On multidimensional periodic arrays

Ivelisse Rubio
Department of Computer Science
University of Puerto Rico, Río Piedras

Carleton Finite Fields eSeminar

March 31, 2021

Outline

Collaborators

- Oscar Moreno - Andrew Tirkel
  - Rafael Arce
  - Francis Castro
  - Domingo Gómez
  - Carlos Hernández
  - Tom Hoholdt
  - José Ortiz
  - Andrés Ramos
  - David Thomson
  - Jaziel Torres

Ongoing work

- Rafael Arce
- Carlos Hernández
- José Ortiz
- Jaziel Torres

The problem

To study constructions and properties of Multidimensional arrays that can be used for applications in

- Digital watermarking
- Code division multiple access (CDMA)
- Multiple target recognition
- Optical orthogonal codes

Properties of the array

- Correlations
- Balance
- Large family size
- Variety of sizes

Properties of the array

- Correlations
- Balance
- Large family size
- Variety of sizes
- Linear complexity (resistance to a Berlekamp-Massey attack)

Constructions proposed by Moreno and Tirkel

- A sequence with good correlation properties and good complexity to construct columns
- A sequence/array with good correlation properties to shift the columns

Constructions proposed by Moreno and Tirkel

- A sequence with good correlation properties and good complexity to construct columns
- A sequence/array with good correlation properties to shift the columns

- Their constructions preserve properties of balance and correlation.

Constructions proposed by Moreno and Tirkel

- A sequence with good correlation properties and good complexity to construct columns
- A sequence/array with good correlation properties to shift the columns

- Their constructions preserve properties of balance and correlation.
- Problems computing complexity

Problem

How to define and compute multidimensional linear complexity?

Results

- Provided a definition and method to compute multidimensional linear complexity of multidimensional periodic arrays.

Results

- Provided a definition and method to compute multidimensional linear complexity of multidimensional periodic arrays.
- Definition generalizes the concept and measure of linear complexity of sequences.

Results

- Provided a definition and method to compute multidimensional linear complexity of multidimensional periodic arrays.
- Definition generalizes the concept and measure of linear complexity of sequences.
- No restrictions on the periods of the array.

Results

- Provided a definition and method to compute multidimensional linear complexity of multidimensional periodic arrays.
- Definition generalizes the concept and measure of linear complexity of sequences.
- No restrictions on the periods of the array.
- A measure more accurate than the one given for multisequences.

Results

- Provided a definition and method to compute multidimensional linear complexity of multidimensional periodic arrays.
- Definition generalizes the concept and measure of linear complexity of sequences.
- No restrictions on the periods of the array.
- A measure more accurate than the one given for multisequences.
- Proved some bounds for the complexity.

Results

- Provided a definition and method to compute multidimensional linear complexity of multidimensional periodic arrays.
- Definition generalizes the concept and measure of linear complexity of sequences.
- No restrictions on the periods of the array.
- A measure more accurate than the one given for multisequences.
- Proved some bounds for the complexity.
- Proved formulas for the exact value of the complexity of some specific arrays.

Results

- Provided a definition and method to compute multidimensional linear complexity of multidimensional periodic arrays.
- Definition generalizes the concept and measure of linear complexity of sequences.
- No restrictions on the periods of the array.
- A measure more accurate than the one given for multisequences.
- Proved some bounds for the complexity.
- Proved formulas for the exact value of the complexity of some specific arrays.
- Implemented our method to compute multidimensional linear complexity.

Results

- Provided a definition and method to compute multidimensional linear complexity of multidimensional periodic arrays.
- Definition generalizes the concept and measure of linear complexity of sequences.
- No restrictions on the periods of the array.
- A measure more accurate than the one given for multisequences.
- Proved some bounds for the complexity.
- Proved formulas for the exact value of the complexity of some specific arrays.
- Implemented our method to compute multidimensional linear complexity.
- Results are compatible with results for sequences and computations with unfolding method.

Results

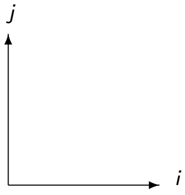
- Provided a definition and method to compute multidimensional linear complexity of multidimensional periodic arrays.
- Definition generalizes the concept and measure of linear complexity of sequences.
- No restrictions on the periods of the array.
- A measure more accurate than the one given for multisequences.
- Proved some bounds for the complexity.
- Proved formulas for the exact value of the complexity of some specific arrays.
- Implemented our method to compute multidimensional linear complexity.
- Results are compatible with results for sequences and computations with unfolding method.
- Computed multidimensional linear complexity of arrays that could not be computed before.

Outline

Multidimensional periodic arrays

Construction of 2-dimensional arrays

Multidimensional periodic arrays



$$(i, j)$$

$$X^i Y^j$$

2-D Composition method (example)

A **column** sequence with good linear complexity:

2-D Composition method (example)

A **column** sequence with good linear complexity:

Legendre ($\mathbb{F}_7$): $c = \ 0, 0, 0, 1, 0, 1, 1, \dots$

2-D Composition method (example)

A **column** sequence with good linear complexity:

Legendre ($\mathbb{F}_7$): $c = 0, 0, 0, 1, 0, 1, 1, \ldots$

A **shift** sequence with good correlation properties:

2-D Composition method (example)

A **column** sequence with good linear complexity:

Legendre ($\mathbb{F}_7$): $c = \; 0, 0, 0, 1, 0, 1, 1, \ldots$

A **shift** sequence with good correlation properties:

Welch: $s_i = 3^i \pmod 7$, $s = \; 1, 3, 2, 6, 4, 5, \ldots$

2-D Composition method (example)

A **column** sequence with good linear complexity:

Legendre ($\mathbb{F}_7$): $c = 0, 0, 0, 1, 0, 1, 1, \ldots$

A **shift** sequence with good correlation properties:

Welch: $s_i = 3^i \pmod 7$, $s = 1, 3, 2, 6, 4, 5, \ldots$

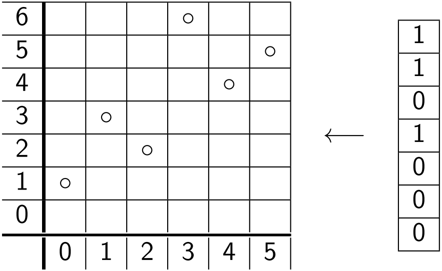
$s =$

6				○		
5						○
4					○	
3		○				
2			○			
1	○					
0						
	0	1	2	3	4	5

2-D Composition method (example)

6				○		
5						○
4					○	
3		○				
2			○			
1	○					
0						
	0	1	2	3	4	5

2-D Composition method (example)



2-D Composition method (example)

6				○		
5						○
4					○	
3		○				
2			○			
1	○					
0						
	0	1	2	3	4	5

$\longleftarrow$

1
1
0
1
0
0
0

$A =$

6	1	1	0	0	0	0
5	0	0	1	1	0	0
4	1	0	0	1	0	1
3	0	0	0	0	1	1
2	0	1	0	1	1	0
1	0	1	1	0	0	1
0	1	0	1	0	1	0
	0	1	2	3	4	5

$p \times p-1$ periodic array

$$A_{i,j} = c_{j-s_i} \pmod p$$

2-D Composition method

- Good things:

2-D Composition method

- Good things:
  1. Balance properties

2-D Composition method

- Good things:
  1. Balance properties
  2. Correlations

2-D Composition method

- Good things:
  1. Balance properties
  2. Correlations
- Problems to solve:

2-D Composition method

- Good things:
  1. Balance properties
  2. Correlations

- Problems to solve:
  1. Need families of arrays

2-D Composition method

- Good things:
  1. Balance properties
  2. Correlations

- Problems to solve:
  1. Need families of arrays
  2. Cannot compute linear complexity of all the arrays

2-D Composition method

- Good things:
  1. Balance properties
  2. Correlations

- Problems to solve:
  1. Need families of arrays
  2. Cannot compute linear complexity of all the arrays

- Solutions:

2-D Composition method

- Good things:
  1. Balance properties
  2. Correlations

- Problems to solve:
  1. Need families of arrays
  2. Cannot compute linear complexity of all the arrays

- Solutions:
  1. Consider other shift sequences

2-D Composition method

- Good things:
  1. Balance properties
  2. Correlations

- Problems to solve:
  1. Need families of arrays
  2. Cannot compute linear complexity of all the arrays

- Solutions:
  1. Consider other shift sequences
  2. Definition and method to compute multidimensional linear complexity

Composition method: Other shift sequences

Exponential quadratic:

$$s_i = A\alpha^{2i} + B\alpha^{i} + C$$

$$A, B, C \in \mathbb{F}_q, \quad \alpha \text{ a primitive element in } \mathbb{F}_q$$

Composition method: Other shift sequences

Exponential quadratic:

$$s_i = A\alpha^{2i} + B\alpha^{i} + C$$

$$A, B, C \in \mathbb{F}_q, \quad \alpha \text{ a primitive element in } \mathbb{F}_q$$

Rational functions:

$$f(x) = \frac{Ax + B}{Cx + D}$$

$$A, B, C, D \in \mathbb{F}_q, \quad AD \neq BC$$

2-D Generalized Legendre (example)

Use the **index table** of the finite field $\mathbb{F}_{p^2}$.

2-D Generalized Legendre (example)

Use the **index table** of the finite field $\mathbb{F}_{p^2}$.

Let $f(x) = x^2 + 2x + 2 \in \mathbb{F}_3[x]$ and $f(\alpha) = 0$.

2-D Generalized Legendre (example)

Use the **index table** of the finite field $\mathbb{F}_{p^2}$.

Let $f(x) = x^2 + 2x + 2 \in \mathbb{F}_3[x]$ and $f(\alpha) = 0$.

$$\alpha^2 = \alpha + 1$$

2-D Generalized Legendre (example)

Use the **index table** of the finite field $\mathbb{F}_{p^2}$.

Let $f(x) = x^2 + 2x + 2 \in \mathbb{F}_3[x]$ and $f(\alpha) = 0$.

$$\alpha^2 = \alpha + 1 \qquad \alpha^k = i\alpha + j = (i, j), \quad i, j \in \mathbb{F}_3$$

2-D Generalized Legendre (example)

Use the **index table** of the finite field $\mathbb{F}_{p^2}$.

Let $f(x) = x^2 + 2x + 2 \in \mathbb{F}_3[x]$ and $f(\alpha) = 0$.

$$\alpha^2 = \alpha + 1 \qquad \alpha^k = i\alpha + j = (i, j), \quad i, j \in \mathbb{F}_3$$

2	α^4	α^7	α^6
1	α^0	α^2	α^3
0	$*$	α^1	α^5
j/i	0	1	2

$\xrightarrow{\log}$

2-D Generalized Legendre (example)

Use the **index table** of the finite field $\mathbb{F}_{p^2}$.

Let $f(x) = x^2 + 2x + 2 \in \mathbb{F}_3[x]$ and $f(\alpha) = 0$.

$$\alpha^2 = \alpha + 1 \qquad \alpha^k = i\alpha + j = (i, j), \quad i, j \in \mathbb{F}_3$$

$$\begin{array}{c|c|c|c|} 2 & \alpha^4 & \alpha^7 & \alpha^6 \\ \hline 1 & \alpha^0 & \alpha^2 & \alpha^3 \\ \hline 0 & * & \alpha^1 & \alpha^5 \\ \hline\hline j/i & 0 & 1 & 2 \end{array} \quad \xrightarrow{\log} \quad W = \begin{array}{c|c|c|c|} 2 & 4 & 7 & 6 \\ \hline 1 & 0 & 2 & 3 \\ \hline 0 & * & 1 & 5 \\ \hline\hline j/i & 0 & 1 & 2 \end{array}$$

$$W_{0,0} = *, \quad W_{i,j} = k, \quad \text{where } \alpha^k = (i, j)$$

Generalized Legendre (example)

Use the **index table** of the finite field $\mathbb{F}_{p^2}$ and take the entries $\pmod 2$:

$$W = \begin{array}{c|ccc} 2 & 4 & 7 & 6 \\ 1 & 0 & 2 & 3 \\ 0 & * & 1 & 5 \\ \hline j/i & 0 & 1 & 2 \end{array} \quad \xrightarrow{\pmod 2}$$

Generalized Legendre (example)

Use the **index table** of the finite field $\mathbb{F}_{p^2}$ and take the entries $\pmod 2$:

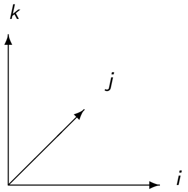
$$W = \begin{array}{c|ccc} 2 & 4 & 7 & 6 \\ 1 & 0 & 2 & 3 \\ 0 & * & 1 & 5 \\ \hline j/i & 0 & 1 & 2 \end{array} \xrightarrow{\pmod 2} A = \begin{array}{c|ccc} 2 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ \hline j/i & 0 & 1 & 2 \end{array}$$

$$A_{0,0} = 0, \quad A_{i,j} = k \pmod 2, \quad \text{where } \alpha^k = (i,j)$$

Multidimensional periodic arrays

Construction of 3-dimensional arrays

Multidimensional periodic arrays



$$(i, j, k)$$

3-D Composition method (example)

A 2-dimensional array with good correlation properties as a **shifting array**:

3-D Composition method (example)

A 2-dimensional array with good correlation properties as a **shifting array**:

Use the **index table** of the finite field $\mathbb{F}_{p^2}$.

Let $f(x) = x^2 + 2x + 2 \in \mathbb{F}_3[x]$ and $f(\alpha) = 0$.

$$\alpha^2 = \alpha + 1 \qquad \alpha^i = i\alpha + j = (i, j), \quad i, j \in \mathbb{F}_3$$

3-D Composition method (example)

A 2-dimensional array with good correlation properties as a **shifting array**:

Use the **index table** of the finite field $\mathbb{F}_{p^2}$.

Let $f(x) = x^2 + 2x + 2 \in \mathbb{F}_3[x]$ and $f(\alpha) = 0$.

$$\alpha^2 = \alpha + 1 \qquad \alpha^i = i\alpha + j = (i, j), \quad i, j \in \mathbb{F}_3$$

$$\begin{array}{c|c|c|c|}
2 & \alpha^4 & \alpha^7 & \alpha^6 \\ \hline
1 & \alpha^0 & \alpha^2 & \alpha^3 \\ \hline
0 & * & \alpha^1 & \alpha^5 \\ \hline
j/i & 0 & 1 & 2
\end{array}
\xrightarrow{\log}
W = \begin{array}{c|c|c|c|}
2 & 4 & 7 & 6 \\ \hline
1 & 0 & 2 & 3 \\ \hline
0 & * & 1 & 5 \\ \hline
j/i & 0 & 1 & 2
\end{array}$$

$$W_{0,0} = *, \quad W_{i,j} = k, \quad \text{where } \alpha^k = (i, j)$$

3-D Composition method (example)

A 2-dimensional array with good correlation properties as a **shifting array**:

Use the **index table** of the finite field $\mathbb{F}_{p^2}$.

Let $f(x) = x^2 + 2x + 2 \in \mathbb{F}_3[x]$ and $f(\alpha) = 0$.

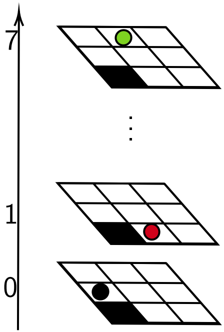
$$\alpha^2 = \alpha + 1 \qquad \alpha^i = i\alpha + j = (i, j), \quad i, j \in \mathbb{F}_3$$

$$\begin{array}{c|c|c|c|} 2 & \alpha^4 & \alpha^7 & \alpha^6 \\ \hline 1 & \alpha^0 & \alpha^2 & \alpha^3 \\ \hline 0 & * & \alpha^1 & \alpha^5 \\ \hline j/i & 0 & 1 & 2 \end{array} \quad \xrightarrow{\log} \quad W = \begin{array}{c|c|c|c|} 2 & 4 & 7 & 6 \\ \hline 1 & 0 & 2 & 3 \\ \hline 0 & * & 1 & 5 \\ \hline j/i & 0 & 1 & 2 \end{array}$$

$$W_{0,0} = *, \quad W_{i,j} = k, \quad \text{where } \alpha^k = (i, j)$$

Entries mark the "floor" where the circles for the shift are placed.

3-D Composition method



4	7	6
0	2	3
*	1	5

NOTE: There are $p^2 - 1$ layers; each layer has a shifting position.

Thanks to Andrés Ramos!

3-D Composition method (example)

* A 2-dimensional array with good correlation properties as a **shifting array**.

4	7	6
0	2	3
*	1	5

* A **column sequence** c of length $p^2 - 1$

3-D Composition method (example)

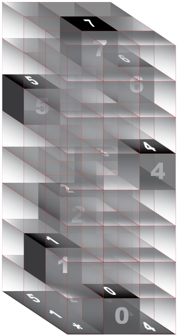
* A 2-dimensional array with good correlation properties as a **shifting array**.

4	7	6
0	2	3
*	1	5

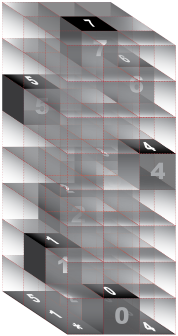
* A **column sequence** c of length $p^2 - 1$

NOTE: There are $p^2 - 1$ layers; each layer has a shifting position.

3-D Composition method (example)



3-D Composition method (example)



$$A_{ijk} = c_{k-\log(i,j) \pmod{p^2-1}} = c_{k-h} \pmod{p^2-1}$$

$$\alpha^h = i\alpha + j$$

Thanks to Jaziel Torres!

3-D Composition method (example)

A 2-dimensional array with good correlation properties as a **shifting array**

A **column sequence** of commensurate length (Sidelnikov)

3-D Composition method (example)

A 2-dimensional array with good correlation properties as a **shifting array**

A **column sequence** of commensurate length (Sidelnikov) or

A 3-dimensional array with good correlation properties as a **shifting array**

A **"floor array"** of commensurate dimensions (Generalized Legendre)

Outline

Periodic sequences

$$s = s_0, s_1, s_2, \ldots$$

Periodic sequences

$$s = s_0, s_1, s_2, \ldots$$

Periodic with period n if

$$s_{i+n} = s_i, \quad \text{for all} \quad i \in \mathbb{N}_0.$$

Periodic sequences

$$s = s_0, s_1, s_2, \ldots$$

Periodic with period n if

$$s_{i+n} = s_i, \quad \text{for all} \quad i \in \mathbb{N}_0.$$

This is a **recurrence relation**.

Periodic sequences

$$s = s_0, s_1, s_2, \ldots$$

Periodic with period n if

$$s_{i+n} = s_i, \quad \text{for all} \quad i \in \mathbb{N}_0.$$

This is a **recurrence relation**.

$$s_n = s_0, \quad s_n - s_0 = 0$$

Periodic sequences

$$s = s_0, s_1, s_2, \ldots$$

Periodic with period n if

$$s_{i+n} = s_i, \quad \text{for all} \quad i \in \mathbb{N}_0.$$

This is a **recurrence relation**.

$$s_n = s_0, \quad s_n - s_0 = 0$$

$$x^n - 1$$

Recurrence relations

$$s_u + \sum_{i<u} c_i s_{i+\beta} = 0$$

Recurrence relations

$$s_u + \sum_{i<u} c_i s_{i+\beta} = 0$$

$$C(x) = \sum_{i \in Supp(C)} c_i x^i$$

Recurrence relations

$$s_u + \sum_{i<u} c_i s_{i+\beta} = 0$$

$$C(x) = \sum_{i \in Supp(C)} c_i x^i$$

Definition

The polynomial C defines a **linear recurrence relation for the sequence** s if the equation

$$\sum_{i \in Supp(C)} c_i s_{i+\beta} = 0 \quad \text{holds for all} \quad \beta \in \mathbb{N}_0.$$

Recurrence relations

$$s_u + \sum_{i<u} c_i s_{i+\beta} = 0$$

$$C(x) = \sum_{i \in Supp(C)} c_i x^i$$

Definition

The polynomial C defines a **linear recurrence relation for the sequence** s if the equation

$$\sum_{i \in Supp(C)} c_i s_{i+\beta} = 0 \quad \text{holds for all} \quad \beta \in \mathbb{N}_0.$$

We say that C is **valid for the sequence** s, $C \in Val(s)$.

Linear complexity

- A recurrence polynomial for the sequence generates the sequence.

Linear complexity

- A recurrence polynomial for the sequence generates the sequence.
- The set of valid polynomials for the sequence $Val(s)$ form an ideal in $\mathbb{F}[x]$.

Linear complexity

- A recurrence polynomial for the sequence generates the sequence.
- The set of valid polynomials for the sequence $Val(s)$ form an ideal in $\mathbb{F}[x]$.
- The minimal polynomial of the sequence is a generator of $Val(s)$.

Linear complexity

- A recurrence polynomial for the sequence generates the sequence.
- The set of valid polynomials for the sequence $Val(s)$ form an ideal in $\mathbb{F}[x]$.
- The minimal polynomial of the sequence is a generator of $Val(s)$.
- The linear complexity measures the resistance to a Berlekamp-Massey attack.

Linear complexity

- A recurrence polynomial for the sequence generates the sequence.
- The set of valid polynomials for the sequence $Val(s)$ form an ideal in $\mathbb{F}[x]$.
- The minimal polynomial of the sequence is a generator of $Val(s)$.
- The linear complexity measures the resistance to a Berlekamp-Massey attack.

Definition

The **linear complexity** of a periodic sequence s is the degree of a minimal polynomial that generates the sequence.

Periodic arrays

$\vdots$			$\vdots$	$\cdots$
$a_{0,3}$	$a_{1,3}$	$a_{2,3}$	$a_{3,3}$	
$a_{0,2}$	$a_{1,2}$	$a_{2,2}$	$a_{3,2}$	$\cdots$
$a_{0,1}$	$a_{1,1}$	$a_{2,1}$	$a_{3,1}$	
$a_{0,0}$	$a_{1,0}$	$a_{2,0}$	$a_{3,0}$	$\cdots$

How can we define (and compute!) the **linear complexity** of a periodic array????

Periodic arrays

$\vdots$			$\vdots$	$\cdots$
$a_{0,3}$	$a_{1,3}$	$a_{2,3}$	$a_{3,3}$	
$a_{0,2}$	$a_{1,2}$	$a_{2,2}$	$a_{3,2}$	$\cdots$
$a_{0,1}$	$a_{1,1}$	$a_{2,1}$	$a_{3,1}$	
$a_{0,0}$	$a_{1,0}$	$a_{2,0}$	$a_{3,0}$	$\cdots$

How can we define (and compute!) the **linear complexity** of a periodic array????

The definition should be consistent both conceptually and numerically with the one dimensional case.

Linear complexity of periodic arrays (unfolding)

Old trick: Transform the problem to the one you know how to solve!

Linear complexity of periodic arrays (unfolding)

Old trick: Transform the problem to the one you know how to solve!

If the periods are relatively prime, one can

- "unfold" the array using the Chinese Remainder Theorem
- construct a sequence from the array

Linear complexity of periodic arrays (unfolding)

Old trick: Transform the problem to the one you know how to solve!

If the periods are relatively prime, one can

- "unfold" the array using the Chinese Remainder Theorem
- construct a sequence from the array
- use the Berlekamp-Massey algorithm to find a minimal generator
- the complexity is the degree of a minimal generator

Linear complexity of periodic arrays (unfolding)

Old trick: Transform the problem to the one you know how to solve!

If the periods are relatively prime, one can

- "unfold" the array using the Chinese Remainder Theorem
- construct a sequence from the array
- use the Berlekamp-Massey algorithm to find a minimal generator
- the complexity is the degree of a minimal generator

Problem: restriction in the period of the array.

Multidimensional periodic arrays

$\vdots$			$\vdots$						$\cdots$
3	1	4	2	3	1	4	2	3	$\cdots$
4	6	0	5	4	6	0	5	4	
2	3	1	5	2	3	1	5	2	$\cdots$
2	4	1	3	2	4	1	3	2	
0	5	3	1	0	5	3	1	0	
3	1	4	2	3	1	4	2	3	$\cdots$
4	6	0	5	4	6	0	5	4	
2	3	1	5	2	3	1	5	2	$\cdots$

$$(n_1, n_2) = (4, 5)$$

$$a_{i+4k_1,\ j+5k_2} = a_{i,j}$$

Multidimensional periodic arrays

Definition

A 2-dimensional array a is said to be 2-**dimensional periodic** if there is a **period vector**, $n = (n_1, n_2) \in \mathbb{N}^2$, such that

$$a_{i+k_1 n_1,\ j+k_2 n_2} = a_{i,j}$$

for all $(i, j), (k_1, k_2) \in \mathbb{N}_0^2$.

Periodic arrays (our approach)

The array is **periodic** with period $n = (n_1, n_2)$ if

$$a_{i+k_1 n_1,\ j+k_2 n_2} = a_{i,j} \quad \text{for all} \quad (i,j), (k_1, k_2) \in \mathbb{N}_0^2.$$

Periodic arrays (our approach)

The array is **periodic** with period $n = (n_1, n_2)$ if

$$a_{i+k_1 n_1,\ j+k_2 n_2} = a_{i,j} \quad \text{for all} \quad (i,j)\,,(k_1,k_2) \in \mathbb{N}_0^2.$$

This is a **recurrence relation**.

$$a_{n_1,0} = a_{0,0} \quad \text{and} \quad a_{n_1,0} - a_{0,0} = 0,$$

Periodic arrays (our approach)

The array is **periodic** with period $n = (n_1, n_2)$ if

$$a_{i+k_1 n_1,\ j+k_2 n_2} = a_{i,j} \quad \text{for all} \quad (i,j)\,,(k_1,k_2) \in \mathbb{N}_0^2.$$

This is a **recurrence relation**.

$$a_{n_1,0} = a_{0,0} \quad \text{and} \quad a_{n_1,0} - a_{0,0} = 0,$$

$$x^{n_1} - 1 \in Val(a)$$

Periodic arrays (our approach)

The array is **periodic** with period $n = (n_1, n_2)$ if

$$a_{i+k_1 n_1,\ j+k_2 n_2} = a_{i,j} \ \text{ for all } \ (i,j), (k_1,k_2) \in \mathbb{N}_0^2.$$

This is a **recurrence relation**.

$$a_{n_1,0} = a_{0,0} \quad \text{and} \quad a_{n_1,0} - a_{0,0} = 0,$$

$$x^{n_1} - 1 \in Val(a)$$

$$a_{0,n_2} = a_{0,0} \quad \text{and} \quad a_{0,n_2} - a_{,0} = 0.$$

$$y^{n_2} - 1 \in Val(a)$$

Recurrence relations

Definition

The polynomial C defines a **linear recurrence relation for the array** a if the equation

$$\sum_{\alpha \in Supp(C)} c_\alpha a_{\alpha+\beta} = 0 \quad \text{holds for all} \quad \beta \in \mathbb{N}_0^2,$$

where $\alpha \in \mathbb{N}_0^2$.

We say that C is **valid for the array** a,

$$C \in Val(a).$$

Linear complexity

- The polynomials that are valid in the array form a polynomial ideal $I = Val(a)$.

Linear complexity

- The polynomials that are valid in the array form a polynomial ideal $I = Val(a)$.
- To generate the array we might need more than one polynomial.

Linear complexity

- The polynomials that are valid in the array form a polynomial ideal $I = Val(a)$.
- To generate the array we might need more than one polynomial.
- A generating set for $I = Val(a)$ generates the array.

Linear complexity

- The polynomials that are valid in the array form a polynomial ideal $I = Val(a)$.
- To generate the array we might need more than one polynomial.
- A generating set for $I = Val(a)$ generates the array.
- The **linear complexity** measures the resistance to find a generating set for $I = Val(a)$.

Linear complexity (for sequences)

Definition

The **linear complexity of a periodic sequence** s is the degree of a minimal polynomial that generates the sequence.

Linear complexity (for sequences)

Definition

The **linear complexity of a periodic sequence** s is the degree of a minimal polynomial that generates the sequence.

How can we generalize this concept for arrays?

Linear complexity

- for sequences: degree of minimal generating polynomial g

Linear complexity

- for sequences: degree of minimal generating polynomial g
- for arrays: more than one generating polynomial $g_1, \ldots, g_l$

Linear complexity

- for sequences: degree of minimal generating polynomial g
- for arrays: more than one generating polynomial $g_1, \ldots, g_l$

- for sequences: number of monomials smaller than $LM(g)$

Linear complexity

- for sequences: degree of minimal generating polynomial g
- for arrays: more than one generating polynomial $g_1, \ldots, g_l$

- for sequences: number of monomials smaller than $LM(g)$
- for arrays: need a monomial ordering and deal with more polynomials

Linear complexity

- for sequences: degree of minimal generating polynomial g
- for arrays: more than one generating polynomial $g_1, \dots, g_l$

- for sequences: number of monomials smaller than $LM(g)$
- for arrays: need a monomial ordering and deal with more polynomials

- for sequences: number of monomials not divisible by $LM(g)$

Linear complexity

- for sequences: degree of minimal generating polynomial g
- for arrays: more than one generating polynomial $g_1, \ldots, g_l$

- for sequences: number of monomials smaller than $LM(g)$
- for arrays: need a monomial ordering and deal with more polynomials

- for sequences: number of monomials not divisible by $LM(g)$
- for arrays: number of monomials not divisible by $LM(g_i)$ of any of the generating polynomials g_i is not invariant for any generating set.

Linear complexity

- for sequences: degree of minimal generating polynomial g
- for arrays: more than one generating polynomial $g_1, \dots, g_l$

- for sequences: number of monomials smaller than $LM(g)$
- for arrays: need a monomial ordering and deal with more polynomials

- for sequences: number of monomials not divisible by $LM(g)$
- for arrays: number of monomials not divisible by $LM(g_i)$ of any of the generating polynomials g_i is not invariant for any generating set. We need a special type of generating set:

Linear complexity

- for sequences: degree of minimal generating polynomial g
- for arrays: more than one generating polynomial $g_1, \dots, g_l$

- for sequences: number of monomials smaller than $LM(g)$
- for arrays: need a monomial ordering and deal with more polynomials

- for sequences: number of monomials not divisible by $LM(g)$
- for arrays: number of monomials not divisible by $LM(g_i)$ of any of the generating polynomials g_i is not invariant for any generating set. We need a special type of generating set:

a Gröbner basis!

Gröbner bases

Definition

Let $G = \{g_1, \ldots, g_l\} \subset I$, I an ideal in $\mathbb{F}[\mathbf{x}]$. One says that G is a **Gröbner basis** for I with respect to $\leq_T$ if

$$\langle LM(g_1), \ldots, LM(g_l) \rangle = \langle LM\,(I) \rangle \,.$$

Gröbner bases

Definition

Let $G = \{g_1, \ldots, g_l\} \subset I$, I an ideal in $\mathbb{F}[\mathbf{x}]$. One says that G is a **Gröbner basis** for I with respect to $\leq_T$ if

$$\langle LM(g_1), \ldots, LM(g_l) \rangle = \langle LM\,(I) \rangle \,.$$

$$I = \langle x+1, x \rangle = \langle 1 \rangle = \mathbb{F}[x]$$

Gröbner bases

Definition

Let $G = \{g_1, \ldots, g_l\} \subset I$, I an ideal in $\mathbb{F}[\mathbf{x}]$. One says that G is a **Gröbner basis** for I with respect to $\leq_T$ if

$$\langle LM(g_1), \ldots, LM(g_l)\rangle = \langle LM(I)\rangle .$$

$$I = \langle x+1, x\rangle = \langle 1\rangle = \mathbb{F}[x]$$

$$\langle x\rangle = \langle LM(x+1), LM(x)\rangle \neq \langle LM(I)\rangle = \langle 1\rangle$$

Properties of Gröbner bases

- A Gröbner basis for an ideal generates the ideal.

Properties of Gröbner bases

- A Gröbner basis for an ideal generates the ideal.
- There are algorithms for computing Gröbner bases. (Most of them depend on having a basis to start from)

Properties of Gröbner bases

- A Gröbner basis for an ideal generates the ideal.
- There are algorithms for computing Gröbner bases. (Most of them depend on having a basis to start from)
- $G = \{g_1, \ldots, g_l\} \subset I$ is a Gröbner basis for I if and only if for any $f \in I$,

$$LM(g_i) | LM(f)$$

for some $g_i \in G$.

Lead monomials

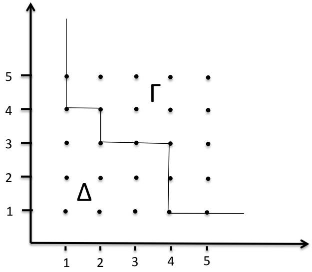


Figure: $\langle x^4y, x^2y^3, xy^4 \rangle$

Back to linear complexity

- **Complexity for sequences**

  degree of minimal generating polynomial g

  $=$ number of monomials not divisible by $LM(g)$

Back to linear complexity

- **Complexity for sequences**

  degree of minimal generating polynomial g

  $=$ number of monomials not divisible by $LM(g)$

- **Complexity for arrays**

  number of monomials not divisible by $LM(g_i)$ for $g_i \in GB$

  $=$ the size of the Delta set!!!

Delta sets

- The Delta set of an ideal is not unique.
- The size of a Delta set is invariant

$$|\Delta_I| = dim_{\mathbb{F}}\left(\mathbb{F}[x,y]/I\right)$$

Linear complexity of arrays

Definition

Let a be an m-dimensional periodic array and $Val(a)$ be the ideal of recurrence relations valid on the array. We define the m-**dimensional linear complexity** $\mathcal{L}$ of the array a as the size of the delta set of $Val(a)$,

$$\mathcal{L} = \left|\Delta_{Val(a)}\right|.$$

Linear complexity of arrays

Definition

Let a be an m-dimensional periodic array and $Val(a)$ be the ideal of recurrence relations valid on the array. We define the m-**dimensional linear complexity** $\mathcal{L}$ of the array a as the size of the delta set of $Val(a)$,

$$\mathcal{L} = \left|\Delta_{Val(a)}\right| .$$

- Invariant measure

Linear complexity of arrays

Definition

Let a be an m-dimensional periodic array and $Val(a)$ be the ideal of recurrence relations valid on the array. We define the m-**dimensional linear complexity** $\mathcal{L}$ of the array a as the size of the delta set of $Val(a)$,

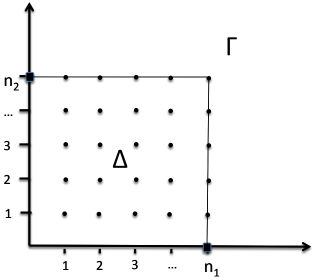
$$\mathcal{L} = \left|\Delta_{Val(a)}\right|.$$

- Invariant measure
- Generalization of measure for sequences

Delta sets and complexity of periodic arrays

$$Val(a) = \{ \text{ linear recurrence relations on a periodic array } a, \; n = (n_1, n_2)\}$$

$$x^{n_1} - 1 \in Val(a), \quad y^{n_2} - 1 \in Val(a)$$



Normalized linear complexity of arrays

Definition

Let a be a periodic array with period $(n_1, \ldots, n_m)$. The **normalized** m**-dimensional linear complexity** $\mathcal{L}_n$ of the array a is

$$\mathcal{L}_n = \frac{\mathcal{L}}{n_1 n_2 \cdots n_m}.$$

$$0 \leq \mathcal{L}_n \leq 1$$

2D Results

Proposition

Let $(a_{i,j})$ be an array constructed using the composition method by shifting columns from a sequence (c_j) cyclically, where the shifts are given by a sequence with period n_1. If $\mathcal{L}(c)$ is the linear complexity of the sequence (c_j) and $\mathcal{L}(a)$ is the linear complexity of the array $(a_{i,j})$, then

$$\mathcal{L}(a) \leq n_1 \mathcal{L}(c).$$

This bound is tight.

2D Results

Corollary

Let $(a_{i,j})$ be an array constructed using the composition method by shifting columns from a sequence (c_j) cyclically, where the shifts are given by a sequence with period n_1. If $\mathcal{L}(c)$ is the linear complexity of the sequence (c_j) and $\mathcal{L}(a)$ is the linear complexity of the array $(a_{i,j})$, then

$$\mathcal{L}_n(a) \leq \mathcal{L}_n(c).$$

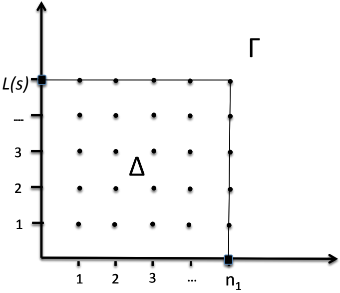
This bound is tight.

Delta set of composition method

$$g \in Val(c)$$

$$Val(a) = \{ \text{ linear recurrence relations on a periodic array } a, \;\; n = (n_1, n_2)\}$$

$$g \in Val(a), \quad y^{n_2} - 1 \in Val(a)$$



2D Results

Proposition

Let $(a_{i,j})$ *be an array constructed using the composition method by shifting columns from a sequence* (c_j) *cyclically, where the shifts are given by a sequence with period* n_1*. If the minimal polynomial of* (c_j)*,* $C(y)$*, is divisible by* $y-1$*,* $\mathcal{L}(c)$ *is the linear complexity of the sequence* (c_j) *and* $\mathcal{L}(a)$ *is the linear complexity of the array* $(a_{i,j})$*, then*

$$\mathcal{L}(a) \leq n_1 \left(\mathcal{L}(c) - 1\right) + 1.$$

This bound is tight.

2D Results

Proposition

Let $(a_{i,j})$ be an array constructed using the composition method by shifting columns from a sequence (c_j) cyclically, where the shifts are given by a sequence with period n_1. If the minimal polynomial of (c_j), $C(y)$, is divisible by $y-1$, $\mathcal{L}(c)$ is the linear complexity of the sequence (c_j) and $\mathcal{L}(a)$ is the linear complexity of the array $(a_{i,j})$, then

$$\mathcal{L}(a) \leq n_1\left(\mathcal{L}(c)-1\right)+1.$$

This bound is tight.

More accurate than the multisequence approach.

2D Results

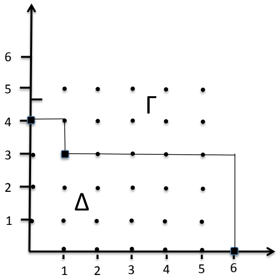
Corollary

Let $(a_{i,j})$ be an array constructed using the composition method by shifting columns from a sequence (c_j) cyclically, where the shifts are given by a sequence with period n_1. If the minimal polynomial of (c_j), $C(y)$, is divisible by $y-1$, $\mathcal{L}(c)$ is the linear complexity of the sequence (c_j) and $\mathcal{L}(a)$ is the linear complexity of the array $(a_{i,j})$, then

$$\mathcal{L}_n(a) \leq \mathcal{L}_n(c) - \frac{1}{n_2} + \frac{1}{n_1 n_2}.$$

This bound is tight.

Example - Delta set of composition method



$$\left|\Delta_{Val(a)}\right| = 19$$

2D Experimental asymptotic results

Sequences	Array Dim.		Column N. Comp	M-T N. Comp	Our N. Comp
Welch		$p \equiv 1, 7 \pmod 8$	.5		.5
Legendre	$p \times p - 1$	$p \equiv 3, 5 \pmod 8$	1		1
Quadratic		$p \equiv 1, 7 \pmod 8$	.5	.5	.5
Legendre	$p \times p - 1$	$p \equiv 3, 5 \pmod 8$	1	1	1

Array	Dim.	M-T N. Comp	Our N. Comp
Gen. Leg. Ternary	$p \times p$	-	1
Gen. Leg. Binary	$p \times p$	-	.5

Conjecture

Let $\mathcal{L}(s)$ be the complexity of a Legendre sequence for p. The normalized linear complexity $\mathcal{L}(a)$ of an array constructed with columns from Legendre and a shift sequence of period $n_1 = p - 1$ is

$$\mathcal{L}_n(a) = \begin{cases} \mathcal{L}_n(s) - \frac{n_1 - 1}{n_1 p} & p \equiv 3 \pmod 4 \\ \mathcal{L}_n(s) & p \equiv 1 \pmod 4 \end{cases}$$

3D Results

Proposition

Let $(a_{i,j,k})$ be a 3D array constructed using the composition method by defining the columns as cyclic shifts up of a sequence (c_j) with period $n_1^2 - 1$, where the shifts are given by a 2D square array with period n_1. If $\mathcal{L}(c)$ is the linear complexity of the sequence (c_j) and $\mathcal{L}(a)$ is the linear complexity of the array $(a_{i,j,k})$, then

$$\mathcal{L}_n(a) \leq \mathcal{L}_n(c).$$

This bound is tight.

3D Results

Proposition

Let $(a_{i,j,k})$ be a 3D array constructed using the composition method by defining the columns as cyclic shifts up of a sequence (c_j) with period $n_1^2 - 1$, where the shifts are given by a 2D square array with period n_1. If $\mathcal{L}(c)$ is the linear complexity of the sequence (c_j) and $\mathcal{L}(a)$ is the linear complexity of the array $(a_{i,j,k})$, then

$$\mathcal{L}_n(a) \leq \mathcal{L}_n(c).$$

This bound is tight.

The same is true for composition with "floors"!!

3D Experimental asymptotic results

Shift Array/ Floor Array	3D Array Dim.	Floor N. Comp	Our 3D N. Comp
3D Welch 2D Gen. Leg. Tern.	$p \times p \times p^2 - 1$	1	1
3D Welch 2D Gen. Leg. Bin.	$p \times p \times p^2 - 1$	.5	.5
3D Quadratic 2D Gen. Leg. Bin.	$p \times p \times p^2 - 1$	.5	.5

3D Experimental asymptotic results

Shift Array/ Floor Array	3D Array Dim.	Floor N. Comp	Our 3D N. Comp
3D Welch 2D Gen. Leg. Tern.	$p \times p \times p^2 - 1$	1	1
3D Welch 2D Gen. Leg. Bin.	$p \times p \times p^2 - 1$	.5	.5
3D Quadratic 2D Gen. Leg. Bin.	$p \times p \times p^2 - 1$	.5	.5

Complexity of 3D Welch with Sidelnikov columns $\longrightarrow$ Complexity of Sidelnikov.

3D Experimental asymptotic results

Shift Array/ Floor Array	3D Array Dim.	Floor N. Comp	Our 3D N. Comp
3D Welch 2D Gen. Leg. Tern.	$p \times p \times p^2 - 1$	1	1
3D Welch 2D Gen. Leg. Bin.	$p \times p \times p^2 - 1$	.5	.5
3D Quadratic 2D Gen. Leg. Bin.	$p \times p \times p^2 - 1$	.5	.5

Complexity of 3D Welch with Sidelnikov columns $\longrightarrow$ Complexity of Sidelnikov.

Complexity of 3D Quadratic with Sidelnikov columns $\longrightarrow$ Complexity of Sidelnikov.

Conjectures

- The normalized linear complextity of arrays constructed by composing a shift sequence/array with a column of length commesurable with the shifts approaches the normalized linear complexity of the column sequence.

Conjectures

- The normalized linear complextity of arrays constructed by composing a shift sequence/array with a column of length commesurable with the shifts approaches the normalized linear complexity of the column sequence.

- The normalized linear complextity of arrays constructed by composing a shift array with a "floor" arrays of dimmensions commesurable with the dimensions of the shift array approaches the normalized linear complexity of the "floor array".

Conjectures

- The normalized linear complextity of arrays constructed by composing a shift sequence/array with a column of length commesurable with the shifts approaches the normalized linear complexity of the column sequence.

- The normalized linear complextity of arrays constructed by composing a shift array with a "floor" arrays of dimmensions commesurable with the dimensions of the shift array approaches the normalized linear complexity of the "floor array".

$$\mathcal{L}_n(a) \longrightarrow \mathcal{L}_n(c)$$

Conjectures

- The normalized linear complextity of arrays constructed by composing a shift sequence/array with a column of length commesurable with the shifts approaches the normalized linear complexity of the column sequence.
- The normalized linear complextity of arrays constructed by composing a shift array with a "floor" arrays of dimmensions commesurable with the dimensions of the shift array approaches the normalized linear complexity of the "floor array".

$$\mathcal{L}_n(a) \longrightarrow \mathcal{L}_n(c)$$

Also have conjectures for exact formulas for the complexity of some 3D arrays.

In Progress...

- Study other sequences and arrays for composition method.

In Progress...

- Study other sequences and arrays for composition method.
- Find formulas for the complexity of arrays constructed with composition method.

In Progress...

- Study other sequences and arrays for composition method.
- Find formulas for the complexity of arrays constructed with composition method.
- Study many other questions regarding multidimensional constructions!

Coming Soon!!

WEB APPLICATION FOR COMPUTING LINEAR COMPLEXITY OF MD ARRAYS

THANKS !!!

- Daniel, David and Steve for the invitation.

THANKS !!!

- Daniel, David and Steve for the invitation.
- My students for their results, help and motivation to work harder.

THANKS !!!

- Daniel, David and Steve for the invitation.
- My students for their results, help and motivation to work harder.
- My collaborators Rafa Arce and Cheo Ortiz for sharing their knowledge and expertise on array properties and applications.

THANKS !!!

- Daniel, David and Steve for the invitation.
- My students for their results, help and motivation to work harder.
- My collaborators Rafa Arce and Cheo Ortiz for sharing their knowledge and expertise on array properties and applications.
- The DEGI of the UPR-RP for the FIPI Grant funding.

THANKS !!!

- Daniel, David and Steve for the invitation.
- My students for their results, help and motivation to work harder.
- My collaborators Rafa Arce and Cheo Ortiz for sharing their knowledge and expertise on array properties and applications.
- The DEGI of the UPR-RP for the FIPI Grant funding.
- All of you!