MODELLING THE SHRINKING GENERATOR
IN TERMS OF LINEAR CA

SARA D. CARDELL
Instituto de Matemática, Estatística e Computação Científica
Universidade Estadual de Campinas, R. Sérgio Buarque de Holanda
651 - Cidade Universitária, Campinas - SP, 13083-859, Brazil

AMPARO FÚSTER-SABATER
Instituto de Tecnologías Físicas y de la Información
Consejo Superior de Investigaciones Científicas
C/Serrano 144, 28006, Madrid, Spain

(Communicated by Joachim Rosenthal)

Abstract. This work analyses the output sequence from a cryptographic non-linear generator, the so-called shrinking generator. This sequence, known as the shrunken sequence, can be built by interleaving a unique PN-sequence whose characteristic polynomial serves as basis for the shrunken sequence’s characteristic polynomial. In addition, the shrunken sequence can be also generated from a linear model based on cellular automata. The cellular automata here proposed generate a family of sequences with the same properties, period and characteristic polynomial, as those of the shrunken sequence. Moreover, such sequences appear several times along the cellular automata shifted a fixed number. The use of discrete logarithms allows the computation of such a number. The linearity of these cellular automata can be advantageously employed to launch a cryptanalysis against the shrinking generator and recover its output sequence.

1. Introduction

Private-key ciphers use the same key for encryption and decryption [18]. Thus, the key exchange between the two parts involved in the communication plays an important role. There are two types of private-key ciphers: stream ciphers and block ciphers. This work focuses on the first type. Stream ciphers are the fastest among the encryption procedures so they are implemented in many technological applications. Assume the messages are binary sequences, then stream ciphers encrypt bits individually [20]. The ciphertext is obtained XOR-ing the original message and a generated keystream sequence. The decryption is performed in the same way; XOR-ing the ciphertext and the same keystream sequence. The main problem in stream...
ciphers is to generate from a short key a long keystream sequence that looks as random as possible. Maximal-length Linear Feedback Shift Registers (LFSRs) \[11\] generate PN
dsequences with good cryptographic properties, but their linearity make them vulnerable. Due to the Berlekamp-Massey algorithm \[15\], these PN-
dsequences must never be used as keystream sequences, but as basis for other more complex structures. On the other hand, some one-dimensional linear cellular automata (CA) generate the same PN-sequences as those generated by maximal-length LFSRs \[4\]. Therefore, CA can be considered as alternative generators to LFSRs. Moreover, some keystream generators can be modelled as linear CA \[3, 9, 10\].

In \[9\], the authors showed that the output sequence of a cryptographic sequence generator, the shrinking generator, can be obtained from a linear model based on CA that uses rules 150 and 90. In this work, we propose a different family of CA that uses rules 102 or 60 and generates the same sequence, that is the shrunken sequence. Furthermore, we provide some properties about the shrunken sequence and the interleaved PN-sequences contained in it. We also analyse the other different sequences generated by the CA.

This paper is organized as follows: Section 2 gives some basic notions in order to understand the rest of the work. In Section 3, the main results of this work are introduced. This section is divided into two parts; Subsection 3.1 where some properties about the characteristic polynomial and the interleaved PN-sequences in the shrunken sequence are presented, and Subsection 3.2 where a characterization of the sequences obtained through the linear CA that generate the shrunken sequence is provided. In Sections 4, some conclusions complete the work.

2. Preliminaries

In this section, the main concepts of this work are introduced. In Section 2.1, we remind the notion of shrinking generator and shrunken sequence. In Section 2.2, we also define cellular automata and rules 102 and 60.

2.1. The shrinking generator. Before reminding the definition of shrinking generator, we need to introduce the concept of decimation, which will be used several times throughout this paper. Let \( \{v_i\} \) be a linear recursive sequence over a finite field. The decimation of the sequence \( \{v_i\} \) by \( d \) is a new sequence obtained by taking every \( d \)-th term of \( \{v_i\} \) \[8\].

The shrinking generator was first introduced by Coppersmith et al. in \[5\]. It is composed of two maximal-length LFSRs, \( R_1 \) and \( R_2 \), with lengths \( L_1 \) and \( L_2 \), respectively, and \( \gcd(L_1, L_2) = 1 \). The PN-sequence \( \{a_i\} \) generated by \( R_1 \) decimates the PN-sequence \( \{b_i\} \) produced by the other register \( R_2 \). The decimation rule is very simple; given two bits \( a_i \) and \( b_i \), \( i = 0, 1, 2, \ldots \) from both PN-sequences, the output sequence \( \{s_j\} \) is obtained as follows:

\[
\begin{align*}
\text{If } a_i &= 1 \text{ then } s_j = b_i. \\
\text{If } a_i &= 0 \text{ then } b_i \text{ is discarded.}
\end{align*}
\]

The sequence \( \{s_j\} \) is called the shrunken sequence and its period is \( T = (2^{L_2} - 1)2^{L_1-1} \). The linear complexity of this sequence, denoted by \( LC \), satisfies \( L_22^{L_1-2} < LC \leq L_22^{L_1-1} \) and its characteristic polynomial has the form \( p(x)^m \), where \( 2^{L_1-2} < m \leq 2^{L_1-1} \) and \( p(x) \) is a primitive polynomial of degree \( L_2 \) \[9\]. Besides, the sequence is almost balanced as its number of 1s is \( 2^{L_1+L_2-2} \). This generator is easy
to implement and has good cryptographic properties, so it is suitable for applications in stream ciphers. Notice that the shrunken sequence is obtained by irregular decimation of a PN-sequence.

**Example 1.** Consider \( R_1 \) the LFSR with characteristic polynomial \( p_1(x) = 1 + x + x^2 \) and initial state 10. Consider also \( R_2 \) the LFSR with characteristic polynomial \( p_2(x) = 1 + x + x^3 \) and initial state 100. The shrunken-sequence can be computed in the following way:

\[
\begin{align*}
R_1 : & \quad 1 \ 0 \ 1 \ 1 \ 0 \ 1 \ 1 \ 0 \ 1 \ 1 \ 0 \ 1 \ 1 \ 0 \ 1 \ 1 \ 0 \\
R_2 : & \quad 1 \ \underline{1} \ 0 \ 1 \ \underline{1} \ 1 \ \underline{X} \ 0 \ 0 \ \underline{X} \ 0 \ 1 \ \underline{X} \ 1 \ 0 \ \underline{X} \ 1 \\
\{s_j\} : & \quad 1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 0 \ 1 \ 1 \ 0 \ 1 \ 0 \ 1 \ 1 \ 1
\end{align*}
\]

The sequence has period 14 and it is not difficult to check that its characteristic polynomial is \( p(x)^2 = (1 + x^2 + x^3)^2 \), consequently its linear complexity equals 6.

### 2.2. Cellular Automata

**Cellular automata** (CA) are devices composed of a finite number of cells whose content is updated according to a rule or function with \( k \) variables [6]. The state of the cell in position \( i \) at time \( t + 1 \), notated \( x_{i+1}^t \), depends on the state of the \( k \) neighbour cells at time \( t \). If these rules are composed exclusively of XOR operations, then the CA are linear. Here, the CA we consider are regular (every cell follows the same rule), cyclic (extreme cells are adjacent) and one-dimensional. For \( k = 3 \), rules 102 and 60 are given by:

**Rule 102:** \( x_{i+1}^t = x_i^t + x_{i+1}^t \)

**Rule 60:** \( x_{i+1}^t = x_{i-1}^t + x_i^t \)

The numbers 01100110 and 00111100 are the binary representations of 102 and 60, respectively. This is the reason why they are called rule 102 and rule 60. In Figure 1, it is possible to see these rules using the terminology introduced by Stephen Wolfram [21], where a white square represents the digit 0 and a black square represents the digit 1. Figure 2 shows the AC-images generated by these rules after applying 15 iterations to the one-dimensional CA. It is possible to see the symmetry between both rules. Notice that, according to Wolfram’s terminology, both rules are considered for \( k = 3 \), but with one null coefficient. For rule 102 the coefficient corresponding to the first component is null, that is, \( x_{i+1}^t = 0 \cdot x_{i-1}^t + x_i^t + x_{i+1}^t \).

In Table 1a we find an example of a linear, regular, cyclic cellular automata that uses rule 102. Note that the linear, regular, cyclic automata with the same length that uses rule 60 provides the same sequences, but these appear in reverse order (see Table 1b).

Due to speed and randomness in their sequences, CA are a very good basis for stream ciphers. Furthermore, their hardware implementation is simple and their regular structure makes possible to find an efficient software implementation. The
3. Main results

Let $\mathbb{F}_2$ be the Galois field of 2 elements. From now on, we consider two registers $R_1$ and $R_2$, with characteristic polynomials $p_1(x), p_2(x) \in \mathbb{F}_2[x]$, lengths $L_1$ and $L_2$, $\gcd(L_1, L_2) = 1$, and $T_1 = 2^{L_1} - 1$ and $T_2 = 2^{L_2} - 1$ the periods of the corresponding PN-sequences, respectively. Besides, the PN-sequences generated by both registers are $\{a_i\}$ and $\{b_i\}$, respectively. We assume without loss of generality that $a_0 = 1$.

3.1. Characteristic polynomial and interleaved PN-sequences. In this section, we outline some of the most important properties of the shrunken sequence. In fact, we prove that the shrunken sequence is constructed interleaving a unique PN-sequence. We also determine the characteristic polynomial of the interleaved sequences and that of the shrunken sequence.

**Theorem 3.1.** The sequences obtained decimating by $2^{L_1-1}$ the shrunken-sequence are PN-sequences with period $T_2$. We call these sequences the **interleaved PN-sequences** of the shrunken sequence.

**Proof.** The sequence $\{a_i\}$ contains $2^{L_1-1}$ ones in the first $T_1$ bits. Suppose the location of these ones in $\{a_i\}$ is $\{0, j_1, j_2, \ldots, j_{2^{L_1-1}-1}\}$. Then, according to the decimation rule, the shrunken sequence is given by

$\begin{align*}
\{b_0, b_{j_1}, b_{j_2}, \ldots, b_{j_{2^{L_1-1}-1}}, b_{T_1}, b_{T_1+j_1}, b_{T_1+j_2}, \ldots, b_{T_1+j_{2^{L_1-1}-1}}, b_{(T_2-1)T_1}, \\
b_{(T_2-1)T_1+j_1}, b_{(T_2-1)T_1+j_2}, \ldots, b_{(T_2-1)T_1+j_{2^{L_1-1}-1}}, \ldots\}
\end{align*}$
The sequence \( \{b_0, b_{T_1}, b_{2T_1}, \ldots, b_{(T_2-1)T_1}\} \) is obtained decimating by \( 2^{L_1-1} \) the shrunken sequence. At the same time, this sequence can be obtained decimating by \( T_1 \) the PN-sequence \( \{b_i\} \). Since \( \gcd(L_1, L_2) = 1 \), it is not difficult to check that \( \gcd(T_1, T_2) = 1 \) and according to [11, Ch. 4, Th. 2], this sequence is a PN-sequence with the same period as \( \{b_i\} \), that is, \( T_2 \).

Using the same procedure, if we decimate the shrunken sequence starting in \( b_{j_1} \), for \( i = 1, 2, \ldots, 2^{L_1-1} - 1 \), then we obtain \( 2^{L_1-1} - 1 \) PN-sequences with period \( T_2 \), the remaining interleaved PN-sequences.

It is worth noticing that the \( 2^{L_1-1} \) interleaved PN-sequences of the shrunken sequence correspond to shifted versions of the same PN-sequence.

Now, we introduce a minor result that will be useful to get later results.

**Lemma 3.2.** If \( p(x) \) is the primitive polynomial that generates the interleaved PN-sequences of the shrunken sequence, then the polynomial \( p(x)^2L_1-1 \) generates the shrunken sequence.

**Proof.** Let \( p(x) = \sum_{i=0}^{L_2} p_i x^i \) be the primitive polynomial that generates the interleaved PN-sequences of the shrunken sequence. These sequences have the following form

\[
\{s_j, s_{2L_1-1+j}, s_{2L_1-1+j}, \ldots, s_{(2^{L_2-2})2^{L_1-1}+j}\}, \quad \text{for } j = 0, 1, 2, \ldots, 2^{L_1-1} - 1.
\]

These PN-sequences have to fulfill the linear recurrence

\[
s_{L_22^{L_1-1}+j} = \sum_{i=0}^{L_2-1} p_i s_{2^{L_1-1}+j}, \quad \text{for } j = 0, 1, 2, \ldots, 2^{L_1-1} - 1.
\]

Thus, we deduce that the polynomial \( \sum_{i=0}^{L_2} p_i x^{2^{L_1-1}} \), which is \( p(x)^{2^{L_1-1}} \), generates the shrunken sequence.

Notice that \( p(x)^{2^{L_1-1}} \) always generates the shrunken sequence but this polynomial might not be the characteristic polynomial. In some cases the characteristic polynomial have the form \( p(x)^m \), with \( 2^{L_1-2} < m < 2^{L_2-1} \).

Now, we focus on the form of the primitive polynomial \( p(x) \).

**Theorem 3.3.** The primitive polynomial \( p(x) \) that generates the interleaved PN-sequences of the shrunken sequence can be computed as

\[
p(x) = (x + \alpha^{T_1})(x + \alpha^{2T_1})(x + \alpha^{4T_1}) \cdots (x + \alpha^{2^{L_2-1}T_1}),
\]

where \( \alpha \in \mathbb{F}_2^{L_2} \) is root of \( p_2(x) \).

**Proof.** We know that the characteristic polynomial \( p_2(x) \) can be seen as

\[
p_2(x) = (x + \alpha)(x + \alpha^2)(x + \alpha^4) \cdots (x + \alpha^{2^{L_2}-1}),
\]

where \( \alpha \in \mathbb{F}_{2^{L_2}} \) is a primitive element in this field as well as a root of \( p_2(x) \). We also know that any bit of the PN-sequence \( \{b_i\} \) can be written as

\[
b_i = A_0 \alpha^i + A_2^0 \alpha^{2i} + A_4^0 \alpha^{4i} + \cdots + A_0^{2^j-1} \alpha^{2^j-1},
\]

with \( A_0 \in \mathbb{F}_{2^{L_2}} \) [14]. When \( A_0 = 1 \), we say the PN-sequence is in its characteristic phase.

The first interleaved PN-sequence of the shrunken sequence is

\[
\{b_0, b_{T_1}, b_{2T_1}, \ldots, b_{(T_2-1)T_1}, \ldots\}.
\]
As we saw before, any bit of this new PN-sequence can be calculated as follows

\[ b_{iT_j} = A_0 \alpha^{iT_1} + A_2^2 \alpha^{2iT_1} + A_0^4 \alpha^{4iT_1} + \cdots + A_0^{4^{j-1}} \alpha^{4^{j-1}iT_1}, \]

where \( \alpha^{2^i} (j = 0, 1, 2, \ldots, L_2 - 1) \) are the roots of \( p_2(x) \). If \( u_i = b_{iT_1} \) and \( \beta = \alpha^{T_1} \), then any bit in the PN-sequence \( \{u_i\} \) can be obtained as follows

\[ u_i = A_0 \beta^i + A_0^2 \beta^{2i} + A_0^4 \beta^{4i} + \cdots + A_0^{4^{j-1}} \beta^{4^{j-1}i}. \]

Therefore, the generator polynomial of the PN-sequence \( \{u_i\} \), the first interleaved sequence, is \( (x + \beta)(x + \beta^2)(x + \beta^4) \cdots (x + \beta^{2^{j-1}}) \), or what is the same

\[ p(x) = (x + \alpha^{T_1})(x + \alpha^{2T_1})(x + \alpha^{4T_1}) \cdots (x + \alpha^{2^{j-1}T_1}). \]

Note that \( p(x) \) only depends on \( p_2(x) \) while the polynomial \( p_1(x) \) affects \( m \). In this way, given a fixed polynomial \( p_2(x) \), every primitive polynomial with degree \( L_1 \) would provide the same \( p(x) \).

Given a polynomial \( q(x) = q_0 + q_1x + q_2x^2 + \cdots + q_n x^n \in \mathbb{F}_2[x] \), its reciprocal polynomial is given by \( q^*(x) = q_n + q_{n-1}x + q_{n-2}x^2 + \cdots + q_0 x^n \in \mathbb{F}_2[x] \). For example, the reciprocal polynomial of \( q(x) = 1 + x + x^4 \) is \( q^*(x) = 1 + x^3 + x^4 \). Now, we can introduce the following result.

**Corollary 1.** If \( L_2 = L_1 + 1 \), then the polynomial

\[ p(x) = (x + \alpha^{T_1})(x + \alpha^{2T_1})(x + \alpha^{4T_1}) \cdots (x + \alpha^{2^{j-1}T_1}) \in \mathbb{F}_2[x] \]

is the reciprocal polynomial of \( p_2(x) \).

**Proof.** Let \( \alpha \) be a root of \( p_2(x) \), then \( \alpha^{-1} \) is a root of \( p_2^*(x) \). We know that \( \alpha^{2^{j-1}-1} = \alpha^0 = \alpha \cdot \alpha^{-1} \), thus \( \alpha^{-1} = \alpha^{2^{j-2}} = \alpha^{2(2^{j-1}-1)} \). If we take \( \gamma = \alpha^{2^{j-1}-1} \), then \( \alpha^{-1} = \gamma^2 \) and, as a consequence, \( \gamma^2 \) is root of \( p_2^*(x) \). The remaining roots of \( p_2^*(x) \) are:

\[ \gamma^2, (\gamma^2)^2, (\gamma^2)^4, \ldots, (\gamma^2)^{2^{j-2}-1}. \]

Furthermore, \( (\gamma^2)^{2^{j-2}} = \gamma^{2^{j-1}} = \gamma \) and, therefore, \( \gamma \) is root of \( p_2^*(x) \). Thus, the roots of \( p_2^*(x) \) are:

\[ (\alpha^{2^{j-1}-1}), (\alpha^{2^{j-1}-1})^2, (\alpha^{2^{j-1}-1})^4, (\alpha^{2^{j-1}-1})^8, \ldots, (\alpha^{2^{j-1}-1}), \]

which are at the same time the roots of \( p(x) \). Consequently, \( p(x) = p_2^*(x) \).

### 3.2. Characterization of the CA

In this section we propose a family of linear, regular, periodic CA that generate the shrunken sequence in the 0-th column (the most-left column). We also study the properties of the other vertical sequences generated by this CA.

Consider now a cellular automaton that only uses rule 102. If the sequence in the 0-th column is \( \{w_i\} \), then the structure of the remaining sequences can be seen in Table 2. It is easy to check that the sequences in columns whose indices are \( 2^j \), for \( j = 0, 1, 2, \ldots \), have the form \( \{w_i + w_{i+2}\} \). If we consider rule 60, the same sequences appear but in reverse order.

In this section we study the relation between the different generated sequences in the CA. Let us start with a minor result about PN-sequences.

**Lemma 3.4.** Let \( p(x) \in \mathbb{F}_2[x] \) be a primitive polynomial of degree \( L_2 \). For the PN-sequence \( \{w_i\} \) generated by \( p(x) \), there exists a number \( D \in \{2, 3, \ldots, 2^{L_2} - 2\} \) such that if the sequence \( \{w_i\} \) is shifted \( D \) positions, then the obtained sequence is \( \{w_i + w_{i+1}\} \).
Proof. Any bit of the PN-sequence \{w_i\} can be computed as

\[ w_i = A_0 \alpha^i + A_0^2 \alpha^{2i} + A_0^4 \alpha^{4i} + \cdots + A_0^{2^{t-1}} \alpha^{2^{t-1}i}, \]

where \( A_0 \in \mathbb{F}_{2^t} \), and \( \alpha \in \mathbb{F}_{2^t} \) is a root of \( p(x) \). Then, we have that

\[ w_i + w_{i+1} = A_0 \alpha^i (1 + \alpha) + A_0^2 \alpha^{2i} (1 + \alpha^2) + A_0^4 \alpha^{4i} (1 + \alpha^4) + \cdots + A_0^{2^{t-1}} \alpha^{2^{t-1}i} (1 + \alpha^{2^{t-1}}). \]

Since \( \mathbb{F}_{2^t} \) is a field and \( \alpha \) is a primitive element, the sum of two elements in the field must be another element in the same field, that is, \( 1 + \alpha = \alpha^D \), for some \( D \in \{2, 3, 4, \ldots, 2^L - 2\} \). Therefore,

\[ w_i + w_{i+1} = A_0 \alpha^{i+D} + A_0^2 \alpha^{2i+2D} + A_0^4 \alpha^{4i+4D} + \cdots + A_0^{2^{t-1}} \alpha^{2^{t-1}i+2^{t-1}D} = w_{i+D}, \]

and the lemma is proven.

Below, we remind the notion of Zech logarithm in a finite field.

**Definition 3.5.** Let \( \alpha \in \mathbb{F}_{2^t} \) be a primitive element. The Zech logarithm with basis \( \alpha \) is the application \( Z_\alpha : \mathbb{Z}_{2^t-2} \to \mathbb{Z}_{2^t-2} \cup \{\infty\} \), such that each element \( t \in \mathbb{Z}_{2^t-2} \) corresponds to \( Z_\alpha(t) \), attaining \( 1 + \alpha^t = \alpha^{Z_\alpha(t)} \).

Now, we can introduce the following result.

**Theorem 3.6.** For a PN-sequence \( \{w_i\} \) generated by a primitive polynomial \( p(x) \in \mathbb{F}_2[x] \), there exists a unique \( D \in \{2, 3, \ldots, 2^L - 2\} \) such that \( w_i + w_{i+1} = w_{i+D} \) and this number is \( D = Z_\alpha(1) \).

**Proof.** According to Lemma 3.4 and Definition 3.5, we know that \( D = Z_\alpha(1) \) and this number is unique for the properties of the Galois fields.

From now on, we say that \( D \) is the number associated to the polynomial \( p(x) \) for the translation.

In Table 2 we saw that if we put a PN-sequence \( \{w_i\} \) in the 0-th column, the next sequence corresponds to \( \{w_i + w_{i+1}\} \). We deduce that the remaining sequences are obtained shifting this PN-sequence \( D, 2D, 3D, \ldots \) positions (mod \( 2^L - 1 \)), respectively.

**Example 2.** For the polynomial \( p(x) = 1 + x^2 + x^3 \), we have that \( \alpha^5 = 1 + \alpha \). Therefore, \( D = Z_\alpha(1) = 5 \). If we consider the initial state 100, the PN-sequence has the form 1001110. If this PN-sequence is located in the 0-th column of the CA (see Table 3), then the sequence in the first column is the same PN-sequence, but it starts in position 5. The next PN-sequence (in the second column) starts in

| 102 | 102 | 102 | 102 | 102 | ... | 102 | ...
|-----|-----|-----|-----|-----|-----|-----|-----|
| \( w_0 \) | \( w_0 + w_1 \) | \( w_0 + w_2 \) | \( w_0 + w_1 + w_2 + w_3 \) | \( w_0 + w_4 \) | \( w_0 + w_8 \) | \( w_0 + w_9 \) | \( w_0 + w_10 \) |
| \( w_1 \) | \( w_1 + w_2 \) | \( w_1 + w_3 \) | \( w_1 + w_2 + w_3 + w_4 \) | \( w_1 + w_5 \) | \( w_1 + w_9 \) | \( w_1 + w_{10} \) | \( w_1 + w_{11} \) |
| \( w_2 \) | \( w_2 + w_3 \) | \( w_2 + w_4 \) | \( w_2 + w_3 + w_4 + w_5 \) | \( w_2 + w_6 \) | \( w_2 + w_{10} \) | \( w_2 + w_{11} \) | \( w_2 + w_{12} \) |
| \( w_3 \) | \( w_3 + w_4 \) | \( w_3 + w_5 \) | \( w_3 + w_4 + w_5 + w_6 \) | \( w_3 + w_7 \) | \( w_3 + w_{11} \) | \( w_3 + w_{12} \) | \( w_3 + w_{13} \) |
| \( \vdots \) | \( \vdots \) | \( \vdots \) | \( \vdots \) | \( \vdots \) | \( \vdots \) | \( \vdots \) | \( \vdots \) |
position $2 \cdot 5 \mod 7 = 3$. The PN-sequence in the third column starts in position $3 \cdot 5 \mod 7 = 1$ and so on (see Table 3).

The following result provides the associated number $D^*$ for reciprocal polynomials.

**Corollary 2.** Let $p(x)$, $p^*(x) \in \mathbb{F}_2[x]$ be two reciprocal primitive polynomials with degree $L_2$ and let $D, D^*$ be the associated numbers for the translation, respectively. Then, we have that $D = 2^{L_2} - D^*$.

**Proof.** Let $p(x) = p_0 + p_1 x + p_2 x^2 + \cdots + p_{L_2-1} x^{L_2-1} + p_{L_2} x^{L_2}$, with $p_0 = p_{L_2} = 1$. The first $2^{L_2} - 1$ bits of the PN-sequence $\{w_i\}$ generated by $p(x)$ have the form

\[
\begin{align*}
&\left\{w_0, w_1, w_2, \ldots, w_{L_2-1}, \sum_{i=0}^{L_2-1} p_i w_i, \sum_{i=0}^{L_2-1} p_i w_i^2, \ldots, \sum_{i=0}^{L_2-1} p_i w_i^{L_2-L_2-2}\right\} \cdot \\
&\left\{\sum_{i=0}^{L_2-1} p_i w_i^{L_2-L_2-1}, \sum_{i=0}^{L_2-1} p_i w_i^{L_2-L_2-3}, \sum_{i=0}^{L_2-1} p_i w_i^{L_2-L_2-2}\right\}.
\end{align*}
\]

The reciprocal polynomial has the form $p^*(x) = p_{L_2} + p_{L_2-1} x + p_{L_2-2} x^2 + \cdots + p_1 x^{L_2-1} + p_0 x^{L_2}$. We consider the initial state

\[\{u_0, u_1, u_2, \ldots, u_{L_2-1}\} = \{w_2^{L_2-2}, w_2^{L_2-3}, w_2^{L_2-4}, \ldots, w_2^{L_2-L_2-1}\}.
\]

The next bit of the PN-sequence $\{u_i\}$ can be generated using $p^*(x)$ as follows

\[u_{L_2} = \sum_{i=0}^{L_2-1} p_{L_2-i} u_i = w_2^{L_2-2} + p_{L_2-1} w_2^{L_2-3} + \cdots + p_1 w_2^{L_2-L_2-1} + p_0 w_2^{L_2-2}.
\]

On the other hand, we know that

\[w_2^{L_2-2} = p_0 w_2^{L_2-2} + p_1 w_2^{L_2-2} + \cdots + p_{L_2-1} w_2^{L_2-2}.
\]

From expressions (1) and (2), we deduce that $u_{L_2} = w_2^{L_2-L_2-2}$. Following the same procedure, we obtain that the first $2^{L_2} - 1$ bits of the PN-sequence $\{u_i\}$ generated

| 102 | 102 | 102 | 102 | 102 | 102 |
|-----|-----|-----|-----|-----|-----|
| 1   | 1   | 1   | 0   | 0   | 1   |
| 0   | 1   | 0   | 1   | 1   | 1   |
| 0   | 1   | 1   | 1   | 0   | 0   |
| 1   | 0   | 0   | 1   | 0   | 1   |
| 1   | 0   | 0   | 1   | 1   | 1   |
| 0   | 1   | 0   | 1   | 1   | 0   |
by $p^*(x)$ are
\[
\{ w_{2^{L_2-2}}, w_{2^{L_2-3}}, \ldots, w_{L_2}, w_{L_2-1}, \ldots, w_2, w_1, w_0 \}.
\]

According to Theorem 3.6 there exists a number $D$ such that $w_i + w_{i+1} = w_{i+D}$. Then, $u_i + u_{i+1} = w_i + 2w_{i-2} + w_{i+2} = w_{i+D} + 2w_{i-2} - 3 \mod (2^{L_2-1}) = w_{i+D} - 2$.

Besides, we know that $u_i = w_{2^{L_2-2}-i}$ and thus $u_{i+D-2} = u_{i+2^{L_2}-D}$, that is, $u_i + u_{i+1} = u_{i+2^{L_2}-D}$. Consequently, $D^* = 2^{L_2} - D$.

Next lemma gives the value for the number associated to $p(x)^2$.

**Lemma 3.7.** According to Theorem 3.6, there exists a number $D$ such that $w_i + w_{i+1} = w_{i+D}$, for the PN-sequence $\{w_i\}$ generated by a primitive polynomial $p(x) \in \mathbb{F}_2[x]$. Then, given the polynomial $p(x)^2$, there exists an associated number $D_2 = 2D$, such that $t_i + t_{i+2} = t_{i+D}$, for every sequence $\{t_i\}$ generated by $p(x)^2$.

**Proof.** Let $p(x) = p_0 + p_1 x + p_2 x^2 + \cdots + p_{L_2-1} x^{L_2-1} + x^{L_2}$. The first $2^{L_2-1}$ bits of the PN-sequence $\{w_i\}$ can be obtained as
\[
\left\{ w_0, w_1, w_2, \ldots, w_{L_2-1}, \sum_{i=0}^{L_2-1} p_i t_{2i}, \sum_{i=1}^{L_2} p_i t_{2i-1}, \sum_{i=1}^{2^{L_2-3}} p_{i-2} t_{2i-3} + 2^{L_2-2} \right\}.
\]
There exists $D \in \{2, 3, \ldots, 2^{L_2}-2\}$, such that $w_i + w_{i+1} = w_{i+D}$.

On the other hand, $p(x)^2 = p_0 + p_1 x + p_2 x^2 + \cdots + p_{L_2-1} x^{2L_2-2} + x^{2L_2}$ and the first $2^{L_2-1}$ bits of the sequence $\{t_i\}$ generated by this polynomial can be obtained as
\[
\left\{ t_0, t_1, \ldots, t_{2^{L_2}-2}, t_{2^{L_2}-1}, \sum_{i=0}^{L_2-1} p_i t_{2i}, \sum_{i=1}^{L_2} p_i t_{2i-1}, \sum_{i=1}^{2^{L_2-3}} p_{i-2} t_{2i-3} + 2^{L_2-2} \right\}.
\]
If we take all elements with even index, $\{t_0, t_2, \ldots\}$, and substitute $t_{2i} = u_i$, we have the sequence
\[
\left\{ u_0, u_1, u_2, \ldots, \sum_{i=0}^{L_2-1} p_i u_i, \sum_{i=1}^{L_2} p_i u_{i-1}, \sum_{i=1}^{L_2+1} p_i u_i, \sum_{i=2}^{2^{L_2-3}} p_{i-2} u_{i-2} + 2^{L_2-2} u_i \right\}.
\]
which is a PN-sequence generated by $p(x)$.

On the other hand, if we take elements with odd index, $\{t_1, t_3, \ldots\}$, and substitute $t_{2i+1} = v_i$ and $j = i - 1$, we obtain the PN-sequence
\[
\left\{ v_0, v_1, v_2, \ldots, \sum_{j=0}^{L_2-1} p_j v_j, \sum_{j=1}^{L_2} p_j v_{j-1}, \sum_{j=2}^{L_2+1} p_j v_j, \sum_{j=2}^{2^{L_2-3}} p_{j-2} v_{j-2} + 2^{L_2-2} v_j \right\}.
\]
The polynomial $p(x)$ generates both PN-sequences and, as a consequence, we have that $u_i + u_{i+1} = u_{i+D}$ and $v_i + v_{i+1} = v_{i+D}$. Then, the sequence $\{t_i\}$ generated by $p(x)^2$ can be generated in the following way,
\[
\{ u_0, v_0, u_1, v_1, \ldots, u_D, v_D, u_0, v_0, \ldots \}.
\]
In this case the number $D_2$ associated to $p(x)^2$ is $D_2 = 2D$ and it is easy to check that this sequence is repeated every two columns along the whole CA.

It is possible to extend this proof for higher powers of 2.

**Theorem 3.8.** According to Theorem 3.6, there exists a number $D$ such that $w_i + w_{i+1} = t_{i+D}$, for the PN-sequence $\{w_i\}$ generated by a primitive polynomial $p(x) \in \mathbb{F}_2[x]$. For the polynomial $p(x)^2$, there exists a number $D_2 = 2D$, such that $t_i + t_{i+2^r} = t_{i+D_2}$, for every sequence $\{t_i\}$ generated by this polynomial.

**Proof.** In this case, the same procedure considered in Lemma 3.7 is followed to prove this result.

The characteristic polynomial of the shrunken sequence has the form $p(x)^m$, with $2^{L_1-2} < m \leq 2^{L_1-1}$ and $p(x)$ as given in Theorem 3.3. This means that the polynomial $p(x)^{2^{L_1-1}}$ can also generate the shrunken sequence (see Lemma 3.2). If we locate the shrunken sequence $\{s_j\}$ in the 0-th column of the CA, the $2^{L_1-1}$-th column will have the form $\{s_j + s_{j+2^{L_1-1}}\}$. According to previous results, this sequence is the shrunken sequence starting in position $D_{2L_1-1}$. Therefore, we have that the shrunken sequence appears in the columns whose indices are multiples of $2^{L_1-1}$.

Let us prove now that the characteristic polynomial of the remaining sequences in the CA is $p(x)^m$ as well.

**Theorem 3.9.** The characteristic polynomial of the sequences generated by the CA is that of the shrunken sequence, that is, $p(x)^m$, where $p(x)$ has the form given in Theorem 3.3.

**Proof.** Let $p(x)^m = q_0 + q_1x + q_2x^2 + \cdots + q_{L_2-1}x^{L_2-1} + x^{L_2}$ be the characteristic polynomial of the shrunken sequence

$$\left\{ s_0, s_1, s_2, \ldots, s_{L-1}, \sum_{j=0}^{L-1} q_js_j, \sum_{j=0}^{L-1} q_js_{j+1}, \sum_{j=0}^{L-1} q_js_{j+2}, \ldots \right\}.$$ 

The sequence in the first column of the CA is

$$\left\{ s_0 + s_1, s_1 + s_2 + s_3, \ldots, s_{L-2} + s_{L-1}, s_{L-1} + \sum_{j=0}^{L-1} q_js_j, \sum_{j=0}^{L-1} q_js_{j+1}, \ldots \right\}.$$ 

Assume that $u_j = s_j + s_{j+1}$, then the previous sequence can be seen as

$$\left\{ u_0, u_1, u_2, \ldots, u_{L-1}, \sum_{j=0}^{L-1} q_ju_j, \sum_{j=0}^{L-1} q_ju_{j+1}, \ldots \right\}.$$ 

Consequently, the polynomial $p(x)^m$ can also generate this sequence. However, this sequence could also be generated by another polynomial with lower degree (divisor of $p(x)^m$).

Consider the first $2^{L_1-1}+1$ sequences in the CA, $s^0, s^1, s^2, \ldots, s^{2^{L_1-1}}$. We assume $s^0$ is the shrunken sequence. As we saw above, $p(x)^m$ generates $s^1$ as well, but suppose that $p(x)^k$, with $k < m$ generates $s^1$. For the same procedure used above,
Table 4. CA that generates the shrunken sequence in Example 3

| 0  | 1  | 0  | 1  | 1  | 0  | 1  | 0  | 0  | 1  | 1  | 0  |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0  | 1  | 1  | 0  | 1  | 0  | 1  | 0  | 0  | 1  | 1  | 0  |
| 0  | 0  | 1  | 1  | 1  | 0  | 1  | 1  | 0  | 0  | 1  | 0  |
| 0  | 0  | 1  | 1  | 0  | 1  | 0  | 1  | 1  | 0  | 0  | 1  |
| 0  | 0  | 1  | 0  | 1  | 0  | 1  | 0  | 1  | 0  | 0  | 1  |
| 0  | 1  | 0  | 1  | 0  | 1  | 0  | 1  | 0  | 0  | 1  | 0  |
| 0  | 1  | 0  | 0  | 0  | 0  | 1  | 1  | 0  | 1  | 0  | 1  |
| 0  | 0  | 1  | 1  | 1  | 0  | 1  | 1  | 0  | 1  | 0  | 0  |
| 0  | 0  | 1  | 0  | 1  | 1  | 0  | 1  | 1  | 0  | 0  | 1  |
| 0  | 1  | 0  | 1  | 0  | 0  | 0  | 1  | 1  | 0  | 1  | 0  |

\( p(x)^k \) generates \( s^2 \) as well, and following the same idea \( p(x)^k \) generates \( s^{2^L - 1} \).

However, we know that \( s^{2^L - 1} \) is the same sequence as \( 2^D \), the shrunken sequence, but starting in position \( D_{2^L - 1} = 2^{L - 1}D \), where \( D \) is the number associated to \( p(x) \). This is a contradiction, so every sequence in the CA is generated by \( p(x)^m \).

\[ \begin{align*}
\text{Theorem 3.10.} & \quad \text{The length of the CA that generates the shrunken sequence in the} \\
& \text{0-th column is} \\
& \quad \frac{T}{\text{gcd}(2^{L-1} - 1, D)},
\end{align*} \]

where \( T \) is the period of the shrunken sequence and \( D \) has the form given in Theorem 3.6.

**Proof.** The shrunken sequence appears in columns whose indices are multiples of \( 2^{L-1} \), but starting in positions \( D_{2^{L-1}}, 2D_{2^{L-1}}, 3D_{2^{L-1}}, \ldots \), respectively (mod \( T \), the period of the shrunken sequence). Then, the number of times \( N \) that the shrunken sequence appears before starting again in position 0 must hold \( D_{2^{L-1}} \cdot N = 0 \) (mod \( T \)). Since \( D_{2^{L-1}} < T \), we have that \( N = T / \text{gcd}(T, D_{2^{L-1}}) \). Finally, we have \( 2^{L-1} \) different sequences, including the shrunken sequence, then the length of the CA is \( N \cdot 2^{L-1} = (T \cdot 2^{L-1}) / \text{gcd}(T, D_{2^{L-1}}) \).

Now, taking into account that \( T = 2^{L-1}(2^{L-1} - 1) \) and \( D_{2^{L-1}} = 2^{L-1} \cdot D \) (according to Theorem 3.8), we have that \( \text{gcd}(T, D_{2^{L-1}}) = 2^{L-1}\text{gcd}(2^{L-1} - 1, D) \) and the length of the automata is

\[ \frac{T}{\text{gcd}(2^{L-1} - 1, D)}. \]

\[ \square \]

**Example 3.** Consider the shrunken sequence obtained in Example 1. The characteristic polynomial of this shrunken sequence is \( p(x)^2 = (1 + x^2 + x^3)^2 \). The number associated to \( p(x)^2 \) is \( D_2 = 2 \cdot D = 2 \cdot 5 = 10 \). In Table 4 there is an example of a cyclic regular linear CA that generates this sequence. This CA has length 14 and it generates two different sequences, the shrunken sequence and another sequence, both with the same period 14. These two sequences appear 7 times but translated. For example, the shrunken sequence appears in columns 0, 2, 4, 6, 8, 10. In column 2 the shrunken sequence appears but starting in position 10, in column 4 in position 20 mod 14 = 6 and so on. The other sequence appears seven times as well, in columns, 1, 3, 5, 7, 9, 11, 13 with the same shifts.
3.3. APPLICATIONS. The shrinking generator and other generators based on irregular decimation \([13, 17]\) were created to increase the linear complexity of the PN-sequences. However, in Section 3.1 we have seen that the shrunken sequence generated by two maximal-length LFSRs of length \(L_1\) and \(L_2\), respectively, is composed of \(2^{L_1-1}\) interleaved PN-sequences (which are the same PN-sequence but shifted). Furthermore, in Section 3.2, we can check that the shrunken sequence can be always computed as the output sequence of a linear cellular automata. We can take advantage of such linear structures and their properties to design a cryptanalysis against the shrinking generator. According to the previous ideas, in \([1, 2]\) the authors proposed an attack based on an exhaustive search over the initial state of the first register. Thus, the complexity of the brute-force attack is reduced by a factor \(2^{L_2}\). We see that this generator can be easily broken, which attempts against the security of the information encrypted by this structure.

3.4. COMPARISON WITH 90/150 CA. In \([9]\), the authors proposed a family of linear, null, hybrid CA that also generated the shrunken sequence. These CA were a combination of both rules 90 and 150. Given two maximal-length LFSRs, it is necessary to perform the algorithm given in \([4]\) and carry out a concatenation procedure to find the CA that generates the shrunken sequence. This fact makes it impossible to predict the form of the CA. In this work, since the CA are regular and we only use rule 102 (or 60) we do not need to look for the CA form. On the other hand, the length of our CA is, at the most, \(T = 2^{L_1-1}(2^{L_2} - 1)\), which is greater than \((2^{L_1} - 1)L_2\) (the length of the CA given in \([9]\)). However, our CA generate \(2^{L_1-1}\) different sequences, the remaining sequences in the CA are the same sequences, which is an advantage compared to the 90/150 CA.

As a conclusion, we can say that our CA are longer but this disadvantage becomes less relevant when we notice the complex process developed in \([9]\) to obtain the CA. Besides, this length is reduced to \(2^{L_1-1}\), since we have this number of sequences repeated along the CA.

4. CONCLUSIONS

The introduction of decimation to break the linearity of the PN-sequences has been useless, since the shrunken sequence can be modelled as the output sequence of a linear model based on CA. In this work, we analyse a family of one-dimensional, linear, regular and cyclic CA based on rule 102 that describes the behaviour of the shrinking generator, designed as a non-linear generator. These CA generate a family of sequences with the same characteristic polynomial and period as those of the shrunken sequence. Besides, the shrunken sequence is composed of sequences whose characteristic polynomial serves as basis for the characteristic polynomial of the shrunken sequence. Therefore, this sequence is sensitive to suffer a cryptanalysis that takes advantage of the linearity and similarity between the sequences generated by the CA.

REFERENCES

[1] S. D. Cardell and A. Füster-Sabater, Cryptanalysing the shrinking generator, Proc. Comp. Sci., 51 (2015), 2893–2897.
[2] S. D. Cardell and A. Füster-Sabater, Performance of the cryptanalysis over the shrinking generator, in Int. Joint Conf. CISIS’15 and ICEUTE’15 (eds. A.H. et al.), Springer, 2015, 111–121.
Modelling the shrinking generator in terms of linear CA

[3] S. D. Cardell and A. Fúster-Sabater, Linear models for the self-shrinking generator based on CA, *J. Cell. Autom.*, 11 (2016), 195–211.

[4] K. Cattell and J. C. Muzio, One-dimensional linear hybrid cellular automata, *IEEE Trans. Comp.-Aided Des.*, 15 (1996), 325–335.

[5] D. Coppersmith, H. Krawczyk and Y. Mansour, The shrinking generator, in *Adv. Crypt. – CRYPTO ’93*, Springer-Verlag, 1993, 23–39.

[6] A. K. Das, A. Ganguly, A. Dasgupta, S. Bhawmik and P. P. Chaudhuri, Efficient characterisation of cellular automata, *IEEE Proc. Comp. Dig. Techn.*, 137 (1990), 81–87.

[7] S. Das and D. RoyChowdhury, Car30: A new scalable stream cipher with rule 30, *Crypt. Commun.*, 5 (2013), 137–162.

[8] P. F. Duvall and J. C. Mortick, Decimation of periodic sequences, *SIAM J. Appl. Math.*, 21 (1971), 367–372.

[9] A. Fúster-Sabater and P. Caballero-Gil, Linear solutions for cryptographic nonlinear sequence generators, *Phys. Lett. A*, 369 (2007), 432–437.

[10] A. Fúster-Sabater, M. E. Pazo-Robles and P. Caballero-Gil, A simple linearization of the self-shrinking generator by means of cellular automata, *Neural Netw.*, 23 (2010), 461–464.

[11] S. W. Golomb, *Shift Register-Sequences*, Aegean Park Press, Laguna Hill, California, 1982.

[12] J. Jose, S. Das and D. RoyChowdhury, Inapplicability of fault attacks against trivium on a cellular automata based stream cipher, in *11th Int. Conf. Cell. Autom. Res. Ind. ACRI 2014*, Springer-Verlag, 2014, 427–436.

[13] A. Kanso, Modified self-shrinking generator, *Comp. Electr. Engrn.*, 36 (2010), 993–1001.

[14] R. Lidl and H. Niederreiter, *Introduction to Finite Fields and their Applications*, Cambridge Univ. Press, New York, NY, 1986.

[15] J. L. Massey, Shift-register synthesis and BCH decoding, *IEEE Trans. Inform. Theory*, 15 (1969), 122–127.

[16] W. Meier and O. Staffelbach, Analysis of pseudo random sequences generated by cellular automata, in *Adv. Crypt. – EUROCRYPT ’91*, Springer-Verlag, Berlin, 1991, 186–199.

[17] W. Meier and O. Staffelbach, The self-shrinking generator, in *Adv. Crypt. – EUROCRYPT 1994*, Springer-Verlag, 1994, 205–214.

[18] A. J. Menezes, P. C. van Oorschot and S. A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, Boca Raton, FL, 1996.

[19] M. Mihaljević, Y. Zheng and H. Imai, A fast and secure stream cipher based on cellular automata over GF(q), in *Proc. Global Telecom. Conf. GLOBECOM 1998*, 1998, 3250–3255.

[20] C. Paar and J. Pelzl, *Understanding Cryptography*, Springer, Berlin, 2010.

[21] S. Wolfram, Cellular automata as simple self-organizing system, Calttech preprint, CALT-68-938, 1982.

[22] S. Wolfram, Cryptography with cellular automata, in *Adv. Crypt. – EUROCRYPT 1985*, Springer-Verlag, 1985, 429–432.

Received December 2014; revised June 2016.

E-mail address: sdcardell@ime.unicamp.br
E-mail address: amparo@iec.csic.es