Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.12785/amis/080204

Generation of Cryptographic Sequences by means of Difference Equations

A. Fúster-Sabater[*]

Institute of Physical and Information Technologies (ITEFI), C.S.I.C., Serrano 144, 28006 Madrid, Spain

Received: 8 Mar. 2013, Revised: 9 Jul. 2013, Accepted: 11 Jul. 2013
Published online: 1 Mar. 2014

Abstract: In the present work, it is shown that the sequences obtained from cryptographic generators based on decimation are just particular solutions of a kind of linear difference equations. Moreover, all these sequences are simple linear combinations of a class of basic sequences (binomial sequences). Cryptographic parameters of decimated sequences, e.g. period, linear complexity or balancedness, can be analyzed in terms of solutions to linear equations. In brief, difference equations are useful tools for the generation of new cryptographic sequences with application in stream ciphers.

Keywords: sequence, linear difference equation, sequence generator, stream cipher, cryptography

1 Introduction

Confidentiality of sensitive information makes use of an encryption function currently called *cipher* that converts the *plaintext* or original message into the *ciphertext*. Symmetric key ciphers are usually divided into two large classes: stream ciphers and block-ciphers depending on whether the encryption function is applied either to each individual bit or to a block of bits, respectively. Stream ciphers are the fastest among the encryption procedures so they are implemented in many technological applications e.g. the encryption algorithm RC4 [16] used in Wired Equivalent Privacy (WEP) as a part of the IEEE 802.11 standards, the encryption function E0 in Bluetooth specifications [1] or the recent proposals HC-128 or Rabbit coming from the eSTREAM Project [17] and included in the latest release versions of CyaSSL (lightweight open source embedded implementation of the SSL/TLS protocol) [18].

Stream ciphers try to imitate the mythic *one-time pad cipher* or *Vernam cipher* [14] and are designed to generate a long sequence (*keystream sequence*) of pseudorandom bits [4]. This keystream sequence is bit-wise added with the plaintext (in emission) in order to obtain the ciphertext or with the ciphertext (in reception) in order to recover the original plaintext. Most keystream generators are based on maximal-length Linear Feedback Shift Registers (LFSRs) [7] whose output sequences, the so-called *m*-sequences, are combined by means of nonlinear functions to produce pseudorandom sequences of cryptographic application. Combinational generators, nonlinear filters, clock-controlled generators or irregularly decimated generators are just some of the most popular keystream sequence generators [5], [8], [14].

Inside the family of irregularly decimated generators, we can enumerate: a) the *shrinking generator* proposed by Coppersmith, Krawczyk and Mansour [2] that includes two LFSRs, b) the *self-shrinking generator* designed by Meier and Staffelbach [13] involving only one LFSR and c) the *generalized self-shrinking generator* proposed by Hu and Xiao [9] that can be considered as a specialization of the shrinking generator as well as a generalization of the self-shrinking generator. Irregularly decimated generators produce good cryptographic sequences characterized by long periods, good correlation features, excellent run distribution, balancedness [6], simplicity of implementation, etc. The underlying idea of this kind of generators is the irregular decimation of a *m*-sequence according to the bits of another one. The result of this decimation process is a binary sequence that will be used as keystream sequence in the cryptographic procedure.

In this work, it is shown that the sequences generated by irregularly decimated generators are particular solutions of binary coefficient homogeneous linear difference equations. In fact, all those sequences are just linear combinations of binomial sequences weighted by

[*] Corresponding author e-mail: amparo@iec.csic.es

binary coefficients. Cryptographic parameters of such sequences (e.g. period, linear complexity or balancedness) can be analyzed in terms of solutions to linear equations. It must be noticed that, although these sequences are irregularly decimated, in practice they are simple solutions to linear equations. This fact establishes a subtle link between irregular decimation and linearity that could be conveniently exploited in cryptanalytic terms.

At the same time, other sequences that are equation solutions but that are not included in the previous families also exhibit good properties for their application in cryptography. Thus, computing the solutions of linear difference equations provides one with new binary sequences whose cryptographic parameters can be easily guaranteed. In brief, linear difference equations can contribute very efficiently to the generation of keystream sequences for stream ciphers.

2 Preliminaries

In this section, we provide some basic notation and definitions that will be used throughout the paper.

Let p be a prime, $q = p^m$, and let $\mathbb{F}_q$ denote a finite field with q elements. The order of an element $\alpha \in \mathbb{F}_q$ is the smallest positive integer k such that $\alpha^k = 1$, denoted by ord(α). An element α with order $q-1$ is called a primitive element in $\mathbb{F}_q$. The primitive elements are exactly the generators of $\mathbb{F}_q^*$, the multiplicative group consisting of the nonzero elements of $\mathbb{F}_q$. Thus, a finite field $\mathbb{F}_q$ consists of 0 and appropriate powers of a primitive element.

Let $\{s_n\} = (s_0, s_1, s_2, \ldots)$ $n \geq 0$ be a sequence over $\mathbb{F}_p$ if $s_n \in \mathbb{F}_p$, $\forall n$. The sequence $\{s_n\}$ is periodic if and only if there exists an integer $T > 0$ such that $s_{n+T} = s_n$ holds for all $n \geq 0$.

Let r be a positive integer, and let $c_0, c_1, \ldots, c_{r-1}$ be given elements of the finite field $\mathbb{F}_p$. A sequence $\{s_n\}$ of elements of $\mathbb{F}_p$ satisfying the relation

$$s_{n+r} = c_1 s_{n+r-1} + c_2 s_{n+r-2} + \ldots + c_{r-1} s_{n+1} + c_r s_n, \quad (1)$$

is called a *rth*-order linear recurring sequence in $\mathbb{F}_p$. The terms $s_0, s_1, \ldots, s_{r-1}$, which determine uniquely the rest of the sequence, are referred to as the initial values. A relation of the form given in (1) is called a *rth*-order homogeneous linear recurrence relation. The monic polynomial of degree r

$$f(x) = x^r + c_1 x^{r-1} + c_2 x^{r-2} + \ldots + c_{r-1} x + c_r \in \mathbb{F}_p[x] \quad (2)$$

is called the characteristic polynomial of the linear recurring sequence and the sequence $\{s_n\}$ is said to be generated by $f(x)$. The minimal polynomial of $\{s_n\}$ is the polynomial of least degree whose linear recurrence relation is satisfied by such a sequence. For a survey of linear recurring sequences over finite fields, the reader is referred to [11].

The generation of linear recurring sequences can be implemented on Linear Feedback Shift Registers (LFSR). These devices with r memory cells (stages) handle information in the form of elements of $\mathbb{F}_p$ and they are based on shifts and linear feedback. The output of the LFSR is the string of elements $(s_0, s_1, s_2, \ldots)$ received in intervals of one time unit. If the characteristic polynomial of the linear recurring sequence is primitive, then the LFSR is called maximal-length LFSR and its output sequence has period $2^r - 1$, see [7]. This output sequence is called *PN*-sequence (pseudo-noise sequence) or *m*-sequence (maximal sequence).

The linear complexity (*LC*) of a sequence $\{s_n\}$ is defined as the length of the shortest LFSR that can generate such a sequence or equivalently the order of the shortest linear recurrence relation satisfied by such a sequence. In a general sense, linear complexity is related with the amount of sequence that is needed to determine the whole sequence. In cryptographic applications, linear complexity must be as large as possible. The recommended value is approximately half the period $LC \simeq T/2$.

In the remaining of this paper, we will consider sequences defined exclusively over the binary field ($p = 2$ and $q = 2^m$) denoted by $GF(2)$ where the extension field will be denoted by $GF(2^m)$. It should be noticed that the analysis provided here can be extended to sequences over any prime extension.

3 Cryptographic Generators Based on Decimations

The most important examples of irregularly decimated sequence generators are next introduced.

The *shrinking generator* is a binary sequence generator [2] composed by two maximal-length LFSRs: a control register R_1 that decimates the sequence produced by the other register R_2. In fact, the *m*-sequence $\{a_n\}$ generated by R_1 controls the bits of the *m*-sequence $\{b_n\}$ generated by R_2. The output sequence $\{z_n\}$ (the so-called shrunken sequence) is obtained according to the following decimation rule:

– If $a_n = 1 \Longrightarrow z_j = b_n$
– If $a_n = 0 \Longrightarrow b_n$ is discarded.

In brief, the output sequence $\{z_n\}$ produced by the shrinking generator is an irregular decimation of $\{b_n\}$ in terms of the bits of $\{a_n\}$. In addition, the sequence $\{z_n\}$ is the keystream sequence in the stream cipher procedure.

The *self-shrinking generator* [13] was designed as a variation of the shrinking generator for potential use in stream cipher applications. This generator consists of a maximal-length LFSR whose *m*-sequence $\{a_n\}$ is self decimated giving rise to the *self-shrunken sequence* $\{z_n\}$ or output sequence of such a generator. The decimation rule is quite simple. In fact, let (a_{2n}, a_{2n+1}) $n \geq 0$ be pairs

of consecutive bits of the sequence $\{a_n\}$, then we proceed as follows:

– If $a_{2n} = 1 \Longrightarrow z_j = a_{2n+1}$
– If $a_{2n} = 0 \Longrightarrow a_{2n+1}$ is discarded.

In fact, period, linear complexity and statistical properties of the self-shrunken sequence $\{z_n\}$ [13] make such a sequence very adequate for their application in stream cipher. In brief, the self-shrinking generator is a simplified version of the shrinking generator that satisfies the same decimation rule as before but includes only one maximal-length LFSR.

Finally, the most representative element of the class of irregularly decimated generators is the *generalized self-shrinking generator* [9] that generates a family of binary sequences for cryptographic purposes. This generator can be described as follows:

–It makes use of two sequences: a *m*-sequence $\{a_n\}$ and a shifted version of such a sequence denoted by $\{v_n\}$.
–It relates both sequences by means of a simple decimation rule to generate an output sequence.

In mathematical terms, the family of generalized self-shrinking sequences can be defined as follows [9]:

Definition 1.*Let $\{a_n\}$ be a m-sequence over $GF(2)$ with period 2^r-1 generated from a maximal-length LFSR of r stages. Let G be a r-dimensional binary vector defined as:*

$$G = (g_0, g_1, g_2, \ldots, g_{r-1}) \in GF(2)^r. \quad (3)$$

The n-th element of the sequence v_n is defined as:

$$v_n = g_0 a_n + g_1 a_{n+1} + g_2 a_{n+2} + \ldots + g_{r-1} a_{n+r-1}, \quad (4)$$

where the sub-indexes of the sequence $\{a_n\}$ are reduced mod 2^r-1. For $n \geq 0$ the following decimation rule is applied:

– *If $a_n = 1$, then v_n is output.*
– *If $a_n = 0$, then v_n is discarded and there is no output bit.*

In this way, an output sequence $(b_0, b_1, b_2, \ldots)$ denoted by $\{b_n\}$ or $\{b(G)\}$ is generated. Such a sequence is called a generalized self-shrinking sequence. We call the sequence family $B(a) = \{\{b(G)\}, G \in GF(2)^r\}$, the family of generalized self-shrinking sequences based on the m-sequence $\{a_n\}$.

Remark that the sequence $\{v_n\}$ is nothing but a shifted version of the sequence $\{a_n\}$. The 2^r-1 nonzero choices of G over $GF(2)^r$ result in the 2^r-1 distinct shifts of $\{v_n\}$ regarding the *m*-sequence $\{a_n\}$. For each new sequence $\{v_n\}$ a new generalized self-shrinking sequence is generated. Let us see a simple example.

Example 1: For the 4-degree and the *m*-sequence $\{a_n\} = \{011110101100100\}$ whose characteristic polynomial is $x^4 + x^3 + 1$, we get 16 generalized self-shrinking sequences based on $\{a_n\}$ (see [9]):

0. $G = (0000), \{b(G)\} = 00000000 \sim$
1. $G = (1000), \{b(G)\} = 11111111 \sim$
2. $G = (0100), \{b(G)\} = 11100100 \sim$
3. $G = (0010), \{b(G)\} = 11011000 \sim$
4. $G = (0001), \{b(G)\} = 10101010 \sim$
5. $G = (1001), \{b(G)\} = 01010101 \sim$
6. $G = (1101), \{b(G)\} = 10110001 \sim$
7. $G = (1111), \{b(G)\} = 01101001 \sim$
8. $G = (1110), \{b(G)\} = 11000011 \sim$
9. $G = (0111), \{b(G)\} = 10010110 \sim$
10. $G = (1010), \{b(G)\} = 00100111 \sim$
11. $G = (0101), \{b(G)\} = 01001110 \sim$
12. $G = (1011), \{b(G)\} = 10001101 \sim$
13. $G = (1100), \{b(G)\} = 00011011 \sim$
14. $G = (0110), \{b(G)\} = 00111100 \sim$
15. $G = (0011), \{b(G)\} = 01110010 \sim$

Recall that the generated sequences are not 16 different sequences as some of them are shifted versions of a unique sequence, e.g. sequences (3, 6, 12, 13) or sequences (2, 10, 11, 15).

It must be noticed that in the class of generalized self-shrinking sequences there will always appear [9] the identical zero sequence $\{000000\cdots\}$ for $G = (00\cdots00)$ and the identical one sequence $\{1111\cdots\}$ for $G = (10\cdots00)$ as well as the sequences $\{010101\cdots\}$ and $\{101010\cdots\}$. Moreover, apart from the identical zero and identical one sequences, the rest of generalized self-shrinking sequences are balanced, see [9].

Now an interesting relation between the sequences $\{a_n\}$ and $\{v_n\}$ can be pointed out.

Lemma 1.*Let $\{s_n\}$ be a m-sequence over $GF(2)$ generated by a maximal-length LFSR of r stages. The shift between the two decimated sequences $\{c_i\} = \{s_{2i}\}$ and $\{b_i\} = \{s_{2i+1}\}$ for $i \geq 0$ equals $\tau = 2^{r-1}$.*

Proof. The result follows from the fact that $c_{i+2^{r-1}} = s_{2(i+2^{r-1})} = s_{2i+2^r} = s_{2i+1+2^r-1} = s_{2i+1} = b_i$. □

This result allows one to characterize the self-shrinking generator as an element of the generalized self-shrinking generator family.

Lemma 2.*The self-shrunken sequence is an element of the class of generalized self-shrinking sequences.*

Proof. If $\{a_n\}$ and $\{v_n\}$ are taken as $\{a_n\} = \{a_{2i}\}$ and $\{v_n\} = \{a_{2i+1}\}$ for $i \geq 0$ respectively, then according to Lemma 1 the shift between both sequences is 2^{r-1}. Thus, if $\{v_n\}$ is shifted 2^{r-1} positions regarding $\{a_n\}$, then the resulting generalized self-shrinking sequence is the self-shrunken sequence. Consequently, the self-shrinking generator is an element of the generalized self-shrinking generator family for this particular value of shift between both sequences. □

In the previous example, when $\{v_n\}$ is shifted 2^3 positions regarding $\{a_n\}$ that is $\{v_n\} = \{110010001111010\}$, then the generalized

self-shrinking sequence $\{b(0111)\}$ corresponds to the self-shrunken sequence.

As the generalized self-shrinking generator is the most general among the irregularly decimated generators, cryptographic parameters of the different sequences obtained from this generator will be analyzed as solutions of linear difference equations.

4 Linear Difference Equations

In this section, the kind of linear difference equations we are dealing with will be introduced.

The linear recurrence relation given in (1) can be expressed as a linear difference equation

$$(E^r + \sum_{j=1}^{r} c_j\, E^{r-j})\, z_n = 0, \qquad n \geq 0 \tag{5}$$

where $z_n \in GF(2)$ is the n-th term of a binary sequence $\{z_n\}$ that satisfies the previous equation, E is the shifting operator which operates on the terms z_n of a solution sequence, i.e. $E^j z_n = z_{n+j}$, the coefficients c_j are binary coefficients $c_j \in GF(2)$, r is an integer and the operations of (5) are the defined operations over $GF(2)$; namely, addition and multiplication modulo 2. The r-degree characteristic polynomial of (5) is

$$f(x) = x^r + \sum_{j=1}^{r} c_j\, x^{r-j}, \tag{6}$$

that coincides with the expression defined in (2). If $f(x)$ is an irreducible polynomial in $GF(2)[x]$ of degree r, then $f(x)$ has a root α in $GF(2^r)$. Furthermore, all the roots of $f(x)$ are simple and are given by the r distinct elements in the extension field ([11], Th. 2.14, pp. 49-50):

$$\alpha,\ \alpha^2,\ \alpha^{2^2}, \ldots,\ \alpha^{2^{(r-1)}} \in GF(2^r). \tag{7}$$

In this case, $A_0\alpha^n$ is a solution of (5) where $A_0 \in GF(2^r)$ is an arbitrary constant. Since the polynomial (6) has r roots, there are r linearly independent solutions to (5), and the general solution is a linear combination of these solutions with r arbitrary constants $A_0, A_1, \ldots, A_{r-1} \in GF(2^r)$ determined by the initial values $z_0, z_1, \ldots, z_{r-1}$. Thus, the general solution can be written as

$$z_n = \sum_{j=0}^{r-1} A_j\, \alpha^{2^j n}, \qquad n \geq 0 \tag{8}$$

where $A_j = (A_{j-1})^2$ $(j = 1, 2, \ldots, r-1)$ for z_n to be in $GF(2)$, see [3]. Therefore, the equation (8) is simplified to

$$z_n = \sum_{j=0}^{r-1} A^{2^j}\, \alpha^{2^j n}, \qquad n \geq 0 \tag{9}$$

with $A \in GF(2^r)$.

Let us generalize the previous linear difference equation to a more complex kind of linear difference equation whose roots have a multiplicity greater than 1. In fact, we are going to consider difference equations of the form

$$(E^r + \sum_{j=1}^{r} c_j\, E^{r-j})^p\, z_n = 0, \qquad n \geq 0 \tag{10}$$

p being an integer $p > 1$. The characteristic polynomial $f_c(x)$ of this kind of equation is

$$f_c(x) = f(x)^p = (x^r + \sum_{j=1}^{r} c_j\, x^{r-j})^p. \tag{11}$$

In this case, the roots of $f_c(x)$ are the same as those of the polynomial $f(x)$, that is $\alpha, \alpha^2, \alpha^{2^2}, \ldots, \alpha^{2^{(r-1)}}$, but with multiplicity p. If α is a root of multiplicity p, then the expression $\binom{n}{i} A_i\, \alpha^n$ with $i = 0, 1, 2, \ldots, p-1$ provides the p linearly independent solutions of (10) associated with the root α, where $\binom{n}{i}$ is a binomial coefficient reduced modulo 2 and the arbitrary constants $A_i \in GF(2^r)$, see [10]. Therefore, the general solution of the equation (10) is a linear combination of the $p \cdot r$ independent solutions that can be written as follows

$$z_n = \sum_{i=0}^{p-1} \binom{n}{i} A_i\, \alpha^n + \sum_{i=0}^{p-1} \binom{n}{i} (A_i)^2\, \alpha^{2n} + \ldots + \sum_{i=0}^{p-1} \binom{n}{i} (A_i)^{2^{r-1}}\, \alpha^{2^{r-1} n}, \tag{12}$$

where each term corresponds to the p independent solutions associated with the root α^{2^j} $(j = 0, 1, \ldots, r-1)$, respectively. As before the same relation among arbitrary constants applies here for z_n to be in $GF(2)$. In a more compact way, the equation (12) can be written as

$$z_n = \sum_{i=0}^{p-1} \left(\binom{n}{i} \sum_{j=0}^{r-1} A_i^{2^j}\, \alpha^{2^j n} \right), \qquad n \geq 0 \tag{13}$$

where $A_0, A_1, \ldots, A_{p-1} \in GF(2^r)$.

In brief, the n-th term of a solution sequence $\{z_n\}$ of (10) is the addition of the n-th term of each one of the p sequences $\{\sum_{j=0}^{r-1} A_i^{2^j}\, \alpha^{2^j n}\}$ $(0 \leq i < p)$ weighted by binomial coefficients.

4.1 Analysis of the Binomial Coefficients

Let us now analyze the binomial coefficients modulo 2 that appear in (13).

It is a well known fact that the binomial coefficient $\binom{n}{i}$ is the coefficient of the x^i term in the polynomial expansion of the binomial power $(1+x)^n$. In addition, the binomial coefficients satisfy:

Table 1: Binomial coefficients, binomial sequences and periods T_i

Bin. coef.	Binomial sequences	T_i
$\binom{n}{0}$	$S_0 = \{1,1,1,1,1,1,1,1 \sim\}$	$T_0 = 1$
$\binom{n}{1}$	$S_1 = \{0,1,0,1,0,1,0,1 \sim\}$	$T_1 = 2$
$\binom{n}{2}$	$S_2 = \{0,0,1,1,0,0,1,1 \sim\}$	$T_2 = 4$
$\binom{n}{3}$	$S_3 = \{0,0,0,1,0,0,0,1 \sim\}$	$T_3 = 4$
$\binom{n}{4}$	$S_4 = \{0,0,0,0,1,1,1,1 \sim\}$	$T_4 = 8$
$\binom{n}{5}$	$S_5 = \{0,0,0,0,0,1,0,1 \sim\}$	$T_5 = 8$
$\binom{n}{6}$	$S_6 = \{0,0,0,0,0,0,1,1 \sim\}$	$T_6 = 8$
$\binom{n}{7}$	$S_7 = \{0,0,0,0,0,0,0,1 \sim\}$	$T_7 = 8$
...	...	...

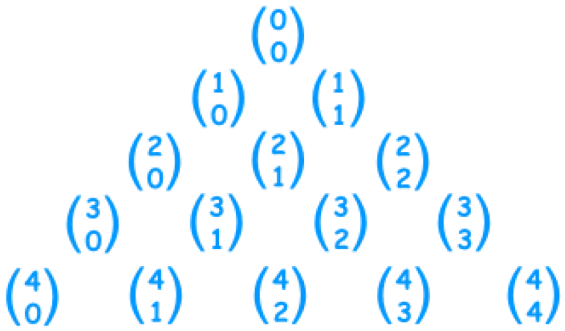

Fig. 1: Binomial coefficients arranged to form Pascal's triangle

$$\binom{n}{0} = 1 \text{ for all integers } n \geq 0,$$
$$\binom{n}{i} = 0 \text{ for all integers } n < i.$$

For n taking successive values $n \geq 0$, each binomial coefficient $\binom{n}{i}$ defines a *binomial sequence* $\{S_i\}$ with constant period T_i. In Table 1, the first binomial coefficients with their corresponding binomial sequences and periods are depicted.

On the other hand, arranging the binomial coefficients into rows for successive values of n gives a triangular array called Pascal's triangle, see Figure 1. The first (leftmost) diagonal of the triangle is the sequence identically 1, the second diagonal corresponds to the natural counting numbers 1, 2, 3, 4, ..., the third diagonal corresponds to the triangular numbers 1, 3, 6, 10, ... as well as many other fascinating relations (tetrahedral numbers, pentatope numbers, hexagonal numbers, Finonacci sequence, etc) [15] that can be found into the diagonals of Pascal's triangle.

The pattern obtained by coloring only the odd numbers in Pascal's triangle and shading out all the other spaces becomes the fractal called the Sierpinski's triangle, see Figure 2. Coming back to the equation (13), we can see that the binomial sequences $\{S_i\}$ correspond to the diagonals of the Sierpinski's triangle reduced modulo 2

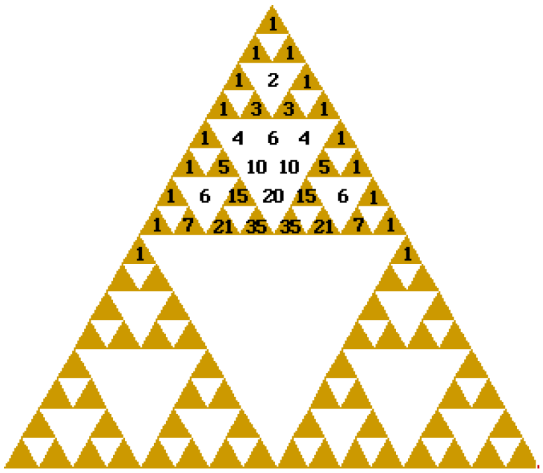

Fig. 2: Sierpinski's triangle with the numerical coefficients of Pascal's triangle

plus additional zeros at the beginning of each sequence for the values $\binom{n}{i}$ with $i > n$.

In brief, the solution sequence $\{z_n\}$ obtained from (13) is a linear combination of a m-sequence weighted by other sequences that are the successive diagonals of the Sierpinski's triangle reduced modulo 2. In an algebraic way, the generation of such diagonals, sequences $\{S_i\}$, follows a simple formation rule. Indeed, the $\{S_i\}$ binomial sequence associated with $\binom{n}{i}$ for $(2^k \leq i < 2^{k+1})$ (k being an integer) has a period $T_i = 2^{k+1}$ and its digits are:

1. The first 2^k bits are 0's.
2. The remaining 2^k bits are the first 2^k bits of the binomial sequence $\{S_{i-2^k}\}$.

According to this rule, binomial sequences can be easily generated.

5 Cryptographic Sequences as Solutions of Linear Difference Equations

Now the main results concerning generalized self-shrinking sequences and linear difference equations are introduced.

Theorem 1. *The family of generalized self-shrinking sequences B(a) based on the m-sequence $\{a_n\}$ are particular solutions of the homogeneous linear difference equation:*

$$(E+1)^p z_n = 0, \qquad p = 2^{L-1}, \tag{14}$$

whose characteristic polynomial is $(x+1)^p$ and L is the degree of the characteristic polynomial of the m-sequence $\{a_n\}$.

Proof: According to [9], the periods of the generalized self-shrinking sequences $B(a)$ are $T \in \{1, 2, 2^{L-1}\}$. Thus, the period T of any generalized self-shrinking sequence divides 2^{L-1}, i.e. it is a power of 2. Hence over $GF(2)$, $x^T + 1 = (x+1)^T$. On the other hand, if $f(x)$ is the characteristic polynomial of the shortest linear recurrence relation satisfied by a generalized self-shrinking sequence, then the condition $f(x)|x^T + 1$ implies that $f(x)$ is of the form:

$$f(x) = (x+1)^{LC} \tag{15}$$

where LC is its linear complexity (the order of the shortest linear recurrence relation satisfied by such a sequence). At the same time, it is a well known fact that the linear complexity of a periodic sequence is $\leq$ its period [7], [10]. Thus, for any generalized self-shrinking sequence $LC \leq 2^{L-1}$ and the polynomial of the shortest linear recurrence relation $f(x)$ divides the characteristic polynomial of (14). Therefore, the generalized self-shrinking sequences satisfied the equation (14) and are particular solutions of this homogeneous linear difference equation. □

Recall that the cryptographic sequences obtained from a nonlinear procedure such as decimation turn to be solutions of linear equations. Now the characteristics of the sequences that satisfy the linear difference equation (14) are analyzed in detail. In fact, the general solution given in (13) particularized to the equation (14) can be written as

$$z_n = \sum_{i=0}^{p-1} \binom{n}{i} A_i = \binom{n}{0} A_0 + \binom{n}{1} A_1 + \ldots + \binom{n}{p-1} A_{p-1},\ n \geq 0 \tag{16}$$

where $\alpha = 1$ is the unique root of $f_c(x)$ in the splitting field of f_c over $GF(2)$. Thus, 1 is the unique root with multiplicity p of the polynomial $(x+1)$ with $r = 1$, $p = 2^{L-1}$ and $A_i \in GF(2)$. Recall that the sequence $\{z_n\}$ is just the addition of binomial sequences weighted by the corresponding coefficients A_i.

It must be noticed that not all the solutions $\{z_n\}$ of (14) are generalized self-shrinking sequences although all the generalized self-shrinking sequences are solutions of (14). From the equation (16), particular features of the solution sequences and consequently of the generalized self-shrinking sequences can be easily determined. All of them are related with the choice of the p-tuple of binary coefficients $(A_0, A_1, A_2, \ldots, A_{p-1})$.

5.1 Periods of the Solution Sequences:

According to Table 1, the period of any binomial sequences S_i is just a power of 2. Moreover, according to (16) $\{z_n\}$ is the addition of binomial sequences with different periods all of them being powers of 2. Thus, the period of $\{z_n\}$ is the maximum period of the binomial sequences included in (16), that is the T_i corresponding to the binomial sequence with the greatest index i $(0 \leq i < p)$ such that $A_i \neq 0$.

Analyzing the periods of the generalized self-shrinking sequences in terms of the solutions of (14), we have:

–Two generalized self-shrinking sequences:

$$\{b(G)\} = 00000000 \sim$$
$$\{b(G)\} = 11111111 \sim$$

with period $T = 1$ corresponding to the coefficients $(A_i = 0, \forall\ i)$ and $(A_0 = 1,\ A_i = 0\ \forall\ i > 0)$ in (16), respectively.

–Two generalized self-shrinking sequences:

$$\{b(G)\} = 10101010 \sim$$
$$\{b(G)\} = 01010101 \sim$$

with period $T = 2$ corresponding to the coefficients $(A_0 = A_1 = 1,\ A_i = 0\ \forall\ i > 1)$ and $(A_0 = 0, A_1 = 1,\ A_i = 0\ \forall\ i > 1)$ in (16), respectively.

–The rest of generalized self-shrinking sequences with period $T = 2^{L-1}$ correspond to p-tuples of coefficients A_i in (16) with any $A_i \neq 0$ in the interval $(2^{L-2} \leq i < 2^{L-1})$.

5.2 Linear Complexity of the Solution Sequences:

As it has been previously seen, the linear complexity of a sequence equals the number and multiplicity of the roots of the characteristic polynomial $f(x)$ in its shortest linear recurrence relation. Therefore coming back to (16) and analyzing the coefficients A_i, the linear complexity of $\{z_n\}$ can be computed. In fact, we have a unique root 1 with maximal multiplicity p. Thus, if i is the greatest index $(0 \leq i < p)$ for which $A_i \neq 0$, then the linear complexity LC of the sequence $\{z_n\}$ will be:

$$LC = i_{max} + 1 \tag{17}$$

as it will be the multiplicity of the root 1.

Concerning the generalized self-shrinking sequences, the main result related to their linear complexity can be stated as follows:

Theorem 2. *The linear complexity LC of the generalized self-shrinking sequences with period $T_i = 2^{L-1}$ satisfies the inequality:*

$$2^{L-2} < LC \leq 2^{L-1}. \tag{18}$$

Proof: The result follows from the fact that those generalized self-shrinking sequences with $T_i = 2^{L-1}$ include at least a binomial sequences $\binom{n}{i}$ for $A_i \neq 0$ with i in the range $2^{L-2} \leq i < 2^{L-1}$. Thus, according to (17) the range of values of their corresponding linear complexity

is given by the equation (18). □

According to Theorem 2, the linear complexity of the generalized self-shrinking sequences with $T = 2^{L-1}$ is adequate for cryptographic purposes. In the case of generalized self-shrinking sequences with $T = 1$ or $T = 2$, their corresponding LC are 1 and 2, respectively.

In brief, the handling of coefficients A_i allows one to generate binary sequences with controllable period and linear complexity.

5.3 An Illustrative Example

A simple example to clarify the results of the previous sections is now introduced. In fact, according to (16) for a maximal-length LFSR of $L = 4$ stages and $p = 2^3 = 8$, the different 8-tuples $(A_0, A_1, \ldots, A_7)$ determine the characteristics not only of the generalized self-shrinking sequences but also those of other solution sequences not included in the previous family. Due to the size of this example, all the possible choices can be analyzed.

In fact, for the 4-degree *m*-sequence introduced in Section 3:

$$\{a_n\} = \{011110101100100\},$$

the family of generalized self-shrinking sequences $B(a)$ are solutions of the equation:

$$(E+1)^p b_n = 0, \ p = 2^3, \tag{19}$$

whose general form is:

$$b_n = \binom{n}{0} A_0 + \binom{n}{1} A_1 + \cdots + \binom{n}{7} A_7, \ n \geq 0 \tag{20}$$

Different choices of the 8-tuple $(A_0, A_1, \ldots, A_7)$ can be considered:

1. For the sequences $\{b_n\} = 00000\cdots$, $\{b_n\} = 11111\cdots$, $\{b_n\} = 10101010\cdots$ and $\{b_n\} = 01010101\cdots$, the choice of the p-tuple has been explained in subsection (5.1).
2. For $A_2 \neq 0$, $A_i = 0 \ \forall \ i > 2$, there is a unique and balanced solution sequence $\{b_n\}$ with period $T_2 = 4$ and $LC_2 = 3$.
   In this case, there is no generalized self-shrinking sequence with such characteristics as there is no generalized self-shrinking sequence with $T = 4$.
3. For $A_3 \neq 0$, $A_i = 0 \ \forall \ i > 3$, there are two non-balanced different sequences with period $T_3 = 4$ and $LC_3 = 4$.
   In this case, there is not generalized self-shrinking sequence with such characteristics neither.
4. For $A_4 \neq 0$, $A_i = 0 \ \forall \ i > 4$, there are two balanced different sequences with period $T_4 = 8$ and $LC_4 = 5$.
   For example, the 5-tuple $(A_0 = 0, A_1 = 0, A_2 = 1, A_3 = 0, A_4 = 1)$ generates $\{b_n\} = \{00111100\cdots\}$ that corresponds to the generalized self-shrinking sequence:

$$G = (0110), \{b(G)\} = 00111100\cdots.$$

   Moreover, a shifted version of this sequence $\{b_n\} = \{11000011\cdots\}$ for the 5-tuple (1,0,1,0,1) corresponds to the generalized self-shrinking sequence:

$$G = (1110), \{b(G)\} = 11000011\cdots.$$

   The 5-tuple $(A_0 = 0, A_1 = 1, A_2 = 1, A_3 = 0, A_4 = 1)$ generates $\{b_n\} = \{01101001\}$ that corresponds to the generalized self-shrinking sequence:

$$G = (1111), \{b(G)\} = 01101001\cdots.$$

   Moreover, a shifted version of this sequence $\{b_n\} = \{10010110\cdots\}$ for the 5-tuple (1,1,1,0,1) corresponds to the generalized self-shrinking sequence:

$$G = (0111), \{b(G)\} = 10010110\cdots.$$

   The last two sequences are shifted versions of the self-shrunken sequence associated with $\{a_n\}$.
5. For $A_5 \neq 0$, $A_i = 0 \ \forall \ i > 5$, there are four not all balanced different sequences with period $T_5 = 8$ and $LC_5 = 6$.
   For example, the 6-tuple $(A_0 = 0, A_1 = 1, A_2 = 1, A_3 = 1, A_4 = 0, A_5 = 1)$ generates the sequence $\{b_n\} = \{01110010\cdots\}$ that corresponds to the generalized self-shrinking sequence:

$$G = (0011), \{b(G)\} = 01110010\cdots.$$

   Moreover, shifted versions of this sequence correspond to the generalized self-shrinking sequences:

$$G = (0101), \{b(G)\} = 01001110\cdots,$$
$$G = (0100), \{b(G)\} = 11100100\cdots,$$
$$G = (1010), \{b(G)\} = 00100111\cdots.$$

   The 6-tuple $(A_0 = 1, A_1 = 0, A_2 = 1, A_3 = 1, A_4 = 0, A_5 = 1)$ generates $\{b_n\} = \{11011000\cdots\}$ that corresponds to the generalized self-shrinking sequence:

$$G = (0010), \{b(G)\} = 11011000\cdots.$$

   Moreover, shifted versions of this sequence correspond to the generalized self-shrinking sequences:

$$G = (1101), \{b(G)\} = 10110001\cdots,$$
$$G = (1011), \{b(G)\} = 10001101\cdots,$$
$$G = (1101), \{b(G)\} = 00011011\cdots.$$

   There are two other non-balanced solution sequences $\{b_n\} = \{00000101\cdots\}$ and $\{b_n\} = \{11111010\cdots\}$ that do not correspond to any generalized self-shrinking sequences although they satisfy the same cryptographic characteristics as far as period and linear complexity are concerned.
6. For $A_6 \neq 0$ and $A_7 = 0$, there are eight not all balanced different sequences with period $T_4 = 8$ and $LC_6 = 7$. None of them corresponds to generalized self-shrinking sequences.
   There are four balanced solution sequences $\{b_n\} = \{01010110\cdots\}$, $\{b_n\} = \{10101001\cdots\}$,

$\{b_n\} = \{01011100\cdots\}$ and $\{b_n\} = \{10100011\cdots\}$ with the same period, the same autocorrelation values and greater linear complexity than that of the generalized self-shrink- ing sequences described in choices (4) and (5).

7. For $A_7 \neq 0$, there are sixteen different and unbalanced solution sequences with period $T_7 = 8$ and $LC_7 = 8$. None of them corresponds to generalized self-shrinking sequences. Nevertheless, it must be noticed that any generalized self-shrinking sequence in choices (4) and (5) becomes a solution sequence of this class just by complementing the last digit, as the binomial sequence corresponding to $A_7 = 1$ is 00000001. For example, the sequence $\{b_n\} = \{00111101\cdots\}$ corresponds to the one-bit complementation of $G = (0110)$, $\{b(G)\} = 00111100$ or $\{b_n\} = \{01101000\cdots\}$ corresponds to the one-bit complementation of $G = (1111)$, $\{b(G)\} = 01101001$ both described in choice (4). The same applies for the generalized self-shrinking sequences in choice (5).

5.4 Generation of Cryptographic Sequences in terms of Binomial Sequences

From the previous section, it can be deduced that the addition modulo 2 of a correct choice of binomial sequences (the p-tuple $(A_0, A_1, A_2, \ldots, A_{p-1})$) results in the generation of sequences with controllable period and linear complexity. Nevertheless, from a cryptographic point of view balancedness must be taken into account.

In this sense, it must be noticed that the complementation of the last bit of a generalized self-shrinking sequence with period 2^{L-1} means that the resulting sequence includes the binomial sequence

$$\binom{n}{2^{L-1}-1} \qquad (n \geq 2^{L-1}-1) \tag{21}$$

That is the identically null sequence except for the last element that is 1. This implies that the obtained sequence will have period $T = 2^{L-1}$, maximum linear complexity $LC = 2^{L-1}$ and quasi-balancedness as the difference between the number of 1's and 0's will be just one. For a cryptographic range $L = 128$, this difference is negligible. In brief, the selection of coefficients A_i allows one to control period, linear complexity and balancedness of the solution sequences.

6 An Algorithm to Compute Period and Linear Complexity of Cryptographic Sequences

An efficient algorithm to determine the period and linear complexity of any binary sequence obtained from a generalized self-shrinking generator is now described.

Algorithm 1 Binomial Sequence Generation

```
01: procedure Binomial (...)
02:     if num1=num2 or num1=1 then out=1;
03:     else
04:         if num1>num2 then out=0;
05:         else
06:             if num2>2*halfperiod then
07:                 c2=(num2-1) mod (2*halfperiod)+1;
08:                 Binomial(num1,c2);
09:             else
10:                 if (num1<>2) then
11:                     c1=num1-halfperiod;
12:                     c2=num2-halfperiod;
13:                     Binomial(c1,c2);
14: end procedure
```

Algorithm 2 Binomial Sequence Composition

```
01: function Composition (...)
02:     for i=1 to lseq do
03:         if seq[i]<>Sum[i] then
04:             for j=i to lseq do
05:                 Binomial(i,j);
06:                 Subseq[i][j]=out;
07:                 Sum[j]=(Sum[j]+Subseq[i][j]) mod 2;
08: end function
```

The mathematical background of the proposal was introduced in the previous sections.

Algorithm 1 finds recursively the binomial sequences that are the components of any given generalized self-shrin- king sequence by means of a bit-wise analysis of the input sequence. In such an algorithm, the variable *halfperiod* takes the value of half the period of the corresponding binomial sequence.

The iterative Algorithm 2 computes the discrepancy between the bit of the sequence b_n and the corresponding bit Σ_n of the addition of binomial sequences synthesized with Algorithm 1 till then. If there is discrepancy, then it calls Algorithm 1. In this way, Algorithm 2 takes as input a generalized self-shrinking sequence of length *lseq*, calls Algorithm 1 when it is necessary to synthesize binomial sequences and produces as output the addition of such binomial sequences.

A rough asymptotic analysis of Algorithm 2 shows the following: each iteration of the inner loop takes a constant amount of time. As there are *lseq* iterations in the outer loop as well as $lseq - i + 1$ iterations in the inner loop, then the total runtime can be expressed by the sum of terms of an arithmetic progression, $lseq(lseq+1)/2$, plus lower order terms. Disregarding lower order terms, we can conclude that the algorithm is efficient as its runtime is $O(lseq^2)$.

Compared with the Berlekamp-Massey algorithm [12], it is a well known fact that such an algorithm must store $2 \cdot LC$ bits of the generalized sequence, while the

Table 2: Average, maximal, minimal *LC* and Periods of Generalized Sequences

L	LC_{ave}	LC_{max}	LC_{min}	T
3	3	3	3	4
4	5	5	5	8
5	12	13	10	16
6	27	28	25	32
7	57	59	54	64
8	120	122	118	128
9	246	249	243	256
10	501	504	498	512
11	1012	1015	1009	1024
12	2035	2038	2031	2048
13	4079	4085	4072	4096
14	8175	8180	8170	8192

algorithm here proposed allows one to compute period and linear complexity with less amount of input sequence. In addition, when the binomial sequence corresponding to the binomial coefficient $\binom{n}{i}$ with $i \geq 2^{L-2}$ is achieved, then the period of the sequence is guaranteed to be $T = 2^{L-1}$ and the linear complexity satisfies the inequality (18). In this way, although not all the bits of the generalized sequence have been processed a lower bound on the linear complexity is already guaranteed. Moreover, this lower bound is exponential in the length L of the LFSR thus adequate for cryptographic purposes.

Table 2 shows the results obtained experimentally when the previous algorithm is applied. More precisely, it depicts the integer approximations of the average, maximal and minimal linear complexities as well as periods of all generalized self-shrinking sequences produced with LFSR from $L = 3$ till $L = 14$. Recall that different maximal-length LFSR with the same number of stages L can generate generalized self-shrinking sequences with different linear complexities. The complexity range is given by the previous equation (18) but the numerical results are close to the upper bound. In any case, the average linear complexity for each L is near the period. In brief, generalized self-shrinking sequences have a linear complexity quite close to their periods what means a good cryptographic quality to prevent cryptanalytic attacks.

7 Conclusions

In this work, it is shown that the sequence obtained from generators based on decimation are particular solutions of homogeneous linear difference equations with binary coefficients. At the same time, there are other many solution sequences not included in the previous class that can be used for cryptographic purposes. The choice of tuples of coefficients allows one:

1.To get all the solutions of the above linear difference equations, among them there are sequences with application in stream cipher.
2.To obtain sequences with controllable period, linear complexity and balancedness.

It must be noticed that, although generalized self-shrinking sequences and self-shrinking sequences are generated from LFSR by irregular decimation, in practice they are simple solutions of linear equations. This subtle paradox between irregular decimation as a procedure to break linearity and linear equations that generate sequences supposed nonlinear can be conveniently exploited in the cryptanalysis of such keystream generators. In fact, such a contradiction confirms the cryptographically celebrated words: Linearity is the curse of the cryptographer (J.L. Massey, Crypto89).

A natural extension of this work is the generalization of this procedure to many other cryptographic sequences, the so-called interleaved sequences, as they present very similar structural properties to those of the sequences obtained from irregular decimation generators.

Acknowledgement

Work supported by Ministry of Science and Innovation and European FEDER Fund under Project TIN2011-25452/TSI.
The author wishes to thank the anonymous referees for their comments and suggestions during the preparation of this manuscript.

References

[1] Bluetooth, *Specifications of the Bluetooth system*, Version 1.1, available at http://www.bluetooth.com/
[2] D. Coppersmith, H. Krawczyk and Y. Mansour, The Shrinking Generator. Proc. of CRYPTO'93. LNCS, **773**, 22-39 (1994).
[3] L. E. Dickson, Linear Groups with an Exposition of the Galois Field Theory. New York: Dover, 3-71 (1958). An updated reprint can be found at http://www-math.cudenver.edu/ wcherowi/courses/finflds.html
[4] eSTREAM, the ECRYPT Stream Cipher Project, Call for Primitives, available at http://www.ecrypt.eu.org/stream/
[5] A. Fúster-Sabater and P. Caballero-Gil, Strategic Attack on the Shrinking Generator, Theoretical Computer Science, **409**, 530-536 (2009) .
[6] A. Fúster-Sabater, P. Caballero-Gil and O. Delgado-Mohatar, Deterministic Computation of Pseudorandomness in Sequences of Cryptographic Application. Proc. of ICCS 2009, Part I, LNCS, **5544**, 621-630 (2009).
[7] S. W. Golomb, Shift Register-Sequences, (Aegean Park Press, Laguna Hill, (1982)) .
[8] G. Gong, Theory and Applications of q-ary Interleaved Sequences, IEEE Trans. Information Theory, **41**, 400-411 (1995).

[9] Y. Hu and G. Xiao, Generalized Self-Shrinking Generator, IEEE Trans. Inform. Theory, **50**, 714-719 (2004).
[10] E. L. Key, An Analysis of the Structure and Complexity of Nonlinear Binary Sequence Generators, IEEE Trans. Informat. Theory, **22**, 732-736 (1976).
[11] R. Lidl and H. Niederreiter, Introduction to Finite Fields and Their Applications, (Cambridge, England: Cambridge University Press, (1986)) .
[12] J. L. Massey, Shift-Register Synthesis and BCH Decoding, IEEE Trans. Informat. Theory, **15**, 122-127 (1969).
[13] W. Meier and O. Staffelbach, The Self-Shrinking Generator, in Proc. EUROCRYPT94. LNCS, **950**, 205-214 (1995).
[14] A. J. Menezes *et al.*, Handbook of Applied Cryptography, (New York:CRC Press, (1997)) .
[15] Mathematical Forum, Pascal's triangle, available at http://mathforum.org/dr.math/faq/faq.pascal.triangle.html
[16] R. L. Rivest, The RC4 Encryption Algorithm. (RSA Data Sec., Inc., March 98) .
[17] M. Robshaw, O. Billiet, New Stream Cipher Designs: The eSTREAM Finalist, Lecture Notes in Computer Science, **4986**, (2008) .
[18] Yet Another SSL (YASSL), available at http://www.yassl.com

Amparo Fúster-Sabater received the B.S. and Ph.D. degrees in physics from the Universidad Complutense, Madrid, Spain, in 1980 and 1985, respectively. Since 1988, she has been with the Spanish Higher Council for Scientific Research (C.S.I.C.) in the Institute of Physical and Information Technologies (ITEFI) at the Information Processing and Coding Department. She is the author of many articles in reputed international journals of mathematical and engineering sciences, conference papers and several books. She has served as referee for different SCI journals and international conferences. Her current research interests include cryptanalysis, stream ciphers and pseudorandom sequences.