Analysis and Computation of Multidimensional Linear Complexity of Periodic Arrays ⋆

Rafael Arce[1], Carlos Hernández[1], José Ortiz[1], Ivelisse Rubio[1], and Jaziel Torres[2]

[1] Department of Computer Science,
[2] Department of Mathematics,
Univ. of PR, 17 Ave. Univ. STE 1701, SJ, Puerto Rico 00925-2537
rafael.arce, carlos.hernandez34, jose.ortiz23, ivelisse.rubio, jaziel.torres@upr.edu

Abstract. Linear complexity is an important parameter for arrays that are used in applications related to information security. In this work we present new results on the multidimensional linear complexity of periodic arrays obtained using the definition and method proposed in [2, 6, 11]. The results include a generalization of a bound for the linear complexity, a comparison with the measure of complexity for multisequences, and computations of the complexity of arrays with periods that are not relatively prime for which the "unfolding method" does not work. We also present conjectures for exact formulas and the asymptotic behavior of the complexity of some array constructions.

Keywords: Linear complexity · Multidimensional arrays · Information security

1 Introduction

Multidimensional periodic arrays are useful in applications such as digital watermarking, multiple target recognition and communications [1, 4, 7, 13–15, 20]. It is desirable to have arrays with a variety of sizes. Depending on the particular application, the array should satisfy properties such as good auto and cross correlation, balance, and complexity. Randomly generated arrays pose problems to provide properties such as periodicity and orthogonality. Precomputed arrays are stored in memory, which imposes a heavy memory burden on some systems. Hence, it is important to provide algebraic constructions for arrays that have the desired properties and are easily implemented. Several constructions have been proposed and their properties analyzed over the years.

Since some of the applications are related to information security, it is particularly important that the arrays have good complexity, meaning that they are resistant to Berlekamp-Massey types of attacks, where the complete array might be deduced from knowing some of its entries. The linear complexity of

⋆ This research was funded by the "Fondo Institucional Para la Investigación (FIPI)" from the University of Puerto Rico, Río Piedras.

sequences has been widely studied [3, 8, 9]. However, not much work has been done on the analysis of the complexity of multidimensional arrays. A definition of the complexity of 2-dimensional arrays viewed as multisequences was given in [10]. The computation of multidimensional linear complexity of 2-dimensional arrays with periods that are relatively prime was done by "unfolding" the array into a sequence and applying the Berlekamp-Massey algorithm in [7, 14]. A new definition and theory for the computation of multidimensional linear complexity of arrays was proposed in [2, 6, 11]. This definition applies to any number of dimensions, does not have the restrictions of the unfolding method, and it is more accurate than the joint linear complexity defined for multisequences.

Given that there are few sequences with known formulas for their complexity, it is expected that formulas for the exact value of the complexity of arrays would be hard to find. In this work we present a generalization of a bound for the linear complexity of arrays presented in [2] and conjectures for exact formulas and the asymptotic behavior of the complexity of some array constructions. It is also proved that the definition of multidimensional linear complexity in [2, 6, 11] is more accurate than the definition of joint linear complexity of multisequences. In addition, we present new computations of the complexity of families of multidimensional arrays for wireless communications and watermarking applications presented in [13, 14] for which the complexity was unknown.

2 Multidimensional Linear Complexity of Periodic Arrays

We consider periodic arrays with entries over a finite field $\mathbb{F}_q$, $q = p^r$, p a prime, and denote the set of non-negative integers by $\mathbb{N}_0$. A sequence $\mathbf{S} = s_0, s_1, \ldots$ is a 1-dimensional array and has period $n \in \mathbb{N}$ if n is the smallest such that $s_{i+n} = s_i$ for all $i \in \mathbb{N}_0$. A polynomial $f(x) = \sum_{i \in Supp(f)} f_i x^i$ defines a linear recurrence relation on the sequence $\mathbf{S}$ if $\sum_{i \in Supp(f)} f_i s_{i+\beta} = 0$ for all $\beta \in \mathbb{N}_0$, where $Supp(f)$ is the set of indices of the non-zero terms of f. We say that these recurrence polynomials are **valid** for the sequence $\mathbf{S}$ and the set of all valid polynomials for $\mathbf{S}$, $Val(\mathbf{S})$, forms an ideal. The **linear complexity** of $\mathbf{S}$, $\mathcal{L}(\mathbf{S})$, is the degree of the minimal (monic) generator of $Val(\mathbf{S})$, $m(x)$, which can be found using the well-known Berlekamp-Massey algorithm. For the generalization to multiple dimensions it is important to note that $\mathcal{L}(\mathbf{S})$ is also the number of monomials that are not divisible by the lead monomial of $m(x)$. Since the sequence has period n, the polynomial $x^n - 1$ is in $Val(\mathbf{S})$ and hence $\mathcal{L}(\mathbf{S}) \leq n$.

A 2-dimensional infinite array over $\mathbb{F}_q$ is a function $\mathbf{A} : \mathbb{N}_0^2 \to \mathbb{F}_q$, and we denote $\mathbf{A}(i, j)$ by a_{ij}. We say that $\mathbf{A}$ is **periodic** with period vector $(n_1, n_2) \in \mathbb{N}^2$ if $a_{i+n_1k_1, j+n_2k_2} = a_{i,j}$ for $k_1, k_2 \in \mathbb{N}_0$ and all $(i, j) \in \mathbb{N}_0^2$. These arrays can be represented by a subarray of dimensions $n_2 \times n_1$ and we do so by associating its entries to the integer coordinates of the first quadrant of the Cartesian plane.

A polynomial $f(x, y) = \sum_{i,j \in Supp(f)} f_{i,j} x^i y^j$ defines a **linear recurrence relation on the array A** if $\sum_{i,j \in Supp(f)} f_{i,j} a_{i+\beta_1, j+\beta_2} = 0$ for all $\beta_1, \beta_2 \in \mathbb{N}_0$. We say that these polynomials are **valid on the array A** and the set of all valid polynomials for $\mathbf{A}$, $Val(\mathbf{A})$, forms an ideal. This ideal might not be generated by a single polynomial but it has finite generating sets. In particular, $Val(\mathbf{A})$ is generated by a Gröbner basis with respect to a monomial ordering $\leq_T$ that can be computed using Sakata's algorithm or the RST algorithm described in [16]. We restricted our description to 2-dimensional arrays in order to simplify the notation but the previous discussion applies to higher dimensions.

Let $\Delta_{Val(\mathbf{A}),\leq_T}$ denote the set of all monomials that are not divisible by any lead monomial in $Val(\mathbf{A})$ with respect to $\leq_T$. As a result of the Gröbner bases properties, $\Delta_{Val(\mathbf{A}),\leq_T}$ is also the set of all monomials that are not divisible by any lead monomial in a Gröbner basis for $Val(\mathbf{A})$ with respect to $\leq_T$, and hence can be computed from the Gröbner basis. The size of $\Delta_{Val(\mathbf{A}),\leq_T}$ is invariant under monomial orderings.

Definition 1. *Let $\mathbf{A}$ be a multidimensional periodic array and $Val(\mathbf{A})$ be the ideal of recurrence relations valid on the array. Define the* **multidimensional linear complexity** $\mathcal{L}(\mathbf{A})$ *of the array $\mathbf{A}$ as the size of any delta set of $Val(\mathbf{A})$; this is,* $\mathcal{L}(\mathbf{A}) = \left|\Delta_{Val(\mathbf{A})}\right|$.

If the m-dimensional array $\mathbf{A}$ has period $(n_1, \ldots, n_m)$, the polynomials $x_1^{n_1} - 1, \ldots, x_m^{n_m} - 1$ are in $Val(\mathbf{A})$ and hence $\left|\Delta_{Val(\mathbf{A})}\right| \leq n_1 n_2 \cdots n_m$. With this we can define the **normalized linear complexity** of the array as $\mathcal{L}_n(\mathbf{A}) = \mathcal{L}(\mathbf{A}) / (n_1 n_2 \cdots n_m)$, a measure that let us compare the complexity of arrays of different dimensions and periods.

2.1 Other measures for complexity

A well studied definition for the complexity of a 2-dimensional array with period (n_1, n_2) is given by considering the array as an n_1**-fold multisequence** $\mathbf{S} = (\mathbf{S}_1, \ldots, \mathbf{S}_{n_1})$, that is, a sequence $\mathbf{S}$ of sequences $\mathbf{S}_1, \ldots, \mathbf{S}_{n_1}$ with period n_2. The **joint linear complexity** of $\mathbf{S}$, $J\mathcal{L}_n(\mathbf{S})$, is the degree of a minimal polynomial that is valid for each $\mathbf{S}_i$. As we will see in Example 2 on Section 4.1, this definition is not as accurate as Definition 1 because it might miss relations among the entries of different columns. In addition, our approach can be used in arrays of higher dimensions.

The linear complexity of some of the 2-dimensional arrays presented in [7, 14] was computed by "unfolding" the array using the Chinese Remainder Theorem in order to construct a sequence, and then compute the complexity of the resulting sequence using the Berlekamp-Massey algorithm. This method has the limitation that the periods of the array must be relatively prime. This constrain is why the complexity of some of the arrays in those papers could not be computed (see Section 3.2). Our approach to multidimensional linear complexity is consistent with the unfolding method [2, 5] but in addition it allows the computation of the complexity of any periodic array, advancing the complexity analysis of constructions of multidimensional arrays.

3 Constructions of multidimensional arrays

In Section 3.2 we present families of arrays from [7, 14] for which the linear complexity could not be computed using the unfolding method. To make this paper self contained, we first present general constructions for these arrays. For the sake of simplicity we only consider 2 and 3 dimensional arrays but the methods can be extended to higher dimensions. Some of the constructions use the index table $\mathbf{W}$ for a finite field $\mathbb{F}_{p^2}$, with respect to a primitive element α. The entries of $\mathbf{W}$ are defined by $w_{i,j} = k$ if $\alpha^k = i\alpha + j$. Since 0 is not a power of α, an $*$ is placed as the entry $w_{0,0}$.

3.1 Two dimensional Legendre arrays

A **binary 2-dimensional Legendre array** $\mathbf{F}_1$ with period (p, p) is constructed from an index table $\mathbf{W}$ for the finite field $\mathbb{F}_{p^2}$ by setting $f_{0,0} = 0$ and taking all other entries of $\mathbf{W}$ modulo 2. Similarly, a **ternary 2-dimensional Legendre array** $\mathbf{F}_2$ with period (p, p) is constructed from $\mathbf{W}$ by setting $f_{0,0} = 0$ and mapping the even entries of $\mathbf{W}$ to 1 and the odd entries to -1. This construction produces solitary Legendre arrays but they will be used by the composition method as "floors" to construct families of 3-dimensional arrays (see Section 3.2).

3.2 The composition method

Multidimensional arrays can be constructed by composing a shift sequence/array with a column sequence or an array of suitable dimension [13, 18]. For example, a **2-dimensional array A** with period vector (n_1, n_2) can be constructed using a shift sequence $\mathbf{S}$ with entries in $\mathbb{Z}_{n_2}$ and period n_1 to define circular shifts of columns defined by a sequence $\mathbf{C}$ of period n_2: $a_{i,j} = c_{j-s_i \pmod{n_2}}$. The entries of the shift sequence might also contain an extra symbol, i.e. the entries can be in $\mathbb{Z}_{n_2} \cup \{*\}$. In this case, the columns corresponding to an undetermined shift $*$ will consist of a sequence of a constant value (see Figure 1).

A **3-dimensional array A** with period vector (n_1, n_2, n_3) can be constructed using a 2-dimensional shift array $\mathbf{S}$ with entries in $\mathbb{Z}_{n_3} \cup \{*\}$ and period vector (n_1, n_2) to define circular shifts of columns given by a sequence $\mathbf{C}$ of period n_3: $a_{i,j,k} = c_{k-s_{i,j} \pmod{n_3}}$. Again, the columns corresponding to an undetermined shift $*$ will consist of a sequence of a constant value.

Similarly, one can construct a **3-dimensional array A** with period vector (n_1, n_2, n_3) by using a shift sequence $\mathbf{S}$ with entries in $\mathbb{Z}_{n_1} \times \mathbb{Z}_{n_2}$ and period n_3 to define circular shifts in both dimensions of "floors" defined by an array $\mathbf{F}$ with period (n_1, n_2): $a_{i,j,k} = f_{(i,j)-s_k}$, where $(i, j) - s_k$ is taken modulo (n_1, n_2).

Shift sequences/arrays: Some of the shift sequences that can be used to construct 2-dimensional arrays are: exponential quadratic, logarithmic quadratic, and Moreno-Maric sequences.

To define an **exponential quadratic sequence over** $\mathbb{Z}_p$ with period $p-1$, consider a quadratic polynomial $f(x) = ax^2 + bx + c \in \mathbb{Z}_p[x]$, $a \neq 0$, a primitive element α in $\mathbb{Z}_p$, and take the values of f in $p-1$ consecutive powers of α: $\mathbf{S} = f\left(\alpha^0\right), f\left(\alpha^1\right), \ldots, f\left(\alpha^{p-2}\right)$. For example, for $\alpha = 3 \in \mathbb{Z}_7$, the quadratic polynomial $f(x) = x^2 + x + 1 \in \mathbb{Z}_7[x]$ gives the exponential quadratic sequence $\mathbf{S} = f\left(\alpha^0\right), \ldots, f\left(\alpha^5\right) = 3, 6, 0, 1, 0, 3$.

One can also use quadratic polynomials $f(x)$ over $\mathbb{F}_q$ to define shift sequences of period $q-1$ but, since we need the sequence to have entries in $\mathbb{Z}_n$, we have to map the values of $f(x)$ from $\mathbb{F}_q$ to $\mathbb{Z}_n$. The **logarithmic quadratic sequence over** $\mathbb{F}_q$ with period $q-1$ is defined by writing the non-zero values in $f\left(\alpha^0\right), f\left(\alpha^1\right), \ldots, f\left(\alpha^{q-2}\right)$ as powers of the primitive root α, $f\left(\alpha^i\right) = \alpha^j$, and letting $s_i = \log_\alpha\left(f\left(\alpha^i\right)\right) = \log_\alpha\left(\alpha^j\right) = j$. If $f\left(\alpha^i\right) = 0$, set $s_i = *$. For example, the quadratic polynomial $f(x) = x^2 + x + 2\alpha$ over $\mathbb{F}_{3^2}$ with $\alpha^2 = \alpha + 1$ has values $f\left(\alpha^0\right), \ldots, f\left(\alpha^7\right) = \alpha^3, \alpha^6, \alpha^2, 0, 0, \alpha^5, \alpha^3, \alpha^2$ and gives the logarithmic quadratic shift sequence $\mathbf{S} = 6, 2, *, *, 5, 3, 2, 3$ used in array $\mathbf{A}_2$ of Figure 1.

Under certain conditions on α, the composition $f^n(x)$ of a rational function $f(x) = \alpha/(x+\beta)$ over $\mathbb{F}_p$ with itself, evaluated in $\mathbb{F}_p \cup \{\infty\}$, produces a cycle of length $p+1 : 0, f^1(0), f^2(0), \cdots, f^{p-1}(0), f^p(0) = \infty$, [5, 12, 14, 17]. The **Moreno-Maric sequence over** $\mathbb{Z}_p$ with period $p+1$ is $\mathbf{S} = 0, f^1(0) = \alpha, \ldots, f^{p-1}(0) = -1, *$. For example, $\mathbf{S} = 0, 3, 2, 1, -1, *$ is the Moreno-Maric sequence over $\mathbb{Z}_5$ with $f(x) = 3/(x+1)$. Since the $*$ is always at the end of $\mathbf{S}$, one can remove it to obtain a **shortened Moreno-Maric sequence** of length p.

To construct 3-dimensional arrays $\mathbf{A}$ with period vector (p, p, p^2-1) one can use the **index table W** for the finite field $\mathbb{F}_{p^2}$, which has entries in $\mathbb{Z}_{p^2-1}$, and $w_{0,0} = *$. The column placed in position $(0, 0)$, that is, all entries $a_{0,0,k}$, will be a sequence of a constant value.

Other 3-dimensional arrays with period vector (p, p, p^2-1) can be constructed from the index table $\mathbf{W}$ for the finite field $\mathbb{F}_{p^2}$ but using a **vector shift sequence S** where $s_k = (i, j)$ and $\alpha^k = i\alpha + j$.

Column sequences and "floor" arrays: A good option for column sequences of period $p > 2$ are **Legendre sequences** with respect to $\mathbb{Z}_p$, which are defined as $c_i = 1$ if i is a quadratic nonresidue mod p, and $c_i = 0$ otherwise. For example, $\mathbf{C} = 0, 0, 0, 1, 0, 1, 1$ is the Legendre sequence with respect to $\mathbb{Z}_7$. **Sidelnikov sequences** with respect to $\mathbb{F}_q$, q odd, are defined as $c_i = 1$ if $\alpha^i + 1$ is a quadratic nonresidue in $\mathbb{F}_q$, where α is a primitive element, and $c_i = 0$ otherwise. These sequences can be used as columns of length $q-1$, where q is odd. For example, $\mathbf{C} = 0, 0, 1, 0, 0, 1, 1, 1$ is the Sidelnikov column sequence with respect to $\mathbb{F}_{3^2}$ and $\alpha^2 = \alpha + 1$ used in array $\mathbf{A}_2$ of Figure 1.

The **2-dimensional Legendre arrays** obtained from the index table $\mathbf{W}$ for a finite field $\mathbb{F}_{p^2}$ have period vector (p, p) and can be used as floor arrays.

Constructions for which the complexity was unknown: The unfolding method was used in [13, 14] to compute the multidimensional linear complexity of several constructions. However, it cannot be used to compute the complexity of some arrays described in the same papers. For example, it cannot be used for arrays of dimensions $p \times p$ such as $\mathbf{F}_1$: 2-dimensional binary Legendre arrays,

$\mathbf{F}_2$: 2-dimensional ternary Legendre arrays, or $\mathbf{A}_1$: shortened Moreno-Maric shift sequences composed with Legendre column sequences. The unfolding method can neither be used to compute the linear complexity of arrays of dimensions $(q-1)\times(q-1)$ such as $\mathbf{A}_2$: logarithmic quadratic shift sequences composed with Sidelnikov column sequences (see Figure 1).

$\mathbf{C} =$

1
1
1
0
0
1
0
0

$\mathbf{A}_2 =$

0	1	0	0	1	0	1	0
0	0	0	0	0	0	0	0
1	0	0	0	0	1	0	1
1	1	0	0	1	0	1	0
1	0	0	0	1	0	0	0
0	0	0	0	1	1	0	1
0	1	0	0	0	1	1	1
1	1	0	0	0	1	1	1

$\mathbf{S} =$

6	2	$*$	$*$	5	3	2	3

Fig. 1: Array $\mathbf{A}_2$ constructed composing logarithmic quadratic shift sequence $\mathbf{S}$ with Sidelnikov column sequence $\mathbf{C}$ and $0,\dots,0$ in the columns corresponding to $*$.

The linear complexity of 3-dimensional arrays with dimensions $p\times p\times(p^2-1)$ such as $\mathbf{A}_3$: index table shift array composed with a Sidelnikov sequence, $\mathbf{A}_4$: vector shift sequences composed with 2-dimensional binary Legendre arrays or $\mathbf{A}_5$: vector shift sequences composed with 2-dimensional ternary Legendre arrays cannot be computed using the unfolding method.

4 Results on complexity

4.1 Theoretical results

The following result generalizes a bound for the complexity of arrays presented as Theorem 1 in [2] to include shift sequences with unknown values. This is, shift sequences with elements in $\mathbb{Z}_{n_2}\cup\{*\}$. Define array $\mathbf{A}$ by

$$a_{i,j} = \begin{cases} c_{j-s_i} \pmod{n_2}, & s_i \neq * \\ 0, & s_i = * \end{cases}$$

Theorem 1 *Let* $\mathbf{S}$ *be a shift sequence over* $\mathbb{Z}_{n_2}\cup\{*\}$ *with period* n_1*,* $\mathbf{C}$ *be a column sequence over* $\mathbb{F}_q$ *with period* n_2*, and* $\mathbf{A}$ *be the 2-dimensional array constructed with the composition method where the column corresponding to* $*$ *consists of* 0*'s. Then,* $\mathcal{L}_n(\mathbf{A}) \leq \mathcal{L}_n(\mathbf{C})$*, where* $\mathcal{L}_n(\cdot)$ *is the normalized linear complexity.*

Proof. Let $m(y)$ be the minimal polynomial of $\mathbf{C}$, $m'(x,y) = \sum_{j\in Supp(m)} m'_{0,j} y^j = m(y)$, and $\gamma = (\gamma_1,\gamma_2)\in\mathbb{N}_0^2$. If $s_{\gamma_1}\neq *$, then, since $m\in Val(\mathbf{C})$ implies that $\sum_{j\in Supp(m)} m_j c_{j+\beta} = 0$ for all $\beta\in\mathbb{N}_0$, we have

$$\sum_{(0,j)\in Supp(m')} m'_{0,j} a_{(0,j)+\gamma} = \sum_{j\in Supp(m)} m_j a_{\gamma_1, j+\gamma_2}$$

$$= \sum_{j\in Supp(m)} m_j c_{j+\gamma_2 - s_{\gamma_1}} = \sum_{j\in Supp(m)} m_j c_{j+\beta} = 0,$$

where $\beta = \gamma_2 - s_{\gamma_1}$, and the indices of $\mathbf{C}$ are considered modulo n_2.

If $s_{\gamma_1} = *$, then $a_{\gamma_1, j+\gamma_2} = 0$, and

$$\sum_{(0,j)\in Supp(m')} m'_{0,j} a_{(0,j)+\gamma} = \sum_{j\in Supp(m)} m_j a_{\gamma_1, j+\gamma_2} = 0.$$

Hence, for any $\gamma \in \mathbb{N}_0^2, \sum_{(0,j)\in Supp(m')} m'_{0,j} a_{(0,j)+\gamma} = 0$ and $m'(x,y) = m(y) \in Val(\mathbf{A})$. This implies that $\Delta_{Val(\mathbf{A})}$ cannot contain exponents that are multiples of $y^{\deg(m)}$. Since $\mathbf{S}$ has period $n_1, x^{n_1} - 1 \in Val(\mathbf{A})$ and $\Delta_{Val(\mathbf{A})}$ cannot contain exponents that are multiples of x^{n_1}. Therefore, $\mathcal{L}_n(\mathbf{A}) = \mathcal{L}(\mathbf{A})/(n_1 n_2) \leq n_1 \deg(m)/(n_1 n_2) = \mathcal{L}_n(\mathbf{C})$.

Remark 1. The above proposition is also true for 3-dimensional arrays.

The selection of the values in the column corresponding to the undefined shift $(*)$ affects the complexity of the array as we can see in the next example.

Example 1. Consider the 5×6 array $\mathbf{A}$ constructed by composing the shift Moreno-Maric sequence over $\mathbb{Z}_5$, $\mathbf{S} = 0, 3, 2, 1, 4, *$ with the column Legendre sequence $\mathbf{C} = 0, 0, 1, 1, 0$. If the column corresponding to the undetermined shift $*$ is replaced with a column of constant 1's, then $\mathcal{L}_n(\mathbf{A}) = 13/15$, while $\mathcal{L}_n(\mathbf{C}) = 4/5$. In this case $\mathcal{L}_n(\mathbf{A}) \not\leq \mathcal{L}_n(\mathbf{C})$.

The following refinement for the bound in Theorem 1 for the cases where $y-1$ divides the minimal polynomial of the sequence $\mathbf{C}$ and $\mathbf{S}$ is a shift sequence over $\mathbb{Z}_{n_2}$, was proved in [2].

Proposition 1 *Let $\mathbf{S}$ be a shift sequence over $\mathbb{Z}_{n_2}$ with period n_1, $\mathbf{C}$ be a column sequence over $\mathbb{F}_q$ with period n_2 and minimal polynomial $m(y)$, and $\mathbf{A}$ be the 2-dimensional array constructed with the composition method. If $y-1$ divides $m(y)$, then $\mathcal{L}_n(\mathbf{A}) \leq \mathcal{L}_n(\mathbf{C}) - \frac{n_1-1}{n_1 n_2}$, where $\mathcal{L}_n(\cdot)$ is the normalized linear complexity.*

Comparison of the linear complexity of an array A with the joint linear complexity of A as a multisequence: As it was mentioned before, our definition of multidimensional linear complexity is more accurate than the joint linear complexity for multisequences.

Proposition 2 *Let $\mathbf{A}$ be a periodic 2-dimensional array with period (n_1, n_2). Then, the normalized linear complexity of $\mathbf{A}$, $\mathcal{L}_n(\mathbf{A})$, is smaller or equal than the normalized joint linear complexity of $\mathbf{A}$ considered as a multisequence, $J\mathcal{L}_n(\mathbf{A})$. This is, $\mathcal{L}_n(\mathbf{A}) \leq J\mathcal{L}_n(\mathbf{A})$.*

Proof. Let $m(y)$ be the joint minimal polynomial of an n_1-fold multisequence $\mathbf{A}$. Then, $m(y)$ is valid for each of the columns $a_{k,0}, a_{k,1}, \ldots, a_{k,n_2-1},\ 0 \leq k <$

n_1, and $\sum_{j \in Supp(m)} m_j a_{k,j+\gamma_2} = 0$ for each $0 \le k < n_1$ and all $\gamma_2 \in \mathbb{N}_0$, where $j + \gamma_2$ is considered modulo n_2. Let $\gamma = (\gamma_1, \gamma_2) \in \mathbb{N}_0^2$ and $m'(x,y) = \sum_{Supp(m)} m'_{i,j} x^i y^j$, where $m'_{i,j} = m_j$ for a fixed $0 \le i < n_1$. Then,

$$\sum_{(i,j) \in Supp(m')} m'_{i,j} a_{(i,j)+\gamma} = \sum_i \sum_{j \in Supp(m)} m_j a_{i+\gamma_1, j+\gamma_2} = \sum_i 0 = 0,$$

since $k = i + \gamma_1$ is fixed in the inner sum. This implies $m'(x,y) \in Val(\mathbf{A})$, and $\Delta_{Val(\mathbf{A})}$ cannot contain $x^i y^{\deg(m)}$ for any $0 \le i < n_1$. Hence, $\mathcal{L}(\mathbf{A}) = |\Delta_{Val(\mathbf{A})}| \le n_1 \deg(m)$ and $\mathcal{L}_n(\mathbf{A}) \le \deg(m)/n_2 = J\mathcal{L}_n(\mathbf{A})$.

There are examples of arrays $\mathbf{A}$ for which $\mathcal{L}_n(\mathbf{A})$, is strictly smaller than $J\mathcal{L}_n(\mathbf{A})$. When an array $\mathbf{A}$ constructed with columns from shifts of the same column sequence $\mathbf{C}$ is considered as a multisequence, the minimal polynomial of $\mathbf{C}$ is valid for all the columns. Hence, the normalized joint linear complexity of $\mathbf{A}$ is equal to the normalized linear complexity of $\mathbf{C}$, $J\mathcal{L}_n(\mathbf{A}) = J\mathcal{L}_n(\mathbf{C})$. From Proposition 1 when $y - 1$ divides the minimal polynomial of $\mathbf{C}$ one has $\mathcal{L}_n(\mathbf{A}) < J\mathcal{L}_n(\mathbf{A})$. The joint linear complexity of $\mathbf{A}$ misses some relations among entries of different columns.

Example 2. Consider the 7×6 array $\mathbf{A}$ constructed by composing the exponential quadratic sequence $\mathbf{S} = 3, 6, 0, 1, 0, 3$ with the column Legendre sequence with respect to $\mathbb{Z}_7$, $\mathbf{C} = 0, 0, 0, 1, 0, 1, 1$. Our definition gives normalized linear complexity $\mathcal{L}_n(\mathbf{A}) = 19/42$. If one considers $\mathbf{A}$ as a multisequence, the normalized joint linear complexity is $J\mathcal{L}_n(\mathbf{A}) = 4/7$ which is larger than $\mathcal{L}_n(\mathbf{A})$.

4.2 Computational results and conjectures

Our computational results focus on arrays for which the linear complexity could not be computed with the unfolding method in [7, 14] because the periods of the arrays were not relatively prime. We computed the complexity of 2-dimensional $p \times p$ arrays from constructions $\mathbf{F}_1, \mathbf{F}_2$ and 3-dimensional $p \times p \times (p^2 - 1)$ arrays from constructions $\mathbf{A}_3, \mathbf{A}_4, \mathbf{A}_5$. The complexity was computed using a C++ implementation [19] of the RST algorithm [16]. All the examples satisfy the conjectured formulas for the normalized linear complexity in Table 1.

Recall that construction $\mathbf{A}_3$ uses Sidelnikov column sequences $\mathbf{C}$, $\mathbf{A}_4$ uses array $\mathbf{F}_1$ as floor, and $\mathbf{A}_5$ uses array $\mathbf{F}_2$ as floor. As seen in Table 1, both $\mathbf{A}_4, \mathbf{A}_5$ have the same normalized complexity as their corresponding floor array. In the case of $\mathbf{A}_3$, one can see that as the size of the array increases (size depends on p), the value $\mathcal{L}_n(\mathbf{A}_3)$ approaches the value of $\mathcal{L}_n(\mathbf{C})$.

We do not have a conjecture of a formula for the complexity of arrays from constructions $\mathbf{A}_1, \mathbf{A}_2$. However, these arrays have the same behaviour of arrays from constructions $\mathbf{A}_3, \mathbf{A}_4, \mathbf{A}_5$, in the sense that their complexities approach the complexity of the column/floor sequence/array. This can be seen in Figure 2, where the graphs show that the difference of the normalized linear complexity of

Table 1: Conjectured formulas for the normalized linear complexity.

Construction	Conjectured $\mathcal{L}_n(\cdot)$	Verified for
$\mathbf{F}_1$	$\frac{1}{2} - \frac{1}{2p^2}$	$p \leq 251$
$\mathbf{F}_2$	$1 - \frac{1}{p^2}$	$3 < p \leq 109$
$\mathbf{A}_3$	$\mathcal{L}_n(\mathbf{C})[1 - \frac{1}{p^2}]$	$p \leq 19$
$\mathbf{A}_4$	$\frac{1}{2} - \frac{1}{2p^2} = \mathcal{L}_n(\mathbf{F}_1)$	$p \leq 23$
$\mathbf{A}_5$	$1 - \frac{1}{p^2} = \mathcal{L}_n(\mathbf{F}_2)$	$3 < p \leq 17$

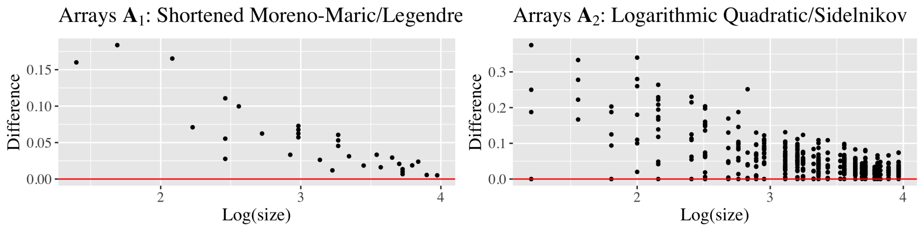

Fig. 2: Difference of normalized complexities $\mathcal{L}_n(\mathbf{C}) - \mathcal{L}_n(\mathbf{A})$ as a function of the log of the size of $\mathbf{A}$, $\log(n_1 n_2)$, where each dot represents an array $\mathbf{A}$ with period vector (n_1, n_2).

arrays composed with column sequences and the normalized linear complexity of the column approaches 0 as the size of the array increases.

Based on the above results and the results from [7, 14], we have the following general conjecture.

Conjecture 1. If $\mathbf{A}$ is an array constructed by composing a shift sequence/array with a column sequence $\mathbf{C}$ or floor array $\mathbf{F}$ of suitable dimensions, then as the size of $\mathbf{A}$ increases, $\mathcal{L}_n(\mathbf{A})$ approaches the normalized complexity of the column $\mathcal{L}_n(\mathbf{C})$ or of the floor $\mathcal{L}_n(\mathbf{F})$.

We validated Conjecture 1 by computing the normalized complexity of arrays constructed by composing a randomly generated shift sequence with a randomly generated binary column sequence, each of period n, for n a multiple of 5, $5 \leq n \leq 100$. We sampled 25 arrays for each n and computed the normalized complexity of each corresponding random column sequence.

References

1. Alderson, T.: n-dimensional optical orthogonal codes, bounds and optimal constructions. Applicable Algebra in Engineering, Communication and Computing **30**(5), 373–386 (2019)
2. Arce-Nazario, R., Castro, F., Gomez-Perez, D., Moreno, O., Ortiz-Ubarri, J., Rubio, I., Tirkel, A.: Multidimensional linear complexity analysis of periodic arrays.

Applicable Algebra in Engineering, Communication and Computing **31**(1), 43–63 (2020)

3. Ding, C., Helleseth, T., Shan, W.: On the linear complexity of Legendre sequences. IEEE Trans. Inform. Theory **44**(3), 1276–1278 (1998). https://doi.org/10.1109/18.669398, http://dx.doi.org/10.1109/18.669398
4. Gómez-Pérez, D., Gómez, A.I., Tirkel, A.: Large families of sequences for cdma, frequency hopping, and uwb. Cryptography and Communications pp. 1–15 (2019)
5. Gomez-Perez, D., Gomez, A.I., Tirkel, A.: Arrays composed from the extended rational cycle. Advances in Mathematics of Communications **11**(2), 313 (2017)
6. Gomez-Perez, D., Hoholdt, T., Moreno, O., Rubio, I.: Linear complexity for multidimensional arrays - a numerical invariant. In: Information Theory (ISIT), 2015 IEEE International Symposium on. pp. 2697–2701 (June 2015). https://doi.org/10.1109/ISIT.2015.7282946
7. Leukhin, A., Moreno, O., Tirkel, A.: Secure CDMA and frequency hop sequences. In: Wireless Communication Systems (ISWCS 2013), Proceedings of the Tenth International Symposium on. pp. 1–5. VDE (2013)
8. MacWilliams, F.J., Sloane, N.J.: Pseudo-random sequences and arrays. Proceedings of the IEEE **64**(12), 1715–1729 (1976)
9. Massey, J.L.: Shift-register synthesis and BCH decoding. IEEE Trans. Information Theory **IT-15**, 122–127 (1969)
10. Meidl, W., Niederreiter, H.: The expected value of the joint linear complexity of periodic multisequences. Journal of Complexity **19**(1), 61–72 (2003)
11. Moreno, O., Høholdt, T., Rubio, I.: Security of multidimensional arrays (62/131616 (March, 2015) and 62/174973 (June, 2015))
12. Moreno, O., Maric, S.V.: A new family of frequency-hop codes. IEEE Transactions on Communications **48**(8), 1241–1244 (2000)
13. Moreno, O., Tirkel, A.: Multi-dimensional arrays for watermarking. In: Information Theory Proceedings (ISIT), 2011 IEEE International Symposium on. pp. 2691–2695. IEEE (2011)
14. Moreno, O., Tirkel, A.: New optimal low correlation sequences for wireless communications. In: Sequences and their applications—SETA 2012, Lecture Notes in Comput. Sci., vol. 7280, pp. 212–223. Springer, Heidelberg (2012)
15. Ortiz-Ubarri, J.: New asymptotically optimal three-dimensional wavelength/space/time optical orthogonal codes for ocdma systems. Cryptography and Communications pp. 1–10 (2020)
16. Rubio, I.M., Sweedler, M., Heegard, C.: Finding a gröbner basis for the ideal of recurrence relations on m-dimensional periodic arrays. In: Contemporary Developments in Finite Fields and Applications, pp. 296–320. World Scientific (2016)
17. Tirkel, A., Hall, T.: New matrices with good auto and cross-correlation. IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences **89**(9), 2315–2321 (2006)
18. Tirkel, A.Z., Osborne, C.F., Hall, T.H.: Steganography-applications of coding theory. In: IEEE Information Theory Workshop, Svalbard, Norway. pp. 57–59 (1997)
19. Torres, J.: RST_Complexity. https://github.com/jazieltorres/RST_Complexity.git (2019)
20. Yao, J., Jiang, R., Heng, W.: Algebraic construction of optimal frequency hopping patterns based on welch costas arrays. IEEE Transactions on Vehicular Technology **69**(2), 1841–1854 (2019)