Samples of Unit-Memory Codes Over $\mathbf{Z}_4$

A. Ashikhmin and V. V. Zablov
Institute for Problems of Information Transmission
Russian Academy of Sciences
Ermolova 19, Moscow GSP-4, 101447, Russia

Abstract — **The samples of unit-memory codes over $\mathbf{Z}_4$ are proposed. They show that unit-memory codes over $\mathbf{Z}_4$ can have better performances than unit-memory codes over $GF(2)$.**

I. Introduction

It is known that unit-memory convolutional codes can have a better error correcting capability than usual multimemory convolutional codes with the same rate and number of elements [1],[2]. Recently, it was shown that block $\mathbf{Z}_4$-linear codes may have better performances than block linear codes over $GF(2)$ [3],[4]. For example, the Kerdock, Preparata, and Delsarte-Goethals codes are $\mathbf{Z}_4$-linear ones. In this paper we obtain two unit-memory codes over $\mathbf{Z}_4$ that have larger free distance then the best known before unit-memory codes over $GF(2)$ of the same length and rate.

II. $\mathbf{Z}_4$-Linear Unit-Memory Codes Over $\mathbf{Z}_4$

A time-invariant $\mathbf{Z}_4$-linear $[n; k_1, k_2]$ unit memory code, with input $\mathbf{x}^{(t)} = (x_1, x_2, \ldots, x_{k_1}, x_{k_1+1}, \ldots, x_{k_1+k_2})$ $x_i \in \{0,1,2,3\}, i = \overline{1, k_1}$, $x_j \in \{0,2\}, j = \overline{k_1+1, k_1+k_2}$, and output $\mathbf{y}^{(t)} = (y_1, y_2, \ldots, y_n)$, $y_i \in \{0,1,2,3\}, i = \overline{1,n}$, at discrete time t, is represented by the encoding rule

$$\mathbf{y}^{(t)} = \mathbf{x}^{(t)} G_0 + \mathbf{x}^{(t-1)} G_1 . \qquad (1)$$

The matrixes G_0, G_1 have the form

$$\begin{pmatrix} A_1 \\ 2A_2 \end{pmatrix},$$

where A_1, A_2 are $(k_1 \times n)$ and $(k_2 \times n)$ $\mathbf{Z}_4$-matrices. All operations in (1) are fulfilled over $\mathbf{Z}_4$. A codeword of length r is defined as a sequence of r output bytes, or a path in the trellis starting in the zero state and returning to the zero state for the first time after r input bytes. The Lee weights of $0,1,2,3 \in \mathbf{Z}_4$ are $0,1,2,1$ respectively, and the Lee weight of $\mathbf{a} \in \mathbf{Z}_4^n$ is the rational sum of the Lee weights of its components. Let us denote by d_r^L the minimum Lee weight of a codeword of length r. Then free distance of $\mathbf{Z}_4$-linear unit-memory code in the Lee metric is calculated as follows

$$d_{free}^L = \min_{r \geq 2} \{d_r^L\}.$$

Let C be a $[n; k_1, k_2]$-$\mathbf{Z}_4$-linear unit-memory code. Denote by ϕ a Gray map from $\mathbf{Z}_4$ to $\mathbf{Z}_2^2$, i.e. $(0,0) = \phi(0), (0,1) = \phi(1), (1,1) = \phi(2), (1,0) = \phi(3)$. We will say that C^* is the binary image of C, if $C^* = \phi(C)$. The code C^* is in general a nonlinear trellis code of length $2n$ and cardinality $4^{k_1} 2^{k_2}$. It is follows from [3],[4] that $d_{free}(C^*) = d_{free}^L(C)$.

III. Local Search Algorithm

The unit-memory codes are defined from a larger number of elements than corresponding multimemory codes; and even for a small number of memory elements, finding these codes by exhaustive search may be practically impossible. To reduce the search efforts we use modification of local search algorithm [5].

Denote by $G = [\mathbf{h}_1, \ldots, \mathbf{h}_{2n}]$ a generator matrix of a $\mathbf{Z}_4$-linear block code of length $2n$ and type $4^{k_1} 2^{k_2}$ [3] with minimum distance in Lee metric d_{min}^L. Here $\mathbf{h}_i = (h_1^{(i)}, h_i^{(i)}, \ldots, h_{k_1+k_2}^{(i)})^\top, i = \overline{1, 2n}$ are columns of length $k_1 + k_2, h_j \in \{0,1,2,3\}, j = \overline{1, k_1}$ and $h_l \in \{0,2\}, l = \overline{k_1+1, k_1+k_2}$. Let $g_i^{(0)} = h_i, i = \overline{1,n}$, and $g_j^{(1)} = h_j, j = \overline{n+1, 2n}$, where $g_i^{(0)}$ and $g_j^{(1)}$ are columns of matrixes G_0 and G_1 respectively (G_0, G_1 are generator matrixes of unit-memory code that we want to construct). Interchanging columns in matrix G and multiplying them by units of $\mathbf{Z}_4$, we will obtain different linear unit-memory $[n; k_1, k_2]$-$\mathbf{Z}_4$-codes with $d_{free}^L \leq d_{min}^L$. Assume that we have a some matrix G. Then we can employ the Viterbi algorithm to calculate the truncated distance profile in Lee metric, $\{d_2^L(G), d_3^L(G), \ldots, d_s^L(G)\}$, of a corresponding unit-memory code. It is also possible to find the number of codewords of length r with Lee weight equal to $d_r^L(G)$. These numbers define the sequence $\{\nu_2^L(G), \nu_3^L(G), \ldots \nu_s^L(G)\}$. In order to find the best possible solution, a "reference" distance profile defined by $\{p_2^L, p_3^L, \ldots p_s^L\}$, with expected optimum values of $d_r^L(G)$ is also needed. Let us introduce a function F as follows

$$F(x,y) = \begin{cases} |y+1-x|^\beta & \text{if } x \leq y \\ 0 & \text{otherwise.} \end{cases}$$

In our findings we used the exponent β equal to 16. Let us introduce a objective function as follows

$$\xi(G) = \sum_{r=2}^{s} w_r \nu_r(G) F(d_r^L(G), p_r^L).$$

The parameters w_r in the objection function are used as penalty "weights" for each component of the distance profile with the purpose to enforce enforcing the maximization of the free distance and, secondly, to maximize the distance profile growth rate.

Denote a desire minimum free distance in Lee metric by D_{free}^L. The reference distance profile sequence $\{p_r\}$ can be initially chosen as $p_r = D_{free}^L + (r-2)\alpha$, where $\alpha = nH^{-1}(1 - (2k_1 + k_2)/(2n))$, H^{-1} is the inverse binary entropy function. The set of weights $\{w_r\}$ form a decreasing sequence. In our findings we used $w_r = -4^r$.

Let us introduce a permutation function $\mathrm{P}_G(i, j, z_1, z_2), i, j = \overline{1, 2n}$, $z_1, z_2 \in \mathbf{Z}_4$ in the following way

$$\mathrm{P}_G(i,j,z_1,z_2) = \begin{cases} t_i = z_2 t_j, \ t_j = z_1 t_i & \text{if } i \neq j \\ t_i = z_1 t_i & \text{if } i = j. \end{cases}$$

The following algorithm allows us to find "good" $\mathbf{Z}_4$-linear unit-memory codes.

Local Search Algorithm

1) take a $\mathbf{Z}_4$-linear block code of length n and type $4^{k_1}2^{k_2}$ with minimum distance $d^L_{min} \geq D^L_{free}$. Denote its generator matrix by G;

2) chose an initial order of columns of matrix G. A random selection can provide fairly good initial solutions;

3) for each i, $i = \overline{1, 2n}$, find the numbers j, z_1, z_2, such that permutation $\mathrm{P}(i, j, z_1, z_2)$ minimizes the objection function $\xi(G)$;

4) if the objection function decreases, then fulfil $\mathrm{P}(i, j, z_1, z_2)$ and go to step 3, otherwise go to step 5;

5) if a noncatastrophic code with $d^L_{free} \geq D^L_{free}$ is found then stop, else go to step 2 to restart the search from another initial solution.

IV. $[4; 3, 0]$-$\mathbf{Z}_4$-linear unit-memory code with $d^L_{free} = 6$.

It is known [3]-[4] that the Nordstrom-Robinson code can be constructed as $\mathbf{Z}_4$-linear code of length 8 and type 4^4 with $d^L_{min} = 6$. The generator matrix of Nordstrom-Robinson code can be written as follows [4]

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 2 & 3 & 1 \\ 0 & 0 & 1 & 0 & 3 & 3 & 3 & 2 \\ 0 & 0 & 0 & 1 & 2 & 3 & 1 & 1 \end{pmatrix}.$$

Let us construct generator matrix G of a block $\mathbf{Z}_4$-linear code of length 8 and type 4^3 with $d^L_{min} = 6$ by the following way: 1) eliminate an all-one row from generator matrix of Nordstrom-Robinson code. As result we will obtain a matrix containing an all-zero column; 2) replace the all-zero column by column $(1\ 3\ 0)^{\mathsf{T}}$. Then employing local search algorithm, we obtain $\mathbf{Z}_4$-linear unit-memory $[4; 3, 0]$-code with $d^L_{free} = 6$. The generator matrixes, G_0 and G_1 of this code have the form

$$G_0 = \begin{pmatrix} 1 & 3 & 3 & 1 \\ 0 & 2 & 1 & 3 \\ 0 & 3 & 2 & 0 \end{pmatrix},$$

$$G_1 = \begin{pmatrix} 3 & 2 & 0 & 0 \\ 3 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix}.$$

Using the Gray map, we obtain a nonlinear binary trellis code of length 8, cardinality $M = 2^6$ with $d_{free} = 6$. It is not difficult to verify that this code has the growth of distance profile equal to $\alpha = 0.5$. Note that the best known binary convolutional unit-memory $[8, 6]$-code has $d_{free} = 5$.

Note that since constructed code is $\mathbf{Z}_4$-linear it can be decoded with syndrome trellis [6]. For this code syndrome trellis has not greater than 256 states and the number of brunches coming to a node does not exceed 4.

V. Binary nonlinear trellis code of length 8, cardinality $M = 128$ and with $d_{free} = 4$.

Let matrixes G_0, G_1 have the following form

$$G_0 = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 3 & 0 & 0 & 1 \\ 3 & 0 & 1 & 2 \\ 1 & 1 & 0 & 1 \end{pmatrix},$$

$$G_1 = \begin{pmatrix} 1 & 1 & 3 & 1 \\ 0 & 2 & 3 & 1 \\ 0 & 3 & 0 & 3 \\ 0 & 3 & 0 & 2 \end{pmatrix}.$$

We will say that output word $\mathbf{y}^{(t)}$ at discrete time t is encoded by the rule

$$\mathbf{y}^{(t)} = \mathbf{x}^{(t)}G_0 + \mathbf{x}^{(t-1)}G_1. \qquad (2)$$

Here $\mathbf{x}^{(t)} = (x_1, x_2, x_3, x_4)$, where $x_1 \in \{0, 1\}$ and $x_2, x_3, x_4 \in \mathbf{Z}_4$. Note that since $x_1 \notin \{2, 3\}$ vectors $\mathbf{y}$ don't form a linear unit-memory code over $\mathbf{Z}_4$. So, encoding rule (2) defines the nonlinear trellis unit-memory code of over $\mathbf{Z}_4$ of length 4 and cardinality $M = 128$ with $d^L_{free} = 4$. Using the Gray map, we obtain a nonlinear binary trellis code of length 8 and cardinality $M = 128$ with $d_{free} = 4$. It is not difficult to verify that this code has the growth of distance profile equal to $\alpha = 2/9$ and therefore is noncatastrophic. Note that up to now binary noncatastrophic (linear or nonlinear) unit-memory code of length 8, cardinality 128 with $d_{free} = 4$ was not known.

An idea [4] of the Gray mapping of $\mathbf{Z}_4-$linear block codes allows to construct good binary block codes. The present paper gives first examples of using this idea for $\mathbf{Z}_4-$linear unit-memory convolutional codes. We hope that these examples can be extended and that it is possible to construct other good binary trellis codes from $\mathbf{Z}_4-$linear unit-memory and multimemory convolutional codes.

References

[1] L. N. Lee, "Short unit-memory byte oriented binary convolutional codes having maximal free distance," *IEEE Trans.Inform.Theory,* vol.IT-22, pp.349–352, May.,1976.

[2] C. Thommesen and J. Justesen, "Bounds on Distance and Error Exponents of Unit-Memory Codes, *IEEE Trans.Inform.Theory,* vol.IT-29, pp.637–649, sep., 1983.

[3] A. A. Nechaev, "Kerdock Code in a Cyclic Form," *Discrete Math.*, vol. 1, no.4, pp.123–139, 1989. (in Russian)

[4] A. R. Hammons, P. V. Kumar, A. R. Calderbank, N. J. A. Sloane, and P. Solé, "The $\mathbf{Z}_4$-Linearity of Kerdock, Preparata, Goethals and Related Codes," *IEEE Trans. Inform. Theory*, in press.

[5] A. Said and R. Palazzo, "Using Combinatorial Optimization to Design Good Unit-Memory Convolutional Codes," *IEEE Trans. Inform. Theory*, vol.IT-39, pp.1100–1108, May, 1993.

[6] V. V. Zyablov and V. R. Sidorenko, "Maximum Likelihood Decoding with Soft Decision of Partial Unit-Memory Codes," *Prob. Peredachi Inf.* vol.28, pp.22–27, 1992.