcal{S}$ has period $2^l - 1$ (see [8]). Further, each of the sequences $s_i^{(t)}$, $1 \leq i \leq l$ also has period $2^l - 1$. This is the maximum possible period that can be obtained from a linear machine.

The most popular implementation of an LFSM is by a Linear Feedback Shift Register (LFSR). We will also consider implementation using Cellular Automata (CA). Below we briefly describe both LFSR and CA.

3.1.1 Linear Feedback Shift Register (LFSR)

For an LFSR, the matrix M is the companion matrix of $p(x)$ and as a result the following two relations hold.

$$\left.\begin{array}{rcll} s_{j+1}^{(t+1)} & = & s_j^{(t)} & t \geq 0, 1 \leq j < l, \\ s_1^{(t+1)} & = & \bigoplus_{i=0}^{l-1} a_{l-1+i} s_{i+1}^{(t)}. & \end{array}\right\} \tag{3}$$

Each of the sequences $s_i^{(t)}$, $1 \leq i \leq t$ satisfy a linear recurrence whose characteristic polynomial is $p(x)$ (see [8]). For $i \geq 1$ the sequence $s_{i+1}^{(t)}$ is obtained from the sequence $s_i^{(t)}$ by a single shift in the time domain. We record this as follows.

Fact 1 *The relative shift between two sequences $s_i^{(t)}$ and $s_j^{(t)}$ extracted from a single LFSR is $|i - j|$.*

An LFSR is simple to implement in hardware using an l-bit register and $l_1 = |\{i : a_i = 1, 0 \leq i \leq l-1\}|$ XOR gates. The initial condition $S^{(0)}$ is loaded into the register to start operation. The next state is determined by (3).

3.1.2 Cellular Automata (CA)

In case of CA the matrix M is a tridiagonal matrix. If the upper and lower subdiagonal entries of M are all 1 then the CA is called a 90/150 CA. We will only consider 90/150 CA. Let $c_1 \ldots c_l$ be the main diagonal entries of M. Then the following relations hold for the sequence of vectors $S^{(0)}, S^{(1)}, \ldots$.

$$\left.\begin{array}{rclclcll} s_1^{(t+1)} & = & & & c_1 s_1^{(t)} & \oplus & s_2^{(t)}, & \\ s_i^{(t+1)} & = & s_{i-1}^{(t)} & \oplus & c_i s_i^{(t)} & \oplus & s_{i+1}^{(t)} & \text{for } 2 \leq i \leq l-1, \\ s_l^{(t+1)} & = & s_{l-1}^{(t)} & \oplus & c_l s_l^{(t)}. & & & \end{array}\right\} \tag{4}$$

A CA can be implemented in hardware using an l-bit register and l XOR gates. The initial condition $S^{(0)}$ is loaded into the register for the CA to start operation. The next state of the CA is obtained using (4).

For $1 \leq i < j < l$, the shift between the sequences $s_i^{(t)}$ and $s_j^{(t)}$ depends upon the CA being used. A general algorithm to compute these shifts have been obtained in [15]. Observations suggest that these shifts can be exponential in l. In Section 7, we discuss this point in detail and conclude that this feature is an important security advantage of CA over LFSR.

3.2 Boolean Functions

An n-variable Boolean function $f(x_1, \ldots, x_n)$ can be represented by a unique multivariate polynomial over $\mathbb{F}_2$. Thus $f(x_1, \ldots, x_n)$ can be written as

$$f(x_1, \ldots, x_n) = \bigoplus_{(i_1, \ldots, i_n) \in \mathbb{F}_2^n} g(i_1, \ldots, i_n) x_1^{i_1} \ldots x_n^{i_n} \tag{5}$$

where $g(x_1, \ldots, x_n)$ is another n-variable Boolean function. The representation of f in (5) is called the *algebraic normal form* (ANF) of f. The *degree* of f, $deg(f)$ is defined to be $\max\{wt(i_1 \ldots i_n) : g(i_1, \ldots, i_n) = 1\}$.

The *weight* of an n-variable Boolean function f is denoted by $wt(f)$ and is defined as $wt(f) = |\{(i_1, \ldots, i_n) \in \mathbb{F}_2^n : f(i_1, \ldots, i_n) = 1\}|$. The function f is *balanced* if $wt(f) = 2^{n-1}$. The *distance* between two n-variable functions f and g is denoted by $d(f,g)$ and is defined as $d(f,g) = |\{(i_1, \ldots, i_n) \in \mathbb{F}_2^n : f(i_1, \ldots, i_n) \neq g(i_1, \ldots, i_n)\}|$. The probability that f and g are unequal is given by $Prob[f \neq g] = \frac{d(f,g)}{2^n}$.

The *Walsh transform* of f is an integer valued function $W_f : \{0,1\}^n \rightarrow [-2^n, 2^n]$ defined as $W_f(u) = \sum_{x \in \mathbb{F}_2^n} (1)^{f(x) \oplus \langle u, x \rangle}$, where $\langle u, x \rangle = u_1 x_1 \oplus \ldots \oplus u_n x_n$ is the inner product of u and x considered as vectors over $\mathbb{F}_2$.

The notion of correlation immune (CI) functions was introduced by Siegenthaler [17]. A characterization of correlation immunity in terms of Walsh transform was obtained in [20]. We present this characterization as our definition. An n-variable function f is said to be *correlation immune of order m (m-CI)* if $W_f(u) = 0$ for all $1 \leq wt(u) \leq m$. A balanced m-CI function is said to be *m-resilient.*

For $u \in \mathbb{F}_2^n$, let $\lambda_u(x_1, \ldots, x_n)$ be a linear function defined as $\lambda_u(x_1, \ldots, x_n) = \langle u, (x_1, \ldots, x_n) \rangle$. Then $W_f(u) = 2^n - 2 \times d(f, \lambda_u)$. Let $A_n = \{\lambda_u \oplus b : u \in \mathbb{F}_2^n, b \in \{0,1\}\}$ be the set of all n-variable affine functions. The nonlinearity of f is defined to be $nl(f) = \min_{g \in A_n} d(f,g)$. Equivalently, this can be written as $nl(f) = 2^{n-1} - \frac{1}{2} \max_{u \in \mathbb{F}_2^n} |W_f(u)|$. Any function $g \in A_n$ such that $d(f,g) = nl(g)$ is said to be *a best affine approximation* of f.

4 Correlation Attacks

The currently known most powerful class of attacks on both the NF and the NC model is the class of correlation attacks. We describe the basic idea of a correlation attack with reference to the NC model.

In the NC model, n input bit sequences are combined by an n-variable Boolean function $f(x_1, \ldots, x_n)$ to produce the key sequence $z^{(t)}$. For notational convenience we will denote the n input sequence to f by $x_1^{(t)}, \ldots, x_n^{(t)}$. The input sequence $x_i^{(t)}$ is produced by an LFSM of length l_i. For $t \geq 0$, we have

$$z^{(t)} \quad = \quad f(x_1^{(t)}, \ldots, x_n^{(t)}). \tag{6}$$

Suppose $W_f(u) \neq 0$ for some $u \in \mathbb{F}_2^n$, with $wt(u) = 1$. Let i be such that $u_i = 1$ and for $j \neq i$, $u_j = 0$. In this situation first order correlation attacks are applicable. The function $\lambda_u(x_1, \ldots, x_n)$ is equal to x_i. The idea is to use the bias $\beta_u = |Prob(\lambda_u = f) - \frac{1}{2}| = \frac{|W_f(u)|}{2^{n+1}}$ to estimate the sequence $x_i^{(t)}$ from the sequence $z^{(t)}$ (or even from the cipher sequence $c^{(t)}$). This was originally proposed by Siegenthaler [18]. Recently a great deal of work has been done in this area (see [3, 4]).

If $\beta_u = 0$ for all u with $wt(u) = 1$, then it is not possible to directly estimate any $x_i^{(t)}$ from $z^{(t)}$. In this case a higher order attack can be carried out as follows. Suppose f is m-CI but not $(m+1)$-CI. Then there exists $u \in \mathbb{F}_2^n$ with $wt(u) = m+1$ such that $W_f(u) \neq 0$. Let $i_1, \ldots, i_{m+1}$ be such that $u_{i_1} = \ldots = u_{i_{m+1}} = 1$ and $u_j = 0$ for $j \notin \{i_1, \ldots, i_{m+1}\}$. Define β_u as before. Then the bias β_u is used to estimate the sequence $y^{(t)} = x_{i_1}^{(t)} \oplus \ldots \oplus x_{i_{m+1}}^{(t)}$. The individual sequences $x_{i_1}^{(t)}, \ldots, x_{i_{m+1}}^{(t)}$ can be obtained from $y^{(t)}$ by solving a system of linear equations.

Define $L = l_{i_1} + \ldots + l_{i_{m+1}}$. The linear complexity (see Section 8) of the sequence $x_{i_1}^{(t)} \oplus \ldots \oplus x_{i_{m+1}}^{(t)}$ is L (see Lemma 6.1). Let N be the number of key bits required to successfully carry out the attack. The parameter N depends on the bias β_u and the length L. Most work on correlation attacks present only simulation studies. Recently, some theoretical analysis has been done in [4, 3]. We briefly describe the analysis from [4].

$$N \simeq \frac{1}{4} \cdot (2kt!\ln 2)^{\frac{1}{t}} \cdot \beta_u^{-2} \cdot 2^{\frac{L-k}{t}}, \tag{7}$$

where k and t are algorithm parameters. The attack stores certain parity check relations and consists of a precomputation phase and a decoding phase. The complexity of the precomputation phase is approximately $N^{\lceil (t-1)/2 \rceil}$ and requires $N^{\lfloor (t-1)/2 \rfloor}$ memory. The number of parity check relations that need to be stored is roughly $\frac{N^t}{t!} \cdot 2^{-(L-k)}$ and the decoding complexity is 2^k times the number of parity checks. Thus the attack becomes infeasible if either β_u is sufficiently close to 0 or L is sufficiently large.

4.1 Resistance of the NC Model to Correlation Attacks

For $u \in \{0,1\}^n$, define $l(u) = u_1 l_1 + \cdots + u_n l_n$. For an m-resilient function define

$$\alpha_f = \min_{W_f(u) \neq 0, wt(u) = m+1} l(u).$$

The lengths of the LFSMs in the NC model are $l_1, \ldots, l_n$ and the secret key length is $l = l_1 + \cdots + l_n$. However, the complexity of a correlation attack depends on the parameter α_f which is less than l. Thus we obtain the following fact.

Fact 2 *The resistance to correlation attack provided by the NC model is sub-optimal in the secret key length.*

Remark : *A consequence of this fact is that to obtain a desired level of security one has to choose a significantly longer secret key. This is clearly a major shortcoming of the NC model.*

5 The Filter-Combiner (FC) Model

In this section we present our new model - the Filter-Combiner Model. We will call this model the FC model. We present a formal description of the model.

Components of the model : An n-variable Boolean function $f(x_1, \ldots, x_n)$ and k $(1 < k < n)$ LFSMs $\mathcal{M}_1 = (\mathbb{F}_2^{l_1}, M_1), \ldots, \mathcal{M}_k = (\mathbb{F}_2^{l_k}, M_k)$. The characteristic polynomials of $M_1, \ldots, M_k$ are chosen to be primitive and $l_1, \ldots, l_k$ are chosen to be all distinct.

Keystream generation : LFSM $\mathcal{M}_j$ produces l_j bit sequences. Out of these i_j bit sequences $y_{j,1}, \ldots, y_{j,i_j}$ are chosen, where $i_1 + \ldots + i_k = n$. The key stream $z^{(t)}$ is generated as follows.

$$z^{(t)} = f(y_{1,1}^{(t)}, \ldots, y_{1,i_1}^{(t)}, y_{2,1}^{(t)}, \ldots, y_{2,i_2}^{(t)}, \ldots, y_{k,1}^{(t)}, \ldots, y_{k,i_k}^{(t)}) \quad \text{for } t \geq 0. \tag{8}$$

Constraints on the model : Denote the sequences $y_{1,1}^{(t)}, \ldots, y_{1,i_1}^{(t)}, y_{2,1}^{(t)}, \ldots, y_{2,i_2}^{(t)}, \ldots, y_{k,1}^{(t)}, \ldots, y_{k,i_k}^{(t)}$ by $x_1^{(t)}, \ldots, x_n^{(t)}$, where $x_1, \ldots, x_n$ are the input variables to the function f. For each variable x_i define $FSM(x_i) = j$ such that the sequence $x_i^{(t)}$ is one of the sequences $y_{j,1}^{(t)}, \ldots, y_{j,i_j}^{(t)}$. The following conditions must hold on the model.

1. If $W_f(u) \neq 0$, then $\{FSM(x_{i_1}), \ldots, FSM(x_{i_p})\} = \{1, \ldots, k\}$, where $u_{i_1} = \ldots = u_{i_p} = 1$ and $u_j = 0$ for $j \notin \{i_1, \ldots, i_p\}$.
2. For $1 \leq j \leq k$, i_j bit sequences are extracted from LFSM $\mathcal{M}_j$ where $i_1 + \cdots + i_k = n$. Let $n = qk + r = r(q+1) + (k-r)q$, where $0 \leq r < k$. We require $i_1 = \ldots = i_r = \lceil \frac{n}{k} \rceil$ and $i_{r+1} = \ldots = i_k = \lfloor \frac{n}{k} \rfloor$.
3. $i_j \leq \log_2 l_j$ for $1 \leq j \leq k$.
4. If $FSM(x_i) = FSM(x_j) = p$, then the shift between the sequences $x_i^{(t)}$ and $x_j^{(t)}$ must be in the range $[\frac{2^{l_p}}{i_p} - \epsilon_p, \frac{2^{l_p}}{i_p} + \epsilon_p]$ for some $\epsilon_p \ll 2^{l_p}$.
5. The maximum length of a message that should be encrypted by the system is $\min_{1 \leq j \leq k} (\frac{2^{l_j}}{i_j} - \epsilon_j)$.

Proposition 5.1 *Let f be m-resilient and suppose Constraint 1 holds. Then $k \leq m+1$.*

Proof : Suppose $k > m+1$ and $u \in \mathbb{F}_2^n$ be such that $wt(u) = m+1$ and $W_f(u) \neq 0$. Let $u_{i_1} = \ldots = u_{i_{m+1}} = 1$ and $u_j = 0$ for $j \notin \{i_1, \ldots, i_{m+1}\}$. Then $|\{FSM(x_{i_1}), \ldots, FSM(x_{i_{m+1}})\}| \leq m+1 < k$. Hence Constraint 1 is violated. ∎

Proposition 5.2 *Suppose Constraint 2 holds. Then Constraint 3 holds if and only if*

$$\begin{aligned} l_j &\geq 2^{\lceil (n/k) \rceil} && \text{if } 1 \leq j \leq r \\ &\geq 2^{\lfloor (n/k) \rfloor} && \text{if } r+1 \leq j \leq k. \end{aligned} \tag{9}$$

Proposition 5.3 *Suppose Constraints 3 and 4 hold. Then the shift between the sequences* $x_i^{(t)}$ *and* $x_j^{(t)}$ *is at least* $2^{l_p - \log_2 \log_2 l_p} - \epsilon_p$*, where* $FSM(x_i) = FSM(x_j) = p$.

Remark : 1. Proposition 5.3 assures us that the shift between any two sequences obtained from the same LFSM is "exponential" in the length of the LFSM.
2. Constraint 5 guarantees that no bit generated by any LFSM is used more than once.
3. Constraint 4 is to be contrasted with Fact 1 in Section 3.1.1. An immediate consequence is that Constraint 4 cannot be realised using LFSR. In Section 9.2 we show that Constraint 4 can be achieved using CA.

6 Resistance to Correlation Attacks

In this section we show that the resistance to correlation attacks provided by the FC model is optimal in the key length. This is a direct consequence of Constraint 1 in the design criteria. We first prove the following result.

Lemma 6.1 *Let* $x_1^{(t)}$ *and* $x_2^{(t)}$ *be two linear recurring sequences whose characteristic polynomials are* $p_1(x)$ *and* $p_2(x)$ *of degrees* d_1 *and* d_2 *respectively. Assume that* $p_1(x)$ *and* $p_2(x)$ *are both primitive. Then the linear complexity of the sequence* $x^{(t)} = x_1^{(t)} \oplus x_2^{(t)}$ *is* $d_1 + d_2$.

Proof : Let $\alpha_1, \ldots, \alpha_{d_1}$ be the roots of $p_1(x)$ and $\beta_1, \ldots, \beta_{d_2}$ be the roots of $p_2(x)$. We can write (see for example [14])

$$\begin{aligned} x_1^{(t)} &= A_1\alpha_1^t \oplus \ldots \oplus A_{d_1}\alpha_{d_1}^t, \\ x_2^{(t)} &= B_1\beta_1^t \oplus \ldots \oplus B_{d_2}\beta_{d_2}^t. \end{aligned} \tag{10}$$

Here $A_1, \ldots, A_{d_1}$ (resp. $B_1, \ldots, B_{d_2}$) are constants determined solely by the initial d_1 (resp. d_2) bits of $x_1^{(t)}$ (resp. $x_2^{(t)}$). Thus we can write

$$x^{(t)} = A_1\alpha_1^t \oplus \ldots \oplus A_{d_1}\alpha_{d_1}^t \oplus B_1\beta_1^t \oplus \ldots \oplus B_{d_2}\beta_{d_2}^t \tag{11}$$

The roots $\alpha_1, \ldots, \alpha_{d_1}$ and $\beta_1, \ldots, \beta_{d_2}$ are elements of the field $GF(2^{\mathsf{lcm}(d_1,d_2)})$. Since $p_1(x)$ and $p_2(x)$ are primitive it is not difficult to see that $\{\alpha_1, \ldots, \alpha_{d_1}\} \cap \{\beta_1, \ldots, \beta_{d_2}\} = \emptyset$. Hence using (11) it follows that the linear complexity of $x^{(t)}$ is $d_1 + d_2$ (see [14]). ∎

Theorem 6.1 *The FC model provides key length optimal resistance to correlation attacks.*

Proof : Let $f(x_1, \ldots, x_n)$ be the m-resilient Boolean function which combines the input bit sequences. Let $x_{i_1}, \ldots, x_{i_{m+1}}$ be such that $W_f(u) \neq 0$, where $u_{i_1} = \ldots = u_{i_{m+1}} = 1$ and for $j \notin \{i_1, \ldots, i_{m+1}\}$, $u_j = 0$. Using Constraint 1 this implies $\{FSM(x_{i_1}), \ldots, FSM(x_{i_{m+1}})\} = \{1, \ldots, k\}$.

Let the characteristic polynomials of the LFSMs be $p_1(x), \ldots, p_k(x)$ of degrees $l_1, \ldots, l_k$. Let the roots of the polynomial $p_i(x)$ be $\alpha_{i,j}$, $1 \leq j \leq l_i$ in the field $GF(2^a)$, where $a = \mathsf{lcm}(l_1, \ldots, l_k)$. Then any bit sequence $y^{(t)}$ obtained from LFSM $\mathcal{M}_i$ can be written as $y^{(t)} = A_1\alpha_{i,1}^t \oplus \ldots \oplus A_{l_i}\alpha_{i,l_i}^t$, where $A_1, \ldots, A_{l_i}$ are constants dependent on the initial l_i bits of the sequence $y^{(t)}$.

Let $x^{(t)} = x_{i_1} \oplus \ldots \oplus x_{i_{m+1}}$. The characteristics polynomials $p_1(x), \ldots, p_k(x)$ are primitive by model criteria. Hence the roots $\alpha_{i,j}$ are all distinct. Using the fact that $\{FSM(x_{i_1}), \ldots, FSM(x_{i_{m+1}})\} = \{1, \ldots, k\}$, we can write

$$x^{(t)} = \bigoplus_{i=1}^{k} \bigoplus_{j=1}^{l_i} C_{i,j} \alpha_{i,j}. \tag{12}$$

Here $C_{i,j} \in GF(2^a)$ are constants and are completely determined by the bits

$$x_{i_1}^{(1)}, \ldots, x_{i_1}^{(l_1)}, \ldots, x_{i_{m+1}}^{(1)}, \ldots, x_{i_{m+1}}^{l_{m+1}}.$$

Hence the linear complexity of the sequence $x^{(t)}$ is $L = l_1 + \ldots + l_k$. In other words, if we want to obtain $x^{(t)}$ by a linear recurrence, then the degree of the characteristic polynomial of the recurrence is at least L. Thus in any correlation attack, the number of key bits required for a successful attack depends on L. Since the length of the secret key is also L, the resistance to correlation attacks is optimal in the key length. ■

Remark : *Theorem 6.1 shows that with respect to correlation attacks the FC model is superior to the NC model (see Fact 2 in Section 4.1).*

7 Eliminating LFSR Based Weaknesses

In traditional implementation of the NF model, LFSRs are used to implement the LFSMs. This means that more than one sequence is extracted from a single LFSR. Extracting more than one sequence from a single LFSR makes the system vulnerable to certain kinds of attacks. These attacks are based on Fact 1 of Section 3.1.1.

Anderson Leakage : Suppose an LFSR of length l and an n-variable function $f(x_1, \ldots, x_n)$ is used to implement the NF model. Let the l sequences of the LFSR be $s_i^{(t)}$, for $1 \leq i \leq t$. Suppose the sequence $x_j^{(t)} = s_{i_j}^{(t)}$ for $1 \leq j \leq n$. Then the relative shift between two sequences $x_{j_1}^{(t)}$ and $x_{j_2}^{(t)}$ is $|i_{j_1} - i_{j_2}| \leq l$. Since the period of any of the sequences $x_j^{(t)}$ is $2^l - 1$, the relative shifts between the sequences are comparatively small. Thus the inputs to the function f are obtained from the same sequence with small shifts. This results in information leakage from the input to the output even if the function f is resilient. No general algorithm is known which can exploit this attack. However, Anderson [1] has provided convincing evidence of the leakage phenomenon.

Inversion Attack : Golic, Clark and Dawson [7] have shown that the chosen tap positions must form a proper difference set or else the system is vulnerable to an inversion attack. The basic weakness exploited is again the fact that the relative shift in the sequences extracted from two tap positions is equal to the distance between them.

In the FC model multiple sequences are extracted from a single LFSM. Thus we have to consider the applicability of the above attacks to the FC model. This is where we use Proposition 5.3 and Constraint 5. Proposition 5.3 states that the relative shift in the sequences extracted from two tap positions must be "exponential" in the length of the LFSM. Constraint 5 states that the maximum length of the message that should be enciphered by the system is less than the minimum shift between any two sequences obtained from a single LFSM. Thus any bit of an extracted sequence is used at most once to generate the pseudo random key stream. Thus the above mentioned attacks are not applicable to the FC model. We record this fact as follows.

Fact 3 *Anderson leakage and Inversion attacks are not applicable to the FC model.*

Remark : *Combining the results of Sections 6 and 7 we see that the FC model is superior to both the NF and the NC models.*

8 Linear Complexity

Given a bit sequence, a parameter of fundamental importance is its *linear complexity* which is defined to be the length of a minimum length LFSR which can generate the sequence. The linear complexity of a bit sequence generated by an LFSM $\mathcal{M} = (\mathbb{F}_2^l, M)$ is l. Given an arbitrary bit sequence, its linear complexity can be determined using the Berlekamp-Massey algorithm [10]. We record some facts about linear complexity.

Fact 4 *The expected linear complexity of a random string of length L is $\lfloor \frac{L}{2} \rfloor$ (see [11]).*

Fact 5 *In case of the NC model the linear complexity can be determined using a result of Rueppel and Staffelbach [14]. Suppose the lengths of the LFSMs are $l_1, \ldots, l_n$ and the sequences are combined using an n-variable Boolean function $f(x_1, \ldots, x_n)$ whose ANF is $\bigoplus_{(i_1,\ldots,i_n)\in\mathbb{F}_2^n} g(i_1, \ldots, i_n)x_1^{i_1} \ldots x_n^{i_n}$. The linear complexity of $z^{(t)} = x_1^{(t)} \oplus \ldots x_n^{(t)}$ is $\leq \sum_{(i_1,\ldots,i_n)\in\mathbb{F}_2^n} g(i_1, \ldots, i_n)l_1^{i_1} \ldots l_n^{i_n}$ where equality is achieved if the lengths l_i, $1 \leq i \leq n$ are all distinct.*

From Fact 4 we have an upper bound on the maximum possible linear complexity in the NC model. Note that this upper bound is substantially less than the value $2^{l_1+\cdots+l_n}$.

For the NF model, it is more difficult to compute the linear complexity of the generated entire key sequence. Let l be the secret key length. Rueppel [13] has shown that for a class of Boolean functions it is possible to generate a key sequence of guaranteed linear complexity at least $\binom{l}{\lfloor \frac{l}{2} \rfloor}$. However, the functions in this class do not necessarily satisfy the other requirements of high nonlinearity, high correlation immunity (see [5]).

In case of the FC model, it is difficult to compute the linear complexity of the entire sequence. Instead we conducted several experiments with different set ups. We describe two set ups.

1. System 1 used 3 CA of lengths 15,16 and 17 bits whose characteristic polynomials are primitive. Two sequences were extracted from the first two CA and three sequences were extracted from the third CA satisfying Constraint 2 of the FC model. A 7-variable, resiliency 3, degree 3, nonlinearity 48 function was used to combine the extracted sequences satisfying Constraint 1 of the FC model. The secret key length of the system is 48 bits.

2. System 2 used 3 CA of lengths 16,17 and 18 bits with primitive characteristic polynomials. Two sequences were extracted from the first CA and three sequences each were extracted from the last two CA satisfying Constraint 2 of the FC model. A 8-variable, resiliency 4, degree 3, nonlinearity 96 function was used to combine the extracted sequences satisfying Constraint 1 of the FC model. The secret key length is 51 bits.

In each of the above two cases we generated key sequences of lengths L equal to $2^{10}, 2^{11}, 2^{12}, 2^{13}, 2^{14}, 2^{15}$ from randomly chosen secret keys (initial configurations of the CA involved). The linear complexity was obtained in each case using the Berlekamp-Massey algorithm as described in [11]. In all our experiments we obtained linear complexity very close to $\frac{L}{2}$, which is as expected for a random bit sequence.

The secret key sizes of 48 and 51 bits are not sufficient in practical stream ciphers. In practical situations the secret key length would be at least 128 and the generated key sequence would be at most 2^{30} between two key changes. It would have been better to test the linear complexity of key sequences of length around 2^{30}. However, the Berlekamp-Massey algorithm requires L^2 operations to compute the linear complexity of a key sequence of length L (see [11]). Thus computing the linear complexity of a sequence of length 2^{30} would require around 2^{60} operations. This makes it impractical to run such experiments. On the other hand, our experiments confirm the following fact.

Fact 6 *If the length L of the extracted key sequence of the FC model is small compared to 2^l (where l is the secret key length), then the linear complexity of the sequence cannot be distinguished from the linear complexity of a random string.*

9 Realization of the FC Model

There are four main constraints on the model that must be satisfied to build a particular system. The first concerns the Boolean function and the connection of the Boolean function to the LFSMs. The second to fourth concerns the implementation of the LFSMs. We describe methods for satisfying these constraints.

9.1 Satisfying Constraints 1 and 2

For $u \in \{0,1\}^n$, define $A_u = \{i_1, \ldots, i_p\}$, where $u_{i_1} = \ldots = u_{i_p} = 1$ and $u_j \neq 0$ for $j \notin \{i_1, \ldots, i_p\}$.

Constraint 1 asks the following question. Can we construct an n-variable, m-resilient Boolean function such that the variables $x_1, \ldots, x_n$ can be partitioned into k sets $A_1, \ldots, A_k$, where $A_i \cap A_u \neq \emptyset$ for $1 \leq i \leq k$ and for each $u \in \mathbb{F}_2^n$ with $W_f(u) \neq 0$? We now describe a simple solution to this problem. We begin with the following simple result.

Proposition 9.1 *Let $f(x_1, \ldots, x_n)$ be a Boolean function of the form*

$$f(x_1, \ldots, x_n) = x_1 \oplus \ldots \oplus x_k \oplus g(x_{k+1}, \ldots, x_n).$$

If $W_f(u) \neq 0$, then $u_1 = \ldots = u_k = 1$.

Proof : Suppose that for some $j \in \{1, \ldots, k\}$, we have $u_j = 0$. Then the variable x_j does not occur in the linear function $l_u(x_1, \ldots, s_n) = \langle u, (x_1, \ldots, x_n) \rangle$. Thus the function

$$f(x_1, \ldots, x_n) \oplus l_u(x_1, \ldots, x_n) = x_j \oplus h(x_1, \ldots, x_{j-1}, x_{j+1}, \ldots, x_n),$$

for some function h. Hence $f \oplus l$ is a balanced function and so $W_f(u) = 0$. ■

We can now describe our construction. Let f be an n-variable, m-resilient function of the form

$$f(x_1, \ldots, x_n) = x_1 \oplus \ldots \oplus x_k \oplus g(x_{k+1}, \ldots, x_n). \tag{13}$$

Construction : Construct the sets A_j $(1 \leq j \leq k)$ as follows.

1. Put element j in A_j.

2. Distribute the elements $k+1,\ldots,n$ to the sets A_j such that $|A_j| = \lceil (n/k) \rceil$ if $1 \leq j \leq r$ and $|A_j| = \lfloor (n/k) \rfloor$ if $r+1 \leq j \leq k$. Note that this can easily be done.

This construction ensures that for any u such that $W_f(u) \neq 0$, we have $A_j \cap A_u \neq \emptyset$ for all $1 \leq j \leq k$. Thus Constraint 1 is satisfied. We extract the input sequences $x_{i_1}^{(t)},\ldots,x_{i_j}^{(t)}$ from LFSM $\mathcal{M}_i$. By construction, each $|A_j|$ is either $\lfloor (n/k) \rfloor$ or $\lceil (n/k) \rceil$. Hence Constraint 2 is also satisfied.

We briefly comment on the availability of n-variable, m-resilient Boolean functions of the form described in (13). The construction of functions in the form (13) was first described by Siegenthaler [17]. Later work [2, 9] have investigated this construction. Note that a necessary condition is that $k \leq m$. Under this condition it is always possible to get n-variable, m-resilient functions in the form (13). In fact, for certain values of the parameters n and m, it is also possible to get functions in the form (13) which achieve the best possible trade-off among resiliency, degree and nonlinearity (see [16]).

Let us now turn the question around and consider the following problem. We first describe Constraint 1 formally as a decision problem.

Problem : CONS1

Instance : A family $\mathcal{F} = \{A_u \subset \{1,\ldots,n\} : u \in \mathbb{F}_2^n, W_f(u) \neq 0\}$, where f is an n-variable, m-resilient Boolean function and a positive integer k such that $2 \leq k \leq m$.

Question : Is there a k-partition $A_1,\ldots,A_k$ of $\{1,\ldots,n\}$ such that $A_u \cap A_i \neq \emptyset$, for every $A_u \in \mathcal{F}$?

Even though Constraint 1 has been described as a decision problem we are really interested in an actual k-partition $A_1,\ldots,A_k$. If we are able to obtain such a partition, then for each A_i we can assign the variables $x_{j_1},\ldots,x_{j_i}$ to the FSM i. Solving CONS1 does not seem to be easy in general. We describe a modified version of CONS1 which is easily proved to be NP-complete.

Problem : Generalized Set Splitting (GSS)

Instance : A family $\mathcal{F} = \{T \subset \{1,\ldots,n\} : |T| \geq m\}$, and a positive integer k with $2 \leq k < n$ and $k \leq m$.

Question : Is there a k-partition $A_1,\ldots,A_k$ of $\{1,\ldots,n\}$ such that $A_i \cap T \neq \emptyset$ for $1 \leq i \leq k$ and for each $T \in \mathcal{F}$?

The GSS problem is a generalized version of the set splitting problem (see [6, page 221]) and is easily proved to be NP-complete. This does not prove the CONS1 problem to be NP-complete, since in CONS1 the family $\mathcal{F}$ is obtained from the nonzero points of the Walsh transform of a Boolean function whereas in GSS the family $\mathcal{F}$ is an arbitrary collection. Thus it may be possible to use algebraic properties of the Walsh transform of f to solve CONS1 easily even though GSS is NP-complete. However, the NP-completeness of GSS is very strong evidence of the intractibility of solving CONS1.

Remark : Given an n-variable, m-resilient Boolean function it might not be possible to satisfy Constraint 1, i.e., there might not be a proper partition or it might be computationally intractible to find a proper partition. However, we have shown that for a large class of cryptographically significant functions it is always possible to satisfy Constraint 1. Also there are examples of functions not of the type (13) for which it is possible to satisfy Constraint 1. Further research will throw more light on the set of functions which satisfy Constraint 1.

9.2 Satisfying Constraints 3 and 4

Constraint 4 depends on the properties of the $\mathcal{M}_1,\ldots,\mathcal{M}_k$. Suppose sequences $x_{j_1}^{(t)},\ldots,x_{j_i}^{(t)}$ are extracted from $\mathcal{M}_i$. We require the relative shift between two sequences $x_{j_k}^{(t)}$ and $x_{j_p}^{(t)}$ to be exponential in l_i.

We consider the use of CA to implement the LFSMs to satisfy Constraint 4. (Note that from Fact 1 in Section 3.1.1 it follows that LFSRs cannot be used to satisfy Constraint 4.) To do this we need to do the following two things.

1. Given an primitive polynomial $p(x)$ of degree l, we need to construct a 90/150 CA which realizes $\mathcal{M} = (\mathbb{F}_2^l, M)$ such that the characteristic polynomial of M is $p(x)$.
2. Given a 90/150 CA producing l-bit state vectors $S^{(t)} = (s_1^{(t)}, \ldots, s_l^{(t)})$, we need to compute the relative shift between any two sequences $s_i^{(t)}$ and $s_j^{(t)}$.

Based on a result by Mesirov and Sweet [12], an efficient solution to the first problem has been presented in [19]. Further, in [15] an algorithm to solve the second problem has been presented.

Experimental results based on the algorithm of [15] show the following Fact.

Fact 7 *For a 90/150 CA of length l with primitive characteristic polynomial, it is possible to obtain at least p ($\log_2 l \leq p < l$) positions such that the relative shift between any two pair of these p positions is in the range $[\frac{2^l}{p} - \epsilon, \frac{2^l}{p} + \epsilon]$ for some $\epsilon \ll 2^l$.*

Remark : Fact 7 should be contrasted with Fact 1. This underlines the enhanced security features of CA sequences over LFSR sequences.

It immediately follows from Fact 7 that Constraints 3 and 4 can be satisfied using CA. For the purpose of illustration we present a concrete example of a 24-cell 90/150 CA.

Example : Consider a 24 cell CA. Choose $p(x) = x^{24} \oplus x^4 \oplus x^3 \oplus x \oplus 1$ to be the characteristic polynomial of the CA. The polynomial $p(x)$ is primitive (see [11, page 161]). We wish to obtain a 90/150 CA whose characteristic polynomial is $p(x)$. It is enough to obtain the main diagonal entries of the state transition matrix (see Section 3.1.2). The main diagonal entries can be described by a 24-bit string. Using the algorithm of [19], we obtain this string to be 110100111001001111001011. Let the sequences obtained from the 24 cells of the CA be denoted by $s_1^{(t)}, \ldots, s_{24}^{(t)}$. Define integers $b_1 = 0, b_2, \ldots, b_{24}$, such that $s_1^{(t)} = s_i^{(t+b_i)}$ for all $t \geq 0$ and $1 \leq i \leq 24$. Using the algorithm of [15], we obtain the values $(b_1, \ldots, b_{24})$ to be equal to

$$(0, 11662498, 16777213, 3837988, 12949649, 13910896, 13911015, 959496,\\ 3720499, 15512414, 9453076, 13780753, 15184694, 2216344, 15313151, 3521236,\\ 760233, 13711752, 13711633, 12750386, 3638725, 16577950, 11463235, 16577952).$$

We select tap positions 1, 4, 11 and 20 and extract $4 = \lfloor \log_2(24) \rfloor$ bit sequences from the CA. The tuple $(b_1, b_4, b_{11}, b_{20}) = (0, 3837988, 9453076, 12750386)$ represents the shift of the 4 sequences from the first sequence. The value of 2^{24} is 16777216. We have

1. $b_4 = 2^{22} - a_4$, where $a_4 = 356316$.
2. $b_{11} = 2^{23} + a_{11}$, where $a_{11} = 1064468$.
3. $b_{20} = 2^{23} + 2^{22} + a_{20}$, where $a_{20} = 167474$.

We have $\min\{b_4 - b_1, b_{11} - b_4, b_{20} - b_{11}, 2^{24} - 1 - b_{20}\} = \min\{2^{22} - a_4, 2^{22} + a_{11} + a_4, 2^{22} + a_{20} - a_{11}, 2^{22} - a_{20} - 1\} = 2^{22} - 886994$. Thus the system can encrypt messages of length $2^{22} - 886994 > 2^{21}$. ∎

10 Conclusion

In this paper we have introduced new ideas to improve upon the well studied classical models of stream ciphers. An important constituent of our model is the use of cellular automata. We point out an important security advantage of cellular automata over linear feedback shift registers. We believe that our model will form the basic skeleton for designing new practical stream cipher systems.

References

[1] R. J. Anderson. Searching for the optimum correlation attack. In *Fast Software Encryption - FSE 1994*, pp 137-143.

[2] P. Camion, C. Carlet, P. Charpin, and N. Sendrier. On correlation immune functions. In *Advances in Cryptology - CRYPTO'91*, pages 86–100. Springer-Verlag, 1992.

[3] A. Canteaut and M. Trabbia. Improved fast correlation attacks using parity checks equations of weight 4 and 5. Advances in Cryptology - EUROCRYPT 2000, Lecture Notes in Computer Science, pp 573-588.

[4] V. Chepysov, T. Johansson and B. Smeets. A simple algorithm for fast correlation attacks on stream ciphers, In *Fast Software Encryption - FSE 2000*, Lecture Notes in Computer Science.

[5] C. Ding, G. Xiao, and W. Shan. *The Stability Theory of Stream Ciphers.* Number 561 in Lecture Notes in Computer Science. Springer-Verlag, 1991.

[6] M.R. Garey and D.S. Johnson. Computers and Intractibility: A Guide to the Theory of NP-completeness. W.H. Freeman, San Francisco, 1979.

[7] J. D. Golic, A. Clark and E. Dawson. Generalized inversion attack on nonlinear filter generators. IEEE Transactions on Computers, 49(10):1100-1109 (2000).

[8] R. Lidl and H. Niederreiter. Introduction to finite fields and their applications. Cambridge University Press, revised edition, 1994.

[9] S.Maitra and P. Sarkar. Highly nonlinear resilient functions optimizing Siegenthaler's inequality. *Advances in Cryptology - CRYPTO 1999*, Lecture Notes in Computer Science, pp 198-215.

[10] J.L. Massey. Shift register synthesis and BCH decoding. *IEEE Transactions on Information Theory,*, 15(1969), 122-127.

[11] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone. Handbook of Applied Cryptography. CRC Press, 1997.

[12] J. P. Mesirov and M. M. Sweet. Continued fraction expansions of rational expressions for built-in self-test. *Journal of Number Theory*, 27, 144-148 (1987).

[13] R. A. Rueppel. Analysis and Design of Stream Ciphers Springer-Verlag, 1986.

[14] R. A. Rueppel and O. J. Staffelbach. Products of linear recurring sequences with maximum complexity. *IEEE Transactions on Information Theory*, volume IT-33, number 1, pp. 124-131, 1987.

[15] P. Sarkar. Computing Shifts in 90/150 Cellular Automata Sequences. CACR Technical Report CORR 2001-46, University of Waterloo, http://www.cacr.math.uwaterloo.ca

[16] P. Sarkar and S. Maitra. Nonlinearity bounds and constructions of resilient Boolean functions. In *Advances in Cryptology - CRYPTO 2000*, number 1880 in LNCS, pages 515–532. Springer Verlag, 2000.

[17] T. Siegenthaler. Correlation-immunity of nonlinear combining functions for cryptographic applications. *IEEE Transactions on Information Theory*, IT-30(5):776–780, September 1984.

[18] T. Siegenthaler. Decrypting a class of stream ciphers using ciphertext only. *IEEE Transactions on Computers*, C-34(1):81–85, January 1985.

[19] S. Tezuka and M. Fushimi. A method of designing cellular automata as pseudo random number generators for built-in self-test for VLSI. In *Finite Fields: Theory, Applications and Algorithms*, Contemporary Mathematics, AMS, pages 363–367, 1994.

[20] G.-Z. Xiao and J. Massey. A spectral characterization of correlation immune combining functions. *IEEE Transactions on Information Theory*, 34(3):569–571, May 1988.