# Linear complexity analysis of multidimensional periodic arrays

Rafael Arce-Nazario, Francis Castro, Domingo Gomez-Perez, Oscar Moreno,
José Ortiz-Ubarri, Ivelisse Rubio, Andrew Tirkel[§]

December 25, 2020

**Abstract**

The linear complexity of a sequence is an important parameter for many applications, especially those related to information security, and hardware implementation. It is desirable to develop a corresponding measure and theory for multidimensional arrays that are consistent with those of sequences. In this paper we use Gröbner bases to develop a theory for analyzing the linear complexity of general multidimensional periodic arrays. We also analyze arrays constructed using the method of composition and establish tight bounds for their linear complexity.

**Keywords:** linear complexity, periodic arrays, multidimensional arrays, multisequences, Gröbner bases

## 1 Introduction

Many researchers in different areas have studied multidimensional arrays and in recent years there has been wide interest in constructions of two-dimensional arrays with good correlation properties [13, 18, 20, 22, 23], see also the recent survey by Gyarmati et al [16]. Auto correlation and cross correlation are generally considered to be important parameters. However, many other multidimensional array parameters exist and their desirable values are determined by the particular application. Multidimensional arrays can be constructed from known sequences. It would be practical to be able to generate multidimensional arrays with desirable properties by constructing them from sequences that possess those properties. For example, for a 2-dimensional array, one can *compose* two sequences, a sequence with good correlation properties and another sequence with a desired linear complexity and also good correlation properties. To construct the array use cyclic shifts of the sequence with the desired complexity as columns, where the shifts are determined by the sequence with good correlation. One hopes to obtain an array that has both good correlation and the expected linear complexity. The method of composition to construct arrays using a shift sequence was used by Tirkel, Osborne and Hall in [36] and was generalized by Moreno and Tirkel in [25]. Arrays constructed using this method have been proved to preserve the good correlation properties [17, 25, 26, 35], and examples suggested that they also preserve complexity properties [17, 26]. One can also visualize a 2-dimensional array as a sequence of sequences, or a multisequence [8, 27], where the array is a sequence of columns.

The linear complexity of a periodic sequence is the minimum length of a linear shift register that generates the sequence, or, equivalently, the degree of the minimal polynomial that generates the sequence, and measures the resistance to a Berlekamp-Massey type of attack. Special two-dimensional arrays, also known

*R. Arce-Nazario, J. Ortiz-Ubarri, I. Rubio are with the Department of Computer Science, University of Puerto Rico, 17 Ave Universidad STE 1701, San Juan, Puerto Rico, 00925–2537, E-mails: rafael.arce@upr.edu, jose.ortiz23@upr.edu, iverubio@gmail.com.

†F. Castro is with the Department of Mathematics, University of Puerto Rico, 17 Ave Universidad STE 1701, San Juan, Puerto Rico, 00925–2537, E-mail: franciscastr@gmail.com.

‡D. Gomez-Perez is with the Faculty of Sciences, University of Cantabria, E-39071 Santander, Spain. Email: domingo.gomez@unican.es.

§A. Tirkel is with Scientific Technology, 8 Cecil St, East Brighton, 3187, Victoria, Australia. Email: atirkel@bigpond.net.au.

$$\mathbf{a} = \begin{array}{|c|c|c|c|c|} \hline & & \vdots & & \\ \hline a_{0,3} & a_{1,3} & a_{2,3} & a_{3,3} & \\ \hline a_{0,2} & a_{1,2} & a_{2,2} & a_{3,2} & \\ \hline a_{0,1} & a_{1,1} & a_{2,1} & a_{3,1} & \cdots \\ \hline a_{0,0} & a_{1,0} & a_{2,0} & a_{3,0} & \\ \hline \end{array}$$

Figure 1: Labeling of elements in array.

as Cellular automata, have been used to generate sequences used in cryptography, see [3, 6]. These results are based on properties of the linear shift register that generates the two dimensional arrays. We remark that the proofs depend heavily on the linear shift register and there are no attacks using multidimensional arrays.

It is natural to look for a corresponding measure and theory for multidimensional arrays that are consistent with the existing ones for sequences. The analysis of arrays visualized as multisequences is done by computing the joint linear complexity of the multisequence: the degree of the minimal polynomial that generates all the individual sequences in the array [5, 8, 9, 21]. This analysis does not take into consideration possible relations among the sequences (columns) that constitute the array. In [14, 15], Gyarmati, Mauduit and Sárközy proposed a new definition for the linear complexity of two-dimensional binary lattices. The authors mention explicitly that this complexity is difficult to calculate. In [26], the complexity analysis of 2-dimensional arrays was done by "unfolding" the array into a sequence using a method based on the Chinese Remainder Theorem (CRT), and then analyzing the linear complexity of the sequence using the Berlekamp-Massey Algorithm. The use of the CRT imposed two restrictions on the arrays: they must be periodic in the two dimensions, and the periods of the array must be coprime, i. e. the period in one dimension has to be relatively prime to the period in the other dimension.

In [12, 24], a new general theory for analyzing the linear complexity of multidimensional periodic arrays was introduced, providing a definition and a method to compute multidimensional linear complexity that is consistent with the one-dimensional definition and the unfolding method. This analysis takes into consideration relations between the entries of the array, including relations across the sequences that constitute the columns. The new definition and method do not require the periods of the array to be coprime and there is no restriction on the dimension of the array; the only restriction is that each period is not divisible by the characteristic of the field. In addition, the algorithm used in the proposed scheme is based on linear algebra calculations and can be implemented easily. In this paper we develop this theory further and analyze arrays constructed with the method of composition [25, 26, 36], presenting tight bounds for their linear complexity.

## 1.1 Periodic arrays and recurrence relations

Let $\mathbb{F}_q$ be the finite field with $q = p^r$ elements, $\mathbb{F}_q[\mathbf{X}] = \mathbb{F}_q[X_1, \ldots, X_m]$ be the ring of polynomials in $m$ variables and coefficients in $\mathbb{F}_q$, and $\mathbb{N}_0 = \{0, 1, 2, \ldots\}$. Let $\mathbf{a} = (a_{i_1,\ldots,i_m}) \subset \mathbb{F}_q^{\mathbb{N}_0^m}$ be an $m$-dimensional infinite array. A 1-dimensional array is a sequence. To represent 2-dimensional arrays as matrices, we label the rows from bottom to top and the columns from left to right, as in Figure 1. This representation might seem unusual but the choice is convenient when one relates $m$-dimensional finite arrays to polynomials in $m$-variables, and infinite arrays to power series. For example, the 2-dimensional array in Figure 1 corresponds to the power series in the variables $X, Y$, where $a_{i,j}$ is the coefficient of the term with monomial $X^iY^j$ :

$$A(X, Y) = a_{0,0} + a_{0,1}Y + a_{1,0}X + a_{0,2}Y^2 + a_{1,1}XY + a_{2,1}X^2 + \cdots.$$

**Definition 1** *An $m$-dimensional array* $\mathbf{a}$ *is said to be* $m$**-dimensional periodic** *if there is a $m$-tuple, that we call the* **period vector**, $n = (n_1, \ldots, n_m) \in \mathbb{N}^m$, *such that*

$$a_{(\alpha_1,\ldots,\alpha_m)} = a_{(\alpha_1+n_1k_1,\ldots,\alpha_m+n_mk_m)}$$

*for* $k_i \in \mathbb{N}_0$ *and all* $(\alpha_1, \ldots, \alpha_m) \in \mathbb{N}_0^m$.

The definition of linear complexity of arrays that we present in Section 2 depends on the polynomials that define linear recursion relations on the array and the notion of *lead exponents*. Consider the exponent $\alpha = (\alpha_1, \ldots, \alpha_m) \in \mathbb{N}_0^m$ and set $\mathbf{X}^\alpha = X_1^{\alpha_1} X_2^{\alpha_2} \cdots X_m^{\alpha_m}$. Suppose that we have a monomial ordering (see Section 1.2) in the monomials of $\mathbb{F}_q[\mathbf{X}]$ and denote by $LE(C)$ the exponent of the leading monomial of the polynomial $C \in \mathbb{F}_q[\mathbf{X}]$. Also set $\alpha \leq \beta$ if and only if $\alpha_i \leq \beta_i$ for $i = 1, \ldots, m$; this is not a monomial ordering but it defines a partial order in the monomials.

The periodicity of the array gives recurrence relations among its entries and the entire infinite array can be then generated by a finite subarray and the recurrence relations. However, there might be other recurrence relations on the array. These recurrence relations are satisfied by polynomials. Let

$$C(\mathbf{X}) = \sum_{\alpha \in Supp(C)} c_\alpha \mathbf{X}^\alpha \in \mathbb{F}_q[\mathbf{X}], \text{ where } Supp(C) := \{\alpha \mid c_\alpha \text{ is a non-zero coefficient of } C\}.$$

**Definition 2** *Let* $\alpha, u \in \mathbb{N}_0^m$. *The polynomial* $C$ *defines a linear recurrence relation at a point* $a_u$ *of the array* $a$ *if* $LE(C) \leq u$ *and*

$$\sum_{\alpha \in Supp(C)} c_\alpha a_{\alpha + u - LE(C)} = 0. \tag{1}$$

*In this case we say that* $C$ **is valid at the point** $a_u$. *Also set* $C$ *to be valid at* $a_u$, *if* $LE(C) \not\leq u$.

**Definition 3** *A polynomial* $C$ **is valid for the array a** *if the equation*

$$\sum_{\alpha \in Supp(C)} c_\alpha a_{\alpha + \beta} = 0 \tag{2}$$

*holds for all* $\beta \in \mathbb{N}_0^m$. *In this case we also say that* **a satisfies the** $m$**-dimensional linear recurrence relation given by** $C$.

Note that $C$ is a valid polynomial for $\mathbf{a}$ if and only if $C$ is a valid polynomial at every point $a_u$ such that $LE(C) \leq u$. The valid polynomials for a sequence $(s_j)$ are called *characteristic polynomials*, and the unique monic characteristic polynomial of minimal degree is called the *minimal polynomial* of $(s_j)$.

Let $Val(\mathbf{a})$ denote the set of all valid polynomials for the array $\mathbf{a}$. It is easy to check that $Val(\mathbf{a})$ is an ideal in $\mathbb{F}_q[\mathbf{X}]$. If $\mathbf{a}$ is an $m$-dimensional periodic array with period vector $n = (n_1, \ldots, n_m)$, as the ones considered here, then it is also easy to see that $Val(\mathbf{a})$ contains the polynomials $X_1^{n_1} - 1, X_2^{n_2} - 1, \ldots, X_m^{n_m} - 1$, and hence the algebraic variety defined by $Val(\mathbf{a})$ is zero dimensional. In this paper we assume that $p \not| n_i$ for $i = 1, \ldots, m$, and hence $Val(\mathbf{a})$ is a radical ideal. The infinite array $\mathbf{a} = (a_{i_1, \ldots, i_m})$ can be generated using a set of generators for $Val(\mathbf{a})$ and the finite subarray $(a_{i_1, \ldots, i_m})_{i_j < n_j}$ determined by the period vector $n = (n_1, \ldots, n_m)$.

Sakata [33] studied the relation between periodic two dimensional arrays, linear recurrence relations and ideals. He proved that if an array $\mathbf{a}$ is 2-dimensional periodic, then the ideal of linear recurrence relations valid on $\mathbf{a}$ is 0-dimensional, a fact that also follows from the paper of Gianni et al. [11]. Later, Sakata [34] generalized the well known Berlekamp-Massey algorithm [19] to compute the minimal polynomial generating a sequence to an algorithm to compute a Gröbner basis for the ideal $Val(\mathbf{a})$ of valid polynomials generating the multidimensional array $\mathbf{a}$.

The problem of finding linear recurrence relations on multidimensional periodic arrays was reinterpreted in [31, 32], and the approach included an algorithm based on linear algebra that can be implemented easily and is suitable for the applications to compute multidimensional linear complexity presented here.

## 1.2 A brief introduction to Gröbner Bases

To define the linear complexity of multidimensional periodic arrays we first need to review some concepts from Gröbner bases. An excellent reference for more details is [4].

The partial order $\alpha \leq \beta$ defined in the previous section is the partial order of divisibility where $\mathbf{X}^\alpha | \mathbf{X}^\beta$ if and only if $\alpha \leq \beta$. We say that a monomial $\mathbf{X}^\alpha = X_1^{\alpha_1} X_2^{\alpha_2} \cdots X_m^{\alpha_m}$ in a set of monomials is minimal with respect to $\leq$ if there is no other monomial $\mathbf{X}^\beta$ in the set with $\beta < \alpha$.

We say that $<_T$ is a **monomial order** if it is a well ordering in $\mathbb{N}_0^m$ such that $<_T$ is a total order, and $\alpha <_T \beta$ implies that $\alpha + \gamma <_T \beta + \gamma$ for $\alpha, \beta, \gamma \in \mathbb{N}_0^m$. Note that divisibility is not a total order and hence is not a monomial order, but it is compatible with any monomial order in the sense that $\mathbf{X}^\alpha | \mathbf{X}^\beta$ implies that $\mathbf{X}^\alpha \leq_T \mathbf{X}^\beta$.

Define $|\alpha| = \sum_{i=1}^m \alpha_i$. Two common examples of monomial orders are the **lexicographical order**, where $\alpha <_{lex} \beta$ if in $\beta - \alpha$ the left most non-zero entry is positive, and the **graded lexicographical order**, where $\alpha <_{grlex} \beta$ if $|\alpha| < |\beta|$ or if $|\alpha| = |\beta|$ and $\alpha <_{lex} \beta$. We will use the following notation:

Let $\mathbb{F}$ be any field and $C = \sum_\alpha c_\alpha \mathbf{X}^\alpha$ be a nonzero polynomial with each $c_\alpha \neq 0$ and $I \subset \mathbb{F}[\mathbf{X}]$. Then,

1. $LE(C) = leadexp(C)$ is the largest exponent vector $\alpha$ in $C$ with respect to $<_T$.
2. $LM(C)$ denotes the leading monomial of $C$ and it equals $\mathbf{X}^{LE(C)}$.
3. $LC(C)$ denotes the coefficient of $LM(C)$. In other words, the so called leading term of $C$ is $LC(C)LM(C)$.
4. $LE(I) := \{LE(C) \mid 0 \neq C \in I\} \subseteq \mathbb{N}_0^m$. (Note that if $I = \{0\}$, then $LE(I) = \{\}$.)
5. $LM(I) := \{LM(C) \mid 0 \neq C \in I\} = \{\mathbf{X}^\alpha \mid \alpha \in LE(I)\}$. (If $I = \{0\}$, then $LM(I) = \{\}$.)

**Definition 4** *Let $G = \{G_1, \ldots, G_l\} \subset I$, $I$ an ideal in $\mathbb{F}[\mathbf{X}]$. $G$ is a* **Gröbner basis** *for $I$ with respect to $<_T$ if $\langle LM(G_1), \ldots, LM(G_l)\rangle = \langle LM\,(I)\rangle$. If $LC(G_i) = 1$ for $i = 1, \ldots, l$ and $LM(G_i)$ does not divide any term of $G_j$ for $i \neq j$, then $G$ is a* **reduced Gröbner basis** *for $I$ with respect to $<_T$.*

The key concept for the definition and computation of the linear complexity of multidimensional arrays is the concept of a *delta set*. A set $\Delta \subset \mathbb{N}_0^m$ is called a **delta set** if it satisfies 1. in the following lemma:

**Lemma 1** *Let $\Delta, \Gamma \subset \mathbb{N}_0^m$ be set theoretic complements. The following conditions are equivalent:*

1. *For $\beta \in \Delta, \alpha \in \mathbb{N}_0^m$, if $\alpha \leq \beta$ then $\alpha \in \Delta$.*
2. *For $\alpha \in \Gamma, \beta \in \mathbb{N}_0^m$, if $\alpha \leq \beta$ then $\beta \in \Gamma$.*

Obviously the set of exponents of all monomials which occur as leading monomials of an ideal $I$ satisfies 2. of Lemma 1. Hence, the set of exponents of all monomials that do not occur as leading monomials of an ideal $I$ is a delta set (these monomials are called *standard monomials* in [1] and also *excluded point set*). We will denote the delta set of the ideal $I$ as $\Delta_I$. So, $\Delta_I = \mathbb{N}_0^m \backslash LE(I)$.

It is a standard result that a Gröbner basis for an ideal generates the ideal. Also $G = \{G_1, \ldots, G_l\} \subset I$ is a Gröbner basis for $I$ if and only if for any $C \in I$, $LM(G_i) | LM(C)$ for some $G_i \in G$ (see [4]). This implies that, to compute the delta set of an ideal with respect to a monomial order, we just need to compute a Gröbner basis for the ideal with respect to that monomial order and look for the monomials that are not divisible by any $G_i \in G$.

**Proposition 1** *Let $G = \{G_1, \ldots, G_l\} \subset I$ be a Gröbner basis for an ideal $I$ with respect to a monomial order $<_T$. Then,*

$$\Delta_I = \{\alpha \in \mathbb{N}_0^m \mid LM\,(G_i) \nmid \mathbf{X}^\alpha, G_i \in G\}\,.$$

Of course the delta set of an ideal depends on the specific monomial order chosen. However, the size of a delta set is invariant under monomial orderings as we proceed to explain.

When we talk about the *dimension* of an ideal $I \subset R$ we mean the *Krull dimension* of the ring $R/I$. $I \subset \mathbb{F}[\mathbf{X}]$ is a 0-dimensional ideal if and only if $\mathbb{F}[\mathbf{X}]/I$ has finite dimension as a $\mathbb{F}$-vector space. In this case, the dimension of $\mathbb{F}[\mathbf{X}]/\sqrt{I}$, where $\sqrt{I}$ is the radical of $I$, is also the number of common roots in the algebraic closure of $\mathbb{F}$ of the polynomials in $I$. The set of monomials which do not occur as leading monomials of elements in $I$ map one to one to a basis of $\mathbb{F}[\mathbf{X}]/I$. So, $I$ is a 0-dimensional ideal if and only if $\Delta_I$ is a finite set. Since the $\Delta_I$ corresponds to a basis of $\mathbb{F}[\mathbf{X}]/I$ as a $\mathbb{F}$ vector space, its size does not depend on the specific monomial order chosen. This invariance is an important fact for the definition of the linear complexity of multidimensional arrays.

Algorithms for computing Gröbner bases usually assume the knowledge of some basis for the ideal. It is important to note that we are interested in computing the linear complexity of multidimensional periodic arrays and for this we need to compute a Gröbner basis for the ideal of linear recurrence relations on the array without having a generating set for the ideal. Hence, we need algorithms that do not assume the knowledge of a basis for the ideal, like the ones given in [31, 32, 34].

# 2 Linear Complexity

In order to use arrays in cryptography, the arrays need to be robust to attacks and there is a need to measure the complexity of the arrays. For sequences (one-dimensional arrays), the **linear complexity** is defined as the degree of the minimal polynomial, or *feedback polynomial* [19], that generates the sequence, if there is any [7, 27]. If no polynomial generates the sequence, the linear complexity is defined to be $\infty$. If a sequence is periodic, the minimal polynomial always exists, has degree less or equal to the period, and the linear complexity measures how resistant the sequence is to a Berlekamp-Massey type of attack. The linear complexity of random sequences has been studied by Niederreiter in [28].

For two-dimensional periodic arrays where the periods are relatively prime, the linear complexity can be measured by "unfolding" the array into a sequence by using a method based on the CRT, followed by the Berlekamp-Massey algorithm. This method imposes a restriction as it can only be applied to arrays whose periods are relatively prime. The linear complexity of arrays constructed from multisequences can be measured by computing the *joint minimal polynomial* of the sequences. The joint minimal polynomial gives the recurrence relations among the entries within each of the columns, but does not take into consideration possible relations among the different columns that constitute the array. The joint linear complexity of random multisequences have been studied by several authors [10, 29, 30] and tends to increase with the number of sequences. Definition 7, introduced in [12, 24] and analyzed here, works for general multidimensional periodic arrays and Gröbner bases methods can be used to compute it.

The set of all the polynomials that generate a periodic sequence form an ideal, and, since the polynomials are in one variable, this ideal is principal and is generated by the minimal polynomial of the sequence. The arrays considered in this work are periodic and hence there are multivariate polynomials that generate the arrays. The set of all the polynomials that generate the array forms an ideal, but, since the polynomials are multivariate, the ideal might be generated by a set of polynomials instead of just one polynomial. In [12, 24] the theory of Gröbner bases was used to extend the concept of linear complexity of sequences to multidimensional periodic arrays, providing a definition and a method to compute multidimensional linear complexity that is consistent with the one-dimensional definition and the unfolding method, and does not have restrictions among the periods of the array or its dimension.

We now provide all the details for the definition of *multidimensional linear complexity*.

## 2.1 Linear complexity of periodic sequences

As we mentioned before, for a periodic sequence $\mathbf{s}$, the **linear complexity** $\mathcal{L}(\mathbf{s})$ is defined as the degree of the minimal polynomial $M(X)$ that generates the sequence. This is the same as the number of monomials

that are not divisible by the lead monomial of $M(X)$. The set of polynomials that generate the sequence, the valid polynomials, form an ideal, and the ideal is generated by $M(X)$. This is, $Val(\mathbf{s}) = \langle M(X) \rangle$. The minimal polynomial $M(X)$ of the sequence $\mathbf{s}$ can be obtained using the Berlekamp-Massey algorithm and the first $2d$ entries of the array, where $d = deg\,(M(X))$. Of course, one does not know $d$ in advance, but, since the sequence is periodic with period $n$, one knows that $X^n - 1 \in Val(\mathbf{s})$ is a characteristic polynomial, $d \leq n$, and hence can use the first $2n$ entries of $\mathbf{s}$ to compute the minimal polynomial. The linear complexity of the sequence is the degree of the minimal polynomial, which is also the number of monomials that are not lead monomials of any element in the ideal. This is, $\mathcal{L}(\mathbf{s}) = \left|\Delta_{Val(\mathbf{s})}\right|$, where $\Delta_{Val(\mathbf{s})}$ is the set defined after Lemma 1.

## 2.2 Linear complexity of multisequences

As we mentioned before, one can visualize a 2-dimensional array as a sequence of sequences, or a **multisequence** where the array is a sequence of columns [8, 27].

**Definition 5** *For an arbitrary $m \in \mathbb{N}$, an $m$-fold* **multisequence** $\mathbf{S} = (\mathbf{s}_1, \ldots, \mathbf{s}_m)$ *over $\mathbb{F}_q$ is a string of parallel sequences $\mathbf{s}_1, \ldots, \mathbf{s}_m$ over $\mathbb{F}_q$.*

The **joint minimal polynomial** of the multisequence $\mathbf{S}$ is the unique monic polynomial $M \in \mathbb{F}_q[X]$ of smallest degree which is a characteristic polynomial of $\mathbf{s}_i$ for all $1 \leq i \leq m$. The **joint linear complexity** of $\mathbf{S}$ is the degree of $M$. If each of the sequences has period $n$, the joint linear complexity $\mathcal{L}(\mathcal{M}\mathbf{S})$ of the multisequence $\mathbf{S}$ is given by

$$\mathcal{L}(\mathcal{M}\mathbf{S}) = n - \deg\left(\gcd\left(X^n - 1, \mathbf{S}_1(X), \ldots, \mathbf{S}_m(X)\right)\right),$$

where $\mathbf{S}_i(X) = \sum_{j=0}^{n} s_j X^j$ is the polynomial associated to the sequence $\mathbf{s}_i$ [5, 21]. This minimal polynomial is also in the ideal of all valid polynomials in the array. It can be found explicitly given a Gröbner basis of the ideal as we will note after Example 1.

## 2.3 Linear complexity of two-dimensional arrays

In Definition 5 of [14], Gyarmati et al. propose a definition for the linear complexity of two-dimensional arrays over $\mathbb{F}_2$. For the special case of periodic arrays, the definition can be written as follows:

**Definition 6** *Let $\mathbf{a}$ be a two-dimensional periodic array and $Val(\mathbf{a})$ be the ideal of recurrence relations valid on the array. The* **two-dimensional linear complexity** $\mathcal{L}_G(\mathbf{a})$ *of the array $\mathbf{a}$ is defined as*

$$\mathcal{L}_G(\mathbf{a}) = \min\left\{(\deg_X(F) + 1)(\deg_Y(F) + 1) - 1 \mid F(X,Y) \in Val(\mathbf{a})\right\}.$$

The intuitive idea behind this definition is a generalization of the one dimensional case for $q = 2$. The authors remarked that this complexity is difficult to calculate explicitly. In a second paper [15], they calculated the expected value for a random sequence. One can use a Gröbner basis for $Val(\mathbf{a})$ to ease the calculations for $\mathcal{L}_G(\mathbf{a})$. Note that if $GB = \{G_1, \ldots, G_l\}$ is a Gröbner basis for $Val(\mathbf{a})$ with respect to $<_T$, then

$$\mathcal{L}_G(\mathbf{a}) \leq M_{GB} = \min\left\{\left(\deg_X(G_i) + 1\right)\left(\deg_Y(G_i) + 1\right) - 1 \mid G_i \in GB\right\}.$$

To compute the exact value of $\mathcal{L}_G(\mathbf{a})$ one considers all the monomials $M$ with $\left(\deg_X(M) + 1\right)\left(\deg_Y(M) + 1\right) - 1 < M_{GB}$. If $\mathcal{L}_G(\mathbf{a}) < M_{GB}$, then there exists a polynomial depending on these monomials that is an element of $Val(\mathbf{a})$. This can be checked reducing the candidates by the Gröbner basis.

## 2.4 Linear complexity of multidimensional periodic arrays

In the case of a multidimensional array $\mathbf{a}$, the set of polynomials that generate the array, the valid polynomials, form the ideal $Val(\mathbf{a})$. Since this ideal might be generated by more than one polynomial, the natural generalization of the concept of linear complexity to a **multidimensional linear complexity** $\mathcal{L}(\mathbf{a})$ is to define it as the number of monomials that are not divisible by the lead monomial of any element in the ideal $Val(\mathbf{a})$. This is, the number of elements in the delta set $\Delta_{Val(\mathbf{a})}$ that can be obtained by computing a Gröbner basis for $Val(\mathbf{a})$ as stated in Proposition 1. This definition is consistent with the definition of the linear complexity of sequences (one-dimensional arrays) and it is invariant under monomial orderings [12, 24]. The computation of a Gröbner basis for a multivariate polynomial ideal is the generalization of the Berlekamp-Massey algorithm for univariate polynomials.

**Definition 7** *Let* $\mathbf{a}$ *be a multidimensional periodic array and* $Val(\mathbf{a})$ *be the ideal of recurrence relations valid on the array. We define the* **multidimensional linear complexity** $\mathcal{L}(\mathbf{a})$ *of the array* $\mathbf{a}$ *as the size of the delta set of* $Val(\mathbf{a})$*; this is,* $\mathcal{L}(\mathbf{a}) = \left|\Delta_{Val(\mathbf{a})}\right|$.

As it was discussed after Lemma 1, the delta set of an ideal depends on the specific monomial order considered, but the size of the delta set of an ideal is invariant. To compute the linear complexity of an array $\mathbf{a}$, we just need to compute a Gröbner basis with respect to any monomial ordering $<_T$. For example, one could use the *graded lexicographic* ordering $<_T=<_{grlex}$ and all the entries $u = (u_1, \ldots, u_m)$ of $a$ with $|u| \leq 2\,(n_1 + \cdots + n_m) - m$ (See Proposition 1.2 of [31]).

**Scheme to compute the linear complexity $\mathcal{L}(\mathbf{a})$ of a multidimensional periodic array $\mathbf{a}$**

1. Choose a monomial ordering $<_T$.
2. Compute a Gröbner basis with respect to $<_T$ for $Val(\mathbf{a})$, the ideal of linear recurrence relations in the array $\mathbf{a}$, using Sakata's algorithm [34] or the Rubio-Sweedler-Taylor (RST) algorithm [31].
3. The set of exponents of monomials that are not divisible by the lead monomials of the elements in the Gröbner basis form the delta set $\Delta_{Val(\mathbf{a})}$ (Proposition 1).
4. $\mathcal{L}(\mathbf{a}) = \left|\Delta_{Val(\mathbf{a})}\right|$.

The RST algorithm for computing a Gröbner basis for the ideal of valid polynomials for the array $\mathbf{a}$, $Val(\mathbf{a})$, is based on linear algebra computations and can be implemented easily. The first step of the algorithm is to order the elements of the array $\mathbf{a}$ with respect to a chosen monomial ordering. When using the method based on the CRT to compute the linear complexity of an array $\mathbf{a}$, the "unfolding" turns the array into a sequence and therefore it is giving a total order to the elements of the array. This is, the CRT defines a monomial ordering on the elements of $\mathbf{a}$. For arrays with dimensions that are relatively prime, both methods are equivalent and produce the (same) number of monomials that are not divisible by the lead monomials of the polynomials valid in the array. Hence, the definition of the multivariate linear complexity is consistent with the process of "unfolding" to obtain the minimal polynomial. The Gröbner bases method has the advantage that it can be used in any multidimensional periodic array, not only those with dimensions that are relatively prime as it happens with the "unfolding" method.

To be able to compare the complexity of periodic arrays $\mathbf{a}$ of different sizes, we use the *normalized linear complexity*, where the complexity $\mathcal{L}(\mathbf{a})$ is divided by the size of the subarray $(a_{i_1,\ldots,i_m})_{i_j<n_j}$ defined by the period vector $(n_1, \ldots, n_m)$. The normalized linear complexity is also referred in other references [24] as the *relative linear complexity*.

**Definition 8** *Let* $\mathbf{a}$ *be a multidimensional periodic array and* $(n_1, n_2, \ldots, n_m)$ *be its period vector. We define the* **normalized linear complexity** $\mathcal{L}_n(\mathbf{a})$ *of the array* $\mathbf{a}$ *as* $\mathcal{L}_n(\mathbf{a}) = \mathcal{L}(\mathbf{a})/n_1 n_2 \cdots n_m$.

For a periodic array $\mathbf{a}$ with period vector $(n_1, \ldots, n_m)$, we know that $X_1^{n_1} - 1, X_2^{n_2} - 1, \ldots, X_m^{n_m} - 1 \in Val(\mathbf{a})$ and hence the size of the delta set $\Delta_{Val(\mathbf{a})}$ is bounded by $n_1 \cdot n_2 \cdots n_m$. This gives trivial upper bounds on the linear complexity and the normalized linear complexity of any periodic array.

**Proposition 2** *Let* $\mathbf{a}$ *be a periodic array with period vector* $(n_1, \ldots, n_m)$ *and* $\mathcal{L}(\mathbf{a})$ *be the linear complexity of* $\mathbf{a}$*. Then,* $\mathcal{L}(\mathbf{a}) \leq n_1 \cdot n_2 \cdots n_m$*.*

**Corollary 1** *Let* $\mathbf{a}$ *be a periodic array with period vector* $(n_1, \ldots, n_m)$ *and* $\mathcal{L}(\mathbf{a})$ *be the linear complexity of* $\mathbf{a}$*. Then,* $\mathcal{L}_n(\mathbf{a}) \leq 1$*.*

Since the normalized linear complexity of an array considers relations among all the entries of the array, it will always be less or equal to the normalized joint linear complexity of the array (visualized as a multisequence). The normalized linear complexity gives a better analysis of the complexity of the array.

**Example 1** *Consider the binary array in Figure 2. The construction of this array is detailed in Example 2.*

$\mathbf{a} =$

| | | | | | | |
|---|---|---|---|---|---|---|
| *6* | 0 | 0 | 1 | 0 | 1 | 1 |
| *5* | 1 | 1 | 0 | 0 | 1 | 0 |
| *4* | 0 | 1 | 1 | 0 | 0 | 0 |
| *3* | 1 | 0 | 1 | 1 | 0 | 0 |
| *2* | 1 | 0 | 0 | 0 | 0 | 1 |
| *1* | 0 | 0 | 0 | 1 | 1 | 0 |
| *0* | 0 | 1 | 0 | 1 | 0 | 1 |
| | *0* | *1* | *2* | *3* | *4* | *5* |

Figure 2: Binary array.

*We now compute the normalized linear complexity of* $\mathbf{a}$*. The reduced Gröbner basis for* $Val(\mathbf{a})$ *with respect to the graded lexicographical order and* $X > Y$ *is:*

$$G = \left\{X^6 - 1, XY^3 + XY^2 + X + Y^3 + Y^2 + 1, Y^4 + Y^2 + Y + 1\right\},$$

*and* $|\Delta_{Val(\mathbf{a})}| = 19$*. Hence,* $\mathcal{L}_n(\mathbf{a}) = \frac{19}{42} \sim 0.45$*. It is also easy to check that* $\mathcal{L}_G(\mathbf{a}) = 4$*.*

*To compute the normalized linear complexity of array* $\mathbf{a}$ *visualized as a multisequence we compute the* gcd *of* $Y^7 - 1$ *and the polynomials associated to the columns,* $Y^5 + Y^3 + Y^2, Y^5 + Y^4 + 1, Y^6 + Y^4 + Y^3, Y^3 + Y + 1, Y^6 + Y^5 + Y, Y^6 + Y^2 + 1$*, to obtain that* $Y^3 + Y + 1$ *is the* gcd*, and* $\mathcal{L}_n(\mathcal{M}\mathbf{a}) = \frac{7-3}{7} \sim 0.57$*. Note that we divide by the length of the sequences that compose the multisequence.*

*Note that, since* $\mathcal{L}_n(\mathcal{M}\mathbf{a})$ *is not taking into account the recurrence relations among all the entries in array* $\mathbf{a}$*,* $\mathcal{L}_n(\mathbf{a}) < \mathcal{L}_n(\mathcal{M}\mathbf{a})$*. Hence, the normalized linear complexity in Definition 8 gives a more accurate understanding of the complexity of the array. Indeed, it is quite simple to construct arrays* $\mathbf{a}$ *with "low" normalized linear complexity* $\mathcal{L}_n(\mathbf{a})$ *but high normalized joint linear complexity* $\mathcal{L}_n(\mathcal{M}\mathbf{a})$*. Considering an array* $\mathbf{a}$ *generated by the composition method with constant shift of a sequence* $\mathbf{s}$ *gives arrays with normalized joint linear complexity equal to the normalized linear complexity of the sequence,* $\mathcal{L}_n(\mathcal{M}\mathbf{a}) = \mathcal{L}_n(\mathbf{s})$*, but the normalized linear complexity is the normalized joint linear complexity divided by the number of rows,* $n_2$*, (the length of the sequence),* $\mathcal{L}_n(\mathbf{a}) = \frac{\mathcal{L}_n(\mathcal{M}\mathbf{a})}{n_2}$*.*

The arrays that we are considering are periodic arrays and this implies that any Gröbner basis for $Val(\mathbf{a})$ will contain a polynomial that depends only in the variable $Y$. This polynomial has to be valid for the complete array, in particular, for each column. This polynomial is the joint minimal polynomial of the array visualized as a multisequence. Hence, the joint linear complexity can also be obtained when computing the linear complexity.

# 3 Construction and complexity analysis of arrays by the method of composition

Let $\mathbf{s}$ be an infinite sequence with entries in any finite field $\mathbb{F}_q$. An infinite two dimensional array is $\mathbf{a} = (a_{i,j})$, where the first index represents the column and the second represents the row. Hence, an array $\mathbf{a}$ can be thought as a sequence $\mathbf{a} = (\mathbf{s}_0, \mathbf{s}_1, \mathbf{s}_2, \ldots)$, of columns of sequences $\mathbf{s}_i$.

In [25, 26] Moreno and Tirkel presented constructions of 2-dimensional arrays $\mathbf{a}$ by considering circular shifts of a suitable sequence $\mathbf{s}$ in $\mathbb{F}_q$, and constructing the array by using shifts of $\mathbf{s}$ as the columns. The shifts on the columns were determined by another sequence $\mathbf{t}$ defined by a function $t : \mathbb{F}_{q'} \longrightarrow \mathbb{F}_{q'}$ with good correlation properties. Two-valued autocorrelation balanced sequences were preferred for the columns. If $q' = p$ we can illustrate the construction in the following way: To determine the shift on each column select an ordering for the elements in $\mathbb{F}_p$ (for example, considering the non-zero elements of the field as powers of a primitive root and ordering them by their exponents); the shift of column $i$ is the value of $t(X)$ when evaluated in the $i$'th element of $\mathbb{F}_p$. For example, if we order the non-zero elements of $\mathbb{F}_p, \alpha^i$, by the exponents $i$, we can construct a $p \times (p-1)$ array $\mathbf{a} = (a_{i,j})$ by setting column $i$ to be a circular shift up of $\mathbf{s}$ by $t\left(\alpha^i\right)$ entries. This is, $a_{i,j} = s_{j-t_i} = s_{j-t(\alpha^i)}$.

The construction described above produces a solitary array. To obtain families of arrays the exponent of the shift sequence can be changed to a higher degree polynomial or a rational function map, as it was done also in [25, 26].

**Example 2** *Let* $\mathbf{s}$ *be the Legendre sequence with respect to 7,* $\mathbf{s} = (0, 1, 1, 0, 1, 0, 0, \ldots)$ *given by*

$$s_j = \begin{cases} \frac{1+\left(\frac{j}{7}\right)}{2}, & \text{if } j \neq 0 \pmod 7 \\ 0, & \text{otherwise} \end{cases},$$

*where* $\left(\frac{j}{p}\right)$ *is the Legendre symbol [7], and* $(t_i)$ *be the Costas sequence defined by* $t_i = 3^i \pmod 7$.

*To construct the array* $\mathbf{a}$ *with period vector* $(6, 7)$ *of Figure 4, we place the first 7 entries of the Legendre sequence* $\mathbf{s}$ *with a circular shift up of* $1 = t_0 = 3^0 \pmod 7$ *in column 0, the first 7 entries of* $\mathbf{s}$ *with a circular shift up of* $3 = t_1 = 3^1 \pmod 7$ *in column 1, the first 7 entries of* $\mathbf{s}$ *with a circular shift up of* $2 = t_2 = 3^2 \pmod 7$ *in column 2, and so on. These and other constructions are described in [35].*

$\mathbf{s} =$

| | |
|---|---|
| *6* | 0 |
| *5* | 0 |
| *4* | 1 |
| *3* | 0 |
| *2* | 1 |
| *1* | 1 |
| *0* | 0 |

$\mathbf{t} =$

| 1 | 3 | 2 | 6 | 4 | 5 |
|---|---|---|---|---|---|
| *0* | *1* | *2* | *3* | *4* | *5* |

Figure 3: Costas $\mathbf{t}$ and Legendre $\mathbf{s}$ sequences

It is common to represent the shift sequence $\mathbf{t}$ as a doubly periodic array of 0's and 1's (here represented as blanks and '$*$', respectively), where each column has exactly one '$*$'. The number of rows has to be consistent with the period of the sequence that will be used as columns. Figure 4 shows the array corresponding to the Costas sequence $\mathbf{t}$ of Figure 3. Shifts of the other sequence $\mathbf{s}$ will form columns that will be substituted in the array corresponding to $\mathbf{t}$. A '$*$' in column $i$, row $j$ of the array $\mathbf{t}$ means that column $i$ will be replaced by the sequence $\mathbf{s}$ shifted up circularly by $i$ entries.

$\mathbf{a} =$

| | | | | | | |
|---|---|---|---|---|---|---|
| *6* | 0 | 0 | 1 | 0 | 1 | 1 |
| *5* | 1 | 1 | 0 | 0 | 1 | 0 |
| *4* | 0 | 1 | 1 | 0 | 0 | 0 |
| *3* | 1 | 0 | 1 | 1 | 0 | 0 |
| *2* | 1 | 0 | 0 | 0 | 0 | 1 |
| *1* | 0 | 0 | 0 | 1 | 1 | 0 |
| *0* | 0 | 1 | 0 | 1 | 0 | 1 |
| | *0* | *1* | *2* | *3* | *4* | *5* |

Figure 4: Shifts up of $\mathbf{s}$ sequence as columns.

$\mathbf{t} =$

| | | | | | | |
|---|---|---|---|---|---|---|
| *6* | | | | * | | |
| *5* | | | | | | * |
| *4* | | | | | * | |
| *3* | | * | | | | |
| *2* | | | * | | | |
| *1* | * | | | | | |
| *0* | | | | | | |
| | *0* | *1* | *2* | *3* | *4* | *5* |

Figure 5: Array corresponding to the Costas sequence $\mathbf{t}$ of Figure 3.

This construction can be generalized by using any two sequences $\mathbf{s}, \mathbf{t}$ (over any finite field) of period $n_1, n_2$ respectively. For a general finite field $\mathbb{F}_{q'}$ the value of $t_i$ might not be an integer mod $p$ and one cannot take $t_i$ as the shift, but one can write the non-zero elements as powers of a primitive root, $t_i = \alpha^e$, and use the exponent $e$ as the shift up in column $i$. This is, if $l$ is the logarithmic map $l : \mathbb{F}_{q'} \longrightarrow \mathbb{Z}_{q'-1},\ l(\alpha^e) = e$, then the array $\mathbf{a}$ formed from sequences $\mathbf{s}$ and $\mathbf{t}$ is defined by

$$a_{i,j} = s_{j-l(t_i)}, \tag{3}$$

where $j - l\,(t_i)$ is considered modulo $n_2$.

We now study the complexity of arrays constructed using the composition method. For simplicity of the notation we continue with the 2-dimensional case.

## 3.1 Complexity of 2-dimensional arrays constructed using the composition method

In [17] the authors conjectured that the linear complexity of some families of arrays constructed by Moreno and Tirkel using the composition method is the complexity of the column sequence times the number of columns (the period of the shift sequence). To obtain exact formulas for the linear complexity of arrays $\mathbf{a}$ one would need to be able to determine a Gröbner basis for $Val(\mathbf{a})$. If, at least, one can determine some elements of $Val(\mathbf{a})$ one can provide bounds for the size of the delta set, $\left|\Delta_{Val(\mathbf{a})}\right|$, and hence bounds for the linear complexity of $\mathbf{a}$.

By determining some elements of $Val(\mathbf{a})$ we now present bounds for the linear complexity of 2-dimensional arrays $\mathbf{a}$ constructed using the composition method. We prove that the complexity is bounded by the complexity of the sequence $\mathbf{s}$ with desired properties used in the construction of the columns times the period of the shift sequence used. The examples in [17] attain the bounds, and hence prove that the bound is tight.

The following proposition states that any polynomial $C(Y)$ valid on the sequence $\mathbf{s}$ used to construct the columns of the array has corresponding polynomials that are valid on the 2-dimensional array constructed

using the composition method as in (3).

**Proposition 3** *Let* $\mathbf{s}$ *be any sequence over* $\mathbb{F}_q$ *with period* $n_2$ *and* $\mathbf{a}$ *be the array constructed with the composition method as in (3) by defining the columns as cyclic shifts up of* $\mathbf{s}$*, where the shifts are given by a sequence* $\mathbf{t}$ *over* $\mathbb{F}_{q'}$ *with period* $n_1$*. Suppose that* $C(Y) = \sum_{j \in Supp(C)} c_j Y^j \in Val(\mathbf{s})$*. Then* $C'(X,Y) = P(X,Y)C(Y) \in Val(\mathbf{a})$ *for any polynomial* $P(X,Y) \in \mathbb{F}_q[X,Y]$*.*

**Proof:** $C \in Val(\mathbf{s})$ implies that $\sum_{j \in Supp(C)} c_j s_{j+\beta} = 0$ for any $\beta \in \mathbb{N}_0$. Since $Val(\mathbf{a})$ is an ideal, we only have to show $C'(X,Y) = C(Y) \in Val(\mathbf{a})$.

Let $c_{0,j} = c_j$ for $j \in Supp(C)$. Then $C'(X,Y) = \sum_{j \in Supp(C)} c_{0,j} Y^j$ and, for any $\gamma \in \mathbb{N}_0^2$,

$$\sum_{0,j \in Supp(C')} c_{0,j} a_{(0,j)+\gamma} = \sum_{0,j \in Supp(C')} c_{0,j} a_{(\gamma_1, j+\gamma_2)} = \sum_{j \in Supp(C)} c_j s_{j+\gamma_2 - l\left(t_{\gamma_1}\right)} = \sum_{j \in Supp(C)} c_j s_{j+\beta} = 0,$$

where $\beta = \gamma_2 - l\left(t_{\gamma_1}\right) \in \mathbb{N}_0$. This implies that $C'(X,Y) = C(Y) \in Val(\mathbf{a})$ and $C'(X,Y) = P(X,Y)C(Y) \in Val(\mathbf{a})$ for any polynomial $P(X,Y) \in \mathbb{F}_q[X,Y]$. □

**Remark 1** *Since* $X^{n_1} - 1, Y^{n_2} - 1 \in Val(\mathbf{a})$ *one can always reduce* $C'$ *mod* $\langle X^{n_1} - 1, Y^{n_2} - 1 \rangle$ *and therefore consider* $C' \in \mathbb{F}_q[X,Y] / \langle X^{n_1} - 1, Y^{n_2} - 1 \rangle$*.*

The only polynomials not depending on $X$ in $Val(\mathbf{a})$ are those arising from the polynomials that are valid on $\mathbf{s}$. As a consequence we get a bound for the linear complexity of the array that only depends on the linear complexity of the sequence $\mathbf{s}$ and the period of the sequence $\mathbf{t}$.

**Proposition 4** *Let* $\mathbf{s}$ *be any sequence over* $\mathbb{F}_q$ *with period* $n_2$ *and* $\mathbf{a}$ *be the array constructed with the composition method as in (3) by defining the columns as cyclic shifts up of* $\mathbf{s}$*, where the shifts are given by a sequence* $\mathbf{t}$ *over* $\mathbb{F}_{q'}$ *with period* $n_1$*. Then, for any* $i \in \mathbb{N}_0$*,* $C'(X,Y) = X^i C(Y) \in Val(\mathbf{a})$ *if and only if* $C(Y) \in Val(\mathbf{s})$*.*

**Proof:** Let $i \in \mathbb{N}_0$ and $C'(X,Y) = X^i C(Y) \in Val(\mathbf{a})$. Then, $c'_{i,j} = c_j$, and, for any $(\beta_1, \beta_2) \in \mathbb{N}_0^2$,

$$\sum_{j=0}^{n_2-1} c'_{i,j} a_{(i,j)+(\beta_1,\beta_2)} = 0.$$

Let $\gamma \in \mathbb{N}_0$ and fix $\beta_2 = \gamma + l\left(t_i\right) \in \mathbb{N}_0$. Then,

$$\sum_{j=0}^{n_2-1} c_j s_{j+\gamma} = \sum_{j=0}^{n_2-1} c'_{i,j} s_{j+\beta_2 - l(t_i)} = \sum_{j=0}^{n_2-1} c'_{i,j} a_{(i,j+\beta_2)} = \sum_{j=0}^{n_2-1} c'_{i,j} a_{(i,j)+(0,\beta_2)} = 0.$$

Therefore, $C(Y) \in Val(\mathbf{s})$.

The other direction is a special case of Proposition 3. □

Proposition 4 implies that the complexity of arrays constructed using the composition method is bounded by the complexity of the periodic sequence used to construct the columns of $\mathbf{a}$ and the period of the shift sequence.

**Theorem 1** *Let* $\mathbf{s}$ *be any sequence over* $\mathbb{F}_q$ *with period* $n_2$ *and* $\mathbf{a}$ *be the array constructed using the composition method as in (3) by defining the columns as cyclic shifts up of* $\mathbf{s}$*, where the shifts are given by a sequence* $\mathbf{t}$ *over* $\mathbb{F}_{q'}$ *with period* $n_1$*. If* $\mathcal{L}(\mathbf{s})$ *is the linear complexity of the sequence* $\mathbf{s}$ *and* $\mathcal{L}(\mathbf{a})$ *is the linear complexity of the array* $\mathbf{a}$*, then* $\mathcal{L}(\mathbf{a}) \leq n_1 \mathcal{L}(\mathbf{s})$*.*

**Proof:** Let $C(Y)$ be the minimal polynomial of $\mathbf{s}$. Then, $\mathcal{L}(\mathbf{s}) = deg\,(C(Y))$. By Proposition 3, $C(Y) \in Val(\mathbf{a})$. Since $\mathbf{t}$ has period $n_1$, we have that $X^{n_1} - 1 \in Val(\mathbf{a})$. This implies that $\Delta_{Val(\mathbf{a})}$ cannot contain the exponents of any monomial that is a multiple of $Y^{deg(C)}$ or a multiple of $X^{n_1}$. Therefore $\mathcal{L}(\mathbf{a}) = \left|\Delta_{Val(\mathbf{a})}\right| \leq n_1\mathcal{L}(\mathbf{s})$. □

Note that the bound on the complexity of the array has the maximum factor in one of the dimensions and the other factor depends on the complexity of the sequence $\mathbf{s}$; if $\mathcal{L}(\mathbf{s})$ is maximal, then $\mathcal{L}(\mathbf{a})$ attains the trivial upper bound. The examples presented in [17], and many of our own examples attain the bound in Theorem 1, and hence prove that this bound, in general, is tight.

**Corollary 2** *Let* $\mathbf{s}$ *be any sequence over* $\mathbb{F}_q$ *with period* $n_2$ *and* $\mathbf{a}$ *be the array constructed using the composition method as in (3) by defining the columns as cyclic shifts up of* $\mathbf{s}$*, where the shifts are given by a sequence* $\mathbf{t}$ *over* $\mathbb{F}_{q'}$ *with period* $n_1$*. If* $\mathcal{L}_n(\mathbf{s})$ *is the normalized linear complexity of the sequence* $\mathbf{s}$ *and* $\mathcal{L}_n(\mathbf{a})$ *is the normalized linear complexity of the array* $\mathbf{a}$*, then* $\mathcal{L}_n(\mathbf{a}) \leq \mathcal{L}_n(\mathbf{s})$*.*

In the case where the sequence $\mathbf{s}$ has entries in a field of characteristic 2 and the minimal polynomial $C(Y)$ for the sequence $s$ is divisible by $Y - 1$, then the bound in the complexity of the array is reduced by $n_1 - 1$, where $n_1$ is the period of the shift sequence $\mathbf{t}$ (the number of columns in the finite array). This implies that, the equality of the conjecture in [17] cannot be attained in general.

The next lemma and proposition are needed to give a bound for the linear complexity for this case where the sequence $\mathbf{s}$ has entries in a field of characteristic 2 and the minimal polynomial $C(Y)$ for the sequence $\mathbf{s}$ is divisible by $Y - 1$.

**Lemma 2** *Fix* $e_0, e_1 \in \mathbb{N}_0$*. If* $\sum_{j=0}^{h} q_j s_{j+\beta} = \sum_{j=0}^{h} q_j s_{j+\beta+1}$ *for any* $\beta \in \mathbb{N}_0$*, then* $\sum_{j=0}^{h} q_j s_{j+e_0} = \sum_{j=0}^{h} q_j s_{j+e_1}$*.*

**Proof:** We can assume that $e_0 < e_1$. By hypothesis, taking $\beta = e_0$, we have that $\sum_{j=0}^{h} q_j s_{j+e_0} = \sum_{j=0}^{h} q_j s_{j+e_0+1}$. Since the hypothesis is true for any $\beta \in \mathbb{N}_0$, we can now take $\beta = e_0 + 1$ and obtain $\sum_{j=0}^{h} q_j s_{j+e_0} = \sum_{j=0}^{h} q_j s_{j+e_0+1} = \sum_{j=0}^{h} q_j s_{j+e_0+2}$. It is clear that we can continue the argument and obtain that $\sum_{j=0}^{h} q_j s_{j+e_0} = \sum_{j=0}^{h} q_j s_{j+e_1}$. □

In the case when the minimal polynomial $C(Y)$ for the sequence $\mathbf{s}$ is divisible by $Y - 1$, we obtain additional polynomials in $Val(\mathbf{a})$ and, as a consequence, the bound on the linear complexity of $\mathbf{a}$ is tighter.

**Proposition 5** *Let* $\mathbf{s}$ *be any sequence over* $\mathbb{F}_{2^r}$ *and* $\mathbf{a}$ *be the array constructed using the composition method as in (3) by defining the columns as cyclic shifts up of* $\mathbf{s}$*, where the shifts are given by a sequence* $\mathbf{t}$ *over* $\mathbb{F}_{q'}$*. If* $C(Y) = (Y-1)Q(Y) \in Val(\mathbf{s})$*. Then,* $C'(X,Y) = Q(Y)\left(X^i - 1\right) \in Val(\mathbf{a})$ *for any* $i \geq 1$*; in particular,* $Q(Y)\,(X-1) \in Val(\mathbf{a})$*.*

**Proof:** Since $X-1$ divides $X^i-1$ and $Val(\mathbf{a})$ is an ideal, we only have to show $Q(Y)\,(X-1) \in Val(\mathbf{a})$. We need to prove that $\sum_{1,j\in Supp(C')} c'_{1,j} a_{(1,j)+\gamma} = 0 \pmod 2$ for any $\gamma = (\gamma_1, \gamma_2) \in \mathbb{N}_0^2$. Let $\gamma = (\gamma_1, \gamma_2) \in \mathbb{N}_0^2$. Then,

$$\sum_{1,j\in Supp(C')} c'_{1,j} a_{(1,j)+\gamma} = \sum_{j\in Supp(Q)} q_j \left(a_{(1,j)+\gamma} + a_{(0,j)+\gamma}\right)$$

$$= \sum_{j\in Supp(Q)} q_j a_{(1+\gamma_1, j+\gamma_2)} + \sum_{j\in Supp(Q)} q_j a_{(\gamma_1, j+\gamma_2)} = \sum_{j\in Supp(Q)} q_j s_{j+\gamma_2-l\left(t_{1+\gamma_1}\right)} + \sum_{j\in Supp(Q)} q_j s_{j+\gamma_2-l\left(t_{\gamma_1}\right)}.$$

qq Since $\gamma_2 - l\left(t_{1+\gamma_1}\right)$ and $\gamma_2 - l\left(t_{\gamma_1}\right)$ are fixed, let $e_0 = \gamma_2 - l\left(t_{1+\gamma_1}\right)$ and $e_1 = \gamma_2 - l\left(t_{\gamma_1}\right)$. Now, since $(Y-1)Q(Y) \in Val(\mathbf{s})$, we have that $\sum_{j \in Supp(Q)} q_j s_{j+1+\beta} + \sum_{j \in Supp(Q)} q_j s_{j+\beta} = 0$ for any $\beta \in \mathbb{N}_0$. This implies that $\sum_{j \in Supp(Q)} q_j s_{j+1+\beta} = \sum_{j \in Supp(Q)} q_j s_{j+\beta}$. By Lemma 2, $\sum_{j \in Supp(Q)} q_j s_{j+e_0} = \sum_{j \in Supp(Q)} q_j s_{j+e_1}$, and

$$\sum_{j \in Supp(Q)} q_j s_{j+\gamma_2 - l\left(t_{1+\gamma_1}\right)} + \sum_{j \in Supp(Q)} q_j s_{j+\gamma_2 - l\left(t_{\gamma_1}\right)}$$

$$= \sum_{j \in Supp(Q)} q_j s_{j+e_0} + \sum_{j \in Supp(Q)} q_j s_{j+e_1} = 2 \sum_{j \in Supp(Q)} q_j s_{j+e_0} = 0 \pmod 2,$$

and this completes the proof. □

The following proposition gives a bound for the complexity of arrays constructed with the composition method and sequences $\mathbf{s}$ with entries in a field of characteristic 2, and where the minimal polynomial $C(Y)$ for the sequence $\mathbf{s}$ is divisible by $Y - 1$.

**Theorem 2** *Let* $\mathbf{s}$ *be any sequence over* $\mathbb{F}_{2^r}$ *with period* $n_2$ *and* $\mathbf{a}$ *be the array constructed using the composition method as in (3) by defining the columns as cyclic shifts up of* $\mathbf{s}$*, where the shifts are given by a sequence* $\mathbf{t}$ *over* $\mathbb{F}_q$ *with period* $n_1$*. If the minimal polynomial of* $\mathbf{s}$*,* $C(Y)$*, is divisible by* $y - 1$*,* $\mathcal{L}(\mathbf{s})$ *is the linear complexity of the sequence* $\mathbf{s}$ *and* $\mathcal{L}(\mathbf{a})$ *is the linear complexity of the array* $\mathbf{a}$*, then* $\mathcal{L}(\mathbf{a}) \leq n_1\left(\mathcal{L}(\mathbf{s}) - 1\right) + 1$*.*

**Proof:** Let $C(Y) = (Y-1)Q(Y)$ be the minimal polynomial of $\mathbf{s}$. By Proposition 3, $C(Y) \in Val(\mathbf{a})$ and, by Proposition 5, $Q(Y)(X-1) \in Val(\mathbf{a})$. Since $\mathbf{t}$ has period $n_1$, we have that $X^{n_1} - 1 \in Val(\mathbf{a})$. This implies that $\Delta_{Val(\mathbf{a})}$ cannot contain the exponents of any monomial that is a multiple of $Y^{deg(C)}$, of $XY^{deg(C)-1}$, or of $X^{n_1}$. Therefore, since $\mathcal{L}(\mathbf{s}) = deg\left(C(Y)\right)$, we have that $\mathcal{L}(\mathbf{a}) = \left|\Delta_{Val(\mathbf{a})}\right| \leq n_1\left(\mathcal{L}(\mathbf{s}) - 1\right) + 1$. □

In this case, even if $\mathcal{L}(\mathbf{s})$ is maximal, $\mathcal{L}(\mathbf{a})$ will be $n_1 - 1$ smaller than the trivial upper bound. Nevertheless, the bound is tight. This is shown by the examples presented in [17], and our own examples attain these bounds.

**Corollary 3** *Let* $\mathbf{s}$ *be any sequence over* $\mathbb{F}_{2^r}$ *with period* $n_2$ *and* $\mathbf{a}$ *be the array constructed as in (3) by defining the columns as cyclic shifts up of* $\mathbf{s}$*, where the shifts are given by a sequence* $\mathbf{t}$ *over* $\mathbb{F}_{q'}$ *with period* $n_1$*. If the minimal polynomial of* $\mathbf{s}$*,* $C(Y)$*, is divisible by* $Y - 1$*,* $\mathcal{L}_n(\mathbf{s})$ *is the normalized linear complexity of the sequence* $\mathbf{s}$ *and* $\mathcal{L}_n(\mathbf{a})$ *is the normalized linear complexity of the array* $\mathbf{a}$*, then* $\mathcal{L}_n(\mathbf{a}) \leq \mathcal{L}_n(\mathbf{s}) - \frac{n_1 - 1}{n_1 n_2}$*.*

### 3.1.1 Complexity of arrays constructed using the composition method and Legendre sequences

The linear complexity of Legendre sequences is given in [7]; we now apply Theorems 1 and 2 to give tight bounds for the complexity of arrays constructed using the composition method and Legendre sequences as defined in [7].

**Proposition 6** *Let* $\mathbf{s}$ *be the Legendre sequence with respect to* $p$ *and* $\mathbf{a}$ *be the array constructed using the composition method as in (3) by defining the columns as shifts of* $\mathbf{s}$*, where the shifts are given by a sequence with period* $n_1$*. Then, linear complexity of* $\mathbf{a}$*,* $\mathcal{L}(\mathbf{a})$ *is:*

$$\mathcal{L}(\mathbf{a}) \leq \begin{cases} n_1\left(\frac{p-1}{2}\right), & p \equiv 1 \pmod 8 \\ n_1(p-1) + 1, & p \equiv 3 \pmod 8 \\ n_1(p-1), & p \equiv 5 \pmod 8 \\ n_1\left(\frac{p-1}{2}\right) + 1, & p \equiv 7 \pmod 8 \end{cases}.$$

**Proof:** From [7] we know that the linear complexity of the Legendre sequence $\mathbf{s}$ with respect to $p$ is

$$\mathcal{L}(\mathbf{s}) = \begin{cases} \frac{p-1}{2}, & p \equiv 1 \pmod 8 \\ p, & p \equiv 3 \pmod 8 \\ p-1, & p \equiv 5 \pmod 8 \\ \frac{p+1}{2}, & p \equiv 7 \pmod 8 \end{cases}.$$

From the same reference we know that $Y-1$ divides the minimal polynomial of the Legendre sequence with respect to $p$ when $p \equiv 3, 7 \pmod 8$. The result now follows from Theorems 1 and 2. □

At this moment we cannot prove that all the arrays $\mathbf{a}$ constructed with a Legendre sequence attain the bound on linear complexity of Proposition 6 but all the examples that we have computed validate this claim and prove that the bound is tight.

**Conjecture 1** *The linear complexity of the array* $\mathbf{a}$, $\mathcal{L}(\mathbf{a})$, *constructed using the composition method as in (3) by defining the columns as shifts of the Legendre sequence* $\mathbf{s}$, *where the shifts are given by a sequence with period* $n_1$ *is the maximal linear complexity given in Proposition 6.*

**Conjecture 2** *Let* $\mathbf{s}$ *be the Legendre sequence with respect to* $p$ *and* $\mathbf{a}$ *be the array constructed using the composition method as in (3) by defining the columns as shifts of* $\mathbf{s}$, *where the shifts are given by a sequence with period* $n_1$. *Then, the normalized linear complexity of* $\mathbf{a}$ *is:*

$$\mathcal{L}_n(\mathbf{a}) = \begin{cases} \frac{1}{2} - \frac{1}{2p}, & p \equiv 1 \pmod 8 \\ 1 - \frac{1}{p} + \frac{1}{n_1 p}, & p \equiv 3 \pmod 8 \\ 1 - \frac{1}{p}, & p \equiv 5 \pmod 8 \\ \frac{1}{2} - \frac{1}{2p} + \frac{1}{n_1 p}, & p \equiv 7 \pmod 8 \end{cases}.$$

*This is, for large arrays,* $\mathcal{L}_n(\mathbf{a})$ *is close to .5 if* $p \equiv 1, 7 \pmod 8$ *and close to 1 if* $p \equiv 3, 5 \pmod 8$.

# 4 Conclusions and future work

In this paper we developed a theory of complexity for general multidimensional periodic arrays by using concepts from Gröbner bases. Our approach is consistent with the one-dimensional definition of linear complexity and allowed us to make generalised yet precise statements about the linear complexity of multidimensional arrays built using the composition method. Future work includes proving the conjectures posed for arrays composed from Legendre sequences as well as the analysis of constructions that use methods other than the composition.

It is also interesting to study cellular automata under this framework. The example in Figure 6 is taken from [3, Table 5]. The two dimensional array, or cellular automata, has linear complexity equal to 3 and the set of valid polynomials is an ideal generated by $\{Y + X + 1, (Y + 1)^3\}$. The two dimensional array is periodic in one dimension but not purely periodic in the other. However, the polynomial $X^3$ is valid for the array, which means that most of the columns are equal to zero. Viewing the cellular automata as a multidimensional array, whose ideal of valid polynomials can be generated by a Grobner basis, one can give alternative proofs of several results, for example Theorem 4 of [2].

# Acknowledgments

Part of this work was inspired by the late Oscar Moreno. Unfortunately, he passed away before we were able to finish this paper. The research of D. Gomez-Perez is supported by the Ministerio de Economía y Competitividad research project MTM2014-55421-P.

$\mathbf{a} =$

| | | | | | |
|---|---|---|---|---|---|
| *4* | 0 | 1 | 1 | 0 | ... |
| *3* | 1 | 0 | 1 | 0 | ... |
| *2* | 1 | 1 | 1 | 0 | ... |
| *1* | 1 | 0 | 1 | 0 | ... |
| *0* | 0 | 1 | 1 | 0 | ... |
| | *0* | *1* | *2* | *3* | ... |

Figure 6: Celullar automata related with the shrinking generator.

# References


[1] Bruno Buchberger. Bruno Buchberger's PhD thesis 1965: An algorithm for finding the basis elements of the residue class ring of a zero dimensional polynomial ideal. *Journal of Symbolic Computation*, 41(34):475 – 511, 2006. Logic, Mathematics and Computer Science: Interactions in honor of Bruno Buchberger (60th birthday).

[2] Sara D. Cardell and Amparo Fúster-Sabater. Linear models for the self-shrinking generator based on CA. *Journal of Cellular Automata*, 11(2-3):195–211, 2016.

[3] Sara D. Cardell and Amparo Fúster-Sabater. Modelling the shrinking generator in terms of linear CA. *Advances in Mathematics of Communications*, 10(4):797–809, 2016.

[4] David Cox, John Little, and Donal O'Shea. *Ideals, varieties, and algorithms*. Undergraduate Texts in Mathematics. Springer, New York, third edition, 2007. An introduction to computational algebraic geometry and commutative algebra.

[5] Thomas W Cusick, Cunsheng Ding, and Ari R Renvall. *Stream ciphers and number theory*, volume 66. Elsevier, 2004.

[6] Sara D. Cardell and Amparo Fúster-Sabater. Discrete linear models for the generalized self-shrunken sequences. *Finite Fields and Their Applications*, 47:222–241, 2017.

[7] Cunsheng Ding, Tor Helleseth, and Weijuan Shan. On the linear complexity of Legendre sequences. *IEEE Trans. Inform. Theory*, 44(3):1276–1278, 1998.

[8] Cunsheng Ding, Guozhen Xiao, and Weijuan Shan. *The stability theory of stream ciphers*, volume 561. Springer Science & Business Media, 1991.

[9] Fang-Wei Fu, Harald Niederreiter, and Ferruh Özbudak. Joint linear complexity of arbitrary multisequences consisting of linear recurring sequences. *Finite Fields and Their Applications*, 15(4):475–496, 2009.

[10] Fang-Wei Fu, Harald Niederreiter, and Ferruh Özbudak. Joint linear complexity of arbitrary multisequences consisting of linear recurring sequences. *Finite Fields Appl.*, 15(4):475–496, 2009.

[11] Patrizia Gianni, Barry Trager, and Gail Zacharias. Gröbner bases and primary decomposition of polynomial ideals. *J. Symbolic Comput.*, 6(2-3):149–167, 1988. Computational aspects of commutative algebra.

[12] Domingo Gomez-Perez, Tom Hoholdt, Oscar Moreno, and Ivelisse Rubio. Linear complexity for multidimensional arrays - a numerical invariant. In *Information Theory (ISIT), 2015 IEEE International Symposium on*, pages 2697–2701, June 2015.

[13] Katalin Gyarmati, Christian Mauduit, and András Sárközy. Measures of pseudorandomness of finite binary lattices, i. the measures $q\_k$, normality. *Acta Arithmetica*, 144(3):295–313, 2010.

[14] Katalin Gyarmati, Christian Mauduit, and András Sárközy. On the linear complexity of binary lattices. *Ramanujan J.*, 32(2):185–201, 2013.

[15] Katalin Gyarmati, Christian Mauduit, and András Sárközy. On linear complexity of binary lattices, II. *Ramanujan J.*, 34(2):237–263, 2014.

[16] Katalin Gyarmati, Christian Mauduit, and András Sárközy. On finite pseudorandom binary lattices. *Discrete Appl. Math.*, 216(part 3):589–597, 2017.

[17] Anatolii Leukhin, Oscar Moreno, and Andrew Tirkel. Secure CDMA and frequency hop sequences. In *Wireless Communication Systems (ISWCS 2013), Proceedings of the Tenth International Symposium on*, pages 1–5. VDE, 2013.

[18] Kit-Ho Mak. More constructions of pseudorandom lattices of $k$ symbols. *Monatsh. Math.*, 177(2):307–323, 2015.

[19] James L. Massey. Shift-register synthesis and BCH decoding. *IEEE Trans. Information Theory*, IT-15:122–127, 1969.

[20] Christian Mauduit and András Sárközy. Construction of pseudorandom binary lattices by using the multiplicative inverse. *Monatshefte für Mathematik*, 153(3):217–231, 2008.

[21] Wilfried Meidl and Harald Niederreiter. The expected value of the joint linear complexity of periodic multisequences. *Journal of Complexity*, 19(1):61–72, 2003.

[22] László Mérai. Construction of pseudorandom binary lattices based on multiplicative characters. *Period. Math. Hungar.*, 59(1):43–51, 2009.

[23] László Mérai. Construction of pseudorandom binary lattices using elliptic curves. *Proc. Amer. Math. Soc.*, 139(2):407–420, 2011.

[24] Oscar Moreno, Tom Høholdt, and Ivelisse Rubio. Security of multidimensional arrays. 62/131616 (March, 2015) and 62/174973 (June, 2015).

[25] Oscar Moreno and Andrew Tirkel. Multi-dimensional arrays for watermarking. In *Information Theory Proceedings (ISIT), 2011 IEEE International Symposium on*, pages 2691–2695. IEEE, 2011.

[26] Oscar Moreno and Andrew Tirkel. New optimal low correlation sequences for wireless communications. In *Sequences and their applications—SETA 2012*, volume 7280 of *Lecture Notes in Comput. Sci.*, pages 212–223. Springer, Heidelberg, 2012.

[27] Gary L Mullen and Daniel Panario. *Handbook of finite fields*. CRC Press, 2013.

[28] Harald Niederreiter. The probabilistic theory of linear complexity. In *Workshop on the Theory and Application of of Cryptographic Techniques*, pages 191–209. Springer, 1988.

[29] Harald Niederreiter, Michael Vielhaber, and LiPing Wang. Improved results on the probabilistic theory of the joint linear complexity of multisequences. *Sci. China Inf. Sci.*, 55(1):165–170, 2012.

[30] Harald Niederreiter and Li-Ping Wang. The asymptotic behavior of the joint linear complexity profile of multisequences. *Monatsh. Math.*, 150(2):141–155, 2007.

[31] Ivelisse M Rubio, Moss Sweedler, and Chris Heegard. Finding a gröbner basis for the ideal of recurrence relations on m-dimensional periodic arrays. In *Contemporary Developments in Finite Fields and Applications*, pages 296–320. World Scientific, 2016.

[32] Ivelisse María Rubio. *PhD thesis: Gröbner bases for 0-dimensional ideals and applications to decoding*. Cornell University, 1998.

[33] Shojiro Sakata. Finding a minimal set of linear recurring relations capable of generating a given finite two-dimensional array. *J. Symbolic Comput.*, 5(3):321–337, 1988.

[34] Shojiro Sakata. Extension of the Berlekamp-Massey algorithm to $N$ dimensions. *Inform. and Comput.*, 84(2):207–239, 1990.

[35] Andrew Tirkel and Tom Hall. New matrices with good auto and cross-correlation. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, 89(9):2315–2321, 2006.

[36] Andrew Z. Tirkel, Charles F. Osborne, and Tom H. Hall. Steganography-applications of coding theory. In *IEEE Information Theory Workshop, Svalbard, Norway*, pages 57–59, 1997.