Linear Complexity for Multidimensional Arrays - a Numerical Invariant

Domingo Gomez-Perez
Department of Mathematics, Statistics and Computer Science,
Universidad de Cantabria, Santander, Spain
Email: domingo.gomez@unican.es

Tom Høholdt
Department of Applied Mathematics and Computer Science,
Technical University of Denmark, Capital Region, Denmark
Email: tomh@dtu.dk

Oscar Moreno
Gauss Research Foundation,
San Juan, Puerto Rico
Email:moreno@nic.pr

Ivelisse Rubio
Department of Computer Science,
University of Puerto Rico,Río Piedras, Puerto Rico
Email: iverubio@gmail.com

***Abstract*—Linear complexity is a measure of how complex a one dimensional sequence can be. In this paper we extend the concept of linear complexity to multiple dimensions and present a definition that is invariant under well-orderings of the arrays. As a result we find that our new definition for the process introduced in the patent titled "Digital Watermarking" produces arrays with good asymptotic properties.**

***Keywords*– linear complexity, invariance, Groebner bases, correlation, arrays, watermarking**

I. Introduction

The media and entertainment industry is asking for watermarks for synchronized video-audio applications and there have been proposed constructions by Moreno and Tirkel, see [1], [2]. Therefore from an applied point of view our abstract is talking of how we can provide a theory of how secure are the Moreno-Tirkel arrays (or any other multidimensional array). In other words, we are building the foundation for a theory of security for multidimensional watermarks. This is extremely important for the media and entertainment industry. The Moreno-Tirkel 3D arrays are constructed using the method of composition. A family of doubly periodic shift sequences is constructed with constrained auto and cross-correlation. In these shift sequences the array length and width are relatively prime, so they can be converted into equivalent one dimensional sequences sequences using the Chinese Remainder Theorem. But, applying the theory of Groebner bases, other dimensions can be used as well. Such sequences have linear complexities between L/2 and L, where L is the sequence length. High linear complexity is desirable for video watermarks because there are always parts of a video which have low contrast within a frame or have regions which change slowly with time. This exposes sections of a watermark to an attacker, who may try to reverse engineer the whole watermark from such sections. Therefore it is very important to examine our results in linear complexity and their implications, and to use watermark sequences with as high complexity as posible. In this paper we are introducing a method to obtain the complexity of sequences for multimedia applications. The paper is organized as follows: Section II describes the construction of two dimensional logarithmic shift sequences and how they are used to shift a Sidelnikov column sequence to produce a 3D watermark array. Section III explains how an exponential shift sequence can be employed to shift a 2D Legendre array to produce a 3D array. Section IV discusses linear complexity issues for multidimensional watermark arrays, and Section V summarizes the results.

A. Notation

From now and on, p denotes a prime and $q = p^r$ is a prime power. We denote $\mathbf{F}_q$ the finite field of q elements. For $q = p$, we identify the elements of the finite field by integer numbers in the range $\{0, \ldots, p-1\}$ and vice versa. For $\mathbf{F}_q$, we consider the elements $\mathbb{F}_q = \{\xi_0, \xi_1, \ldots, \xi_{q-1}\}$ using an ordered basis $\{\gamma_1, \ldots, \gamma_r\}$ of $\mathbb{F}_q$ over $\mathbb{F}_p$ for $0 \leq n < q$,

$$\xi_n = n_1\gamma_1 + n_2\gamma_2 + \cdots + n_r\gamma_r,$$

if

$$n = n_1 + n_2 p + \cdots + n_r p^{r-1},\ 0 \leq n_i < p, \quad i = 1, \ldots, r.$$

Using this order, it is easy to relate the elements of a finite field with an $r-$dimensional array. The multiplicative group of any finite field is cyclic, so for any generator (also called a primitive element) $\alpha \in \mathbf{F}_q$ and $0 \neq \beta \in \mathbf{F}_q$, we let $s = \log_\alpha \beta$ be the unique integer $0 \leq s \leq q-1$ such that $\alpha^s = \beta$.

II. Log Map in Multi-dimensional grids

The process of constructing 3D arrays is illustrated in Figure 1. The top left table 1(a) shows the powers of a primitive element in $\mathbf{F}_9$ raised to all its powers in the grid format. The vertical arrow shows a logarithmic mapping of the original grid table. This resulting shift array can be used to shift a column sequence of length 8, a Sidelnikov sequence, directly below the shift array. Sidelnikov sequences are chosen for their good qualities, see [3] for definition and a study of its linear complexity.

The shift array and the column sequence are used to generate a three dimensional array, shown in solid form below. The array is generated by first placing a column of length 8 below every entry in the grid. The column contains all zeros if

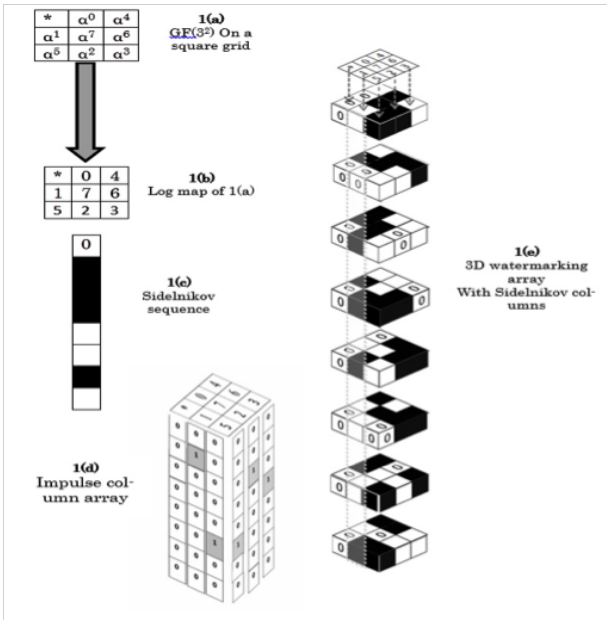

Fig. 1. 3 dimensional Moreno-Tirkel array using logarithmic maps

the entry is not defined and otherwise contains a solitary entry of one (i.e. an impulse column) in the position determined by the entry in the corresponding grid location. This is illustrated in Figure 1(d). The right side of the figure, 1(e), shows the impulse column being replaced by a commensurate Sidelnikov sequence. This produces a 3D watermarking array over +1,-1, and 0. The array in Figure 1 is solitary, so by itself it does not address the requirement of delivering large families of arrays with low off-peak autocorrelation and low cross-correlation. There are several ways of modifying and extending the method used to construct two dimensional arrays, to produce families of three dimensional arrays. The resulting three dimensional arrays differ in their size and shape, the family size and correlation values.

Take, $A, B, C \in \mathbf{F}_{p^2}$, with $A \neq 0$ and define the family of sequences,

$$s_i^{A,B,C} = \log_\alpha(A\xi_i^2 + B\xi_i + C),\ i = 0, \ldots, p^2 - 1. \quad (1)$$

In this family, two different shift sequences $s^{A,B,C}$ and $s^{A',B',C'}$ are equivalent if the arrays they generate are multi-periodical shifts of each other. The number of non-equivalent classes is approximately p^2. A watermarking array is constructed by using s_i belonging to each coordinate on the grid to cyclically shift a binary Sidelnikov sequence of length $p^2 - 1$. Our construction using the quadratic shift as in (1), guarantees that no more than 2 such Sidelnikov columns can match and therefore the worst case autocorrelation and cross-correlation is of the order of $2p^2$.

III. Exponential map on a grid

The $\log_\alpha$ function defines a bijection between the sets $\{1, \ldots, p^r - 1\}$ and $\{0, \ldots, p^r - 2\}$. The digits in base p, as in coding theory, represents a vector with r coordinates.

First, consider a shift sequence $s_i = \alpha^i$ which generates a solitary array. An example of such a construction over $\mathbf{F}_9$

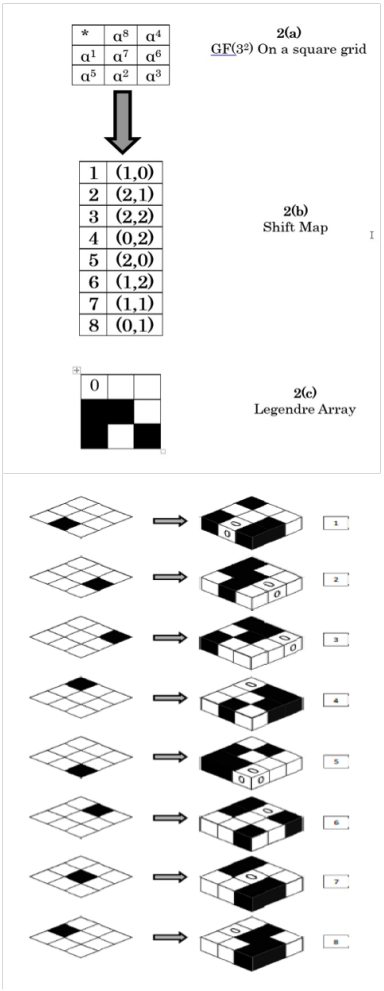

Fig. 2. 3D Moreno-Tirkel array generation using an exponential map

is shown in Figure 2. The grid coordinates are used to shift a 3×3 Legendre Array described in [4]. Figure 2(a) shows the elements of $\mathbf{F}_9$ on the two dimensional grid. Figure 2(b) shows the grid coordinates for ascending powers of α, which is to be used as a two dimensional shift sequence. Figure 2(c) shows a $3\times$ 3 Legendre array, which is commensurate with the shift sequence. Figure 2(d) depicts the complete three Dimensional array constructed plane by plane.

IV. Linear Complexity

In order to be useful in security applications, such as digital watermarking and cryptography, the arrays need to be robust to attack. An attacker can have access to parts of the array and attempt to reconstruct the rest of the array by linear algorithms such as the Berlekamp-Massey. For one dimensional sequences, a measure of linear complexity is simply the degree of the minimum recursion polynomial which can be used to construct the sequence, see [5]. Linear complexity was extended to cover multisequences and a survey of work related to the linear complexity of multisequences was presented in [6], but the generalization to the multidimensional setting is what we are doing here. The purpose of this paper is to introduce a general and invariant measure of the linear complexity of arrays in any number of dimensions.

The set of all polynomials that generate the sequence form a polynomial ideal. The minimum recursion polynomial generating the sequence is a generator for the ideal and its degree is also the number of monomials that are not lead monomials of any element of the ideal. The polynomials in the ideal are "valid" for the sequence (or one dimensional array) as we will describe below.

The natural generalization of the concept of linear complexity for multidimensional arrays is the number of monomials that are not lead monomials of any element of the ideal of "valid polynomials" for the array. However, in the multidimensional case this ideal is generally generated by more than one multivariate polynomial and we need a theory that guarantees invariance in the number of monomials that are "excluded" as leading monomials of the polynomials in the ideal. The theory of Groebner bases provides this invariance.

Sakata in [7] studied the relation between periodic arrays, linear recursion relations and ideals. We now review the necessary concepts to present a definition for the linear complexity of a multidimensional array. Let $\mathbf{Z}_0$ denote the set of nonnegative integers and let $\Sigma_0 = \mathbf{Z}_0^r$. An $r-$dimensional array over a field $\mathbf{F}_q$ is a mapping $s : \Gamma \mapsto \mathbf{F}_q$, $\Gamma \subset \Sigma_0$ and it is written as $s = (s_\alpha)$, where $s_\alpha = s(\alpha),\ \alpha \in \Gamma$, is the "value" of s at α. We consider a monomial ordering $<_\tau$ on $\mathbf{F}_q[X_1, \ldots, X_r]$, the r-variate polynomial ring over $\mathbf{F}_q$. We will look at linear recursion relations which are satisfied by the given array. This can be represented by r-variate polynomials. Any such polynomials can be written as: $f = \sum_{\alpha \in \Sigma_0} f_\alpha X^\alpha$, or using the concept of support, $f = \sum_{\alpha \in supp(f)} f_\alpha X^\alpha$, where $supp(f) = \{\alpha \in \Sigma_0 \mid f_\alpha \neq 0\}$.

The maximum element with respect to $<_\tau$ is called the **lead exponent** and is denoted by $le(f)$. The monomial $X^{le(f)}$ is the **lead monomial** of f. The polynomial f gives a linear recursion at s_k if $le(f) \leq k$, where $\leq$ denotes the partial order defined by $\leq$ in all the components in Σ_0, and

$$\sum_{\alpha \in supp(f)} f_\alpha s_{\alpha+k-le(f)} = 0. \tag{2}$$

If a polynomial f satisfies $\sum_{\alpha \in supp(f)} f_\alpha s_{\alpha+k-le(f)} = 0$ for all $k \in \Sigma_0$ with $le(f) \leq k$, then we say that the polynomial is *valid* for the array s. The set of all polynomials f that are valid for a fixed array s are denoted by $VALPOL(s)$ and is an ideal of $\mathbf{F}_q[X_1, \ldots, X_r]$.

A Groebner basis for an ideal I is a set of polynomials $G = \{g_1, \ldots, g_l\}$ such that if $f \in I$, then there exists $g_i \in G$ such that $le(g_i) \leq le(f)$. We can define the set $le(I) = \{le(f) \mid f \in I\}$, and the complement of this set is called **the delta set, the excluded point set or the footprint**. Hence, the delta set corresponds to all the monomials which are not lead monomials of the elements in the ideal I. The delta set depends on the chosen monomial order $<_\tau$ but the size of the delta set does not change with the order chose, hence it is invariant.

If s is a multidimensional periodic array with period vector $(T_1, \ldots, T_r)$, then $VALPOL(s)$ contains the polynomials $X_1^{T_1} - 1, \ldots, X_r^{T_r} - 1$ and therefore the delta set of $VALPOL(s)$, $\Delta(s)$, is finite. We are now ready to define the **linear complexity of a multidimensional array** s as the number of elements in $\Delta(s)$.

There are several algorithms to compute Groebner bases for the ideal of linear recursion relations $VALPOL(s)$ and hence $\Delta(s)$. For example, Sakata describes in [8] an algorithm for this calculation. The algorithm is an extension of an earlier 2D version [7] which has been used for decoding algebraic geometry codes, see [9], [10]. The thesis of Rubio [11] also contains an algorithm that computes a Groebner bases for $VALPOL(s)$ by reducing a matrix whose rows consist of mappings of the array s. Rubio's algorithm is an analog of an algorithm, due to Moss Sweedler and Lee Taylor, for the computation of Groebner bases for 0-dimensional ideals without necessarily knowing a basis for the ideal.

For completeness, we outlay here how to calculate the linear complexity of a multi-dimensional arrays.

1) Choose a monomial ordering $<_\tau$.
2) Compute a Groebner basis with respect to $<_\tau$ for $VALPOL(s)$, the ideal of linear recursion relations in the array s, using Sakata's algorithm or Rubio's analog to the Sweedler-Taylor algorithm.
3) The set of exponents of monomials that are not multiples of the lead monomials of the elements in the Groebner basis form the delta set $\Delta(s)$.
4) The linear complexity of the array s is the number of elements in the set $\Delta(s)$.

Remark: The delta set, $\Delta(s)$, depends on the monomial ordering that has been chosen. However, the size of $\Delta(s)$ (which gives the linear complexity of the array) depends only in the ideal and not on the set of generators and hence is a numeric invariant, see [12].

A. Known construction

Arrays of dimension 2 have been analyzed directly with respect to their linear complexity using the Groebner bases method, and we have done this in cases where the array dimensions are relatively prime. This was achieved using the Chinese Remainder Theorem (CRT) as described in [1]. CRT converts two dimensional arrays into one dimensional sequences. Then, the one dimensional sequences can be analyzed with regard to their linear complexity, using the Berlekamp-Massey algorithm. The delta set for the sequence contains the monomials of degree less than the degree of the recursion polynomial obtained by the Berlekamp-Massey algorithm. Note that since the Groebner bases method leaves invariant the number of elements in the delta set, for any ordering of the monomials this establishes that cases of dimensions 1 and 2 give the same linear complexity when the CRT method can be applied.

A reverse process to CRT, called folding can be applied to convert one dimensional sequences into two dimensional arrays. Sequence families constructed using Bent, Small Kasami, No-Kumar, Generalized No-Kumar, Gold, and Kerdock can be folded into two dimensional arrays. These have known linear complexities. The Moreno-Tirkel arrays are two dimensional constructions of the type described previously (3D examples given above) except that their dimensions are relatively prime. Their construction is explained below. Their linear complexity is obtained by using the CRT to convert them from two dimensions to one, apply the Berlekamp-Massey algorithm; the complexity is the degree of the recursion polynomial obtained, which is the size of the delta set of the sequence.

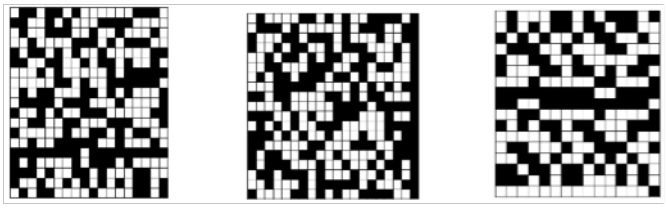

Fig. 3. Examples of Moreno-Tirkel arrays

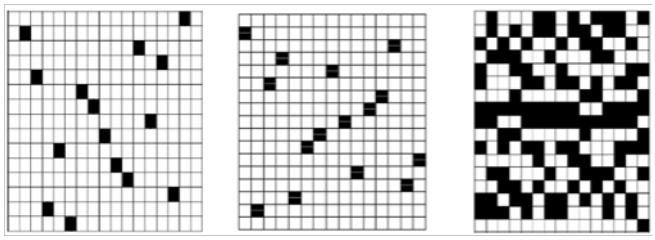

Fig. 4. Moreno-Tirkel construction with family C

B. Moreno-Tirkel arrays

Family A is obtained by applying an exponential quadratic shift sequence to commensurate columns as described before. Families B and C are obtained by applying a similar process to a rational function cycle through a finite field, see [1] for more details. Figure 3 shows the array form (19×18) of a sequence obtained from Family A, an array (19×20) from Family B, and a (15×17) array from Family C, where the rows have been substituted by binary Legendre sequences of length 17. These are the closest values to compare with the array format (15×17) of the previously known constructions. Note that there are no apparent symmetries. The array of Family C is obtained by transposing the original shift array before substitution. This can be done because the original array has at most one dot per column and exactly one dot per row. The process is shown in Figure 4.

V. Results

Using the methods just described, the linear complexity for the known CDMA sequences and the three families of Moreno-Tirkel constructions with Legendre sequence columns have been analyzed and the results are presented in Table I. The comparison has been restricted to sizes for which all constructions exist, so that a quantitative comparison between the constructions can be made. We normalize the linear complexity dividing by the length of the sequence, which we denote by L. This way, we are making a fair comparison between sequences of different lengths.

Note, that the Moreno-Tirkel arrays are available in many more sizes than the classical counterparts. Also, note that some of the arrays can be folded into higher dimensions e.g. 255 can be folded into 15×17 or $3 \times 5 \times 17$, 1023 can be folded into 31×33 or $31 \times 11 \times 3$, and 4095 can be folded into 63×65 or $7 \times 9 \times 65$ or $7 \times 9 \times 5 \times 13$ or various other two, three, or four dimensional arrays. It is evident that the Moreno-Tirkel families are the only ones to have high linear complexity, and that this complexity remains high as the array size tends to infinity. We remark that the linear complexity in one-dimensional and multi-dimensional arrays

Family/Length	255	1023	4095	Assymptote
Bent [14]	0.125	-	0.018	≈ 0
Small Kasami [15]	0.047	0.015	0.004	≈ 0
No-Kumar [16]	0.23	0.15	0.03	≈ 0
Generalized N-K	-	0.13	-	≈ 0
Gold [17]	0.06	0.02	0.002	≈ 0
Kerdock [18]	0.15	0.05	0.01	≈ 0
Moreno-Tirkel (A)	0.5 $L = 342$	0.5 $L = 930$	0.98 $L = 4922$	[1]
Moreno-Tirkel (B)	0.98 $L = 380$	0.46 $L = 992$	0.97 $L = 4556$	[1]
Moreno-Tirkel (C)	0.47 $L = 255$	-	-	[1]

TABLE I. Linear complexity calculations

can be computed in polynomial time in the size of the delta set, so sequences with small delta set can be predicted. Therefore, the Moreno-Tirkel arrays are inherently more secure when used as arrays for watermarking or as sequences for wireless communications. Observation: For Family A and C of the Moreno-Tirkel construction, the recursion polynomial of the long sequence is obtained from that of the column sequence by raising each term to the power equal to the number of columns in the array. Example: Consider the array of Figure 4(c). The column is a binary Legendre sequence of length 17:

$$0, 1, 1, 0, 1, 0, 0, 0, 1, 1, 0, 0, 0, 1, 0, 1, 1.$$

Column recursion polynomial:

$$x^8 + x^7 + x^6 + x^4 + x^2 + x + 1.$$

Recursion polynomial for the long sequence:

$$x^{120} + x^{105} + x^{90} + x^{60} + x^{30} + x^{15} + 1.$$

The number of columns in the array is 15.

Result 1: Our new definition for the process introduced in the patent [13].

Result 2: We extend the concept of linear complexity to multiple dimensions by presenting a definition that is invariant under well-orderings of the arrays, including the Chinese remainder process, as conjectured in [1]. [1]

Acknowledgments

The authors want to thank Andrew Tirkel for his support, ideas and participation in many useful discussions. The paper would have not been possible without him. The authors thankfully acknowledge the computer resources, technical expertise and assistance provided by the the Santander Supercomputing services at the University of Cantabria.

References

[1] O. Moreno and A. Tirkel, "New optimal low correlation sequences for wireless communications," in *Sequences and Their Applications–SETA 2012*. Springer, 2012, pp. 212–223.

[2] S. Blake, O. Moreno, and A. Z. Tirkel, "Families of 3d arrays for video watermarking," in *Sequences and Their Applications-SETA 2014*. Springer, 2014, pp. 134–145.

[3] M. Z. Garaev, F. Luca, I. E. Shparlinski, and A. Winterhof, "On the lower bound of the linear complexity over f˜ p of sidelnikov sequences," *IEEE Transactions on Information Theory*, vol. 52, no. 7, p. 3299, 2006.

[1] The conjectured normalized linear complexity follows this pattern, 0.5 if $p \equiv 3, 5 \bmod 8$ and 1.0 if $p \equiv 1, 7 \bmod 8$ for p sufficiently big.

[4] O. Moreno and A. Tirkel, "Multi-dimensional arrays for watermarking," in *Information Theory Proceedings (ISIT), 2011 IEEE International Symposium on*. IEEE, 2011, pp. 2691–2695.

[5] H. Niederreiter and R. Lidl, *Finite fields*. Addison-Wesley Reading, Mass., 1983, vol. 102.

[6] H. Niederreiter, "Linear complexity and related complexity measures for sequences," in *Progress in Cryptology-INDOCRYPT 2003*. Springer, 2003, pp. 1–17.

[7] S. Sakata, "Finding a minimal set of linear recurring relations capable of generating a given finite two-dimensional array," *Journal of Symbolic Computation*, vol. 5, no. 3, pp. 321–337, 1988.

[8] ——, "Extension of the berlekamp-massey algorithm to¡ i¿ n¡/i¿ dimensions," *Information and Computation*, vol. 84, no. 2, pp. 207–239, 1990.

[9] S. Sakata, H. E. Jensen, and T. Høholdt, "Generalized berlekamp-massey decoding of algebraic geometry codes up to half the feng-rao bound," in *Information Theory, 1994. Proceedings., 1994 IEEE International Symposium on*. IEEE, 1994, p. 153.

[10] T. Høholdt and R. Pellikaan, "On the decoding of algebraic-geometric codes," *IEEE Transactions on Information Theory*, vol. 41, no. 6, pp. 1589–1614, 1995.

[11] I. Rubio, "Groebner bases for 0-dimensional ideals and applications to decoding," Ph.D. dissertation, Cornell University, 1998.

[12] D. A. Cox, J. Little, and D. OSHEA, *Ideals, varieties, and algorithms: an introduction to computational algebraic geometry and commutative algebra*. Springer Science & Business Media, 2007.

[13] O. Moreno and A. Tirkel, "Digital watermarking," Australian Patent PCT/AU210/000 990, 2010.

[14] J. Olsen, R. A. Scholtz, and L. Welch, "Bent-function sequences," *Information Theory, IEEE Transactions on*, vol. 28, no. 6, pp. 858–864, 1982.

[15] T. Kasami, "Weight distribution formula for some class of cyclic codes," DTIC Document, Tech. Rep., 1966.

[16] J.-S. No and P. V. Kumar, "A new family of binary pseudorandom sequences having optimal periodic correlation properties and larger linear span," *Information Theory, IEEE Transactions on*, vol. 35, no. 2, pp. 371–379, 1989.

[17] R. Gold, "Optimal binary sequences for spread spectrum multiplexing (corresp.)," *Information Theory, IEEE Transactions on*, vol. 13, no. 4, pp. 619–621, 1967.

[18] A. A. Nechaev, "Kerdock code in a cyclic form," *Discrete Mathematics and Applications*, vol. 1, no. 4, pp. 365–384, 1991.