Mathematica
Balkanica

New Series Vol. 18, 2004, Fasc. 1-2

Ternary Cyclic and Negacyclic LUEP Codes of Lengths up to 26

Tsonka Baicheva[1], *Irina Gancheva*[2]

Using a computer search, the unequal error protection capabilities of all ternary cyclic and negacyclic codes of lengths up to 26 that have minimum distances at least 3 are found.

AMS Subj. Classification: 68R15.

Key words: Ternary cyclic codes; ternary negacyclic codes; linear unequal error protection codes.

1. Introduction

Most error-correcting block codes considered in the literature have the property that their correcting capabilities are described in terms of the correct reception of the entire message. These codes can successfully be applied in those cases where all positions in a message word require protection against errors. However, many applications exist in which some message positions are more important than others. For example in transmitting numerical data, errors in the sign or in the high-order digits are more serious than the errors in the low-order digits. As another example consider the transmission of message words from different sources simultaneously in only one codeword, where the different sources have different demands concerning the protection against errors. For instance an observer transmitting by a satellite the results of many simultaneous experiments. Some of the experiments may be more important than others and therefore may be deserving of higher error protection. Instead of using a separate code for each experiment, it would usually be more efficient and more desirable to use one code, and thus one encoder and one decoder.

Linear codes that protect some positions in a message word against a larger number of errors than other ones are called linear unequal error protection (LUEP) codes. Masnick and Wolf [10] introduced the concept of unequal error protection. They considered error protection of single positions in codewords and described some properties, found bounds, and gave examples of linear systematic UEP codes. In the following papers ([1], [2], [3], [5], [6], [7], [8], [9]) approaches to the construction of UEP codes based on a generator or parity-check matrix were used.

In this work we consider error protection of single positions in the input message words, following the definition of Dunning and Robbins [4]. They introduced a so-called separation vector to measure the error-correcting capability of a LUEP code. A basic problem is to find a LUEP code with a given dimension and separation vector such that its length is minimal and hence its information rate is maximal. In this work, using a computer search, the unequal error protection of all ternary cyclic and negacyclic codes of lengths up to 26 that have minimum distances at least 3 were found.

2. Definitions and preliminaries

A linear $[n, k]$ code C of length n and dimension k over $GF(q)$ is a k-dimensional linear subspace of $GF(q)^n$. A generator matrix G of this code is a k by n matrix whose rows form a basis of C. Then, if $x = (x_1, \ldots, x_k)$ is an information word, $v(x) = xG = (v_1, \ldots, v_n)$ is a codeword of the code C. For $x \in GF(q)^n$, $wt(x)$ denotes the Hamming weight of v, i.e., the number of nonzero components in v.

Dunning and Robbins [4] have introduced the following formal definition.

Definition 1 For a linear $[n, k]$ code C over the alphabet GF(q) the *separation vector* $s(G) = (s(G)_1, \ldots, s(G)_k)$ of length k, with respect to a generator matrix G of C, is defined by

$$s(G)_i = min\{wt(mG) | m \in GF(q)^k, m_i \neq 0\}, i = 1, \ldots, k$$

This means that for any $\alpha, \beta \in GF(q), \alpha \neq \beta$, the sets $\{mG | m \in GF(q)^k, m_i = \alpha\}$ and $\{mG | m \in GF(q)^k, m_i = \beta\}$ are at distance $s(G)_i$ apart $(i = 1, \ldots, k)$.

If a linear code C has a generator matrix G such that the components of the separation vector $\mathbf{s}(G)$ are not mutually equal, then the code C is called a *linear unequal error protection* (LUEP) code.

In this work we have used a modification of Definition 1 for the separation vector as it has been done in [8]. Let C be an $[n, k_1 + k_2 + \ldots + k_m]$ code for the message space $M_1 \times M_2 \times \ldots \times M_m$, where $M_i = \{0, 1, \ldots q - 1\}^{k_i}$, for $i = 1, 2, \ldots m$. Each message x for C can be written as $x = (x_1, x_2, \ldots, x_m)$, where x_i is a k_i-tuple and is the component message of x for the component message space $M_i, i = 1, 2, \ldots, m$. The separation vector $s = (s_1, s_2, \ldots, s_m)$ is defined by

$$s_i = min\{d[v(x), v(x')] : x = (x_1, \ldots, x_i, \ldots, x_m), x' = (x'_1, \ldots, x'_i, \ldots, x'_m),$$

$$v(x), v(x') \in C, x_l, x'_l \in M_l, l = 1, 2, \ldots, m, x_i \neq x'_i\}.$$

Clearly, the minimum distance of C is

$$d_{min} = min\{s_1, s_2, \ldots, s_m\}.$$

Let $v(x)$ be the transmitted codeword of C and r be the received vector. It can be shown that x_i can be correctly decoded from r if $d[v(x), r] \leq \lfloor (s_i - 1)/2 \rfloor$, for $i = 1, 2, \ldots, m$. Therefore, the separation vector s specifies the UEP capability of C.

The linear code C can be expressed as the direct sum of component linear codes $C_1, C_2, \ldots, C_m$, where C_i is an $[n, k_i]$ code for the message space $M_i, i = 1, 2, \ldots m$. The following theorem shows an easy method of investigating the UEP capability of C.

Theorem 1 *[8] Suppose the minimum distance of C_1 is d_1 and $d_1 < d_2 < \ldots < d_m$. If the minimum weight of any codeword in $C \backslash C_1 \oplus C_2 \oplus \ldots \oplus C_{i-1}$ is $d_i, i = 2, 3, \ldots, m$, then the separation vector s of C for the message space $M_1 \times M_2 \times \ldots \times M_m$ is $s = (d_1, d_2, \ldots, d_m)$.*

It should be noted that the separation vector, or equivalently the UEP capability, of a code is dependent on its encoding method. Hence, for a linear code, its separation vector is dependent on the choice of its generator matrix.

Suppose that C is a q-qry cyclic code. In [10] it has been shown that all cyclic codes in systematic form do not have UEP capabilities. Therefore, all cyclic UEP codes are in nonsystematic form. Let $h(X) = q_1(X)q_2(X) \ldots q_l(X)$ be the parity check polynomial of C, where $q_1(X), q_2(X), \ldots, q_l(X)$ are q-ary irreducible polynomials. The cyclic code V_i with parity check polynomial $q_i(X)$ is a subcode of C. Code C is the direct sum of $V_1, V_2, \ldots, V_l$. The following theorem provides an encoding method to achieve the maximal UEP capability of a cyclic code.

Theorem 2 *[8] The matrix $G = [G_1^T G_2^T \ldots G_l^T]^T$ is a generator matrix for C that provides maximal UEP capability of C, where G_i is a generator matrix of the minimal ideal $V_i, i = 1, 2, \ldots, l$.*

The direct sum of some $V_{i_1}, V_{i_2}, \ldots, V_{i_r}$ is a cyclic subcode of C with parity check polynomial $q_{i_1}(X)q_{i_2}(X) \ldots q_{i_r}(X)$ and generator matrix $G = [G_{i_1}^T G_{i_2}^T \ldots G_{i_r}^T]^T$. Theorem 2 implies the following result.

Corollary 1 *[8] Suppose C has a maximal UEP capability represented by the separation vector $s = (d_1, d_2, \ldots, d_m)$ for the message space $M_1 \times M_2 \times \ldots \times M_m$, where $d_1 < d_2 < \ldots < d_m$. Then, the generator matrix for C that yelds s is in the form of $G = [G_1^T G_2^T \ldots G_l^T]^T$, where G_j is the generator matrix for a cyclic subcode C_j of C with parity check polynomial $h_j(X)$ and $deg(h_j(X)) = k_j, j = 1, 2, \ldots m$ Furthermore, the parity check polynomial for C is $h(X) = h_1(X)h_2(X)\ldots h_m(X)$ and $C = C_1 \oplus C_2 \oplus \ldots \oplus C_m$.*

Computing algorithm. We can use the following search algorithm to find the maximal UEP capability of a linear code C (see [8]).

First step. To find all the codewords of minimum distance d_1 in C and the subcode C_1 spanned by these codewords. If $C = C_1$, then C is not a UEP code. Otherwise - second step.

Second step. To find all the codewords of minimum weight d_2 in $C \backslash C_1$ and the subcode C_2' spanned by these codewords. $C_1 + C_2'$ is equal to $C_1 \oplus C_2$, where C_2 is some subcode of C_2'. If $C = C_1 \oplus C_2$, then C is a two-lewel UEP code with separation vector $s = (d_1, d_2)$, where $d_1 < d_2$. Otherwise, we proceed to the third step.

i-th step. To find all the codewords of minimum weight d_i in $C \backslash C_1 \oplus C_2 \oplus \ldots \oplus C_{i-1}$ and the subcode C_i' spanned by these codewords. Then, $C_1 \oplus C_2 \oplus \ldots \oplus C_{i-1} + C_i' = C_1 \oplus C_2 \oplus \ldots \oplus C_i$, and C is an i-level UEP code with separation vector $s = (d_1, d_2, \ldots, d_i)$, where $d_1 < d_2 < \ldots < d_i$. Otherwise, C is at least an $(i+1)$-level UEP code and we have to proceed to the next step.

3. Results

Using this algorithm, we compute the separation vectors of ternary cyclic and negacyclic codes of lengths up to 26, for which the minimum distances are at least 3. The results are listed in the Tables below. Each code in the table is characterized by exponents of its nonzeros that are also roots of its parity check polynomial. We use semicolons to separate the set of nonzeros into two subsets as all the codes are two level UEP codes. The first subset represents the nonzeros of the subcode C_2 and the second subset represents the nonzeros of the subcode C_1.

Ternary Cyclic Codes

No	$[n,k,d]$	k_1	k_2	s_1	s_2	nonzeros
1.	$[14,7,4]$	1	6	7	4	7;2
2.	$[16,10,4]$	2	8	5	4	10;0,4,5,8
3.	$[16,9,4]$	1	8	6	4	8;2,5,10
4.	$[16,8,5]$	2	6	7	5	10;0,5,8
5.	$[16,7,6]$	3	4	7	6	8,10;5
6.	$[16,7,5]$	1	6	8	5	8;4,5
7.	$[16,6,6]$	2	4	9	6	10;5
8.	$[16,6,4]$	2	4	5	4	10 ;0,4,8

Ternary Cyclic Codes

No.	[n,k,d]	k_1	k_2	s_1	s_2	nonzeros
9.	$[16,5,6]$	1	4	10	6	8;5
10.	$[16,5,4]$	1	4	8	4	8;2,10
11.	$[20,14,4]$	1	13	5	4	10;0,1,4,11
12.	$[20,13,4]$	4	9	6	4	11;2,4,10
13.	$[20,13,4]$	1	12	6	4	10;1,4,11
14.	$[20,12,4]$	7	5	5	4	5,10,11;0,4
15.	$[20,11,4]$	2	9	5	4	10;2,4,5
16.	$[20,11,4]$	6	5	6	4	5,11;2,10
17.	$[20,10,4]$	5	5	6	4	10,11;0,4
18.	$[20,10,4]$	2	8	5	4	0,10;1,11
19.	$[20,9,6]$	1	8	10	4	10;1,11
20.	$[20,9,6]$	4	5	8	4	11;2,10
21.	$[20,8,5]$	4	4	8	5	11;0,5,10
22.	$[20,7,8]$	1	6	10	8	10;5,11
23.	$[20,7,4]$	2	5	10	4	5;2,10
24.	$[20,6,8]$	2	4	10	8	5;4
25.	$[20,6,4]$	1	5	10	4	10;0,4
26.	$[20,5,8]$	1	4	10	8	10;4
27.	$[22,15,4]$	5	10	6	4	7;2,4
28.	$[22,11,6]$	1	10	10	6	11;2,7
29.	$[22,11,4]$	1	10	11	4	11;2,4
30.	$[22,7,10]$	1	6	11	10	0;7,11
31.	$[22,6,12]$	1	5	13	12	11;4

Ternary Negayclic Codes

No.	$[n, k, d]$	k_1	k_2	s_1	s_2	nonzeros
1.	$[20, 14, 3]$	4	10	4	3	6;2,4,5
2.	$[20, 12, 3]$	2	10	5	3	1;2,4,5
3.	$[20, 8, 5]$	4	4	8	5	6;1,2

References

[1] I. M. Boyarinov. On unequal error protection codes, *Proc. Fifth Conf. on Theory of Transmission and Coding of Inform.*, Moskow-Gorki, U.S.S.R., pt. II, 1972, 22-24.

[2] I. M. Boyarinov and G. L. Katsman. On linear unequal error protection codes, *Proc. Seventh Nat. Symp. on Problems of Redundacy in Information System"*, Leningrad, U.S.S.R., pt. I, 1977, 66-70.

[3] I. M. Boyarinov and G. L. Katsman. Linear unequal error protection codes, *Voprosi Kibernetiki*, **34** (1977), 60-91.

[4] L. A. Dunning and W. E. Robbins. Optimal encodings of linear block codes for unequal error protection, *Inform. Contr.*, **37** (1978), 150-177.

[5] V. N. Dynkin and V. A. Togonidze. Cyclic codes with unequal protection of symbols, *Probl. Peredach. Inform.*, **12** (1976), no. 1, 24-28.

[6] W. J. van Gils. Two topics on linear unequal error protection codes: Bounds on their length and cyclic code classes, *IEEE Trans. Inform. Theory*, **29** (1986), no. 6, 866-876.

[7] C. C. Kilgus and W. C. Gore. A class of cyclic unequal-error-protection codes, *IEEE Trans. Inform. Theory*, **18** (1972), 687-690.

[8] M. C. Lin and S. Lin. Cyclic unequal error protection codes constructed from cyclic codes of composite length, *IEEE Trans. Inform. Theory*, **34** (1988), no. 4, 867-871.

[9] D. Mandelbaum. Unequal-error-protection codes derived from difference sets,*IEEE Trans. Inform. Theory*, **18** (1972), 686-687.

[10] B. Masnik and J. Wolf. On linear unequal error protection codes, *IEEE Trans. Inform. Theory*, **13** (1967), 600-607.

Institute of Mathematics and Informatics
Bulgarian Academy of Sciences
P.O. Box 323,
Veliko Tarnovo 5000, BULGARIA
e-mails: [1] *tsonka@moi.math.bas.bg* [2] *irina@moi.math.bas.bg*

Received 30.09.2003