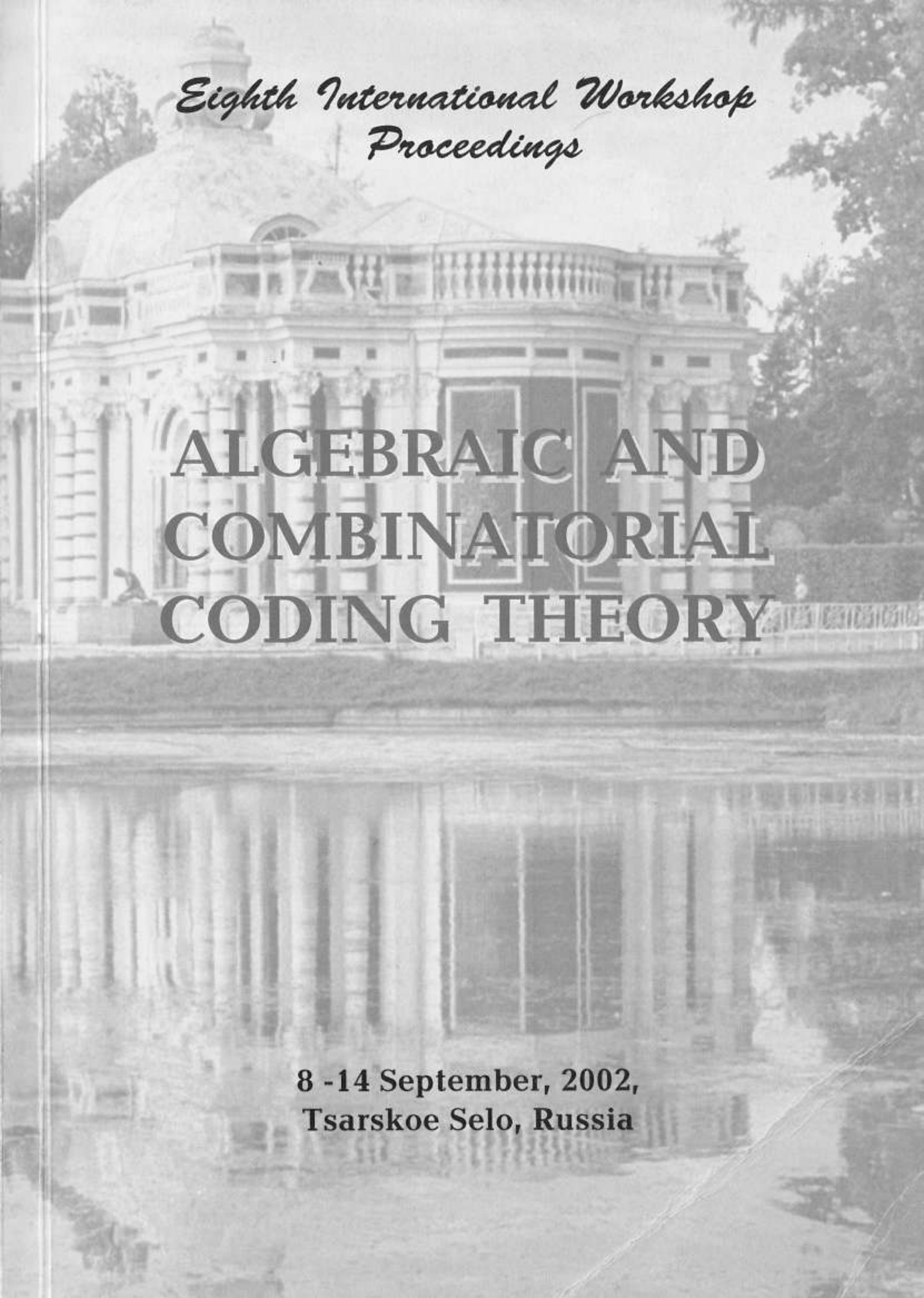


Eighth International Workshop
Proceedings

ALGEBRAIC AND COMBINATORIAL CODING THEORY

8 -14 September, 2002,
Tsarskoe Selo, Russia

Светлане Топаловой

Eighth International Workshop

on

ALGEBRAIC AND COMBINATORIAL CODING THEORY

Proceedings

8 -14 September, 2002, Tsarskoe Selo, Russia

Сергею [illegible]

PREFACE

Eighth International Workshop on Algebraic and Combinatorial Coding Theory (ACCT-VIII) is organized by the Institute for Information Transmission Problems of the Russian Academy of Sciences, St.Petersburg State University of Airspace Instrumentation (SUAI) and the Institute of Mathematics and Informatics of the Bulgarian Academy of Sciences.

The previous workshops were held in Varna, Bulgaria (1988), St.Petersburg, Russia (1990), Voneshta Voda, Bulgaria (1992), Novgorod, Russia (1994), Sozopol, Bulgaria (1996), Pskov, Russia (1998) and Bansko, Bulgaria (2000).

The workshop is supported by Russian Academy of Science (substantially) and the Russian Foundation for Fundamental Research.

Organizing committee

Bassalygo L. (Co-Chairman)
Dodunekov S. (Co-Chairman)

Kapralov S. (Gabrovo)
Kolesnik V. (St.Petersburg)
Kudrjashov B. (St.Petersburg)
Landjev I. (Sofia)
Zyablov V. (Moscow)
Zyapkov N. (Shoumen)

Programme committee

Manev N. (Co-Chairman)
Shehunova N. (Co-Chairman)

Boyvalenkov P. (Sofia)
Kolev E. (Sofia)
Levenstein V. (Moscow)
Mironchikov E. (St.Petersburg)
Yorgov V. (Shoumen)
Zinoviev V. (Moscow)

CONTENTS

Unidirectional Error Control Codes and Related Combinatorial Problems

R. Ahlswede, H. Aydinian, and L.H. Khachatrian

University of Bielefeld, Dept. of Mathematics, POB 100131, D-33501 Bielefeld
E-mail:ayd@mathematik.uni-bielefeld.de

Abstract

q–ary codes capable of correcting all unidirectional errors of certain level $1 \le \ell \le q-2$ are considered. We also discuss some related extremal combinatorial problems.

1 Introduction

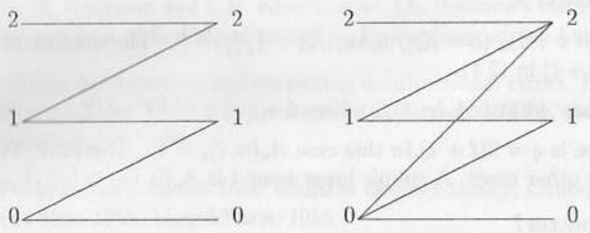
An extensive theory of error control coding has been developed under the assumption of symmetric errors in the data bits; i.e. errors of type 0 → 1 and 1 → 0 can occur in a codeword.However in many digital systems such as fiber optical communications and optical disks the ratio between probability of errors of type 1 → 0 and 0 → 1 can be large. Practically we can assume that only one type of errors can occur in those systems. These errors are called asymmetric. The statistics also shows that in some of the recently developed LSI/VLSI ROM and RAM memories the most likely faults are of the unidirectional type. The unidirectional errors slightly differ from asymmetric type of errors: both 1 → 0 and 0 → 1 type of errors are possible, but in any particular word all the errors are of the same type.The problem of protection against unidirectional errors arises also in designing fault–tolerant sequential machines, in write–once memory systems, in asynchronous systems et al.Codes correcting asymmetric/unidirectional errors are not well studied since they encounter more complicated structures than those for symmetric errors. (for more information see a good collection of papers in [2]). The first construction of nonlinear codes correcting asymmetric single errors was given by Varshamov and Tennengolts [5]. Modifications of VT–codes where used to construct new codes correcting t–asymmetric errors and burst of errors [2]. Very few constructions are known for codes correcting unidirectional errors (see [2]). We call a code of length n, correcting t–asymmetric errors a generalized VT–code if it is given by the set of solutions $(x_1,\dots,x_n) \in \{0,1\}^n$ of a linear congruence of the type

$$\sum_{i=1}^{n} f(i)x_i \equiv a \mod M$$

where $f(i)$ $(i=1,\dots,n)$ is an integer valued function, a and M are integers. There are deep relationships between VT–codes and some difficult problems in Additive Number Theory [6], [3]. In [6] Varshamov introduced a q–ary asymmetric channel. The inputs and outputs of the channel are n–sequences over a q–ary alphabet labelled with integers $\{0,1,\dots,q-1\}$. If the symbol i is transmitted then the only symbols which the receiver can get are $\{i,i+1,\dots,q-1\}$. Thus for any transmitted vector $(x_1,\dots,x_n)$ the received vector is of the form $(x_1+e_1,\dots,x_n+e_n)$ where $e_i \in \{0,\dots,q-1\}$ and $x_i+e_i \le q-1$, $i=1,\dots,n$. Then Varshamov says that t–errors have occured if $e_1+\cdots+e_n = t$. Generalizing the idea of VT–codes Varshamov presented [6] several ingenious constructions of t–error correcting codes for the defined channel. These codes has been shown to be superior to BCH codes correcting t errors for $q \ge 2$ and for large n.

2 ℓ–AUEC–codes and related problems

The number of symmetric errors in real systems is usually limited, while the number of unidirectional/asymmetric errors can be fairly large. This motivated several authors to consider codes that correct a few symmetrical errors and detect/correct all/many unidirectional (asymmetric) errors.We introduce now a special type of asymmetric errors in a q–ary channel. As above the alphabet Q is labelled with integers $\{0,1,\dots,q-1\}$ and for every transmitted vector $x=(x_1,\dots,x_n)$ the output is of the form $(x_1+e_1,\dots,x_n+e_n)$, where "+" denotes real addition, and $x_i+e_i \le q-1$; $i=1,\dots,n$.We say that an asymmetric error $e=(e_1,\dots,e_n)$ is of level $1 \le \ell \le q-1$ if $0 \le e_i \le \ell$. We also say that t asymmetric errors have occured if for the Hamming weight $wt_H(e)=t$. Correspondingly we say that t unidirectional errors have occured, if the output is either $x+e$ or $x-e$. The difference between the channel described above and Varshamov's channel for $q>2$, $l=1$ is seen in the figure below.



Here we concentrate on the case $t=n$. That is we consider q–ary codes correcting all asymmetric or unidirectional errors of given level ℓ. For those we use the abreviations ℓ–AAEC– and ℓ–AUEC–codes, respectively.For given $1 \le \ell \le q-2$ let $A_a(n,\ell)_q$ and $A_u(n,\ell)_q$ denote the maximum number of codewords in a q–ary code of length n, correcting all asymmetric and unidirectional errors, respectively. Clearly $A_u(n,\ell)_q \le A_a(n,\ell)_q$. Define two distances between $x = (x_1, \dots, x_n)$, $y = (y_1, \dots, y_n) \in Q^n = \{0, 1, \dots, q-1\}^n$.

$$d_a(x,y) = \max\{|x_i - y_i| : i = 1, \dots, n\}$$

$$d_u(x,y) = \begin{cases} d_a(x,y), & \text{if } x \geq y \text{ or } x \leq y \\ 2d_a(x,y), & \text{if } x \text{ and } y \text{ are incomparable} \end{cases}$$

where $x \geq y$ means that $x_i - y_i \geq 0$, for $i = 1, \dots, n$.

Proposition 1. Let $\mathcal{C} \subset \{0, \dots, q-1\}^n$. Then

(i) $\mathcal{C}$ is an ℓ–AAEC–code iff for every $x, y \in \mathcal{C}$ holds $d_a(x,y) \geq \ell + 1$

(ii) $\mathcal{C}$ is an ℓ–AUEC–code iff for every $x, y \in \mathcal{C}$ holds $d_u(x,y) \geq 2\ell + 1$.

It turns out that it is very easy to determine $A_a(n,\ell)_q$ for any given parameters $1 \leq \ell \leq q-2$ and n. However this is not the case for unidirectional codes.

Theorem 1. *For $1 \leq \ell \leq q-2$ one has $A_a(n,\ell)_q = \lceil \frac{q}{\ell+1} \rceil^n$.*

Theorem 2. *Given integers $\ell \geq 1$, $q > 2(\ell+1)$ we have $c\left(\frac{q}{\ell+1}\right)^n \leq A_u(n,\ell)_q \leq \lceil \frac{q}{\ell+1} \rceil^n$ for some constant c.*

Write $q = 2m + \varepsilon$, where $\varepsilon \in \{0, 1\}$, and let $Q = \{-m, -m+1, \dots, m+\varepsilon\}$. Let us define X to be the set of solutions $x \in Q^n$ of the equation

$$\sum_{i=0}^{n-1} (\ell+1)^i x_i = a. \tag{2.1}$$

It is easy to see that X is a l–AUEC–code.In a special case when $\ell + 1 | q$ we can maximize $|X|$ over all choices of a.

Theorem 3. *For $\ell + 1 | q$ $(q = |Q|)$* $\max_a |X| = \left(\frac{q}{\ell+1}\right)^{n-1}$. The maximum assumed for any $a \in Q = [-m, m+\varepsilon]$ in (2.1).

What can we say about $A_u(n,\ell)_q$, when $\ell + 2 \leq q \leq 2(\ell+1)$?

The simplest case is $q = 2(\ell+1)$.In this case $A_u(n,\ell)_q = 2^n$. However, we have no "good" lower bounds for other cases. A simple lower bound is $A_u(n,\ell)_q \geq \binom{n}{\lfloor \frac{n}{2} \rfloor}$.

Can we do it better?

The Case: $\ell = 1$

For $q = 3$ we have $A_u(n,1)_3 \geq \binom{n}{\lfloor \frac{n}{2} \rfloor}$.

We believe that one has equality in this case.

For $q = 4$ $A_u(n,1)_4 = 2^n$.

$q = 5$. Simple bounds observed above give us $c(2,5)^n \leq A_u(n,1)_5 \leq 3^n$. However the lower bound can be improved. To this end we look for good constructions of 1–AUEC codes given by means of some equation. Let $Q = \{0, \pm 1, \pm 2\}$. Given integers $a_0, \dots, a_{n-1}, \lambda$ let X be the set of all solutions $x = (x_0, \dots, x_{n-1}) \in Q^n$ of an equation

$$\sum_{i=0}^{n-1} a_i x_i = \lambda. \tag{2.2}$$

Proposition 2. The set X is a 1–AUEC code if all subset sums of $a_0, \dots, a_{n-1}$ are distinct.

Note that for $\lambda = 0$ this is also a necessary condition. Let $\{a_0, \dots, a_n\} \subset \mathbb{N}$ has distinct subset sums. Denote by $LA_u(n)_5$ the maximum possible number of solutions $x \in Q^n$ of the (2.2) over all choices of $a_0, \dots, a_n$ and integer λ. A slightly modified version of this problem was raised by Bohman (see [1]) in connection with a sum packing problem of Erdős [3].

Theorem 4. *For some constants c_1, c_2 one has $c_1(2,538)^n < LA_u(n)_5 < c_2(2,723)^n$.*

Error Detection Problem The detection problems for asymmetric and unidirectional errors are equivalent, i.e. any t–error detecting asymmetric code is also a t–error detecting unidirectional code. In fact the detection problem for unidirectional errors is much easier than the error correction problem. This problem is completely solved for binary channels (see Borden in [2]). That is for any $1 \leq t \leq n$; $t, n \in \mathbb{N}$; an optimal code of length n that can detect up to t errors is constructed. For $t < n$ observe that a code C detects all patterns of t or fewer unidirectional errors, iff whenever a codeword x covers a codeword y then for the Hamming distance $d(x,y) > t + 1$. In this case as an optimal code one has to take as codewords all vectors with Hamming weight $w = \lfloor \frac{n}{2} \rfloor \mod (t+1)$. This follows from a result of Katona [4].The problem is also solved for the Varshamov's channel, however for the channel we described above the problem is open.

References

[1] R. Ahlswede, H. Aydinian and L.H. Khachatrian, On Bohman's conjecture related to a sum packing problem of Erdős, submitted to Proceedings of the Amer. Math. Aoc.

[2] M. Blaum, Codes for detecting and correcting unidirectional errors. Edited by Mario Blaum. IEEE Computer Society Press Reprint Collections. IEEE Computer Society Press, Los Alamitos, CA, 1993.

[3] P. Erdős, Problems and results from additive number theory, Colloq. Theoretic des Nombres, Bruxelles, 1955, Liege&Paris, 1956.

[4] G. Katona, Families of subsets having no subset containing another with small difference, Niew. Arch. Wisk. (3) 20, 54–67, 1972.

[5] P.R. Varshamov and G.M. Tennengolts, A code which corrects single asymmetric errors (Russian) Avtomat. Telemeh. 26, 282–292, 1965.

[6] P.R. Varshamov, A class of codes for asymmetric channels and a problem from the additive theory of numbers, IEEE Trans. Inform. Theory, IT–19, No. 1, 92–95, 1973.

Extrinsic Information Transfer Functions for LDPC and Turbo Codes

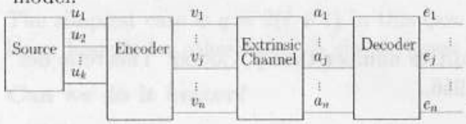
A. Ashikhmin, G. Kramer, and S. ten Brink
Lucent Technologies, Bell Labs
600 Mountain Ave., Murray Hill, NJ 07974, U.S.A.
e-mail: {aea,gkr,stenbrink}@lucent.com

1 Introduction

Extrinsic information transfer (EXIT) charts predict the convergence behavior of iterative decoding and detection scheme [1]. Experience suggests the charts are accurate, but there is a lack of proofs explaining why they work. The main purpose of this work is to show how to use EXIT charts for analyzing behavior of LDPC and Turbo codes and study properties of EXIT charts in the case of the binary erasure channel (BEC).

2 EXIT Charts

Let D_c be an $[n, k_c]$ linear code. Let us assume that D_c is used in the following communication model.



Here the Extrinsic channel is a noisy memoryless channel. The reason why we call it extrinsic is that during iterative decoding of an LDPC or Turbo code constituent decoders transmit extrinsic information from one to another. We assume that the vector $(a_1, \dots, a_n)$ is decoded by a maximum *a posteriori* extrinsic information (MAP) decoder. In other words the decoder computes values

$$e_j = \ln \frac{\Pr(v_j = 0|\underline{a}_{[j]})}{\Pr(v_j = 1|\underline{a}_{[j]})},$$

where $\underline{a}_{[j]} = (a_1, \dots, a_{j-1}, a_{j+1}, \dots, a_n)$.

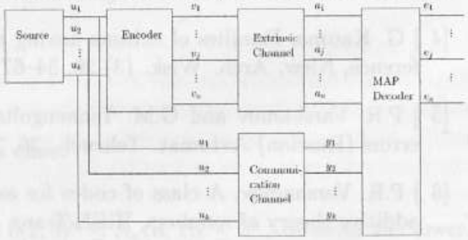
Let us denote by $I_A = \frac{1}{n}\sum_{j=1}^n I(V_j; A_j)$ the average mutual information on the input of the decoder and by $I_E^c = \frac{1}{n}\sum_{j=1}^n I(V_j; E_j)$ the average mutual information on its output. Since we assume a MAP decoder it is semi obvious that

$$I(V_j; E_j) = I(V_j; \underline{A}_{[j]}). \qquad (1)$$

Note that if the decoder is not a MAP one then $I(V_j; E_j) < I(V_j; \underline{A}_{[j]})$.

In what follows we will consider I_E^c as a function of $I_A \in [0..1]$. This function is our first EXIT chart.

Let D_v be an $[n, k_v]$ linear code and let us use it in the following communication model.



Similar to the previous case we introduce notions of average *a priory* information I_A and extrinsic information I_E^v. Again it is semi obvious to see that in our model

$$I(V_j; E_j) = I(V_j; \underline{YA}_{[j]}). \qquad (2)$$

We will consider I_E^v as a function of I_A and this function is our second EXIT chart.

3 EXIT Charts for LDPC codes

We will start with the following generalization of regular [2], [3] and irregular [4] LDPC codes. Let $C_i^v, i = 1, \dots, t$, be a set of $[n_i^v, k_i^v]$ linear codes. Denote $n = \sum_{i=1}^t k_i^v$ and $N = \sum_{i=1}^t n_i^v$. Let $C_i^c, i = 1, \dots, m$, be another set of $[n_i^c, k_i^c]$ linear codes chosen such that $\sum_{i=0}^m n_i^c = N$. We will assume that linear encodings for codes C_i^v and C_j^c are defined.

Let G be a bipartite graph with the vertex set $V \bigcup C$, $|V| = t, |C| = m$, and N edges. We will call vertexes from V variable nodes and vertexes from C check nodes. We assume that edges of G are enumerated and that $E = \{1, \dots, N\}$ is the set of their indexes. Denote by E_i the set of indexes of edges incident to the vertex $i, i \in V$ or C.

Let S be the set of vectors $\underline{w} \in \{0,1\}^N$ such that for any vertex i of V or C entries of $\underline{w}$ with indexes from E_i form a codeword of C_i^v or C_i^c respectively.

Let now $\underline{u}_1, \dots, \underline{u}_t$ be information vectors of codes $C_1^v, \dots, C_t^v$ respectively. Denote by $\underline{w}_1, \dots \underline{w}_t$ corresponding codewords. Now let us define a generalized LDPC code A as the set of vectors $\underline{u} = (\underline{u}_1, \dots, \underline{u}_t)$ such that the vectors $\underline{w} = (\underline{w}_1, \dots, \underline{w}_t)$ belong to S.

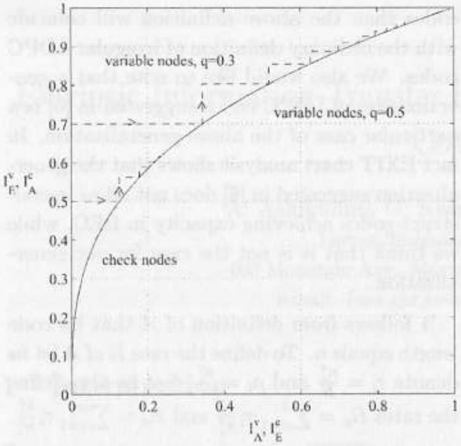
Note that if we choose C_i^v codes to be repetition codes and C_i^c to be single parity check codes then the above definition will coincide with the ordinary definition of irregular LDPC codes. We also would like to note that a generalization of LDPC codes suggested in [6] is a particular case of the above generalization. In fact EXIT chart analysis shows that the generalization suggested in [6] does not allow to construct codes achieving capacity in BEC, while we think that it is not the case for our generalization.

It follows from definition of A that its code length equals n. To define the rate R of A let us denote $\tau_i = \frac{n_i^v}{N}$ and $\rho_i = \frac{n_i^c}{N}$. Let us also define the rates $R_v = \sum_{i=1}^t \tau_i \frac{k_i^v}{n_i^v}$ and $R_c = \sum_{i=1}^m \tau_i \frac{k_i^c}{n_i^c}$.

Proposition 1 $R \geq 1 - \frac{1-R_c}{R_v}$.

Define the code D_c as the $[N, k_c], k_c = \sum_{i=1}^m k_i^c$, linear code formed by the direct sum [5, Ch.2.9] of codes $C_i^c, i = 1, \dots, m$. Similarly define the code D_v as the $[N, k_v], k_v = \sum_{i=1}^t k_i^v$, code formed by the direct sum of codes $C_i^v, i = 1, \dots, t$. Let us assume that N tends to infinity and that connection of vertices of G is chosen randomly. Using (1) and (2), it is not difficult to show that under this assumptions in the case of BEC the behavior of iterative decoding of A can be described by the evolution of the average mutual information in the process of alternating MAP decodings of codes D_c and D_v.

Let us consider an example. Let A be a $(2,4)$ regular LDPC code. In other words C_i^v are repetition codes of lengths 2 and C_i^c are single parity check codes of length 4. We assume that codewords of A are transmitted through BEC with erasure probability q. In this simple example we can compute EXIT charts explicitly as follows: $I_E^c(I_A) = (I_A)^3$ and $I_E^v(I_A) = 1 - q(1 - I_A)$. The EXIT charts for cases of $q = 0.3$ and 0.5 are presented on the following picture.



Using the charts, one can track the evolution of the mutual information. In the case of $q = 0.5$ the charts intersect each other. This means that at some moment the mutual information will stop changing and the iterative decoding will get stuck. On the contrary if $q = 0.3$ the charts do not intersect. Therefore with sufficiently many iterations we can achieve an arbitrary small probability of error.

In the case of BEC we can establish the following important properties of EXIT charts.

Denote $\mathcal{A}_c = \int_0^1 I_E^c \, dI_A$ and $\mathcal{A}_v = \int_0^1 I_E^v \, dI_A$ and let C be the capacity of BEC.

Theorem 2 *$\mathcal{A}_v = 1 - (1 - C)R_v$ and $\mathcal{A}_c = 1 - R_c$.*

From Proposition 1 and Theorem 2 it follows that

$$\frac{1 - \mathcal{A}_v}{\mathcal{A}_c} = \frac{(1 - C)R_v}{1 - R_c} = \frac{1 - C}{(1 - R_c)/R_v} = \frac{1 - C}{1 - R}. \tag{3}$$

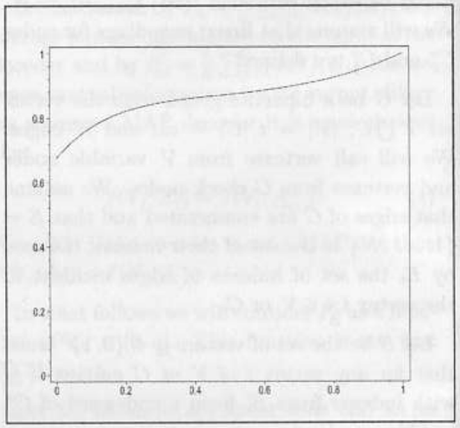
From this it follows that if $R > C$ then the EXIT charts will unavoidably intersect each other and the iterative decoding will fail. It also follows from (3) that the larger gap between the EXIT charts the further the code rate R from the capacity C.

Let $I_E^{c\perp}$ be the EXIT chart of the dual code of D_c.

Theorem 3 *$I_E^{c\perp}(I_A) = 1 - I_E^c(1 - I_A)$.*

A dual relation can be also formulated for the EXIT chart of the code D_v, but it is more complicated and we omit it in the given text.

Let us consider an example of a generalized LDPC code. Let all codes C_i^c be the Hamming codes of length 31. Let 40 percent of codes C_i^v be repetition codes of length 8 and 60 percent of codes C_i^v be single parity check codes of length 11. The corresponding EXIT charts are presented on the following picture.



One can see that EXIT charts for this code have different form compared to EXIT charts of LDPC code from the previous picture. This special form allows one to reduce the number of decoding iterations needed for achieving a desired probability of decoding error. The example also demonstrates a possibility of designing a code on such an way that there is a gap between EXIT charts in the area of small probability of error (upper right corner). We believe that this property will be useful for design of finite length codes. After several iterations of iterative decoding of a finite length code random variable passing by edges of G become dependent. As result the behavior of decoding process begins to deviate from the theoretical prediction. We expect that keeping EXIT charts apart in the area of small error probability will help to delay this effect and to lower the error floor.

EXIT chart analysis can be also conducted for Serially Concatenated Turbo Codes [7]. In the given text we omit details and only would like to mention the following result.

Let R be the rate of a serially concatenated turbo code A. Let R_{out} and R_{in} be rates of outer and inner codes of A respectively. It is well known that $R = R_{out}R_{in}$. The question is how to choose the rates R_{out} and R_{in}. The EXIT chart analysis shows that if we want to approach capacity in the case of BEC we must choose $R_{in} = 1$.

References

[1] S. ten Brink, "Convergence of iterative decoding," *Electron. Lett.*, **35**, pp. 806–808, 1999.

[2] R. G. Gallager, "Low density parity check codes," Sc.D. thesis, Sept. 1960.

[3] R. G. Gallager, "Low-density parity-check codes," *IRE Trans. Inform. Theory*, **8**, pp. 21–28, 1962.

[4] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. A. Spielman, "Efficient erasure correcting codes," *IEEE Trans. Inform. Theory*, **47**, pp.569–584, 2001.

[5] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*, New York: North-Holland, 1977.

[6] M. Lentmaier and K. Sh. Zigangirov, "On Generalized low-density parity-check codes based on Hamming component codes," *IEEE Communication Letters*, **3**, pp.248–250, 1999.

[7] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, "Serial concatenation of interleaved codes: performance analysis, design, and iterative decoding," *IEEE Trans. Inform. Theory*, **44**, pp.909–926, 1998.

On ranks and kernels problem of perfect codes

S. V. Avgustinovich*[†][‡] O. Heden *[§][¶] F. I. Solov'eva*[†][||]

Abstract

In 1998 Etzion and Vardy [6] proposed to clarify which pairs of numbers (r, k) are attainable as the rank r and kernel dimension k of some perfect code of length n. Two switching constructions are used to find an asymptotic decision of the problem.

1. Preliminaries

Denote by E^n n-dimensional metric space over the Galois field $GF(2)$ with the Hamming metric. Let $\mathbf{0}$ ($\mathbf{1}$) be the all-zero (all-one) vector in E^n. *A perfect binary code* C of length n with the code distance 3 (further a perfect code) is a subset in E^n such that for any vector $y \in E^n$ there exists a unique codeword $x \in C$ with $d(x, y) \leq 1$, where d is the Hamming distance. It is well-known that perfect codes of length n with distance 3 exist iff $n = 2^m - 1, m > 1$. *The kernel* $Ker(C)$ of a code C is the set of all its *periods* (all codewords $x \in C$ such that $x + C = C$). The dimension of the kernel is denoted by $k = k(C)$. The dimension of the subspace spanned by a code C is called the *rank* $r = r(C)$ of the code C.

In 1998 Etzion and Vardy [6] proposed to clarify which pairs of numbers (r, k) are attainable as the rank r and kernel dimension k of some perfect code of length n. It will be mentioned further as the *ranks and kernels problem.* Let $\delta(r)$ be such minimal number that $2^{\delta(r)} - \delta(r) - 1 \geq r - n + \log(n + 1)$. Denote by $U(n, r)$ the following

$$U(n, r) = n - \log(n + 1) - \delta(r).$$

In 1994 Etzion and Vardy [5] described a spectrum of ranks and in 1995 Phelps and LeVan [10] described a spectrum of kernel dimension of perfect codes for every admissible length $n \geq 15$. In 1998 Etzion and Vardy [6] established that for full rank perfect codes for every $n \geq 2^m - 1, m > 3$, it is true $k(C) \leq U(n, r)$ and the bound is tight for full rank perfect codes for each $n \geq 2^{10} - 1$. In 2001 Phelps and Villanueva, see [12], using the same technik generalized the result for perfect codes with any rank $r \leq n$ and showed that the bound is tight for perfect codes of rank $r < n$ for any $n = 2^m - 1, m > 3$. Denote by $L(n, r)$ the following

$$L(n,r) = \begin{cases} 2^{n-r}, & \text{if } r > n - \log(n+1) + 1, \\ 2^{n-r} - 1, & \text{if } r \leq n - \log(n+1) + 1. \end{cases}$$

Phelps and Villanueva [12] established that $L(n, r)$ is the exact lower bound of kernel dimension of a perfect code for length $n \geq 15$ and rank r.

For $r < 15$ perfect codes of length 15 for all possible pairs (r, k) are given in [11]. For $n = 15$ full rank perfect codes with any dimension kernel k, $1 \leq k \leq 5$ are known, see [7, 9, 6], for $k \geq 7$ full rank perfect codes do not exist [6, 13]. Full survey of the investigated ranks and kernels problem and related questions can be found in [4].

In the paper we are going to get the following theorem.

Theorem 1. *Let n and r be natural numbers such that $n = 2^m - 1$, $m > 10$, $n - \log(n+1) \leq r \leq n$. Then for any natural number k such that $L(n, r) \leq k \leq U(n, r)$ there exists a perfect code of length n and rank r with kernel dimension k.*

2. Construction I

Let H^n be the Hamming code of length n defined by its parity check matrix with columns given in lexicographic order. A linear subspace R_i^0 of the code H^n is defined as a linear span of all vectors of weight 3 with the ith coordinate equaled to 1, $i \in \{1, \ldots, n\}$. It is called reduced i-component. For any vector $v \in H^n$ a set $R_i^v = R_i^0 + v$ is called an i-component with the representative v. Let us consider a set of pairs $F = \{(u_1, i_1), (u_2, i_2), ..., (u_s, i_s)\}$, where $u_t \in H^n$, $i_t \in \{1, 2, ..., n\}$.

We call a family F *separable* if the following conditions are hold:

1. the set of vectors of length $\log(n + 1)$ corresponding to the binary representation of natural numbers $i_1, i_2, ..., i_s$ is linear independent over $GF(2)$;

2. $\mathbf{0} \notin R_{i_t}^{u_t}$, $i_t \in \{1, 2, ..., n\}$;

3. for all $t \neq l$ it is true that $R_{i_t}^{u_t} \cap R_{i_l}^{u_l} = \emptyset$.

The number s of pairs in the family F is the *size* of the family. A separable family F we call *full rank family* if $s = \log(n + 1)$. Let $M = \{L_1, L_2, \ldots, L_s\}$ be a set of arbitrary linear subspaces of the code H^n. A family F we call *M-separable* if besides of conditions 1 and 2 the following condition is valid

3*. for all $t \neq l$ it is hold $(R_{i_t}^{u_t} + L_t) \cap (R_{i_l}^{u_l} + L_l) = \emptyset$.

Let all spaces L_i in the set M coincide with some space L. Unless otherwise stated a M-separable family F we will call in this case *L-separable.*

Let us consider the set

$$C(F, M) = H^n \setminus \bigcup_{t=1}^{s} (R_{i_t}^{u_t} \oplus L_t) \cup \bigcup_{t=1}^{s} (R_{i_t}^{u_t} \oplus L_t \oplus e_{i_t}),$$

where e_{i_t} is the vector with one in only (i_t)th coordinate.

Let $K(F, M) = \bigcap_{t=1}^{s} (R_{i_t}^0 \oplus L_t)$. Using the same approach as in [10] it can be proved the following fact.

[*] We are grateful to Swedish Institute for supporting this research.

[†] Sobolev Institute of Mathematics, Novosibirsk, Russia, avgust@math.nsc.ru, sol@math.nsc.ru

[‡] This research was partially supported by the RFBR 00-01-00916

[§] Royal Institute of Technology, Stockholm, Sweden, olohed@math.kth.se

[¶] This research was supported by Swedish Research Council

[||] This research was partially supported by the RFBR 00-01-00822

Theorem 2. *Let F be a M-separable family of size s. Then the set $C(F, M)$ is a perfect code of rank $n - \log(n+1) + s$ with kernel $K(F, M)$.*

Corollary 1. *Let F be a L-separable family of size s. Therefore there exist perfect codes of length n of rank $n - \log(n+1) + l$ with any kernel dimension from $dim(K(F))$ to $dim(K(F, L))$.*

Last theorem shows that to prove Theorem 1 it is necessary to construct L-separable family of pairs for appropriate subspaces L. A basis of the construction is given by the following propositions.

Proposition 1. *For all admissible $n > 7$ there exist separable families of any size s, where $s = 1, \ldots, \log(n+1)$.*

Proposition 2. *Let F be a separable family of pairs of size s of the code $H^{(n-1)/2}$ and $v \in H^{(n-1)/2} \setminus \bigcup_{t=1}^{s} R_{i_t}^{u_t}, v \notin \{\mathbf{0}, \mathbf{1}\}$. Then the family $F' = F \cup (v, n)$ is a $R_n^{\mathbf{0}}$-separable family of pairs of size $s+1$ of the code H^n.*

Proposition 3. *Let F be a L-separable family of pairs of size s of the code $H^{(n-1)/2}$ and $v \in H^{(n-1)/2} \setminus \bigcup_{t=1}^{s} (R_{i_t}^{u_t} \oplus L), v \notin L$. Then the family $F' = F \cup (v, n)$ is a $(R_n^{\mathbf{0}} \oplus L)$-separable family of pairs of size $s+1$ of the code H^n.*

Propositions 1 and 2 guarantee the existence of perfect codes of any rank s, $s \leq n$, with minimal possible kernel and with kernel dimension till $(n-1)/2$. A possibility to choose any linear subspaces of the space L for the set M gives continuous variation (adding one with every step) of kernel dimension from the minimal up to $(n-1)/2$. Proposition 3 leaves a hope that the construction I can be useful to construct perfect codes with big kernels. However to complete the proof of Theorem 1 we have to use the construction II.

3. Construction II

In the section we describe the class of perfect codes of length n rank $r \leq n$ and kernel dimension $(n-1)/2 \leq k \leq U(n, r)$. To do it we use well-known iterative Vasil'ev construction [1]. Let us remind it.

Let C' be a perfect code of length $(n-1)/2 = 2^{m-1} - 1, m \geq 2$, and λ be a function from C' to the set $\{0, 1\}$. Let $|u| = u_1 + \cdots + u_{(n-1)/2} \pmod 2$, where $u = (u_1, \ldots, u_{(n-1)/2})$. The set $C^n = \{(u, u+v, |u| + \lambda(v)) : u \in E^{(n-1)/2}, v \in C'\}$ is a perfect code of length n.

Let C be a perfect code of length n rank $n - \log(n+1) \leq r \leq n$ and maximal kernel dimension $U(n, r)$. First we take into account the existence of perfect codes of length 15 with different ranks and known maximal size kernels [6, 9, 11, 13] and full rank perfect code of length $2^{10} - 1$ with maximal kernel (see [6]). Then choosing subspaces of different dimensions in the kernel $Ker(C)$ and an appropriate function λ in iterative Vasil'ev construction we get the following theorem.

Theorem 3. *Let n and r be natural numbers such that $n = 2^m - 1$, $m > 10$, $n - \log(n+1) \leq r \leq n$. Then for any positive integer k such that $(n-1)/2 \leq k \leq U(n, r)$ there exists a perfect code of length n and rank r with kernel dimension k.*

4. Conclusion

From Corollary 1 of Theorem 2 and from Theorem 3 we get Theorem 1. For some pairs (n, r), where $n \leq 2^{10} - 1$ for full rank perfect codes the ranks and kernels problem is still open. Earlier Theorem 1 was announced in [4]. The verification of the theorem required to use many different switching, concatenation and combining constructions, see [2]-[4]. The approach described in Section 3 were used in [2] to construct not full rank perfect codes and in [3] for full ranks codes with big kernels. In the presented paper we use only two switching constructions.

References

[1] *Vasil'ev Y.L.*, On nongroup close-packed codes, Problems of Cybernetics 8 (1962) 375-378 (in Russian).

[2] *Avgustinovich S. V., Heden O., Solov'eva F. I.* On ranks and kernels of perfect codes, Stockholm: Royal Inst. of Technology, 2001. (Preprint / Trita-mat.-2001-13).

[3] *Avgustinovich S. V., Solov'eva F. I., Heden O.*, Perfect codes of full rank with kernels of large dimensions // Diskretn. Anal. Issled. Oper., 2001. Ser. 1 V. 8, No.4, 3-8 (in Russian).

[4] *Avgustinovich S. V., Solov'eva F. I., Heden O.* On ranks and kernels of perfect codes, Proc. of Int. Workshop on Discrete Analysis and Operation Research, Novosibirsk, Russia. June (2002) to appear (in Russian).

[5] *Etzion T., Vardy A.*, Perfect binary codes: constructions, properties and enumeration // IEEE Trans. Inform. Theory. 1994. V. 40. N 3. P. 754–763.

[6] *Etzion T., Vardy A.*, On perfect codes and tilings: problems and solutions // SIAM J. Discrete Math. 1998. V. 11. N 2. P. 205–223.

[7] *Heden O.* A binary perfect code of length 15 and codimension 0 // Des. Codes Cryptogr. 1994. V. 4. N 3. P. 213–220.

[8] *Hergert F.* Algebraische Methoden für Nichtlineare Codes. Thesis Darmstadt, 1985.

[9] *Näslund M.* Steiner triple systems and perfect codes. Master of Sci. thesis // Royal Institute of Technology, Stockholm, Sweden, 1993.

[10] *Phelps K. T., LeVan M. J.* Kernels of nonlinear Hamming codes // Des., Codes and Cryptogr., 1995. V. 6, N 3. P. 247–257.

[11] *Phelps K. T.* An enumeration of 1-perfect binary codes of length 15 // Australas. J. Combin. 2000. V. 21. P. 287–298.

[12] *Phelps K.T., Villanueva M.* On perfect codes: rank and kernel // Des. Codes Cryptogr., to appear.

[13] *Vardy A.* Private communication.

The Newton radius of some binary and ternary cyclic codes

Tsonka Baicheva
Institute of Mathematics and Informatics
Bulgarian Academy of Sciences
P.O. Box 323, 5000 Veliko Tarnovo, Bulgaria
tsonka@moi.math.bas.bg

Abstract

For a linear $[n, k, d]$ code C all errors of weight $t \le (d-1)/2$ are uniquely correctable. However, there are errors of weight $> t$ which are also uniquely correctable. The Newton radius of a code is defined to be the largest weight of a uniquely correctable error. In this work Newton radii of all binary cyclic codes of lengths up to 31 and ternary cyclic and negacyclic codes of lengths up to 22 are determined.

I Introduction

In [1] the code parameter Newton radius of a binary linear code was introduced for the largest weight of a uniquely correctable error. Bounds and exact values of the Newton radius for several classes of binary linear codes were given in [2], [3] and [4]. Later these bounds were generalized for linear codes over arbitrary alphabets in [5].

II Notations and preliminary results

Here we will summarize the results obtained in the papers mentioned above.

Let us denote by $w(x)$ the Hamming weight of the vector x and by $d(x, y)$ the Hamming distance between vectors x and y. If C is a linear $[n, k]$ code let $z + C$ be a coset of C. A coset leader is a vector in the coset of minimal Hamming weight. A coset leader x is unique if it is the only coset leader in the coset, i.e. $w(x) < d(x, c)$ for all nonzero codewords c. An error e is uniquely correctable if and only if it is the unique coset leader in its coset. The Newton radius $\nu(C)$ of C is the largest weight of a uniquely correctable error, i.e $\nu(C) = max\{w(x) | w(x) < d(x, c), \forall c \in C\backslash\{0\}\}$. The covering radius $r(C)$ is defined as the maximal distance of a vector from the code, i.e. $r(C) = max\{w(x) | w(x) \le d(x, c), \forall c \in C\backslash\{0\}\}$. It follows immediately from the definitions that

$$\left\lfloor \frac{d-1}{2} \right\rfloor \le \nu(C) \le r(C). \tag{1}$$

A lower bound on the Newton radius is given in [2]

$$\nu(C) \ge r(C) - k. \tag{2}$$

For an $[n, k; q]$ code without zero coordinates (see [5])

$$\nu(C) < n - \frac{n}{q} \tag{3}$$

$$\nu(C) \le \frac{(q-1)n - k - (q-2) - r(C)}{q-1}. \tag{4}$$

We will note that there is equality in (1) for perfect codes, in (2) for two times repeated binary codes, in (4) for binary codes with $k \le 3$ and for two times repeated binary codes.

III The Newton radius of some binary and ternary cyclic codes

In [6], [7] and [8] all binary cyclic codes of lengths up to 31, ternary cyclic codes of lengths up to 25 and ternary negacyclic codes of lengths up to 26 were classified and their covering radii were determined. These classifications were used as a basis for our investigation. The Newton radii of binary cyclic codes with dimensions 2 and 3 were determined by (4). For the rest of the investigated codes upper and lower bounds for $\nu(C)$ were determined. To find the exact value of $\nu(C)$ computer search was used testing only these vectors whose weights are between the lower and upper bounds for $\nu(C)$. This way, we were able to determine all unique coset leaders for each weight between t and $\nu(C)$. Unfortunately, such straightforward computations can be used only for codes with short lengths.

The codes which have cosets with more than 90 per cent unique coset leaders, the number of these unique leaders and the corresponding weight of the coset are presented in the tables below. All the results of this investigation can be found at http://www.moi.math.bas.bg/∼tsonka.

Table 1. Binary cyclic codes.

No	[n,k,d]	$\nu(C)$	Unique leaders	No	[n,k,d]	$\nu(C)$	Unique leaders
1.	[15,3,5]	6	425, w=3	10.	[27,7,6]	9	2754, w=3
2.	[21,9,3]	4	189, w=2	11.	[27,6,6]	9	2754, w=3
3.	[21,7,3]	7	189, w=2	12.	[27,3,9]	12	80352, w=5
4.	[21,6,7]	6	5880, w=4				288954, w=6
5.	[21,3,7]	9	5880, w=4				825552, w=7
			18816, w=5	13.	[27,2,18]	12	4540968, w=9
6.	[21,2,14]	9	105987, w=7	14.	[31,16,5]	5	4185, w=3
7.	[25,5,5]	10	2250, w=3	15.	[31,11,10]	7	162099, w=5
			11625, w=4	16.	[31,10,10]	7	162099, w=5
8.	[25,4,10]	10	50625, w=5	17.	[31,6,15]	10	7291200, w=8
9.	[27,9,3]	9	324, w=2	18.	[31,5,16]	11	7490220, w=8

Table 2. Ternary cyclic codes.

No	[n,k,d]	ν	Unique leaders	No	[n,k,d]	ν	Unique leaders
1.	[14,2,7]	8	15316, w=4	22.	[20,7,8]	7	73350, w=4
2.	[16,7,6]	4	4160, w=3	23.	[20,7,6]	7	8960, w=3
3.	[16,6,4]	4	432, wt=2				70984, w=4
4.	[16,5,6]	6	4160, w=3	24.	[20,6,10]	7	474960, w=5
5.	[16,5,4]	6	432, w=2	25.	[20,6,8]	8	73500, w=4
6.	[16,4,8]	8	27744, w=4	26.	[20,6,8]	7	76150, w=4
7.	[16,4,6]	6	4160, w=3	27.	[20,6,4]	8	700, w=2
8.	[16,4,4]	8	432, w=2	28.	[20,5,11]	8	2297880, w=6
9.	[16,3,10]	8	135744, w=5	29.	[20,5,8]	9	76150, wt=4
10.	[16,3,8]	8	28296, w=4				445088, wt=5
11.	[16,2,12]	9	505120, w=6	30.	[20,5,4]	10	700, w=2
12.	[16,2,8]	8	28840, w=4	31.	[20,4,12]	10	2425440, w=6
			131712, w=5	32.	[20,4,8]	10	76150, w=4
13.	[20,12,4]	3	700, w=2				449408, w=5
14.	[20,11,4]	4	700, w=2	33.	[20,4,5]	8	8800, w=3
15.	[20,10,4]	4	700, wt=2	34.	[20,3,10]	10	493120, w=5
16.	[20,9,6]	5	8320, w=3				2368200, w=6
17.	[20,9,4]	5	700, w=2	35.	[20,2,10]	12	495120, w=5
18.	[20,8,8]	6	70910, w=4				2442000, w=6
19.	[20,8,5]	6	8800, w=3				9340800, w=7
20.	[20,8,4]	6	700, w=2	36.	[22,2,11]	14	4285560, w=6
21.	[20,7,8]	6	73500,w=4				

Table 3. Ternary negacyclic codes.

No	[n,k,d]	ν	Unique leaders	No	[n,k,d]	ν	Unique leaders
1.	[14,2,7]	8	15316, w=4	6.	[20,6,9]	7	465888, w=5
2.	[18,4,6]	8	6446, w=3	7.	[20,4,5]	8	8800, w=3
3.	[18,2,9]	10	271152, w=5	8.	[20,4,12]	9	2443680, w=6
			1122912, w=6	9.	[20,2,15]	11	31785000, w=8
4.	[20,8,8]	6	69440, w=4	10.	[22,2,11]	14	4285560, w=6
5.	[20,8,5]	6	8800, w=3				

References

[1] T. Helleseth, T. Kløve and V. Levenshtein, The Newton radius of equidistant codes, *Proc. IEEE Intern. Symp. on Inform. Theory and its Applications*, Victoria, B.C., Canada, Sept. 17-30, 1996, pp. 721-722.

[2] T. Helleseth and T. Kløve, The Newton radius of codes, *IEEE Trans. on Inform. Theory*, vol. 43, No. 6, 1997, pp. 1820-1831.

[3] E. Gabidulin and T. Kløve, On the Newton radius, *Reports in Informatics* (Dept. Informatics, Univ. Bergen, Bergen, Norway), no. 130, Feb. 1997.

[4] T. Kløve, Relations between the covering and Newton radii of binary codes, *Discrete Mathematics*, vol. 238, 2001, pp. 81-88.

[5] E. Gabidulin and T. Kløve, On the Newton and covering radii of linear codes, *IEEE Trans. on Inform. Theory*, vol. 45, No. 7, 1999, pp. 2534-2536.

[6] D. Downie, N. J. A. Sloane, The covering radius of cyclic codes of length up to 31, *IEEE Trans. Inf. Theory*, vol. 31, 1985, pp. 446-447.

[7] T. Baicheva, The covering radius of ternary cyclic codes with length up to 25, *Designs, Codes and Cryptography*, vol.13, pp. 223-227, 1998.

[8] T. Baicheva, On the covering radius of ternary negacyclic codes with length up to 26, *IEEE Trans. on Inform. Theory*, vol. 47, No. 1, 2001, pp.413-416.

Algorithm for Computing the Similarity between Binary Vectors

Vladimir B. Balakirsky
"Confident" (Russia)
EIDMA (The Netherlands)
Email : v.b.balakirsky@tue.nl

Abstract — **The algorithm for computing the length of a common subsequence of two binary sequences of length n, which requires a variable number of computations depending on input data, is presented. The basis of the algorithm can be effectively used to obtain the probability distribution of the similarity between random vectors. This point is demonstrated for the case when one of vectors is chosen at random and another one is the alternating vector.**

Let $S(\mathbf{u},\mathbf{v})$ denote the length of the longest sequence occurring as a common subsequence between two given binary sequences, $\mathbf{u}$ and $\mathbf{v}$, of the same length n. The parameter $S(\mathbf{u},\mathbf{v})$ is called the deletion similarity between $\mathbf{u}$ and $\mathbf{v}$ [1], and $n-S(\mathbf{u},\mathbf{v})$ is called the Levenshtein distance between $\mathbf{u}$ and $\mathbf{v}$. This parameter is important for designing insertions–deletions correcting codes. It is also relevant to other applications in cryptology and data compression.

We will proceed with the following definition.

Definition 1. *The similarity* $S(\mathbf{u},\mathbf{v})$ *between two binary vectors,* $\mathbf{u} = (u_1,\ldots,\ u_n)$ *and* $\mathbf{v} = (v_1,\ldots,v_n)$, *is the largest integer* s *such that there exist two vectors,* $(i_1,\ldots,i_s)$ *and* $(j_1,\ldots,j_s)$ *with* $1 \le i_1 < i_2 < \ldots < i_s \le n$ *and* $1 \le j_1 < j_2 < \ldots < j_s \le n$, *such that* $u_{i_d} = v_{j_d}$ *for all* $d = 1,\ldots,s$. *The similarity between the all–zero vector and the all–one vector is defined as* 0.

The algorithms that allow us to compute $S(\mathbf{u},\mathbf{v})$ with the complexity $O(n^2)$ are known [2], [3]. We address the problem of simplification of these algorithms.

Let us denote $[n] \triangleq \{1,\ldots,n\}$. For $i = j = 0$ and for all $i,j \in [n]$ with $u_i = v_j$ introduce $p_{i,j}, q_{i,j} \ge 1$ as the smallest integers such that

$$u_{i+1} = v_{j+q_{i,j}},\ \ v_{j+1} = u_{i+p_{i,j}},$$

where we assume that $u_{i+1} = v_{n+1}$ and $v_{j+1} = u_{n+1}$.

N0. Set $s = 0$ and $\mathcal{N}_0 = \{(0,0)\}$.

N1. Form the sets

$$\mathcal{N}'_{s+1} = \bigcup_{(i,j)\in\mathcal{N}_s} \Big\{ (i{+}1, j{+}q_{i,j}),\ (i{+}p_{i,j}, j{+}1) \Big\}$$

and $\mathcal{N}_{s+1} \subseteq \mathcal{N}'_{s+1}$ in such a way that the pair $(i',j') \in \mathcal{N}'_{s+1}$ does not belong to the set $\mathcal{N}_{s+1}$ if and only if one of the following two conditions are satisfied : (a) $i' > n$ or $j' > n$; (b) there exists a pair $(i^*,j^*) \in \mathcal{N}'_{s+1}\backslash\{(i',j')\}$ either with $i^* < i', j^* = j'$ or with $j^* < j', i^* = i'$.

N2. If $\mathcal{N}_{s+1} \ne \emptyset$, then increase s by 1 and go to N1.

N3. Output s.

The N0–N3 algorithm is illustrated in Table 1.

A "direct algorithm" for finding the similarity between $\mathbf{u}$ and $\mathbf{v}$ can be presented as sequential computations of the elements of an $n \times n$ matrix $\mathbf{S}$ (for example, row by row) using the rules

$$\begin{aligned} u_i \ne v_j \ &\Longrightarrow\ S_{i,j} \triangleq S^*_{i,j}, \\ u_i = v_j \ &\Longrightarrow\ S_{i,j} \triangleq \max\Big\{ S^*_{i,j},\, S_{i-1,j-1}+1 \Big\}, \end{aligned}$$

where

$$S^*_{i,j} \triangleq \max\Big\{ S_{i-1,j},\, S_{i,j-1} \Big\}$$

Table 1: The sets constructed by the N0–N3 algorithm for the pair of vectors $(\mathbf{u},\mathbf{v}) = (001010, 110001)$; $S(\mathbf{u},\mathbf{v}) = 4$ and $C(\mathbf{u},\mathbf{v}) = 10$.

s	$\mathcal{N}_s$	$\mathcal{N}'_{s+1}$
0	(0,0)	(1,3), (3,1)
1	(1,3)	(2,4)
	(3,1)	(4,3), (5,2)
2	(2,4)	(3,6), (4,5)
	(4,3)	(5,6), (6,4)
	(5,2)	(6,3)
3	(3,6)	(4,7)
	(4,5)	(5,6)
	(6,3)	(7,4)
4	(5,6)	(6,7)

and

$$\text{either } i' = 0 \text{ or } j' = 0 \Longrightarrow S_{i',j'} \triangleq 0.$$

This algorithm is illustrated in Figure 1.

For all $i,j \in [n]$, let us denote

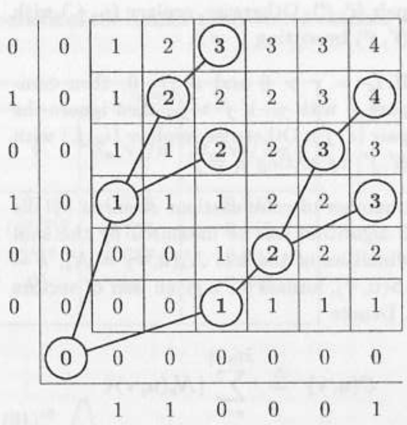
$$T_{i,j} \triangleq \begin{cases} S_{i,j}, & \text{if } S_{i,j} > S^*_{i,j}, \\ 0, & \text{otherwise.} \end{cases}$$

Furthermore, let $T_{0,0} \triangleq 0$. Introduce an oriented graph whose nodes are identified by the pair $(0,0)$ and by all pairs $(i,j) \in [n] \times [n]$ such that $T_{i,j} = S_{i,j}$. Define the edges of the graph as follows : there is an edge connecting the node (i,j) with the node (i',j') if and only if

$$T_{i',j'} = T_{i,j} + 1,\ i' > i,\ j' > j.$$

One can easily check that the graph above is a rooted oriented binary tree, which we call *the similarity tree* and illustrate the construction in Figure 1. The difference between the N0–N3 algorithm and "the direct algorithm" is that we immediately build a tree and form $S(\mathbf{u},\mathbf{v})$ as the largest value of $T_{i,j}$, where the maximum is taken on all pairs $(i,j) \in [n] \times [n]$.

Theorem 1. *For any pair of vectors* $(\mathbf{u},\mathbf{v})$, *the* N0–N3 *algorithm outputs the value of the similarity between* $\mathbf{u}$ *and* $\mathbf{v}$.



0	0	1	2	3	3	3	4
1	0	1	2	2	2	3	4
0	0	1	1	2	2	3	3
1	0	1	1	1	2	2	3
0	0	0	0	1	2	2	2
0	0	0	0	1	1	1	1
	0	0	0	0	0	0	0
		1	1	0	0	0	1

Figure 1: Illustration of the algorithm for computing the similarity between two vectors, $\mathbf{u} = 001010$ and $\mathbf{v} = 110001$, and corresponding similarity tree.

Notice that $p_{i,j}$ and $q_{i,j}$ can be easily found after $\mathbf{u}$ and $\mathbf{v}$ are represented as results of concatenations of the 0–packets and the 1–packets when the lengths of packets are computed in advance and stored in the memory. To "refine" the set $\mathcal{N}'_{s+1}$ at step N1(b) we also need the number of computations, which is linear in $|\mathcal{N}'_{s+1}|$. Namely, let $\mathbf{I} = (I_0,\ldots,I_n)$ and $\mathbf{J} = (J_0,\ldots,J_n)$ be two auxiliary arrays. Suppose that we have inspected some number of entries of the set $\mathcal{N}'_{s+1}$, formed the set $\mathcal{N}_{s+1}$ of cardinality M, and numbered its entries by $(i_1,j_1),\ldots,(i_M,j_M)$. Furthermore, let the arrays $\mathbf{I}$ and $\mathbf{J}$ be filled in such a way that

$$\begin{cases} I_{i_m} = J_{j_m} = m, & m = 1,\ldots,M \\ i \notin \{i_1,\ldots,i_M\} & \Longrightarrow\ I_i = 0 \\ j \notin \{j_1,\ldots,j_M\} & \Longrightarrow\ J_j = 0. \end{cases}$$

Then the inspection of the next entry $(i',j') \in \mathcal{N}'_{s+1}$ proceeds as follows.

1. If $I_{i'} = J_{j'} = 0$, then set $(i_{M+1}, j_{M+1}) = (i',j')$ and $I_{i'} = J_{j'} = M+1$. Increase M by 1.

2. If $I_{i'} = 0$ and $J_{j'} = r > 0$, then compare i' with i_r. If $i' > i_r$, then ignore the

pair (i', j'). Otherwise, replace (i_r, j_r) with (i', j') by setting $i_r = i'$.

3. If $I_{i'} = r > 0$ and $J_{j'} = 0$, then compare j' with j_r. If $j' > j_r$, then ignore the pair (i', j'). Otherwise, replace (i_r, j_r) with (i', j') by setting $j_r = j'$.

The number of computations required by the N0–N3 algorithm can be measured by the sum of cardinalities of the sets $\mathcal{N}_s(\mathbf{u}, \mathbf{v}) = \mathcal{N}_s$, $s = 0, \ldots, S(\mathbf{u}, \mathbf{v})$, formed for a given pair of vectors $(\mathbf{u}, \mathbf{v})$. Denote

$$C(\mathbf{u}, \mathbf{v}) \triangleq \sum_{s=0}^{S(\mathbf{u},\mathbf{v})} |\mathcal{N}_s(\mathbf{u}, \mathbf{v})|.$$

In Table 2 we show the average and the maximum number of computations in accordance with the function C:

$$\overline{C}_n \triangleq 2^{-2n} \sum_{\mathbf{u},\mathbf{v}} C(\mathbf{u}, \mathbf{v}),$$

$$C_{\max,n} \triangleq \max_{\mathbf{u},\mathbf{v}} C(\mathbf{u}, \mathbf{v}).$$

We also show the values of the average similarity $\overline{S}_n$ and the entropy H_n for the probability distribution $(P_n(0), \ldots, P_n(n))$ whose entries are defined as

$$P_n(s) \triangleq 2^{-2n} \Big| \Big\{ (\mathbf{u}, \mathbf{v}) : S(\mathbf{u}, \mathbf{v}) = s \Big\} \Big|,$$

for all $s = 0, \ldots, n$.

Theorem 2. *If $n + 1$ is divisible by 3, then the maximum number of computations can be expressed as*

$$C_{\max,n} = C(\bar{\mathbf{u}}, \bar{\mathbf{v}}) = \frac{(n+1)(2n+5)}{9} \approx \frac{2n^2}{9},$$

where

$$\bar{\mathbf{u}} \triangleq 1^{\ell}(01)^{(n-\ell-1)/2}0,$$
$$\bar{\mathbf{v}} \triangleq 0^{\ell-1}(10)^{(n-\ell+1)/2},$$

and $\ell \triangleq (n+1)/3$.

Table 2: Some parameters of the probability distribution of the similarity between two random binary vectors of length n.

n	$\overline{S}_n/n$	$H_n/\log n$	$\overline{C}_n/n$	$C_{\max,n}/n$
4	0.631	0.813	1.078	1.750
5	0.649	0.752	1.123	2.000
6	0.663	0.711	1.183	2.167
7	0.674	0.684	1.250	2.429
8	0.684	0.662	1.321	2.625
9	0.691	0.646	1.395	2.889
10	0.698	0.632	1.470	3.100
11	0.704	0.621	1.547	3.364
12	0.708	0.612	1.624	3.583
13	0.713	0.604	1.701	3.846
14	0.717	0.596	1.778	4.071
15	0.720	0.590	1.856	4.333
16	0.724	0.585	1.933	4.562

Finding of the analitical expression for the probability distribution of the similarity seems to be a difficult combinatorial problem. One of approaches to attack this problem is the finding of the probability distribution of the similarity between a uniformly distributed random vector $\mathbf{u}$ and a vector $\mathbf{v}$, which is fixed in a regular way. This distribution highly depends on $\mathbf{v}$, as it is illustrated in Table 3 and Figure 2 where we show the values of the average similarity $\overline{S}_n(\mathbf{v})$ and the entropy $H_n(\mathbf{v})$ for the probability distribution $(P_n(0|\mathbf{v}), \ldots, P_n(n|\mathbf{v}))$ whose entries are defined as

$$P_n(s|\mathbf{v}) \triangleq 2^{-n} \Big| \Big\{ \mathbf{u} : S(\mathbf{u}, \mathbf{v}) = s \Big\} \Big|,$$

for all $s = 0, \ldots, n$.

Theorem 3. *Suppose that n is even and denote $\mathbf{v}_2 \triangleq (01)^{n/2}$. The similarity between $\mathbf{v}_2$ and any binary vector $\mathbf{u}$ of length n having $k(\mathbf{u})$ packets can be expressed as*

$$S(\mathbf{u}, \mathbf{v}_2) = \begin{cases} n/2 + \lceil (k(\mathbf{u}) - 1)/2 \rceil, & \text{if } u_1 = 0, \\ n/2 + \lfloor (k(\mathbf{u}) - 1)/2 \rfloor, & \text{if } u_1 = 1, \end{cases}$$

where

$$k(\mathbf{u}) \triangleq 1 + \Big| \Big\{ i \in \{2, \ldots, n\} : u_i \neq u_{i-1} \Big\} \Big|.$$

Table 3: Some parameters of the probability distributions of the similarity between a random vector $\mathbf{u}$ and a fixed vector $\mathbf{v}$ of length n.

$n = 16$			$n = 8$		
$\mathbf{v}$	$\overline{S}_n(\mathbf{v})/n$	$H_n(\mathbf{v})/\log n$	$\mathbf{v}$	$\overline{S}_n(\mathbf{v})/n$	$H_n(\mathbf{v})/\log n$
$(01)^8$	0.734	0.522	$(01)^4$	0.718	0.543
$(0^2 1^2)^4$	0.749	0.502	$(0^2 1^2)^2$	0.714	0.542
$(0^4 1^4)^2$	0.708	0.519	$0^4 1^4$	0.637	0.601
$0^8 1^8$	0.621	0.587	0^8	0.5	0.848
0^{16}	0.5	0.761			

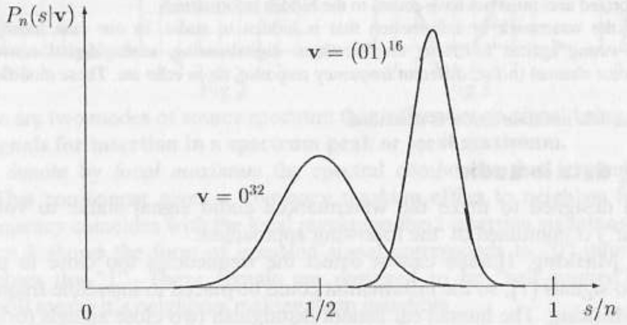


Figure 2: The probability distributions of the similarity between a random vector $\mathbf{u}$ of length $n = 32$ and the vectors $\mathbf{v} = 0^{32}, (01)^{16}$.

Corollary.

$$P_n(s|\mathbf{v}_2) = \binom{n+1}{2s-n} 2^{-n}, \ \textit{for all } s = n/2, \ldots, n$$

and

$$P_n(s|\mathbf{v}_2) = 0, \ \textit{for all } s < n/2.$$

Thus,

$$\overline{S}_n(\mathbf{v}_2) = \frac{3n-1}{4}.$$

References

[1] V. I. Levenshtein, "Binary codes capable to correct deletions, insertions, and reversals", *Doklady Akademii Nauk SSSR*, vol. 163, pp. 845–848, 1965. (in Russian). English translation : *J. Soviet Phys.–Doklady*, vol. 10, pp. 707–710, 1966.

[2] R. A. Wagner and M. J. Fischer, "The string–to–string correction problem", *Journal of the ACM*, vol. 21, no. 1, pp. 168–173, 1974.

[3] W. J. Masek, M. S. Paterson, "How to compute string–edit distance quickly", in : *Time Warps, String Edits, and Macromolecules : The Theory and Practice of Sequence Comparison*, D. Sankoff and J. B. Kruskal (eds.), Addison–Wesley, pp. 337–349, 1983.

Hidden data transmission over the voice channel

Andrey Belogolovy
St-Petersburg State University of Airspace Instrumentation,
Bolshaja Morskaja str, 67, St-Petersburg, 190000, Russia
E-mail: andrew@guap.delfa.net

Hidden data transmission in audio signals is a steganographical term that describes the algorithm of inaudible method of insertion data in audio, i.e. human cannot detect the presence of modifications, but there is a way to extract hidden data anyway [2], [3], [4]. Transmission over the voice channel means that before the extraction of hidden data we will transmit the modified audiosignal from speakers to microphone.

The steganographical method of insertion data in audio signal for transmission over the voice channel declares:

- possibility to recover hiding data from audio without original signal
- an unauthorized user must not be able to detect the presence of hidden information
- an unauthorized user must not have access to the hidden information

Robustness of the watermark or information that is hidden in audio, in our case means that hidden information should be strong against following modifications: digital/analog, analog/digital conversations and transmission over the voice channel (noise, different frequency response, room echo etc. These modifications would be described later)

In this paper the data insertion model is described.

Inaudible hiding data in audio

The method designed to make the watermarked audio signal stable to voice channel (transmission "by air") is combined of the following approaches:

Frequency Masking. Human cannot detect the frequencies too close to peaks, local maxima etc. in audio signals [1], so the information could be placed in inaudible frequencies.

Temporal Masking. The human ear cannot distinguish two close signals (original and it echo) - the echo is perceived as added resonance. [4]

Direct Sequence Spread Spectrum. The way of using special noise like signal as additional information signal. To keep the noise level low and inaudible, the spread code is attenuated to nearly 0.5% of the dynamic range of the host sound signal.

Transmission over the voice channel

"Voice channel" is a channel produced by audio speaker and a microphone. "Transmission over the voice channel" is the term that we use to describe the transmission of audio file as a sound from PC audio speaker to PC-microphone and then its conversion to digital form (ADC). It's clear that the file transmitted over the voice channel differs from the original.

The main known modifications of sound during transmission over the voice channel are: additive noise, phase shift (room effect, echo), changing of amplitude-frequency response, random broad-spectrum interference.

The effects of all modifications described are: unequal distortions of different spectral parts of audio signal; peaks and falls of power spectra could not match in transmitted and received signals. So, there is no way to use frequency spectrum components amplitudes for direct value storing and no way to use peaks and falls in spectra as delimiters or binders.

To solve this problem the fixed-frequencies binding is used. The decoder should know the frequency component for every frame of signal that stores the information. The number of component depends on the frequency response of speakers and microphone being used. For standard microphone the frequency components for insertion belongs to interval [3.5:6.0] kHz.

By *insertion frequency* for the frame we mean the number of frequency component that will be the binder for the frame, i.e. the insertion frequency is the component in which the center of insertion signal would be placed. The detector would find the insertion signal center only in insertion frequency.

The information bits are represented as special insertion signals that have information in their form factor, not amplitude values. To restore synchronization the pseudo-noise sequence (direct sequence spread spectrum approach) is used.

Insertion signals

Lets denote by *insertion signals* the arrays of 7 elements to be inserted in frequency spectrum of the one frame of audio signal. The form of insertion signal depends on information bit and on form of source signal spectrum that corresponds to insertion frequency.

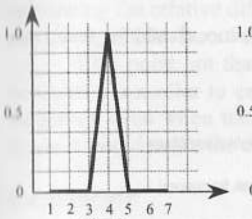


Fig 1.

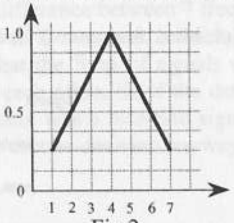


Fig 2

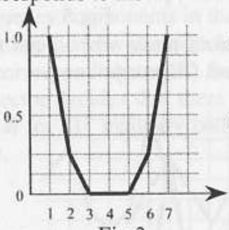


Fig 3

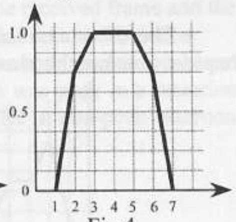


Fig 4

There are two modes of source spectrum that influences on signal being inserted:

1. Signals for insertion in a spectrum peak or local maximum.

Lets denote by *local maximum* the spectral component that is greater than 2 of it's neighbors. This component gives a frequency masking effect to neighbor frequencies. If the insertion frequency coincides with the local maximum the 2 insertion signals are used:

Figure 1 shows the form of insertion signal corresponds to "0" information bit value, Figure 2 shows the "1". These signals are designed to be "self-masked", i.e. the central component still masks it's neighbors making them inaudible.

2. Signals for insertion in non-peak frequencies

By non-peak frequencies we mean the frequency components that give no frequency masking effect to neighbor frequency components. Figure 3 shows the form of insertion signal corresponds to "0" information bit value, Figure 4 shows the "1".

The signals of this pair have to be more different from each other and at the same time "1"'s signal should be look like "1"'s signal for insertion in peak case. The purpose of that would be explained later.

Data insertion process

At first, the coder places the synchro stamp in time area of the signal using direct sequence spread spectrum. It means adding a pseudo-noise sequence with low magnitude but good correlation attributes to make able set up synchronization in the receiving side.

The source audio signal is then split to frames of length 512, than the spectral transformation (FFT) is applied to every frame. Lets describe the insertion progress for one frame in frequency representation meaning that after modifications the invert spectral transformation (IFFT) would be applied and the new audio signal would be reconstructed from separate frames.

After having the frequency frame the coder should decide which insertion mode to use. It analyzes the frequency amplitudes near to insertion frequency. If there are peaks (local maxima) in the range of [insertion frequency-3; insertion frequency+3] than coder switches to "insertion in peak" mode, otherwise it use "non-peak insertion" mode.

1. Insertion in peak mode.

In the best case the insertion frequency coincides with the local maxima. In that case coder selects the information signal from signals for insertion in a spectrum peak corresponding to value of information bit (1 or 0) and scales the signal to be the same height as a peak value (Figure 5).

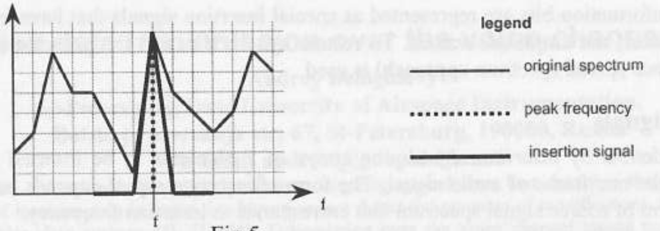


Fig 5.

The other case takes place when peak (or local maximum) is too closed to insertion frequency but not the same. (Figure 6)

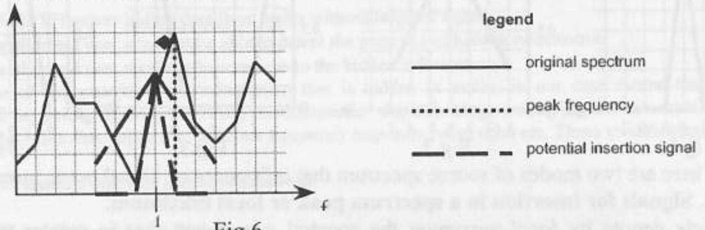


Fig 6.

In this case if the coder tries to insert signal in frequency f_i it would cut the peak and the change become audible. The solution is to move peak value to be at the frequency f_i, this change is inaudible if peak movement described occurs only at one frame in succession, i.e. the peak movement is allowed only to one frame is sequence.

After the movement we have the situation like in case of figure 5, so coder scales insertion signal up to peak value and inserts it.

Lets point out that due to peak moving the insertion frequency f_i should be different for every consequent frames. To do it lets select f_i pseudo randomly, and make this pseudo-random rule known to insertion and extraction processes both.

2. Insertion in non-peak mode.

Lets denote by frequency *masking threshold* the threshold that describes the inaudibility of spectral frequency component if it's value is less than masking threshold. Masking envelope is the curve of masking thresholds for every frequency components.

The coder builds masking envelope for the range of frequencies neighbor to insertion frequency. Than it selects the information signal from signals for insertion in non-peak corresponding to value of information bit (1 or 0) and scales the signal to make it's maximum values equal to values of masking envelope for corresponding frequencies (Figure 7).

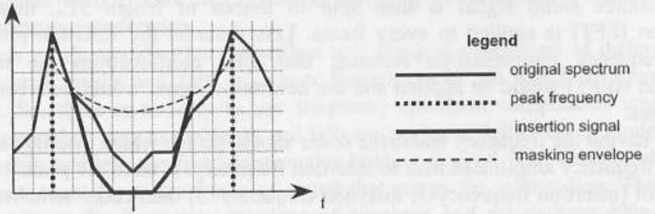


Fig 7.

Detection process

After detecting a synchro stamp (pseudo-noise sequence which was added to source signal) and setting up synchronization the audio stream is split to frames and make a spectral transformation as the coder has done. For every frame detection process tries to decide if there is a peak in insertion frequency. So there is two detection schemes.

1. Detector decides that there was a peak in insertion frequency

The detector should distinguish the signals for insertion in peak. This could be done by analyzing the relative square of the figure bounded with 7 frequency components neighbor to insertion frequency.

2. Detector decides that there was no peak

The detector should distinguish the signals for insertion in non-peak. It could be done by calculating the relative difference between 7 frequency components in the received frame and the pair of signals for insertion in non-peak and selecting the bit value with minimum difference.

Lets point out that the form of signals corresponding to "1" for both modes (peak and non-peak) is similar to each other. So if the detector decides that there was peak in a insertion frequency even when there was a insertion signal for "1" made by coder in non-peak insertion mode it would make the correct decision anyway.

Conclusion

The novelty of method described is that the method is oriented to voice channel, i.e. it could be stable to transmission of signal with hidden data "by air".

Transmission data over the voice channel using the method described above is not reliable itself because of difference in transmitted sound and received one. The detector could make the wrong decisions and the transmitted and received data however would not match. So the whole model could be considered as a upper-level channel with errors in it.

The fact is that the errors arising in upper-level channel are non-symmetrical, i.e. the probability of transition "0-1" is higher then probability of transition "1-0". So implementation of standard well-known codes for correcting symmetrical errors (BCH etc) would not give good results. Now the author works with task of building a special non-symmetrical error oriented code.

References

1. Information technology -- Coding of moving pictures and associated audio for digital storage media at up to 1,5 Mbits/s -- Part3: audio. British standard. BSI, London. October 1993. Implementation of ISO/IEC 11172-3:1993. BSI, London. First edition 1993-08-01.

2. Laurence Boney, Ahmed H. Tewfik and Khaled N. Hamdy , "Digital Watermarks for Audio Signals" in IEEE Int. Conf. On Multimedia Computing and Systems,(Hiroshima, Japan), June 1996.

3. P.Noll, "Wideband speech and audio coding", IEEE Communications Magazine, Nov. 1993, pp.34-44.

4. Gruhl,D., Lu, A., Bender, W.: Echo hiding. In: Anderson, R. (ed.): Information hiding, Cambrige, UK. Lecture Notes in Computer Science, Vol.1174. Springer-Verlag, Berlin/Heidelberg (1996) 295-315.

Isometries for rank distance and permutation group of Gabidulin codes

Thierry P. Berger

LACO, Université de Limoges,
123 av. A. Thomas, 87060 Limoges CEDEX, FRANCE
e-mail: thierry.berger@unilim.fr tel: (33) 5 55 45 73 38

Abstract *The rank distance was introduced in 1985 by E. Gabidulin [1]. He determined a lower bound for the minimum rank distance of a code. Moreover, he constructed a class of codes which meet this bound: the so-called Gabidulin codes.*
In this paper, we first characterize the linear isometries for the rank distance. Then we determine the isometry group and the permutation group of Gabidulin codes of full length (i.e. the length is equal to the degree of the field extension).

I. Isometries for rank distance

Let $K = GF(q^m)$ be an extension of degree m of the finite field $GF(q)$. Let $E = K^n$ be the vector space of dimension n over K.

Definition 1 *For $a \in E$, $a = (a_1, \dots, a_n)$, the rank $rk(a)$ of a is the dimension of the $GF(q)$-vector space generated by $\{a_1, \dots, a_n\}$.*

Let a and b be two elements of E. The relation $d_r(a,b) = rk(a-b)$ defines a distance over E. Following this definition, it is natural to define the minimum rank distance d_r of a code C. Moreover, if d_h denotes the classical Hamming distance, then for all a, b in E, the rank distance satisfies the inequality $d_r(a,b) \leq d_h(a,b)$.

The group of linear isometries for the classical Hamming distance is well-known: it is the monomial group of $n \times n$ matrices over K with one and only one non-zero element on each row and each column [2]. This group is generated by the permutations of the support and the scalar multiplications by invertible elements on each coordinate.

In this section, we characterize the linear transformations that are isometries for the rank distance.

Definition 2 *An isometry for the rank distance is a K-linear automorphism f of E which preserves the rank of the elements of E, i.e. $rk(a) = rk(f(a))$ for all a in E.*

Let $Iso(E)$ be the group of isometries for the rank distance. The following facts are very easy to check:

- The scalar multiplications $h_\lambda : a = (a_1, \dots, a_n) \mapsto \lambda a = (\lambda a_1, \dots, \lambda a_n)$, $\lambda \in GF(q^m)^*$ are isometries for the rank distance.
- For all $M \in GL(n,q)$, the K-linear endomorphism f_M of E defined by $a \mapsto aM$ is an isometry for the rank distance.

The following theorem characterizes the isometries for the rank distance.

Theorem 1 *The isometry group $Iso(E)$ for the rank distance is generated by the scalar multiplications h_λ, $\lambda \in GF(q^m)^*$ and the linear group $GL(n,q)$. This group is isomorphic to the product group $(GF(q^m)^*/GF(q)^*) \times GL(n,q)$.*

Proof : As noticed previously, the scalar multiplications and the transformations associated to $n \times n$ invertible matrices with coefficients in $GF(q)$ are isometries for the rank distance.

Let $f \in GL(n, q^m)$ be an invertible K-linear transformation, and M be its associated matrix in the canonical basis. The i-th row of M is the image of e_i by f.

Suppose that f is an isometry for the rank distance. The rank of each row must be one. Moreover, eventually using a scalar multiplication, it is possible to suppose that the elements of the first row are in $GF(q)$, i.e. $f(e_1) \in GF(q)^n$.

Let $i \in \{2, \dots, n\}$. Following the preceding remarks, there exists a $\mu \in K^*$ such that $\mu^{-1} f(e_i)$ is in $GF(q)^n$. This is $\mu^{-1} f_{i,j} \in GF(q)$ for all $j = 1, \dots, n$.

Let c be $e_1 + e_i$. The rank of c is 1. Its image is $f(c) = (f_{1,1} + f_{i,1}, f_{1,2} + f_{i,2}, \dots, f_{1,n} + f_{i,n})$ and must be of rank 1. There exists at least one non-zero coordinate, for example the first. Set $\nu = f_{1,1} + f_{i,1} \neq 0$. Since the rank of $f(c)$ is 1, for a fixed j there exists a $s \in GF(q)$ such that $f_{1,j} + f_{i,j} = s\nu$, i.e. $f_{1,j} + f_{i,j} = s(f_{1,1} + f_{i,1})$.

From this fact, we deduce $f_{1,j} - sf_{1,1} = -f_{i,j} + sf_{i,1}$. Set $t = f_{1,j} - sf_{1,1}$. This is an element of $GF(q) \cap \mu GF(q)$. Then either μ is in $GF(q)$ and the elements of the i-th row are in $GF(q)$, or $t = 0$, that implies $f_{1,j} = sf_{1,1}$ for all j: the i-th row is deduced from the first by multiplication by s. This is not possible, since the matrix M is invertible. This proves the fact that al the $f_{i,j}$ are in $GF(q)$ and f is in $GL(n,q)$.

To complete the proof, we first remark that the scalar multiplications h_λ commute with all the linear transformations. Moreover, the intersection of the linear group $GL(n,q)$ and the group of scalar multiplications is the subgroup of scalar multiplications for which λ is in $GF(q)^*$. This implies that $Iso(E)$ is isomorphic to the direct product $(K^*/GF(q)^*) \times GL(n,q)$. □

II. Gabidulin codes

In this paragraph, we restrict ourselves to Gabidulin codes of full length, i.e. of length $n = m$, where m is the degree of extension of $K = GF(q^m)$ over the base field $GF(q)$. The Gabidulin codes where introduced in [1]. These codes are MRD (Maximum Rank Distance): they meet the best possible rank distance $d_r = n + 1 - k$, where k is the dimension of the code.

We first recall the definition and the main properties needed in the next section. Let $g = (g_1, \dots, g_m)$ be a basis of K over $GF(q)$. Let $\mathcal{G}_{k,g}$ be the matrix defined by

$$\mathcal{G}_{k,g} = \begin{pmatrix} g_1 & g_2 & \cdots & \cdots & g_m \\ g_1^{[1]} & g_2^{[1]} & \cdots & \cdots & g_m^{[1]} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ g_1^{[k-1]} & g_2^{[k-1]} & \cdots & \cdots & g_m^{[k-1]} \end{pmatrix}$$

with the convention $a^{[i]} = a^{q^i}$.

Definition 3 *The Gabidulin code of dimension k relatively to the basis g is the code $G_{k,g}$ of length m over K generated by the matrix $\mathcal{G}_{k,g}$.*

In [1], E.M. Gabidulin proved that these codes are MRD.

Now, we present a more precise characterization of the dual of a Gabidulin code under their restriction that the length is exactly m.

Let $g = (g_1, \ldots, g_m)$ be a basis of K over $GF(q)$, and $h = (h_1, \ldots, h_m)$ be the trace-orthogonal basis of g: h is the unique basis such that $Tr(g_i h_j) = \sum_{\ell=0}^{m-1} g_i^{[\ell]} h_j^{[\ell]} = \delta_{i,j}$.

Clearly, this relation is equivalent to $\mathcal{G}_{m,g}\mathcal{G}^t_{m,h} = I$.

Lemma 1 *The dual of the Gabidulin code $G_{k,g}$ is the Gabidulin code $G_{m-k,h^{[k]}}$.*

Proof : The Gabidulin code $G_{m-k,h^{[k]}}$ is the dual of $G_{k,g}$ if and only if $< g^{[i]}, h^{[j]} >= 0$ for all $i = 0, \ldots,k-1$ and all $j = k, \ldots,m-1$. The result follows from the relation $\mathcal{G}^t_{m,h}\, \mathcal{G}_{m,g} = (< g^{[i]}, h^{[j]} >)_{i,j} = \mathcal{G}_{m,g}\mathcal{G}^t_{m,h} = I$. □

Proposition 1 *Suppose $k < m$. Let $G_{k,g}$ be a Gabidulin code and h be the trace-orthogonal basis associated to g. A Gabidulin code $G_{m-k,h'}$ is the dual of $G_{k,g}$ if and only if there exists a scalar $a \in K^*$ such that $h' = ah$.*

Proof : If the Gabidulin code $G_{m-k,h'}$ is the dual of $G_{k,g}$ then $< g^{[i]}, h'^{[j]} >= 0$ for all $i = 0, \ldots,k-1$ and all $j = 0, \ldots,m-k-1$. These relations are equivalent to $< g^{[i-j]}, h' >= 0$ for all $i = 0, \ldots,k-1$ and all $j = 0, \ldots,m-k-1$, i.e. $< g^{[s]}, h' >= 0$ for all $s \in [0, m[\backslash\{k\}$. Set $a =< g^{[s]}, h' >$.

Our conditions become $\mathcal{G}_{m,g}h'^t = ae^t_k$. The matrix $\mathcal{G}_{m,g}$ is invertible, and then the solution is unique. Moreover, $\mathcal{G}_{m,g}(h^{[k]})^t = e^t_k$, and then $h' = ah^{[k]}$. Clearly $a \neq 0$, since h' is not 0.

The reverse part of this proposition is trivial. □

From this proposition, we can deduce a characterization of the distinct bases who give the same Gabidulin code.

Theorem 2 *Suppose $k < m$. Two Gabidulin codes $G_{k,g}$ and $G_{k,g'}$ are equal if and only if if there exists a scalar $a \in K^*$ such that $g' = ag$.*

Proof : This result is the previous proposition applied to the dual $G_{m-k,h}$ of the Gabidulin code $G_{k,g}$. □

Remark: This result cannot be directly extended to Gabidulin codes of length n less than m.

III. Isometry group and permutation group of Gabidulin codes

The scalar multiplication $h_\lambda : (c_1, \ldots, c_m) \mapsto (\lambda c_1, \ldots, \lambda c_m)$, $\lambda \in K^*$, is clearly an element of the isometry group of every K-linear code of E.

Let $M \in GL(m, q)$ be a $m \times m$ invertible matrix with coefficients in $GF(q)$ and $f \in Iso(E)$ be its associated isometry.

Lemma 2 *Let $g = (g_1, \ldots, g_m)$ be a basis of K over $GF(q)$. Let $g' = (g'_1, \ldots, g'_m) = gM$ be its image by f. The image of the Gabidulin code $G_{k,g}$ by f is the Gabidulin code $G_{k,g'}$.*

Proof : Since the coefficient of M are in $GF(q)$, the image of the i-th row $g^{[i-1]}$ by f is $g'^{[i-1]}$. The image of the basis $(g^{[0]}, \ldots, g^{[k-1]})$ of $G_{k,g}$ by f is then the basis $(g'^{[0]}, \ldots, g'^{[k-1]})$ of $G_{k,g'}$. □

Theorem 3 *Let $1 \leq k < m$. The isometry group of the Gabidulin code $G_{k,g}$ is the group of scalar multiplications, isomorphic to K^*.*

Proof : The isometry group $Iso(G_{k,g})$ of the Gabidulin code $G_{k,g}$ contains the scalar multiplications.

Reciprocally, let $f \in Iso(E)$ be an element of $Iso(G_{k,g})$. Recall that $Iso(E)$ is generated by the scalar multiplications and the linear group $GL(m, q)$. Multiplying eventually f by a scalar λ, we can suppose that f is in $GL(m, k)$. Let $g' = f(g)$. If A is the matrix associated to f, this gives $\mathcal{G}_{m,g'} = \mathcal{G}_{m,g}A$, and then $A = \mathcal{G}^{-1}_{m,g}\mathcal{G}_{m,g'}$.

Moreover, using Lemma 2 and Theorem 2, there exists a scalar $a \in K^*$ such that $g' = ag$. This implies $g'^{[i]} = a^{[i]}g^{[i]}$ for all $i = 0, \ldots,m-1$. Let A' be the diagonal matrix such that $a'_{i,i} = a^{[i]}$. We obtain $\mathcal{G}_{m,g'} = \mathcal{G}_{m,g}A'$, and then $A = A'$. Recall that the elements of A are in $GF(q)$, then $a^{[i]} = a$ for all i and the transformation f is the scalar multiplication by $a \in GF(q)^*$. This completes the proof. □

Corollary 1 *For $1 \leq k < m$, the permutation group of the Gabidulin code $G_{k,g}$ is trivial, i.e. it contains only the identity.*

Proof : The permutation group is a subgroup of $Iso(G_{k,g})$. The only scalar multiplication which is a permutation is the identity. □

Conclusion In this paper, we characterized the isometries for the rank distance. Using this property, we have been able to find the permutation group and the isometry group of Gabidulin codes.

However, the problem of the classical automorphism group of Gabidulin codes for the Hamming distance remains open.

For example, if $k = 1$, it is easy to construct a permutation followed by appropriated scalar multiplications on each component who leaves the code globally invariant.

We never obtained such a non-trivial example for $2 \leq k < m-1$.

References

[1] E.M. Gabidulin: *Theory of codes with maximum rank distance*,
Problemy Peredachi Informatsii, vol.21, n.1, p. 1-12, 1985.

[2] W.C. Huffman: *Groups and codes*,
In V.S. Pless & W.C. Huffman Editors, Handbook of Coding Theory, Vol.II, Chapter 17, Amsterdam: Elsevier 1998.

An application of $\{\delta(q+1), \delta; n+1, q\}$-minihypers on generalized quadrangles

J. De Beule
(joint work with **M. R. Brown** and **L. Storme**)
Ghent University
Dept. Pure Mathematics and Computer Algebra, Krijgslaan 281
B 9000 Gent, Belgium
jdebeule@cage.rug.ac.be

Abstract

Minihypers in finite projective spaces have been used greatly to study the problem of linear codes meeting the Griesmer bound; thereby showing their importance for coding theory. But they are also important for a great variety of geometrical problems. Using the classification of $\{\delta(q+1), \delta; n+1, q\}$-minihypers we obtain results on spreads of certain finite generalized quadrangles. We discuss both the application and the result.

1 Introduction

In this section we introduce the concept of a generalized quadrangle, or shortly, a GQ.

Definition 1.1 *A (finite)* generalized quadrangle *(GQ) is an incidence structure* $\mathcal{S} = (\mathcal{P}, \mathcal{B}, \mathrm{I})$ *in which* $\mathcal{P}$ *and* $\mathcal{B}$ *are disjoint non-empty sets of objects called* points *and* lines *(respectively), and for which* $\mathrm{I} \subseteq (\mathcal{P} \times \mathcal{B}) \cup (\mathcal{B} \times \mathcal{P})$ *is a symmetric point-line incidence relation satisfying the following axioms:*

(i) Each point is incident with $1+t$ *lines* $(t \geqslant 1)$ *and two distinct points are incident with at most one line.*

(ii) Each line is incident with $1+s$ *points* $(s \geqslant 1)$ *and two distinct lines are incident with at most one point.*

(iii) If x *is a point and* L *is a line not incident with* x*, then there is a unique pair* $(y, M) \in \mathcal{P} \times \mathcal{B}$ *for which* $x \,\mathrm{I}\, M \,\mathrm{I}\, y \,\mathrm{I}\, L$*.*

The integers s *and* t *are the* parameters *of the GQ and* $\mathcal{S}$ *is said to have order* (s,t)*. If* $s = t$*, then* $\mathcal{S}$ *is said to have order* s*.*

Examples: Classical examples are the symplectic space $\mathrm{W}_3(q)$ in $\mathrm{PG}(3,q)$, the hyperbolic quadric $\mathrm{Q}^+(3,q)$ in $\mathrm{PG}(3,q)$, the parabolic quadric $\mathrm{Q}(4,q)$ in $\mathrm{PG}(4,q)$, the elliptic quadric $\mathrm{Q}^-(5,q)$ in $\mathrm{PG}(5,q)$, and the Hermitian varieties $\mathrm{H}(3,q^2)$ and $\mathrm{H}(4,q^2)$ in $\mathrm{PG}(3,q^2)$ and $\mathrm{PG}(4,q^2)$, which have respectively order q, $(q,1)$, q, (q,q^2), (q^2,q) and (q^2,q^3). The non-classical examples of Tits are given in the following definition:

Definition 1.2 *Let* $n = 2$ *(respectively,* $n = 3$*) and let* $\mathcal{O}$ *be an oval (respectively, an ovoid) of* $\mathrm{PG}(n,q)$*. Furthermore, let* $\mathrm{PG}(n,q)$ *be embedded as a hyperplane in* $\mathrm{PG}(n+1,q)$*. Define points as*

(i) the points of $\mathrm{PG}(n+1,q) \setminus \mathrm{PG}(n,q)$*,*

(ii) the hyperplanes X *of* $\mathrm{PG}(n+1,q)$ *for which* $|X \cap \mathcal{O}| = 1$*, and*

(iii) one new symbol (∞)*.*

Lines are defined as

(a) the lines of $\mathrm{PG}(n+1,q)$ *which are not contained in* $\mathrm{PG}(n,q)$ *and which meet* $\mathcal{O}$ *(necessarily in a unique point), and*

(b) the points of $\mathcal{O}$*.*

Incidence is inherited from $\mathrm{PG}(n+1,q)$*, whereas the point* (∞) *is incident with no line of type (a) and with all lines of type (b).*

It is straightforward to show that these incidence structures are GQ's with parameters $s = q$, $t = q^{n-1}$.

Definition 1.3 *A* spread *of a GQ* $\mathcal{S}$ *of order* (s,t) *is a set* S *of lines such that every point of* $\mathcal{S}$ *is incident with exactly one element of* S*. A spread necessarily contains* $1+st$ *lines. A* partial spread *is a set* S *of lines for which every point is incident with at most one line of* S*. A partial spread is called* maximal *if* S *is not contained in a larger partial spread. If the size of a partial spread is* $1+st-\delta$*, then* δ *is said to be the* deficiency *of the partial spread.*

The natural question is whether a partial spread with certain deficiency can be maximal, or, in other words, can a partial spread with small deficiency be extended? Using minihypers we can give answers to this question for partial spreads of the GQ's $T_2(\mathcal{O})$ and $T_3(\mathcal{O})$.

2 The minihypers

Definition 2.1 *An* $\{f, m; N, q\}$*-minihyper is a pair* (F, w)*, where* F *is a subset of the point set of* $\mathrm{PG}(N,q)$ *and where* w *is a weight function* $w\colon \mathrm{PG}(N,q) \rightarrow \mathbb{N}\colon x \mapsto w(x)$*, satisfying*

1. $w(x) > 0 \iff x \in F$,
2. $\sum_{x\in F} w(x) = f$, and
3. $\min\{\sum_{x\in H} w(x) \| H \in \mathcal{H}\} = m$, where $\mathcal{H}$ is the set of hyperplanes of $\mathrm{PG}(N,q)$.

Related to certain minihypers are blocking sets of $\mathrm{PG}(2,q)$. The following theorem about blocking sets is used for the final theorem.

Theorem 2.2 (A. Blokhuis, L. Storme, and T. Szőnyi [3]) *Let B be a blocking set in $\mathrm{PG}(2,q)$, $q = p^h$, p prime, of size $q+1+c$. Let $c_2 = c_3 = 2^{-1/3}$ and $c_p = 1$ for $p > 3$.*

1. *If $q = p^{2d+1}$ and $c < c_p q^{2/3}$, then B contains a line.*
2. *If $4 < q$ and q is a square and $c < c_p q^{2/3}$, then B contains a line or a Baer subplane.*

To establish the connection between blocking sets and certain minihypers we need one more definition.

Definition 2.3 *Let $\mathcal{A}$ be the set of all lines of $\mathrm{PG}(N,q)$. A sum of lines is a weight function w: $\mathcal{A} \to \mathbb{N}$: $L \mapsto w(L)$. A sum of lines induces a weight function on the points of $\mathrm{PG}(N,q)$, which is given by $w(x) = \sum_{L\in\mathcal{A}, x\in L} w(L)$. In other words, the weight of a point is the sum of the weights of the lines passing through that point. A sum of lines is said to be a sum of n lines if the sum of all the weights of the lines is n.*

The connection is finally expressed in the following theorem, which will be of direct use for our application

Theorem 2.4 (Govaerts and Storme [1]) *Let (F,w) be a $\{\delta(q+1),\delta;N,q\}$-minihyper, $q > 2$, satisfying $0 \leqslant \delta < \epsilon$, where $q+\epsilon$ is the size of the smallest non-trivial blocking set in $\mathrm{PG}(2,q)$. Then w is a weight function induced on the points of $\mathrm{PG}(N,q)$ by a sum of δ lines.*

3 The application

Considering an arbitrary partial spread of $T_n(\mathcal{O})$, we will define an $\{\delta(q+1),\delta;n+1,q\}$-minihyper.

Definition 3.1 *Let S be a partial spread of a GQ. A hole with respect to S is a point of the GQ which is not incident with any line of S.*

Consider a partial spread S of $T_n(\mathcal{O})$, $n = 2$ or $n = 3$, of size $q^n + 1 - \delta$. Referring to the definition of the GQ $T_n(\mathcal{O})$, let $\pi_0 = \mathrm{PG}(n,q)$ which contains $\mathcal{O}$ and which is embedded in $\mathrm{PG}(n+1,q)$ as a hyperplane. We remark that a partial spread contains at most one line of type (b) of the GQ, because all lines of type (b) intersect in (∞).

Definition 3.2 *Let S be a partial spread of $T_n(\mathcal{O})$ ($n = 2$ or $n = 3$). Define w_S: $\mathrm{PG}(n+1,q) \to \mathbb{N}$ as follows:*

(i) if $x \in \mathrm{PG}(n+1,q) \setminus \pi_0$ and x is a hole with respect to S, then $w_S(x) = 1$, otherwise $w_S(x) = 0$,

(ii) suppose $x \in \mathcal{O}$, define $w_S(x) = \delta_x$, with $q - \delta_x$ the number of lines of S through x.

(iii) $w_S(x) = 0, \forall x \in \pi_0 \setminus \mathcal{O}$.

This weight function determines a set F of points of $\mathrm{PG}(n+1,q)$. We will denote the defined minihyper by (F, w_S).

We can now prove

Lemma 3.3 *Let S be a partial spread of $T_n(\mathcal{O})$ ($n = 2$ or 3) which covers (∞) and which has deficiency $\delta < q$. Then w_S is the weight function of a $\{\delta(q+1),\delta;n+1,q\}$-minihyper (F, w_S).*

This lemma leads immediately to

Theorem 3.4 (M.R. Brown, J. De Beule and L. Storme [2]) *Let S be a partial spread with deficiency δ of $T_n(\mathcal{O})$ ($n = 2$ or 3) covering (∞). If $\delta < \epsilon$, with $q + \epsilon$ the size of the smallest non-trivial blocking set in $\mathrm{PG}(2,q)$, $q > 2$, we can always extend S to a spread.*

References

[1] P. Govaerts and L. Storme. On a particular class of minihypers and its applications. I. The result for general q. *Des. Codes Cryptogr.*, accepted.

[2] M.R. Brown, J. De Beule and L. Storme. Partial spreads of $T_2(\mathcal{O})$ and $T_3(\mathcal{O})$. *European J. Combin.*, accepted.

[3] A. Blokhuis, L. Storme and T. Szőnyi. Lacunary polynomials, multiple blocking sets and Baer subplanes. *J. London Math. Soc. (2)*, 60(2):321–332, 1999.

A subclass of generalized q-ary Goppa codes with good parameters

Sergei Bezzateev and Natalia Shekhunova
St.Petersburg University of Airspace Instrumentation
Bolshaya Morskaya st. 67, St.Petersburg, 190000 Russia
email: bsv@aanet.ru

A construction of generalized q-ary Goppa codes with good parameters is given. Based on Brouwer's table [1], some good codes are obtained from our construction.

1 Introduction

To construct a generalized Goppa codes[2,3] over the finite field F_q, one has to choose a set of locators L of polynomials $\{\Psi_i(x), \Phi_i(x)\}, i = 0,..,n$ from $F_q[x]$ such that for any $i,j : GCD(\Psi_i(x), \Psi_j(x)) = 1$, $\deg(\Phi_i(x)) < \deg(\Psi_j(x)) \leq l$ and $GCD(\Psi_i(x), \Phi_i(x)) = 1$. The second object that determine Goppa code is Goppa polynomial - $G(x)$. $G(x)$ is the polynomial degree t over F_q and $GCD(G(x), \Psi_i(x)) = 1$ for all $i = 0,..,n-1$. Vector $a = (a_0, a_1, ..., a_{n-1})$ with elements from $F_q$is a codeword of generalized (L, G) Goppa code if and only if

$$\sum_{i=0}^{n-1} a_i \frac{\Phi_i(x)}{\Psi_i(x)} = 0 \; mod \; G(x)$$

The minimum distance d of the generalized (L, G) codes estimated by:

$$d \geq \frac{t-l}{l} + 1$$

and dimension $k \geq n - t$.

In this paper we construct a class of q-ary linear codes by using approach to obtain Goppa codes with good parameters introduced by authors for binary and ternary codes in [2,3]. As result we obtain quasi cyclic codes with good parameters.

2 Construction

Let's consider a case, when as the polynomials for locator set are used all irreducible over F_q polynomials with degree l and $\deg \Phi_i(x)) \leq l-1$ for all $i = 0,..,n$ a is element from F_q. In this case length of a code n is determined by number of irreducible polynomials degree l over F_q :

$$I_q(l) = \frac{1}{l} \sum_{d \mid l} \mu(d)\, q^{l/d}$$

where $\mu(d)$ is the Mobius function [4].

Similarly as in [3] let's choose

$$G(x) = x^{q^{l-1}+q^{l-2}+ \, ... \, +q+1},$$

then the i-th column h_i of a check matrix H of such code can be written as follows:

$$h_i = \begin{bmatrix} \frac{1}{\beta_i^t} + \frac{1}{\beta_i^{qt}} + ... + \frac{1}{\beta_i^{q^{s-1}t}} \\ \frac{\beta_i}{\beta_i^t} + \frac{\beta_i^q}{\beta_i^{qt}} + \; ... \; + \frac{\beta_i^{q^{s-1}}}{\beta_i^{q^{s-1}t}} \\ ... \\ \frac{\beta_i^{t-1}}{\beta_i^t} + \frac{\beta_i^{(t-1)q}}{\beta_i^{qt}} + \; ... \; + \frac{\beta_i^{(t-1)q^{s-1}}}{\beta_i^{q^{s-1}t}} \end{bmatrix}$$

where $\beta_i \in F_{q^l}$ and $\beta_i, \beta_i^q, \; ... \;, \beta_i^{q^{l-1}}$ – roots of $\Psi_i(x)$, and F_{q^l} is the extension over F_q of degree l.

By executing transformations in each line of h_i we have:

$$h_i = \begin{bmatrix} \frac{T_l(1)}{\beta_i^t} \\ \frac{T_l(\beta_i)}{\beta_i^t} \\ ... \\ \frac{T_l(\beta_i^{l-1})}{\beta_i^t} \end{bmatrix}$$

where $T_l(\alpha)$ is a trace of element α from F_{q^l} . Using the technique described by the authors earlier in [3,4] it is easy to show, that in a check matrix H of this code composed by the columns h_i , $i = 0,..,n$ it is exist not more than $q^{l-1} - q^{l-2} - \; ... \; - q$ linearly independent rows with elements from $F_q$if $T(1) \neq 0$ and not more than $q^{l-1} - q^{l-2} - \; ... \; - q - 1$ if $T(1) = 0$. Therefore the dimension k of such generalized (L, G) Goppa code is :

$$k \geq n - \frac{q^l - 2q^{l-1} + q}{q-1}$$

if $T(1) \neq 0$ and

$$k \geq n - \frac{q^l - 2q^{l-1} + 1}{q-1}$$

if $T(1) = 0$.

Using an estimation for the minimum distance d of the generalized (L, G) codes we have:

$$d \geq \frac{q^l - 1}{(q-1)\, l} + \frac{1}{l}$$

It is easy to show that for $l \leq \sqrt{q}$:

$$\frac{r}{n} + \frac{d}{n} \geq 1 - \frac{1}{\sqrt{q}} + \frac{1}{\sqrt{q}\,(q-1)}$$

3 Example

In this section, we show example of codes from our subclass in case $l = 2$, $q = 8$ and $\mathrm{G}(x) = x^{i(q+1)}, i = 1, 2, 3, 4$. Let's choose a set of locators L of polynomials $\{\Psi_j(x), \Phi_j(x)\}$, $j = 0, .., n-1$ from $F_q[x]$ such that

$$\frac{\Phi_j(x)}{\Psi_j(x)} = \frac{\alpha_j + \alpha_j^q}{(x + \alpha_j) \cdot (x + \alpha_j^q)} ,$$

where $\alpha_j \in F_{q^2}$ and $\alpha_j \neq \alpha_j^q$.

It is easy to show that for these codes

$$r \leq i\left(q - \frac{i+1}{2}\right);\ n = \frac{q^2 - q}{2};\ d \geq \frac{i(q+1)}{2} + 1.$$

$n = 28$

i	k	d	d_B [2]
1	21	6	6
2	15	10	10
3	10	15	15
4	6	19	19
5	3	24	24 *optimal*

Remark. It should be note that some of this codes was obtained as punctured Reed-Solomon codes by Chaoping Xing and San Ling[5].

References

[1] A. Brouwer, "Bounds on the minimum distance of linear code," http://www.win.tue.nl/~aeb/voorlincod.html

[2] S.V. Bezzateev and N.A.Shekhunova , "One Generalization of Goppa codes", ISIT-97, Proceedings, p.299, Ulm, Germany, 1997.

[3] S.V. Bezzateev and N.A.Shekhunova , "A subclass of binary Goppa codes with improved estimation of the code dimension", Design, Codes and Cryptography, v.14,p.23-38, 1998.

[4] F. J. MacWilliams and N. J. A. Sloane, "*The theory of error-correcting codes*", Amsterdam: North-Holland, 1977.

[5]C.Xing and S.Ling, "A class of linear codes with good parameters", preprint,1999

Trellis representations for quasi-cyclic codes

I. E. Bocharova and B. D. Kudryashov
Department of Information Systems
St.-Petersburg University on Aerospace Instrum.
St.-Petersburg, 190000, Russia
e-mail: irina@it.lth.se; boris@it.lth.se

M. Handlery and R. Johannesson
Department of Information Technology
Lund University, P.O. Box 118
SE-22100 Lund, Sweden
e-mail: marc@it.lth.se; rolf@it.lth.se

Abstract

A new approach to search for conventional trellis representations for quasi-cyclic codes is presented. It is based on using parameters of the tailbiting trellises for these codes. Some newly found conventional trellis representations for quasi-cyclic codes are tabulated.

I. Introduction

It is well-known that a trellis representation of a linear block code may be used for efficient trellis-based decoding procedures. So-called conventional trellises for linear block codes have been investigated by many authors [1],[2],[3],[4]. Tailbiting trellises were introduced in [5] and have been rather intensively investigated in recent years [6],[7],[8]. Since a tailbiting code can be obtained by terminating a convolutional code [5],[6] it can be represented by a regular structure. The convolutional encoder is initialized with the proper starting state such that the encoder, after encoding a block of information bits, will end in the starting state. A rate $R = b/c$ convolutional code with memory m can be terminated into a rate $R = (bl)/(cl)$ block code. We call this quasi-cyclic linear (cl, bl) block code a tailbiting (TB) code with tailbiting length l and rate $R = b/c$. The maximal state complexity of its trellis representation is equal to bm.

The problem of searching for new TB codes has very high computational complexity even compared to searching for convolutional codes. To search for the best TB code with maximal state complexity bm, length l and rate $R = b/c$ we should test $2^{(m+1)bc}$ sets of bc generator polynomials of degree m (length $m + 1$). For each set of generator polynomials we have to determine the minimum distance of the code. In order to reduce the search complexity we first need efficient methods to reject weak code candidates, then we need an efficient search procedure for finding a codeword of minimal weight in a TB code [8].

Due to the nonregular structure of conventional trellises, the problem of finding good linear block codes of rate $R = b/c$ with conventional trellises having low state complexity is even more complicated than the problem of finding quasi-cyclic codes and their TB trellis representations. Thus, only a few conventional trellises for rather short linear block codes [4] and for special classes of codes [9],[10] have been published. However, if we already have a tailbiting trellis representation of a linear, $R = b/c$ code with state complexity less than or equal to bm, it is rather easy to construct a minimal conventional trellis with regular structure and maximal state complexity less than or equal to $2bm$. By using the regular structure of this conventional trellis representation it is possible to significantly simplify the code descriptions and implementations of maximum-likelihood decoding procedures [11].

First, we briefly review some notions of tailbiting and conventional trellises and their complexity criteria. Then we describe an approach to construct conventional trellises via tailbiting trellises. Parameters of some newly found conventional trellises are tabulated.

II. Trellis representations for linear block codes – Definitions.

Let us consider the generator matrix G^{tb} of a rate $R = b/c$ tailbiting code $\mathcal{C}$ with tailbiting length l that is generated by a memory m convolutional encoder. We assume with no essential loss of generality that the generator matrix of the parent convolutional code is realized in controller canonical form. Then, the $bl \times cl$ matrix G^{tb} has the following form

$$G^{tb} = \begin{pmatrix} * & * & * & * & \dots & * & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & * & * & * & * & \dots & * & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & * & * & * & * & \dots & * & \mathbf{0} & \mathbf{0} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & * & * & * & * & \dots & * \\ * & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & * & * & * & \dots & * \\ * & * & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & * & * & \dots & * \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ * & * & \dots & * & \mathbf{0} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & * \end{pmatrix}, \qquad (1)$$

where the $(m + 1)$ nontrivial blocks of G^{tb} marked by asterisks represent binary $b \times c$ matrices and $\mathbf{0}$ denotes the trivial all-zero block of the same size. It is known [8] that a tailbiting trellis for a linear code $\mathcal{C}$ with generator matrix G^{tb} may be obtained by the product of the elementary tailbiting trellises for each row $\mathbf{r}_i^{tb}$, $i = 1, 2, \dots, bl$, of G^{tb}. The state complexity of the tailbiting trellis for the code $\mathcal{C}$ at each time instant depends on the number of rows in G^{tb} that are active [8]. A row is active at time j if its jth element is nontrivial and it is not the last nontrivial (in circular sense) element in this row. Moreover, for tailbiting codes obtained from convolutional codes with generator matrices realized in controller canonical form, the maximal state complexity μ^{tb} is determined [8] as

$$\mu^{tb} = \max_{i=1,\dots,cl} \{\alpha_i\} = bm,$$

where α_i denotes the number of active positions of the ith column of G^{tb}.

The minimal conventional trellis for a linear block code $\mathcal{C}$ of rate $R = K/N$ can be obtained according to [3] as a product of the elementary trellises for each row of its generator matrix $G^{conv}{=}(\mathbf{r}_i^{\mathbf{conv}})$, $i = 1, 2, \dots, K$, if G^{conv} is in minimal span form or, in other words, if both $start(\mathbf{r}_i^{\mathbf{conv}}) \neq start(\mathbf{r}_j^{\mathbf{conv}})$ and $end(\mathbf{r}_i^{\mathbf{conv}}) \neq end(\mathbf{r}_j^{\mathbf{conv}})$ if $i \neq j$, where $start(\mathbf{x})$ denotes the first nontrivial position of vector $\mathbf{x}$ and $end(\mathbf{x})$ denotes the last nontrivial position of vector $\mathbf{x}$. The state complexity of the minimal conventional trellis at each time instant is determined by the number of active rows in G^{conv}.

III. Constructing conventional trellises via tailbiting trellises

As mentioned in the introduction, the search complexity for tailbiting trellises (codes) is rather high. To speed up the search of tailbiting trellises we rejected weak code candidates using two types of the tests described in [8] and choose as candidates only codes with an active distance-slope [12] greater than a predetermined threshold. For finding the minimum distance of the code we used BEAST, a bidirectional efficient algorithm for searching a tree, modified for searching tailbiting codes [11]. Moreover, we performed the search over the reduced set of the subcodes of the tailbiting code that have to be checked to assure that every codeword or a quasi-cyclic shift of it has been considered [11].

The use of the described search technique made it possible to find new quasi-cyclic TB codes some of which have larger minimum distances than the previously best known codes with the same parameters. Tables of some newly found codes of rate $R = 1/c$ are presented in [8] and [11]. Applying the described search technique to time-varying convolutional codes of rate $R = 1/c$ with period T, we found new rate $R = T/(Tc)$ TB codes with less maximal state complexities. Moreover, we found rate $R = T/(Tc)$ TB codes with conventional trellis representations that have lower maximal state complexity than the conventional trellis representations of the best time-invariant $R = 1/c$ TB codes with the same length N, dimension K, and minimum distance d_{min}. In Table I, we present the new TB codes obtained via time-varying convolutional codes of rate $R = 1/2$. The parameter $\widehat{d}_{\min}$ in Table I describes the lower and upper bounds on the minimum distance from the Brouwer-Verhoeff table [13], and μ^{tb} and $\widehat{\mu}^{tb}$ are the maximal state complexity of the TB trellis and its lower bound [8], respectively. The generators are given in octal form.

This research was supported in part by the Swedish Academy of Science in cooperation with the Russian Academy of Sciences and in part by the Swedish Research Council under Grant 2001-3117.

TABLE I

New tailbiting codes obtained via time-varying convolutional codes of rate $1/2$

K	$d_{\min}(\hat{d}_{\min})$	$\mu^{tb}(\widehat{\mu}^{tb})$	T	Generators
18	8(8-9)	5(4)	2	(13,15), (57,75)
30	12(12-14)	8(8)	2	(507,675)),(573,705)
38	14(14-18)	10(10)	2	(2375,2457),(2515,3073)

Let us show how to construct the minimal conventional trellis for a linear block code $\mathcal{C}$ with given generator matrix G^{tb}. Consider again the generator matrix G^{tb} of a rate $R = b/c$ tailbiting code $\mathcal{C}$ with tailbiting length l that is generated by a memory m convolutional encoder. It may be represented in the form $G^{tb} = \binom{G^u}{G^l}$ where G^u consists of the first $b(l-m)$ rows of G^{tb} generating a zero-tail terminated convolutional code [11] represented by the trellis $\mathcal{T}_u$ with maximal state complexity equal to bm. Matrix G^l consists of the last bm rows of G^{tb}. The corresponding trellis $\mathcal{T}_l$ obtained as the product of the elementary trellises corresponding to the rows of G^l has maximal state complexity less than or equal to bm. By taking the trellis product $\mathcal{T}_u\mathcal{T}_l$ we obtain a conventional trellis that has maximal state complexity less than or equal to $2bm$. Clearly, using the algorithm proposed by Kschischang and Sorokine [3], we can reduce G^{tb} to the minimal span form matrix G^{conv} and obtain a minimal conventional trellis for $\mathcal{C}$ with equal or less maximal state complexity.

We are now going to show that, due to the regular structure of the initial TB trellis, we can represent the obtained conventional trellis by a few generator polynomials and their shifts. Consider for example a linear block code with parameters $N = 20$, $K = 10$, and $d_{min} = 6$. Its tailbiting trellis representation has maximal state complexity $\mu^{tb} = 3$, and its conventional trellis representation has maximal state complexity $\mu^{conv} = 6$. The corresponding generator matrices G^{tb} and G^{conv} have the form

$$G^{tb} = \begin{pmatrix} 11110110000000000000 \\ 00111101100000000000 \\ 00001111011000000000 \\ 00000011110110000000 \\ 00000000111101100000 \\ 00000000001111011000 \\ 00000000000011110110 \\ 10000000000000111101 \\ 01100000000000001111 \\ 11011000000000000011 \end{pmatrix} \text{ and } G^{conv} = \begin{pmatrix} 11110110000000000000 \\ 00111101100000000000 \\ 00001111011000000000 \\ 00000011110110000000 \\ 00000000111101100000 \\ 00000000001111011000 \\ 00000000000011110110 \\ 01011000001100010000 \\ 00010110000011000100 \\ 00000101100000110001 \end{pmatrix}.$$

It is easy to see that G^{conv} may be represented as 7 shifts to the right by 2 positions of generator 11110110 ($F6$ in the hexadecimal form) and 3 shifts to the right by 2 positions of generator 0101100000110001 (5831 in the hexadecimal form) (see Table II). Note that, according to [4], this conventional trellis is optimal in the sense of complexity. It can be shown that for TB codes of rate $R = 1/c$ the matrix G^{conv} always can be represented as shifts of c generators.

In Table II, we present the new conventional trellises for linear block codes of rate $R = 1/2$. The parameters μ^{conv} and $\widehat{\mu}^{conv}$ denote state complexity of the conventional trellis and its lower bound, respectively [2]. We represent the generator polynomials in hexadecimal form, and in parentheses we point out the number of shifts of the given generator and the number of positions by which it is shifted. Note that by reducing the matrices G^{tb} obtained from time-varying convolutional codes to minimal span form it is possible to get a conventional trellis with maximal state complexity less than $2bm$. For example, a linear block code with parameters $N = 28$, $K = 14$, and $d_{min} = 8$ has been obtained from a time-varying convolutional code of rate $R = 1/2$ and period $T = 2$. It has maximal tailbiting trellis complexity $\mu^{tb} = 5$ and its conventional trellis has maximal state complexity $\mu^{conv} = 9$.

TABLE II

New conventional trellises for rate $1/2$ linear block codes

K	$d_{\min}(\hat{d}_{\min})$	$\mu^{conv}(\widehat{\mu}^{conv})$	Generators
10	6(6)	6(6)	F6(7,2), 5831(3,2)
13	7(7)	10(7)	FCD(8,2), 5EF38(5,2)
14	8(8)	9(8)	BDC(5,4),73B(5,4), 3B00CE(1,0),127748(2,4),0297786(1,0)
15	8(8)	10(8)	E75(10,2), 594E78(5,2)
18	8(8)	9(7)	BDC(7,4), 73B(7,4),12700748(4,2),3B0000CE(1,0), 029700786(1,0)
19	8(8-9)	10(7)	E75(14,2), 59400E78(5,2)
20	9(9-10)	12(11)	31EB(14,2), 7FDC00C8(6,2)
22	10(10)	12(12)	3B5B(16,2), 69B000038(6,2)
30	12(12-14)	16(13)	EE69C(11,4),39677(11,4), 00114B70000D7E(2,4), 7EB0000ED288(2,12), 01F948B700D7E(1,0), 0047D7700ED288(1,0), 0706FFB7ED288(1,0), 11A68FB70D7E(1,0)
46	16(16-22)	26(20)	DEB2F12(33,2),7288E4800003405C9(13,2)

IV. Conclusion

In this paper, we have presented a new approach for finding conventional trellis representations for linear block codes via their tailbiting trellis representations. The newly found conventional trellises have low state complexity, and due to their regular structure, a simple description that allows efficient implementations of maximum-likelihood decoding procedures. We tabulated the newly found tailbiting and conventional trellises.

References

[1] G.D. Forney, Jr. "Dimension/length profiles and trellis complexity of linear block codes," *IEEE Trans. Inform. Theory.*, vol. 34, pp. 1152–1187, 1988.

[2] A. Lafourcade and A. Vardy, "Lower bounds on the trellis complexity of block codes," *IEEE Trans. Inform. Theory.*, vol. 41, pp. 1938–1954, 1995.

[3] F. R. Kschischang and V. Sorokine, "On the trellis structure of block codes," *IEEE Trans. Inform. Theory*, vol. 41, pp. 1924–1937, Nov. 1995.

[4] P. Schuurman, "A table of state complexity bounds for binary linear codes," *IEEE Trans. Inform. Theory*, vol. 42, pp. 2034–2042, Nov. 1996.

[5] G. Solomon and H. C. A. van Tilborg, "A connection between block and convolutional codes," *SIAM J. Appl. Math.*, vol. 37, pp. 358–369, 1979.

[6] J. H. Ma and J. K. Wolf, "On tail-biting convolutional codes," *IEEE Trans. Commun.*, vol. 34, pp. 104–111, Feb. 1986.

[7] A. R. Calderbank, G. D. Forney, Jr., and A. Vardy, "Minimal tail-biting trellises: The Golay code and more," *IEEE Trans. Inform. Theory*, vol. 45, pp. 1435–1455, Jul. 1999.

[8] I. E. Bocharova, R. Johannesson, B. D. Kudryashov, and P. Ståhl, "Tailbiting codes: Bounds and search results," *IEEE Trans. Inform. Theory*, vol. 48, pp. 137–148, Jan. 2002.

[9] A. Vardy, and Y. Be'ery "Maximum-likelihood soft decision decoding of BCH codes," *IEEE Trans. Inform. Theory.*, vol. 40, pp. 546–554, 1994.

[10] A. Trachtenberg, "Designing lexicographic codes with a given trellis complexity," *IEEE Trans. Inform. Theory.*, vol. 48, pp. 89–100, 2002.

[11] I. E. Bocharova, M. Handlery, R. Johannesson, and B. D. Kudryashov, "A BEAST for prowling in trees," *submitted to IEEE Trans. Inform. Theory*, June, 2002.

[12] I. E. Bocharova, M. Handlery, R. Johannesson, and B. D. Kudryashov, "Tailbiting codes obtained via convolutional codes with large slope," *accepted for publication, IEEE Trans. Inform. Theory*, Mar. 2002.

[13] A. E. Brouwer and T. Verhoeff, "An updated table of minimum-distance bounds for binary linear codes," *IEEE Trans. Inform. Theory*, vol. 39, pp. 662–677, Mar. 1993, see also *http://www.win.tue.nl/˜aeb/voorlincod.html.*

Bounds for Quaternary Equidistant Constant Weight Codes

Galina Bogdanova, Teodora Yorgova
Institute of Mathematics and Informatics, BAS, Bulgaria
galina@moi.math.bas.bg teda@moi.math.bas.bg

Abstract

We explore the problem of finding bounds for quaternary equidistant constant weight codes. For $d = 3$, all bounds are determined for all values of n and w. A table of the best known bounds for quaternary equidistant constant weight codes with parameters $2 \le w < n$ and $4 \le n \le 10$ is presented.

1 Introduction

Let Z_q denote the set $\{0, 1, ..., q-1\}$ and let Z_q^n be the set of all n-tuples over Z_q. A code is called *equidistant* if all the distances between distinct codewords are d. An $(n, M, d)_q$ equidistant code is a code over Z_q of length n, cardinality M and distance d. Let $B_q(n, d)$ denote the largest possible value of M when the other parameters are fixed. An $(n, M, d, w)_q$ code is called *equidistant constant weight* code (ECWC) if in an equidistant code all codewords have the same weight w. Let $B_q(n, d, w)$ denote the largest possible value of M in a ECWC when the other parameters are fixed. Codes with such parameters are called optimal.

Equidistant codes have been investigated by a large number of authors. Some works published on this topic are [3], [4], [6], [8]. A few papers study ECWC, for example [2], [5], [7] and [1].

2 General Bounds

For $d > 2w$, $B_q(n, d, w) = 1$, so we consider only codes with distance $d \le 2w$. Some bounds for equidistant codes are given by the following theorems:

Theorem 1 $B_q(n, d) = 1 + B_q(n, d, d)$.

Theorem 2 (Delsarte) $B_q(n, d) \le (q-1)n + 1$.

Theorem 3 $B_q(n, n, w) \le q$,

$$B_q(n+1, d, w) \ge B_q(n, d, w), \qquad B_q(n+1, d, w+1) \ge B_q(n, d, w).$$

According to [5] $B_q(q+1, q, q-1) \le (q^2+q)/2$.

Theorem 4 (the Johnson bounds for ECWC) *The maximum number of codewords in a q-ary ECWC satisfy the inequalities:*

$$B_q(n, d, w) \le \frac{n}{n-w} B_q(n-1, d, w),$$

$$B_q(n, d, w) \le \frac{n(q-1)}{w} B_q(n-1, d, w-1).$$

Theorem 5 *[2] For $k = 1, 2, ..., n$, if $P_k^2(w) > P_k(d) P_k(0)$, then*

$$B_q(n, d, w) \le \frac{P_k^2(0) - P_k(d) P_k(0)}{P_k^2(w) - P_k(d) P_k(0)}.$$

Here $P_k(x)$ is the Krawtchouk polynomial defined by

$$P_k(x) = \sum_{i=0}^{k} \binom{x}{i} \binom{n-x}{k-i} (-1)^i (q-1)^{k-i} \quad \textit{and} \quad P_k(0) = \binom{n}{k} (q-1)^k.$$

Theorem 6 *[8] The optimal equidistant $(n, qt, d)_q$ codes and RBIB designs $(v = qk, b, k, r, \lambda)$ are equivalent to one another and their parameters are connected by the conditions $v = M$, $b = nq$, $k = t$, $r = n$, $\lambda = n - d$.*

Theorem 7 *If there exists an $(n, M, d, w)_q$ code, then there exists a $(\lambda n, M, \lambda d, \lambda w)_q$ code for all integers $\lambda \ge 1$.*

3 Combinatorial bounds and constructions

For $d = 3$, all bounds are determined for all values of $n \ge 4$ and $w \ge 2$. The following combinatorial bounds and families of ECWC are obtained:

Proposition 8 *There exists a family of optimal ECWC over an alphabet of four elements with parameters $(n, 3, 3, 2)_4$ for every integer $n \ge 4$.*

Proof: Let u be a fixed codeword with length n and weight 2. Considering how many codewords are at distance exactly 3 from u we obtain $B_4(n, 3, 2) = 3$. ∎

Proposition 9 *The maximum number of codewords in a $(n, M, 3, 3)_4$ ECWC satisfy the inequality*

$$B_4(n, 3, 3) \le 8.$$

Proof: We may assume without loss of generality (wlog) that all codewords are ordered in lexicographic way and the first codeword is 00...0111.
(i) We cannot have more than one symbol 0 in the last four coordinate positions in a certain codeword. Otherwise there will be a codeword which will differ from the first one in $d \neq 3$ positions, which is a contradiction. Hence in the first $n-4$ coordinate positions we have only zeros. After shortening with respect to the symbol 0 $n-4$ times we get a code which is equivalent to a $(4, M, 3, 3)_4$ code.
(ii) $B_4(4,3,3) \leq 8$, which follows from the next reasoning: We may assume wlog that in the first coordinate position we have 0001... and the first codeword is 0111. After shortening with respect to the symbol 0 we get a code which does not contain a symbol 0 and is equivalent to an optimal $(3,3,3)_3$ code. Hence the fourth codeword will differ from the first three in two positions and $d \leq 2$, which is a contradiction. Therefore in each coordinate position there are not more than two 0s and $B_4(n,3,3) \leq 8$; ■

Proposition 10 *There exists a family of optimal ECWCs with parameters* $(4+\lambda, 8, 3, 3)_4$ *for every integer* $\lambda \geq 0$.

Proof:
(i) We construct a $(4,3,3)_4$ ECWCs 0111, 0222, 1(012), 2(021) and hence $B_4(4,3,3) \geq 8$, where (...) denotes all cyclic shifts of the symbols inside the parentheses.
(ii) Let us denote the code, constructed in (i) with C. From C we construct a family of $(4+\lambda, 8, 3, 3)_4$ ECWCs in the following way:

$$\left\{ \left(\underbrace{0...0}_{\lambda}, c \right) |c \in C \right\}$$

where $\lambda \geq 0$.
(iii) From proposition 9 and (ii) we obtain $B_4(n,3,3) = 8$. ■

Proposition 11 *There exists a family of optimal ECWCs with parameters* $(4+\lambda, 9, 3, 4+t)_4$ *for every integer* $\lambda \geq 0$ *and* $0 \leq t \leq n-4$.

Proof:
(i) From the code C, constructed in Proposition 10 we construct the code C' in the following way: we add the all-zero codeword and apply the permutation: 1 → 3, 0 → 1 and we obtain the code 1111, 1222, 1333, 2(123), 3(132). Hence $B_4(4,3,4) \geq 9$.
(ii) We construct the code

$$\left\{ \left(\underbrace{0...0}_{\lambda-t} \underbrace{\alpha...\alpha}_{t}, c \right) |c \in C' \right\},$$

where $\lambda \geq 0$, $0 \leq t \leq n-4$ and $\alpha = 1, 2, 3$.
In a similar way as in Proposition 9 and from (ii) we obtain $B_4(n,3,4) = 9$. ■

4 A Table of $B_4(n,d,w)$

The best known upper and lower bounds (and exact values when these coincide) for ECWC over an alphabet of four elements, of length $n \leq 10$ are displayed in Table 1.

For codes of small size we apply combinatorial reasoning. All the numbers in column $d = 3$ are obtained by Proposition 8, Proposition 10 and Proposition 11. For the rest of the values of M we use our own, specifically developed, computer algorithm (based on exhaustive search), which is of exponential complexity.

If in a certain position only one number occurs, then this number is the exact value of $B_4(n,d,w)$ and the corresponding codes are optimal. If the bound is obtained by our computer algorithm, in this case we omit the index. If two numbers are given, then the right one is the best known upper bound for $B_4(n,d,w)$, received by Theorem 4 and Theorem 5. The left one is the best known lower bound for $B_4(n,d,w)$.

Acknowledgment. The authors wish to thanks V.A.Zinoviev for helpful discussions.

References

[1] G.T.Bogdanova, T.A.Yorgova. Bounds for Ternary Equidistant Weight Codes. *Mathematics and Education in mathematics 31*, Borovec, April 3-6, 2002, pp. 131-135.

[2] F.W. Fu, T. Klove, Y. Luo, and V.K. Wei. On equidistant Constant Weight codes. *In proceedings WCC'2001 Workshop on Coding and Cryptograpy*, Paris, France, Jan 2001, pp. 225-232.

[3] J.I. Hall. Bounds for equidistant codes and partual projective planes. Discrete Math., vol.17, 1977, pp. 85-94.

[4] J.I. Hall, A.J.E.M. Jansen, A.W.J. Kolen, and J.H. van Lint. Equidistant codes with distance 12. Discrete Math., vol.17, 1977, pp. 71-83.

[5] W. Heise and Th. Honold. Some equidistant constant weight codes. http://fatman.mathematik.tu-muenchen.de/ heise/MAT/code_oval.html

[6] J.H. van Lint. A theorem on equidistant codes. *Discrete Math.*, vol. 67, 1973, pp. 353-358.

[7] D.R. Stinson and G.H.J. van Rees. The equivalence of certain equidistant binary codes and symmetric BIBDs. *Combinatorica*, vol. 4, 1984, pp. 357-362.

[8] N. V. Semakov and V. A. Zinoviev. Equidistant q-ary codes with maximal distance and resolvable balanced incoplete block desins. *Problemi peredachi Informatsii*, vol. 4, No. 2, 1968, pp. 3-10.

The rank and kernel of $\mathbb{Z}_4$-linear Preparata-like and Kerdock-like codes[1]

J.Borges, J.Rifà

Computer Science Department, Universitat Autònoma de Barcelona,
08193-Bellaterra, Spain.
E-mail: {joaquim.borges, josep.rifa}@uab.es

K.Phelps

Discrete & Statistical Sciences, Auburn University
Auburn, Al 36849-5307 USA.
E-mail: phelpkt@dms.auburn.edu

V.A. Zinoviev

Institute for Problems of Information Transmission of the Russian Academy of Sciences
Bol'shoi Karetnyi per. 19, GSP-4, Moscow, 101447, Russia.
E-mail: zinov@iitp.ru

Abstract: We say that a binary code of length n is additive if it is isomorphic to a subgroup of $\mathbb{Z}_2^\alpha \times \mathbb{Z}_4^\beta$, where the quaternary coordinates are transformed to binary by means of the usual Gray map and hence $\alpha + 2\beta = n$.

In this paper we prove that any additive extended Preparata-like code always verifies $\alpha = 0$, i.e. it is always a $\mathbb{Z}_4$-linear code. Moreover, we compute the rank and the dimension of the kernel of such Preparata-like codes and also the rank and the kernel of the $\mathbb{Z}_4$-dual of these codes, i.e. the $\mathbb{Z}_4$-linear Kerdock-like codes.

1. Introduction: Let $\mathbb{F}^n$ denote the set of all binary vectors of length n. As usual $d(\cdot,\cdot)$ denotes the *Hamming distance* and $\mathrm{wt}(\cdot)$ denotes the *Hamming weight*. Let e_i denote the vector of weight one with the nonzero coordinate at the i-th position for $i = 1, \ldots, n$. For any vector $v \in \mathbb{F}^n$ we denote by *supp*(v) the set of coordinate positions in which v has nonzero entries.

Let $\star$ be a binary operation such that $(\mathbb{F}^n, \star)$ is a translation-invariant Abelian group, that is, a group with the property that

$$d(x \star v, x \star u) = d(v, u), \quad \forall x, v, u \in \mathbb{F}^n. \tag{1}$$

As can be seen in [2], $(\mathbb{F}^n, \star) \cong (\mathbb{Z}_2^\alpha \times \mathbb{Z}_4^\beta, +)$ where $\alpha + 2\beta = n$ and '+' is the usual addition modulo 2 for the $\mathbb{Z}_2$ coordinates and modulo 4 for the $\mathbb{Z}_4$ coordinates. An isomorphism between $\mathbb{Z}_2^\alpha \times \mathbb{Z}_4^\beta$ and $\mathbb{F}^n$ is given by the map

$$\phi(x_1, \ldots, x_\alpha \mid y_1, \ldots, y_\beta) = (x_1, \ldots, x_\alpha \mid \varphi(y_1), \ldots, \varphi(y_\beta)),$$

where $\varphi(0) = (0,0)$, $\varphi(1) = (0,1)$, $\varphi(2) = (1,1)$ and $\varphi(3) = (1,0)$ is the usual Gray map from $\mathbb{Z}_4$ onto $\mathbb{Z}_2^2$. Now, it is clear that

$$x \star y = \phi(\phi^{-1}(x) + \phi^{-1}(y)), \quad \forall x, y \in \mathbb{F}^n.$$

A *(binary) additive code* (see [2, 3]) $(D, \star)$ of length $n = \alpha + 2\beta$ is a subgroup of $(\mathbb{F}^n, \star)$. An additive code is a particular case of the more general class of *translation-invariant propelinear codes* [2, 8]. Note that the case $\beta = 0$ corresponds to a linear code and the case $\alpha = 0$ corresponds to a $\mathbb{Z}_4$-linear code. In this last case (see [8]) $(D, \star)$ is a group, where each codeword x is associated with a coordinate permutation $\pi_x \in \mathcal{S}_n$ such that $x \star y = x + \pi_x(y)$ for any $y \in D$.

Recall that the rank of any code D is the dimension of the linear span of D, which we denote here $< D >$, and the kernel of D, denoted here by $ker(D)$, is defined by $ker(D) = \{v \in D \mid v + D = D\}$.

Preparata-like codes are not linear. Concerning to their possible algebraic structure, we remark that the original Preparata codes [7] have a group propelinear structure see ([8]) and the extended Preparata-like codes defined in [6] are $\mathbb{Z}_4$-linear and so, according to [8], they are propelinear codes.

In this paper we prove the nonexistence of extended Preparata-like codes with other additive structures different of the $\mathbb{Z}_4$-linear ones. Given a Preparata-like code P^*, it is well known [11] that the code C^* obtained as the union of P^* and the vectors at maximum distance from P^* is a perfect single error correcting code or *1-perfect code*. If P^* is a standard Preparata code, then C^* is linear i.e. a Hamming code. If P is an extended Preparata-like code, then C is an extended 1-perfect code. We show that if P is $\mathbb{Z}_4$-linear, then C is also $\mathbb{Z}_4$-linear. We also prove that, up to a permutation of binary coordinates, the vector with all the quaternary coordinates 1 belongs to the extended Preparta-like code P, to the associated extended 1-perfect code C, and to both extended $\mathbb{Z}_4$-duals K and H. Finally, we compute the ranks and kernels of extended Preparata-like code P and its $\mathbb{Z}_4$-dual, the Kerdock-like code K.

2. Preparata-like codes: A *Preparata-like code* P^* has length $n = 2^{2m} - 1$ $(m \geq 2)$, minimum distance $d = 5$ and $|P^*| = M = 2^{n+1-4m}$ codewords. Such a code satisfies the Johnson bound and therefore it is nearly perfect [5, 10] and strongly uniformly packed and so, completely regular [10] (in particular, distance invariant). If we assume P^* contains the zero codeword, then the codewords of weight 5 form a $2-(n, 5, \lambda)$-design [9] with $\lambda = (n-3)/3$.

The main results of this section are:

[1] Research partially supported by Spanish CICYT Grant TIC2000-0739-c04-01, by Catalan DURSI Grant 2001 SGR 00219 and also by Ministerio de educación, cultura y deporte Grant SAB 2000-0058.

Proposition 1 *Let P^* be any Preparata-like code and let $P_3^* = \{c \in \mathbb{F}^n \mid d(c, P^*) = 3\}$. Code $C^* = P^* \cup P_3^*$ is a 1-perfect error corecting code and the rank of P^* is equal to the rank of C^*.*

Proposition 2 *Let P be an additive extended Preparata-like code and let $P_4 = \{c \in \mathbb{F}^n \mid d(c, P) = 4\}$. Code $C = P \cup P_4$ is an extended 1-perfect code and it is partitioned into cosets of P of weight four.*

Proposition 3 *Let P be an additive extended Preparata-like code, then so is $C = P \cup P_4$.*

Theorem 4 *Let P be an extended additive Preparata-like code and let $K = \phi(\mathcal{P}^{\perp})$ be the corresponding extended additive Kerdock-like code. Then P and K are both $\mathbb{Z}_4$-linear.*

Let q_i denote the quaternary vector in $\mathbb{Z}_4^{2^{2m-1}}$ with only one nonzero coordinate at the i-th position, the value of which is 2. In other words, $\phi(q_i)$ is a binary vector in $\mathbb{Z}_2^{2^{2m}}$ with two nonzero positions, corresponding to the i-th quaternary position.

As a natural extension of Proposition 2, now we can give the evident partition of the extended $\mathbb{Z}_4$-linear 1-perfect code C into the translates of the corresponding $\mathbb{Z}_4$-linear Preparata-like code P.

Theorem 5 *Let P be a $\mathbb{Z}_4$-linear extended Preparata-like code of length $n+1 = 2^{2m}$, and let $C = P \cup P_4$ be the corresponding extended 1-perfect $\mathbb{Z}_4$-linear code. Then for any integer $i = 1, 2, ..., (n+1)/2$, the code C is partitioned into the cosets of P of weight 4 as follows:*

$$C = \bigcup_{j=1}^{(n+1)/2} \{P + \phi(q_i + q_j)\}$$

(recall that "+" means addition modulo 2 for binary vectors and modulo 4 for vectors over $\mathbb{Z}_4$).

Theorem 6 *Let **1** be the quaternary vector with all coordinates 1. After a permutation of binary coordinates vector $\phi(\mathbf{1})$ belongs to C, H, P and K.*

3. The rank and kernel a $\mathbb{Z}_4$-linear Preparata codes and their dual, $\mathbb{Z}_4$-linear Kerdock-like codes:

Proposition 7 *Let C be the extended 1-perfect code corresponding to an extended $\mathbb{Z}_4$-linear Preparata-like code P, of length $n+1 = 2^{2m}$. Then C is the kernel of a group homomorphism θ from $\mathbb{F}^{n+1}$ onto $\mathbb{Z}_2^{\gamma} \times \mathbb{Z}_4^{\delta}$, where $\gamma + 2\delta = 2m+1$ and $\delta = 1$.*

Theorem 8 *Let C be an extended perfect $\mathbb{Z}_4$-linear code. If the length of C is $n+1 = 2^{2m} \geq 16$, then $rank(C) = 2^{2m} - 2m$ for $m > 2$ and $rank(C) = 2^{2m} - 2m - 1$ for $m = 2$.*

Theorem 9 *The rank of an extended $\mathbb{Z}_4$-linear Preparata-like code P of length 2^{2m} is equal to $rank(P) = 2^{2m} - 2m$ for $m > 2$ and $rank(P) = 2^{2m} - 2m - 1$ for $m = 2$.*

The rank of $\mathbb{Z}_4$-linear Kerdock-like code K of length $n+1 = 2^{2m}$ is equal to $rank(K) = 2m^2 + m + 1$.

Theorem 10 *Let P be any $\mathbb{Z}_4$-linear Preparata-like code of length $n+1 = 2^{2m}$ and let K be its $\mathbb{Z}_4$-dual. Then $dim\,(ker(P)) = 2^{2m-1} - 2m + 1$ and $dim\,(ker(K)) = 2m + 1$.*

References:

[1] R.D.Baker, J.H.Van Lint, R.M.Wilson, "On the Preparata and Goethals codes", *IEEE Trans. Inform. Theory*, vol. 29, n. 3, pp. 342-345, May 1983.

[2] J. Borges and J. Rifà, "A characterization of 1-perfect additive codes", *IEEE Trans. on Information Theory*, vol. 45, pp. 1688-1697, 1999.

[3] A.E. Brouwer, A.M. Cohen and A. Neumaier, *Distance Regular Graphs*. Springer-Verlag, 1989.

[4] I.I.Dumer, "Some new uniformly packed codes", Proc. of Moscow Institute of Physics and Technology, pp. 72-78. Moscow, 1976.

[5] J.M. Goethals and S.L. Snover, "Nearly perfect binary codes", *Discrete Math.*, vol. 3, pp. 65-88, 1972.

[6] A.R. Hammons, P.V. Kumar, A.R. Calderbank, N.J.A. Sloane and P. Solé, "The Z_4-linearity of kerdock, preparata, goethals and related codes," *IEEE Trans. on Information Theory*, vol. 40, pp. 301-319, 1994.

[7] F.P. Preparata, "A class of optimum nonlinear double-error correcting codes," *Inform. and Control*, vol. 13, pp. 378-400, 1968.

[8] J. Rifà and J. Pujol, "Translation invariant propelinear codes," *IEEE Trans. Information Theory*, vol. 43, pp. 590-598, 1997.

[9] N.V. Semakov and V.A. Zinoviev, "Balanced codes and tactical configurations," *Problems Inform. Transmission*, vol. 5, pp. 22-28, 1969.

[10] N.V. Semakov, V.A. Zinoviev and G.V. Zaitsev, "Uniformly packed codes," *Problems Inform. Transmission*, vol. 7, pp. 30-39, 1971.

[11] N.V. Semakov, V.A. Zinoviev and G.V. Zaitsev, "Interrelation of Preparata and Hamming codes and extension of Hamming codes to new double-error-correcting codes," *Proc. 2nd Internat. Sympos. Inform. Theory*, Tsakhadsor, Armenia, 1971, Academiai Kiado, Budapest, 1973.

On the asymptotic behaviour of a necessary condition for existence of spherical designs

Silvia Boumova
Institute of Mathematics and Informatics
Bulgarian Academy of Sciences
8 G.Bonchev, 1113 Sofia, BULGARIA
silvi@moi.math.bas.bg

Danyo Danev
Department of Electrical Engineering
Linköping University
S-581 83 Linköping, SWEDEN
danyo@isy.liu.se

Abstract

We investigate the asymptotic consequences of a necessary condition for existence of spherical $(2k-1)$-designs of odd cardinality. Our calculations show that no asymptotic improvement of the known results can be obtained (despite that the condition works well in small dimensions).

1 Introduction

A spherical code is a spherical τ-design if and only if the average value of any real n-variable polynomial $f(x) = f(x_1, x_2, \ldots, x_n)$ of total degree at most τ over the whole sphere is equal to the average value of this polynomial over the code, i.e. the equality

$$\int_{S^{n-1}} f(x)d\mu(x) = \frac{1}{|C|}\sum_{x\in C} f(x) \tag{1}$$

holds. The maximal number τ for which a spherical code C is a τ-design is called strength of C.

One wants to minimize the size of a spherical design provided the dimensions and the strength are fixed. Let

$$B_{odd}(n,\tau) = \min\{|C| : C \in S^{n-1} \text{ is a } \tau\text{-design}, C \text{ is odd}\}$$

be the minimum possible odd cardinality of a τ-design in n dimensions.

Let $\tau = 2k-1$ be fixed integer. It follows from the classical bound of Delsarte-Goethals-Seidel [6] that

$$B_{odd}(n, 2k-1) \geq 2\binom{n+k-2}{n-2} + 1.$$

Some linear programming improvements on the Delsarte-Goethals-Seidel bound were obtained by Boyvalenkov-Nikova [4, 5] and Yudin [9].

For further notations we refer to Levenshtein [8] and [1].

Levenshtein [7, 8] proved that for any $s \in [-1, 1)$ there exist real numbers $-1 \leq \alpha_0 < \alpha_1 < \cdots < \alpha_{k-1} = s$ and non-negative $\rho_0, \rho_1, \ldots, \rho_{k-1}, \rho_k$ such that the formula

$$f_0 = \rho_k f(1) + \sum_{i=0}^{k-1} \rho_i f(\alpha_i)$$

is valid for every real polynomial $f(t)$ of degree at most $2k-1$. If $C \in S^{n-1}$ is a $(2k-1)$-design, then these numbers are uniquely determined by $|C| = 1/\rho_k$.

Theorem 1 ([1]). *If $C \subset S^{n-1}$ be a $(2k-1)$-design with odd cardinality then*

$$\rho_0|C| \geq 2. \tag{2}$$

We express condition (2) in terms of the numbers $\alpha_0, \alpha_1, \ldots, \alpha_{k-1}$.

Corollary 2. *Let $C \subset S^{n-1}$ be a $(2k-1)$-design of odd cardinality. Then*

$$-\frac{(1-\alpha_1^2)(1-\alpha_2^2)\cdots(1-\alpha_{k-1}^2)}{\alpha_0(\alpha_0^2-\alpha_1^2)(\alpha_0^2-\alpha_2^2)\cdots(\alpha_0^2-\alpha_{k-1}^2)} \geq 2. \tag{3}$$

2 Asymptotic consequences of Theorem 1

Let $\tau = 2k-1$ be fixed and let n tend to infinity. We investigate the impact of Theorem 1 on $(2k-1)$-designs of cardinality of order n^{k-1}. We need the asymptotic behaviour of α_0. We also use some identities for orthogonal polynomials due to Levenshtein.

The Delsarte-Goethals-Seidel bound implies that

$$B_{odd}(n, 2k-1) \gtrsim \frac{2n^{k-1}}{(k-1)!},$$

where the inequality $\gtrsim$ should be interpreted as

$$\lim_{n\to+\infty} \frac{B_{odd}(n,2k-1)}{n^{k-1}} \geq \frac{2}{(k-1)!}.$$

Boyvalenkov-Danev-Nikova [3] improve this to

$$B_{odd}(n, 2k-1) \gtrsim \frac{(1+2^{1/\tau})n^{k-1}}{(k-1)!}.$$

In small dimensions, the bounds, obtained by condition from Theorem 1, are better than those of Boyvalenkov-Danev-Nikova [3] for $\tau \geq 5$. It turns out however that both bounds (from [3] and Theorem 1) give the same asymptotic consequences. The reason for this phenomenon is that the asymptotic behaviour of both bounds depends only on the asymptotics of α_0.

Lemma 3. *Let $n \to +\infty$ and k be fixed. Then all roots apart from α_0 of the equation $P_k^{1,0}(t)P_{k-1}^{1,0}(s) - P_k^{1,0}(s)P_{k-1}^{1,0}(t) = 0$ tend to zero and $\alpha_0 \sim \frac{P_k^{1,0}(s)}{P_{k-1}^{1,0}(s)}$.*

Proof. The first assertion follows from $|\alpha_0| > |\alpha_{k-1}| > |\alpha_1| > |\alpha_{k-2}| > \cdots$ (cf. [2, Appendix]) and $s = \alpha_{k-1} \leq t_k$ (the last one tends to zero when $n \to +\infty$ and k is fixed).

Now the behaviour of α_0 can be derived by the Viète formula

$$\begin{aligned} \sum_{i=1}^{k-1} \alpha_i &= -\frac{k}{n+2k-2}\left(1 - \frac{(n+2k-1)(n+k-2)}{k(n+2k-3)} \cdot \frac{P_k^{1,0}(s)}{P_{k-1}^{1,0}(s)}\right) \\ &= -\frac{k}{n+2k-2} + \frac{(n+2k-1)(n+k-2)}{(n+2k-2)(n+2k-3)} \cdot \frac{P_k^{1,0}(s)}{P_{k-1}^{1,0}(s)} \sim \frac{P_k^{1,0}(s)}{P_{k-1}^{1,0}(s)} \end{aligned}$$

as n tends to infinity and k is fixed. □

It follows from Lemma 3 and Corollary 2 that it is enough to find the asymptotic behaviour of the ratio $P_k^{1,0}(s)/P_{k-1}^{1,0}(s)$. This can be done, for example, by using the following identity due to Levenshtein.

Lemma 4 ([8], equality (5.86)). *For $s \in [t_{k-1}^{1,1}, t_k^{0,0})$ we have*

$$L_{2k-1}(n,s) = \left(1 - \frac{P_{k-1}^{1,0}(s)}{P_k^{(n)}(s)}\right) R(n,2k) = \left(1 - \frac{P_k^{1,0}(s)}{P_k^{(n)}(s)}\right) R(n,2k+2).$$

Theorem 5. *Let $n \to +\infty$, k be fixed and $C \subset \mathbf{S}^{n-1}$ be a $(2k-1)$-design of cardinality $|C| \sim R(n,2k+1) + \gamma n^{k-1} \sim \left(\gamma + \frac{2}{(k-1)!}\right) n^{k-1}$, where γ is some constant. Then $\alpha_0 \sim -\frac{1}{1+\gamma(k-1)!}$.*

We are now in a position to describe the asymptotic consequence of Theorem 1.

Theorem 6. *We have*

$$B_{odd}(n, 2k-1) \gtrsim \frac{1+2^{1/(2k-1)}}{(k-1)!} \cdot n^{k-1}.$$

The first three cases (3-, 5- and 7-designs) are $B_{odd}(n,3) \gtrsim 2.2599n$, $B_{odd}(n,5) \gtrsim 1.0743n^2$, and $B_{odd}(n,7) \gtrsim 0.3506n^3$, compared to the bounds $B_{odd}(n,3) \gtrsim 2n$, $B_{odd}(n,5) \gtrsim n^2$, and $B_{odd}(n,7) \gtrsim \frac{n^3}{3}$, which are ensured by the Delsarte-Goethals-Seidel bound.

3 Acknowledgments

The authors would like to thank Peter Boyvalenkov for the helpful discussions.

References

[1] Boyvalenkov, P., Boumova, S., and Danev, D. Necessary conditions for existence of designs in polynomial metric spaces. *European J. of Combinatorics 20* (1999), 213–225.

[2] Boyvalenkov, P., Danev, D., and Landgev, I. On maximal spherical codes II. *J. of Combinatorial Designs 7* (1999), 316–326.

[3] Boyvalenkov, P., Danev, D., and Nikova, S. Nonexistence of certain spherical designs of odd strengths and cardinalities. *Discr. and Comp. Geom. 21* (1999), 143–156.

[4] Boyvalenkov, P., and Nikova, S. New lower bounds for some spherical designs. *Lecture Notes in Computer Science 781* (1994), 207–216.

[5] Boyvalenkov, P., and Nikova, S. Improvements of the lower bounds for some spherical designs. *Mathematica Balkanica 12* (1998), 151–160.

[6] Delsarte, P., Goethals, J.-M., and Seidel, J. Spherical codes and designs. *Geom. Dedicata 6* (1977), 363–388.

[7] Levenshtein, V. Designs as maximum codes in polynomial metric spaces. *Acta Applicandae Math. 25* (1992), 1–83.

[8] Levenshtein, V. Universal bounds for codes and designs. *Chapter 6 (499-648) in Handbook of Coding Theory, Eds. V.Pless and W.C.Huffman, Elsevier Science B.V.* (1998).

[9] Yudin, V. Lower bounds for spherical designs. *Izvestiya: Mathematics 3, 61* (1997), 673–683.

About a Class of Binary Constant-Weight Codes

Iliya Bouyukliev and Veselin Vavrek

Institute of Mathematics, Bulgarian Academy of Sciences, P.O.Box 323, 5000 V. Tarnovo, Bulgaria e-mail: iliya@moi.math.bas.bg

Abstract

We present some results about binary constant-weight codes with parameters $(n = 4t, d = 2t, w = 2t - 1)$ and $M = 4t$ codewords. A complete classification of these codes for $n \leq 16$ and of 1-generated cyclic codes for $n \leq 30$ is given. All codes of a special class with $n < 60$ and the same codes with an additional condition for $n \leq 124$ are found.

1 Introduction

An (n, d, w) constant-weight binary code is a set of binary vectors of length n, such that each vector contains w ones and $n - w$ zeros, and any two vectors differ in at least d positions. For given values of n, d and w, the maximum integer M such that an (n, d, w) code with M codewords exists is denoted by $A(n, d, w)$.

The results related to this function are summarized in the encyclopedic work of Brouwer and al. [3] and Agrel and al. [1]. The existence of optimal $(n = 4t, d = 2t, w = 2t - 1)$ codes with $M = 4t$ has been proved for $t \leq 7$. We suppose that codes with these parameters exist for any t and they are optimal. That's why we consider codes with such parameters.

In this paper we classify the codes with $t \leq 4$, and the cyclic codes with $t \leq 8$. We construct a class of cyclic codes of this type with $t \leq 31$.

2 Classification Results

To classify the codes with $t \leq 4$, we have used the method given in [3] and a backtrack search with pruning techniques presented in [4]. For the code equivalence, we have used the algorithm of [2]. We present the results in the following table:

n	Num	AUT
8	1	1:48;
12	11	1:3; 1:6; 1:240; 2:16; 1:10; 1:12; 1:9; 1:144; 2:24;
16	54	14:1; 8:2; 15:3; 2:4; 2:6; 3:8; 2:12; 3:16; 1:24; 1:96; 1:48; 1:21; 1:672

Let a be a binary vector of length n. We define the cyclic code C_a as the set of all vectors obtained by cyclic shift from a. We classify all cyclic codes C_a with the parameters given above by exhaustive search and present the results in the following table:

n	Num	AUT
8	1	1:48;
12	4	1:12; 1:24; 1:144; 1:240;
16	2	1:16; 1:672;
20	8	7:20; 1:2880;
24	8	2:24; 5:48; 1:2640
28	21	14:28; 4:56; 1:168; 1:4032; 1:4368;
32	12	12:32;

3 Construction of Codes

For larger values of n it is difficult to construct codes C_a, that is why we investigate a more special class of cyclic codes. We consider codes C_a such that there exists exactly one codeword of distance $> d$ from a, and the other codewords are of distance d from a. We denote this class of cyclic codes by E.

Let $\rho_i(a)$ be the binary vector obtained by i-times cyclic shift from a.

The following Lemma holds:

Lemma 1 Let a be a binary vector such that $C_a \in E$. Then:

1. $\mathrm{d}(a, \rho_i(a)) = \mathrm{d}(a, \rho_{n-i}(a))$
2. $\mathrm{d}(a, \rho_{n/2}(a)) = n - 2$
3. $\exists b \in F_2^{n/2-1} : C_a \equiv C_{\overline{0b0(1+b)}}$

Proof: 1. It follows from:

$$\rho_i(a + \rho_{n-i}(a)) = a + \rho_i(a)$$

2. Let us consider the vectors $a + \rho_i(a)$ for $i = 1, \ldots, n-1$. So we obtain vectors of weight d and only one of them has weight $r > d$. The number of 1's in each coordinate is $\frac{n}{2} + 1$ or $\frac{n}{2} - 1$. Hence $r = n - 2$.
3. The vector $c = a + \rho_{n/2}(a)$ has exactly 2 zeros. Since $\rho_{n/2}(c) = c$, it follows that the zeros of c are at distance $n/2$ from each other. ■

We define the map

$$\chi(b) := \overline{0b0(1+b)}$$

Lemma 2. Let $a = (a_0, a_1, \dots, a_n)$, l be an integer relatively prime to n, and $b = (b_0, b_1, \dots, b_n)$ be the vector for which $b_i = a_{li \ mod \ n}, i = 0, \dots, n-1$. If $C_a \in E$ then $C_b \in E$.

Proof: We have $\mathrm{d}(b, \rho_i(b)) = \mathrm{d}(a, \rho_{i.(l^{-1} \ mod \ n)}(a))$ ■

Let $J = (1, 0, 1, 0, \dots, 0, 1)$, and

$$\eta((a_0, a_1, \dots, a_{2t-1})) = (a_{2t-1}, a_{2t-2}, \dots, a_1, a_0)$$

Lemma 3. If $C_{\chi(b)} \in E$ then

$$\{C_{\chi(b+1)}, C_{\chi(\eta(b))}, C_{\chi(b+J)}\} \subset E$$

Proof: To prove the lemma, we have used that $\rho(\chi(b)) = \chi(b+1)$ and lemma 2 when $l = n-1$ and $l = n/2+1$. ■

By a computer search the following lemma was proved:

Lemma 4. If $C_{\chi(b)} \in E$ for $b \in F_2^{2t-1}, t = 1, 2, \dots, 14$ then $b = \eta(b) + J$ when t is even, and $b = \eta(b) + J + 1$ when t is odd. ■

If we suppose that this lemma is true for $t > 14$. we can consider the vectors of length t to obtain codes of length $4t$. In the following table we present vectors b such that $C_{\chi(b)} \in E$.

n	b	Aut	n	b	Aut
4	0	24	56	1B01539	117936
8	1	48	60	04853E8E	48720
12	02	144	64	2B18593F	59520
16	27	672	76	023AC2C122	101232
20	05E	2880	84	0ACE10BA4C0	137760
24	247	2640	88	1E770152269	158928
28	013A	4368	96	3C46085D644B	207552
36	0F894	9792	100	0B403189B2AF0	470400
40	1D109	13680	108	078FBB02B11496	297648
48	302753	24288	120	0CBE11217116BCD	410640
52	0A189A0	62400	124	06CB805829EA90C6	453840

References

[1] E. Agrel, A. Vardy, K. Zeger, "Upper Bounds for Constant Weight Codes", *IEEE Trans, Info. Theory,* vol -46, 2000, pp. 2373-2395.

[2] I. Bouyukliev "An Algorithm for finding isomorphisms of codes", Proceedings of the International Workshop OCRT, Sunny Beach, Bulgaria, June 2001, pp. 35-41.

[3] A.E.Brouwer, J. B. Shearer,N.J.A. Sloane, and W.D. Smith, "A New Table of Constant Weight Codes", *IEEE Trans, Info. Theory,* vol 36, 1990, pp 1334-1380.

Some new results on optimal codes over $GF(4)$

Iliya Bouyukliev
Institute of Mathematics, Bulgarian Academy of Sciences, P.O.Box 323,
5000 V. Tarnovo, Bulgaria e-mail: iliya@moi.math.bas.bg
and Zlatko Varbanov
Faculty of Mathematics and Informatics, St.St.Cyril and Metodij
University, 5000 V. Tarnovo, Bulgaria

Abstract

We prove that $d_4(27,5) = 17$, $d_4(31,5) = 20$, $d_4(24,6) = 14$, $d_4(28,6) = 17$, $d_4(24,7) = 13$, $d_4(102,5) = 74$, $d_4(231,5) = 172$, $d_4(226,5) = 168$, $d_4(221,5) = 164$. Moreover, we classify the quaternary optimal codes for some values of n and k.

1. Introduction

A central problem in coding theory is to find $d_q(n,k)$, the largest value of d for which an $[n,k,d;q]$-code exists. An $[n,k,d_q(n,k);q]$ code is called optimal. Another important problem is to classify all optimal codes with given parameters.

Until now, the following exact values for the function $d_q(n,k)$ have been known: Bouyukliev, Jaffe and Vavrek in [2] finished the problem for $d_2(n,k)$ when $k \leq 8$. Landgev finished this problem for $q = 3$ and $k \leq 5$ [10]. The quaternary case was considered in [1], [7], [8], [11], etc. The last unknown values of $d_4(n,k)$ for $k \leq 4$ were found in [11]. Bounds for $d_q(n,k)$ have been published in Brouwer's tables in [4].

In this paper, we investigate quaternary linear codes. Our main approach is an algorithm for the extension of codes using their residual codes.

There are some classification results of optimal linear codes over $GF(4)$ related to geometrical constructions [8], Near-MDS-codes [6], and [9], [12], etc. We present some new classification results.

2. About Q-EXTENSION

Let G be a generator matrix of a linear $[n, k, d; q]$-code C. Then the **residual code** $Res(C, c)$ of C with respect to a codeword c is the code generated by the restriction of G to the columns where c has a zero entry.

Lemma 2.1 [5] Suppose C is an $[n, k, d]$-code over GF(q) and suppose $c \in C$ has weight w, where $d > w(q-1)/q$. Then $Res(C, c)$ is an $[n-w, k-1, d']$-code with $d' \leq d - w + \lceil w/q \rceil$.

The program contains two main approaches for extension of codes. The first one is the extension up to length which is the construction of an $[n, k, d]$ code on the basis of an $[n-w, k-1, d']$ code as its residual code, or on the base of an $[n-i, k, d']$ code. The second one is the extension up to dimension which is the extension of an $[n, k, d]$ code to $[n+i, k+i, d]$ or $[n+i+1, k+i, d]$ code. If G is a generator matrix for a $[n, k, d]$ code, we extend it to

$$\left(\begin{array}{c|c} * & I_i \\ \hline G & 0 \end{array} \right) \quad or \quad \left(\begin{array}{c|c} * & 1\ I_i \\ \hline G & 0 \end{array} \right) \qquad (1)$$

where I_i is the identity matrix. We take the matrix G in systematic form, thus we can fix k columns more.

More information on this topic can be found in [3].

3. New bounds for $d_4(n, k)$

We have proved the nonexistence of several codes with given parameters and in this way we have found new upper bounds for the function $d_4(n, k)$.

Theorem 3.1. $d_4(27, 5) = 17$ and $d_4(31, 5) = 20$.

Proof: There exists a unique [9,4,5]-code over $GF(4)$. Using Q - Extension, we prove that this code cannot be extended to a [27,5,18;4]- code, therefore $d_4(27, 5) = 17$. There exist exactly two inequivalent [10,4,6;4] codes. Using Q-Extension, we see that no one of them can be extended to a [31,5,21;4]-code, therefore $d_4(31, 5) = 20$.

Corollary 3.2. No code with parameters $[28, 6, 18; 4]$ exists, therefore $d_4(28, 6) = 17$.

Theorem 3.3. $d_4(24, 6) = 14$ and $d_4(24, 7) = 13$.

Proof: There exist exactly 19 optimal [9,5,4]-codes over $GF(4)$. Using Q-Extension, we obtain that none of these codes can be extended to a [24,6,15;4]-code, and hence $d_4(24, 6) = 14$. There exist exactly 23 optimal [10,6,4;4]-codes. Using Q-Extension, we obtain that no one of them can be extended to a [24,7,14;4]- code, therefore $d_4(24, 7) = 13$.

Corollary 3.4. $d_4(25, 7) \leq 14, d_4(25, 8) \leq 13, d_4(26, 8) \leq 14, d_4(26, 9) \leq 13, d_4(27, 9) \leq 14, d_4(27, 10) \leq 13, d_4(28, 10) \leq 14$.

Theorem 3.5. $d_4(102, 5) = 74, d_4(231, 5) = 172, d_4(226, 5) = 168, d_4(221, 5) = 164$.

Proof: We construct by heuristic search codes with parameters: [102,5,74], [231,5,172], [226,5,168], [221,5,164]. The weight functions of the codes are:

$[102, 5, 74] - 1 + 489z^{74} + 306z^{76} + 15z^{80} + 162z^{82} + 30z^{84} + 21z^{90}$

$[231, 5, 172] - 1 + 885z^{172} + 63z^{176} + 75z^{188}$

$[226, 5, 168] - 1 + 825z^{168} + 120z^{172} + 3z^{176} + 75z^{184}$

$[221, 5, 164] - 1 + 768z^{164} + 171z^{168} + 9z^{172} + 75z^{180}$

4. Classification results

We have presented some classification results for $n \leq 32$. The codes with parameters [16,3,12;4], [20,3,15;4] and [21,3,16;4] are McDonalds so each of them is unique. All results are obtained by Q-extension. We summarize them in the table.

$q = 4$	3		4		5		6	
	d	number	d	number	d	number	d	number
10	6	**45**	6	2 [6]	5	**4**	4	**23**
11	7	**25**	6		6	1 [6]	5	1 [6]
12	8	**16**	7		6		6	1 [6]
13	9	**4**	8	**28**	7		6	
14	10	**2**	9	**1**	8	**6**	7	
15	11	1	10	**1**	8		8	**3**
16	12	1 MD	11	**1**	9		8	
17	12	**12**	12	1 BCH	10		9	
18	13	**2**	12		10		10	
19	14	**1**	12		11		10	
20	15	1 MD	13		12		11	
21	16	1 MD	14		13		12	
22	16	**6**	15	**15**	14		12-13	
23	16		16	**3**	15	**2**	13-14	
24	17	**102**	16		16	1 [9]	**14**	
25	18	**27**	17		16		15	
26	19	**5**	18	**48**	16-17		16	
27	20	1	19	**2**	**17**		16-17	
28	20		20	**1**	18		**17**	
29	21	**38**	20		19		17-18	
30	22	**13**	21	**6**	20		18-19	
31	23	**3**	22	**1**	**20**		19-20	
32	24	1	22		21		20	
33	24		23		22		20-21	
34	25		24		22-23		21-22	

References

[1] M.Bhandari, M.Garg, "Optimum codes of dimension 3 and 4 over GF(4)", *IEEE Trans. Info. Theory*, vol. IT-38, pp. 1564–1567, 1992.

[2] I.Bouyukliev, D.Jaffe, and V.Vavrek, The Smallest Length of Eight-Dimensional Binary Linear Codes with Prescribed Minimum Distance, *IEEE Trans. Inform. Theory*, Vol.46, 4,pp.1539-1544, 2000.

[3] I.Bouyukliev and J.Simonis, "Some new results for optimal ternary linear codes", to appear in *IEEE Trans, Info. Theory*.

[4] A.E.Brouwer, "Bounds on the size of linear codes", in *Handbook of Coding Theory*, Edited by V.Pless and W.C.Huffman, Elsevier, Amsterdam etc., ISBN:0-444-50088-X, 1998.

[5] S. Dodunekov, "Minimal block length of a q-ary code with prescribed dimension and code distance", *Problems of Inform. Transmission*, vol. 20, No. 4, pp. 239-249, 1984.

[6] S. Dodunekov, I. Landgev, On the quaternary [11,6,5] and [12,6,6] codes, Applications of Finite Fields (ed. D. Gollmann), IMA Conference Series 59, Clarendon Press, Oxford, pp. 75-84, 1996.

[7] P.P.Greenough and R.Hill, "Optimal linear codes over $GF(4)$", *DiscreteMathematics*, vol.125, pp. 187-199, 1994.

[8] N. Hamada," A survey of recent work on characterization of minihypers in $PG(t,q)$ and nonbinary linear codes meeting the Griesmer bound", *J. Combin. Inform. Syst. Sci.* Vol. 18, pp. 161–191, 1993.

[9] R. Hill and P. Lizak, Extensions of linear codes, Proc. IEEE Int. Symposium on Inform. Theory, Ulm, Germany, p 114, 1997.

[10] I. Landgev, The nonexistence of some ternary fivedimensional codes, *Designs, Codes and Cryptography*, 15, pp. 245-258, 1998.

[11] I. Landgev, T. Maruta and R. Hill, On the nonexistence of quaternary [51,4,37] codes, Finite Fields and Their Applications 2, pp. 96–110, 1996.

[12] P. Östergård, "Classifying subspaces of Hamming spaces", to appear in *Designs, Codes and Cryptography*.

Optimal and asymptotically optimal SCEC two-dimensional array codes

I.M. Boyarinov

Institute for System Analysis
Russian Academy of Sciences
60 years of October ave. 9
117312, Moscow, Russia
e-mail: i.boyarinov@mtu-net.ru

Abstract – Two-dimensional array codes that can correct two-dimensional clusters (or bursts) of errors are presented. Constructions of optimal and asymptotically optimal single cluster-error-correcting (*SCEC*) two-dimensional array codes are given.

1 Introduction

There are data transmission and storage systems with two-dimensional data structures that suffer from two-dimensional clusters of errors. For correction of two-dimensional clusters of errors two-dimensional array codes were constructed (see [1]–[5] and the references therein). According to the Singleton–type bound [4], [6], the redundancy r required for a $b_1 \times b_2$ cluster error-correcting code is

$$r \geq 2b_1 b_2. \tag{1}$$

By analogy with one-dimensional codes array codes that meet the bound (1) are said to be optimal. The ratio $z = \frac{2b_1 b_2}{r}$ can be used as a measure of cluster-error-correcting efficiency of an array code. An optimal array code has the cluster-error-correcting efficiency z equal to 1. The cluster-error-correcting efficiency z of asymptotically optimal array codes approaches 1 as the size $n_1 \times n_2$ of array codes becomes large. Known *SCEC* two-dimensional array codes have the cluster-error-correcting efficiency z equal (or approximately equal) to $\frac{2}{3}$ or less. In this paper we give some constructions of optimal and asymptotically optimal *SCEC* two-dimensional array codes.

2 Optimal *SCEC* codes

For integers $n_i,\ i = 1,2$ we consider the set $V(n_1, n_2)$ of all binary two-dimensional $n_1 \times n_2$ arrays. A linear K–dimensional $(K \leq n_1 n_2)$ subspace $C(n_1, n_2)$ of the space $V(n_1, n_2)$ is called a linear binary two-dimensional array $[n_1 \times n_2,\ K]$ code of size $n_1 \times n_2$ with K information symbols and $r = n_1 n_2 - K$ parity-check symbols.

Lemma 1 *Let α be a root of the polynomial $x^3 + x + 1$ over $GF(2)$ and C be the binary extended Hamming $[8,4]$ code with the parity-check matrix*

$$H = \begin{bmatrix} 1 & \alpha & \alpha^2 & \alpha^3 & \alpha^4 & \alpha^5 & \alpha^6 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \qquad (2)$$

If $c^{(1)} = (c_1^{(1)}, c_2^{(1)}, c_3^{(1)}, c_4^{(1)}, c_5^{(1)}, c_6^{(1)}, c_7^{(1)}, c_8^{(1)})$ and $c^{(2)} = (c_1^{(2)}, c_2^{(2)}, c_3^{(2)}, c_4^{(2)}, c_5^{(2)}, c_6^{(2)}, c_7^{(2)}, c_8^{(2)})$ are code words of the code C, then the 4×4 array

$$v = \begin{bmatrix} c_1^{(1)} & c_2^{(1)} & c_3^{(1)} & c_4^{(1)} \\ c_1^{(2)} & c_2^{(2)} & c_3^{(2)} & c_4^{(2)} \\ c_5^{(1)} & c_6^{(1)} & c_7^{(1)} & c_8^{(1)} \\ c_5^{(2)} & c_6^{(2)} & c_7^{(2)} & c_8^{(2)} \end{bmatrix} \qquad (3)$$

is a code word of the optimal binary two-dimensional array $[4 \times 4, 8]$ code V correcting single error clusters of size 2×2.

Applying the approach of Lemma 1 to binary cyclic Abramson $(2^m - 1,\ 2^m - m - 2)$ codes correcting error burst of length 2 [7] we can construct linear binary two-dimensional array $[n_1 \times n_2,\ K]$ codes correcting single error clusters of size 2×2 where $n_1 n_2 \leq 2^{m+1} - 2$ and $K \leq n_1 n_2 - 2m - 2$.

In Lemma 1 we used the row interleaving scheme with the interleaving degree 2. Using optimal binary one-dimensional array codes correcting single error bursts of length b_1 (for example, codes from Table 9.3 [8]) and row interleaving schemes with the interleaving degree b_2 we can construct optimal binary two-dimensional array codes correcting single error clusters of size $b_1 \times b_2$ for any integers b_1 and b_2.

3 Asymptotically optimal *SCEC* codes

Let $l_i, b_i,\ i = 1,2$ be positive integers and $l_1 l_2 \leq 2^{b_1 b_2}$. Let

$$u = \begin{bmatrix} u_{1,1} & u_{1,2} & \cdots & u_{1,l_2} \\ u_{2,1} & u_{2,2} & \cdots & u_{2,l_2} \\ \cdots & \cdots & \cdots & \cdots \\ u_{l_1,1} & u_{l_1,2} & \cdots & u_{l_1,l_2} \end{bmatrix} \qquad (4)$$

be a two-dimensional $l_1 \times l_2$ array over $GF(2^{b_1 b_2})$ such that $(u_{1,1}, u_{1,2}, \ldots, u_{1,l_2}, u_{2,1}, u_{2,2}, \ldots, u_{2,l_2}, \ldots, u_{l_1,1}, u_{l_1,2}, \ldots, u_{l_1,l_2})$ is a code word of a (shortened) Reed-Solomon $(l_1 l_2,\ l_1 l_2 - 2)$ code over $GF(2^{b_1 b_2})$ correcting single errors.

There is a one-to-one correspondence between elements of the Galois field $GF(2^{b_1 b_2})$ and binary $b_1 \times b_2$ matrices. Representing every element $u_{i,j}$ in u as the binary $b_1 \times b_2$ matrix, we obtain a code word u^* of the optimal binary two-dimensional array $[l_1 b_1 \times l_2 b_2,\ l_1 l_2 b_1 b_2 - 2b_1 b_2]$ code U^* correcting single phased error clusters of size $b_1 \times b_2$.

Given a single-error-correcting (SEC) two-dimensional array $[l_1 \times l_2,\ l_1 l_2 - 2]$ code over $GF(2^{b_1 b_2})$ we construct the two-dimensional array $[\lambda_1 l_1 \times \lambda_2 l_2,\ \lambda_1 \lambda_2 l_1 l_2 - 2\lambda_1 \lambda_2]$ code W over $GF(2^{b_1 b_2})$ by two-dimensional interleaving with row interleaving degree λ_1 and column interleaving degree λ_2. Representing every symbol of a code word w of the code W as the binary $b_1 \times b_2$ matrix, we obtain a code word $w*$ of the binary two-dimensional array $[\lambda_1 l_1 b_1 \times \lambda_2 l_2 b_2,\ \lambda_1 \lambda_2 l_1 l_2 b_1 b_2 - 2\lambda_1 \lambda_2 b_1 b_2]$ code W^*.

Lemma 2 *The binary two-dimensional array code W^* corrects single error clusters of size $((\lambda_1 - 1)b_1 + 1) \times ((\lambda_2 - 1)b_2 + 1)$. As the interleaving degrees λ_1 and λ_2 become large, the cluster-error-correcting efficiency z of the code W^* approaches 1.*

References

[1] H. Imai, "Two-dimensional Fire codes," *IEEE Trans.Inform. Theory,* vol. IT-19, pp. 786–806, May 1973.

[2] K.A.S. Abdel-Ghaffar, R.J. McEliece, and H.C.A. van Tilborg, "Two-dimensional burst identification codes and their use in burst correction," *IEEE Trans.Inform. Theory,* vol. IT-34, pp. 494–504, May 1988.

[3] M. Blaum and P.G. Farrel,"Array codes for cluster-error correction," *Electronics Letters,* vol. 30, pp. 1752–1753, May 1994.

[4] E.M. Gabidulin and V.V. Zanin, "Codes Correcting Two-Dimensional Burst Errors," in *the 3^{th} International Symposium on Communication Theory and Applications (ISTA95)* , Ambleside, UK, 10-14 July 1995, pp. 66–78.

[5] M. Breitbach, M. Bossert, V. Zyablov, and V. Sidorenko, "Array codes correcting a two-dimensional cluster of errors," *IEEE Trans.Inform. Theory,* vol. IT-44, pp. 2025–2031, Sept. 1998.

[6] M. Bossert and V. Sidorenko, "Singleton type bounds for blot correcting codes," *IEEE Trans.Inform. Theory,* vol. IT-42, pp. 1021–1023, May. 1996.

[7] N.M. Abramson, "A class of systematic codes for non-independent errors," *IRE Trans.Inform. Theory,* vol. IT-5, pp. 150–157, Dec. 1959.

[8] S. Lin, and D.J.Jr. Costello, *Error Control Coding: Fundamentals and Applications.* Englewood Cliffs, NJ, Prentice-Hall, 1983.

On spherical (n, M, ℓ, τ)-sets

Peter Boyvalenkov
Institute of Mathematics and Informatics
Bulgarian Academy of Sciences
8 G.Bonchev str., 1113 Sofia, BULGARIA
peter@moi.math.bas.bg

Maya Stoyanova
Faculty of Mathematics and Informatics
Sofia University
5 James Baucher blvd, Sofia, BULGARIA
stoyanova@fmi.uni-sofia.bg

Abstract

We consider spherical codes which admit exactly two different nonzero distances between their points and which are spherical 2- or 3-designs. In the later case we prove that such codes are maximal.

1 Introduction

A spherical code C is a finite nonempty subset of the n-dimensional unit sphere $\mathbb{S}^{n-1}$. Denote by $\ell = \ell(C)$ the number of distinct inner products of different points of C. Then C is called an ℓ-distance spherical set.

A spherical τ-design is a spherical code $C \subset \mathbb{S}^{n-1}$ such that

$$\frac{1}{\mu(\mathbb{S}^{n-1})}\int_{\mathbb{S}^{n-1}} f(x)d\mu(x) = \frac{1}{|C|}\sum_{x\in C} f(x)$$

($\mu(x)$ is the Lebesgue measure) holds for all polynomials $f(x) = f(x_1, x_2, \ldots, x_n)$ of degree at most τ (i.e. the average of f over the set C is equal to the average of f over $\mathbb{S}^{n-1}$). The number τ is called strength of C.

We consider (n, M, ℓ, τ)-sets which are spherical τ-designs on $\mathbb{S}^{n-1}$ with $\ell = \ell(C)$. Since many known maximal ℓ-distance sets are spherical designs of suitable strength, we decided to investigate further this connection. A general bound by Delsarte-Goethals-Seidel [5] says that $\tau \leq 2\ell$.

For $\ell = 2$ and $\tau = 4$ the corresponding codes are tight spherical 4-designs [5, 2, 3]. So we consider the next two cases $\tau = 2$ and $\tau = 3$.

2 Some preliminaries

Let $C \subset \mathbb{S}^{n-1}$ be a spherical code and $x \in C$. Then the system $\{A_t(x) : -1 \leq t < 1\}$ of integers

$$A_t(x) = |\{y \in C : \langle x, y\rangle = t\}|$$

is called distance distribution of C with respect to x. We take only the nonzero entries in the distance distribution. If C is distance regular, i.e. $A_t(x)$ does not depend on $x \in C$, we denote $A_t(x) = A_t$.

The following definition for spherical designs is crucial for our approach. If $C \subset \mathbb{S}^{n-1}$ is a spherical τ-design then for every point $y \in C$ and for every real polynomial $f(t)$ of degree at most τ, the equality

$$\sum_{x \in C\setminus\{y\}} f(\langle x, y\rangle) = f_0|C| - f(1). \tag{1}$$

holds, where

$$f_0 = c_n \int_{-1}^{1} f(t)(1-t^2)^{(n-3)/2}dt, \quad c_n = \frac{\Gamma(n-1)}{2^{n-2}(\Gamma(\frac{n-1}{2}))^2}.$$

(f_0 is the first coefficient in the expansion $f(t) = \sum_{i=0}^{k} f_i P_i^{(n)}(t)$ in terms of the Gegenbauer polynomials [1, Chapter 22]).

3 The case $\tau = 3$

Let $C \subset \mathbb{S}^{n-1}$ be a 2-distance set and a spherical 3-design. Then

$$2n \leq |C| \leq \frac{n(n+3)}{2}.$$

If the upper bound is attained, than C is already 4-design. Since all feasible parameter sets of 4-designs are determined [2, 3] we assume that $|C| < n(n+3)/2 - 1$. Then we consider the whole range (despite the feasible codes with fewer than $n(n+1)/2$ points would not be maximal 2-distance sets).

Theorem 1. C is a spherical $(n, M, 2, 3)$-set if and only if it is a maximal code which attains the Levenshtein bound $L_3(n, s)$ [6].

Scetch of proof. The "$\Leftarrow$" direction is well known. For the "$\Rightarrow$" direction, we observe that C is distance regular and compute its distance distribution. This gives a connection between the inner products and the size $M = |C|$ which implies that $M = L_3(n, s)$, where $s = s(C)$ is the larger inner product of C.

Corollary 2. (Lloyd-type theorem) If C is a spherical $(n, M, 2, 3)$-set then both inner products are rationals.

Proof. This follows from the Llyod-type theorem in [4, Section 3] which states that all spherical codes which attain the Levenshtein bound $L_3(n, s)$ have rational inner products.

4 The case $\tau = 2$

Let $C \subset \mathbb{S}^{n-1}$ be a 2-distance set and a spherical 2-design. Then

$$n + 1 \leq |C| \leq \frac{n(n+3)}{2}$$

and C is distance regular. Denote $M = |C| = n + 1 + k$, where $0 \leq k \leq (n-1)(n+2)/2$ is integer.

The following lemma can be proved by applying a little algebra on the equations which are given by (1) for suitable polynomials.

Lemma 3. If C is a spherical $(n, M, 2, 2)$-set then its inner products t_1 and t_2 are

$$t_1 = -\frac{\sqrt{D}}{n(n+k)A_{t_1}} - \frac{1}{n+k}, \quad t_2 = \frac{\sqrt{D}}{n(n+k)A_{t_2}} - \frac{1}{n+k},$$

where $D = nk(n+k+1)A_{t_1}A_{t_2}$ is integer.

We proceed by investigation of the derived codes of C. These codes are 1-designs and 2-distance sets, i.e. they are also distance regular. Now calculations of their distance distribution and Lemma 3 give the following:

Theorem 4. (Lloyd-type theorem) If C is a spherical $(n, M, 2, 2)$-set then both inner products are rationals or n is even, $M = 2n + 1$ and the inner products are $t_{1,2} = \frac{-1\pm\sqrt{2n+1}}{2n}$.

Interestingly, the later case in Theorem 4 can be realized. We have a construction of a $(6, 13, 2, 2)$-set with inner products $t_{1,2} = \frac{-1\pm\sqrt{13}}{12}$. Moreover, our construction (as in [3, Section 4]) shows that this code is unique up to isometry.

Acknowledgment. This research was partially supported by the Framework Cultural Agreement between University of Perugia (Italy) and Institute of Mathematics of Bulgarian Academy of Sciences (Bulgaria) and the Bulgarian NSF under Contract MM-901/99.

References

[1] M. Abramowitz, I.A. Stegun, *Handbook of Mathematical Functions*, Dover, New York, 1965.

[2] E. Bannai, R.M. Damerell, Tight spherical designs I, *J. Math. Soc. Japan* 31, 1979, 199-207.

[3] P.G. Boyvalenkov, Computing distance distribution of spherical designs, *Lin. Alg. Appl.* 226/228, 1995, 277-286.

[4] P.G. Boyvalenkov, I.N. Landgev, On maximal spherical codes I, Proc. XI Intern. Symp. AAECC, Paris 1995; Springer–Verlag *Lect. Notes Comp. Sci.* 948, 158-168.

[5] P. Delsarte, J.-M. Goethals, J.J. Seidel, Spherical codes and designs, *Geom. Dedicata* 6, 1977, 363-388.

[6] V.I. Levenshtein, Universal bounds for codes and designs, Chapter 6 (499-648) in *Handbook of Coding Theory*, Eds. V. Pless and W.C. Huffman, Elsevier Science B.V., 1998.

Complete classification of the doubly perfect binary additive superimposed codes

Danyo Danev
Department of Electrical Engineering
Linköping University
SE-581 83 Linköping, SWEDEN
danyo@isy.liu.se

Abstract

The classification of the so called doubly perfect binary additive superimposed codes has been an open problem for about ten years. In this paper we prove that the only sets of parameters for which such codes could exist are these already known.

1 Introduction

Superimposed codes were originally considered by Kautz and Singleton [5]. They used a binary "OR" operation as a superposition mechanism. Here we study another type which was first defined by Chien and Frazer in [2]. The superposition mechanism for this type is an addition modulo two (i.e. binary "XOR") of the codewords. A possible application of this scheme is in identification systems.

As usual with $H(n,2)$ we denote the set of all n-tuples with elements from the alphabet $GF(2) = \{0,1\}$. The Hamming distance $d_H(x,y)$ between two elements x and y is defined to be the number of positions where they differ. Let C be an arbitrary subset of $H(n,2)$. We denote by C_m^* the multi-set defined as

$$C_m^* \triangleq \{x_1 + \cdots + x_s : 0 \leq s \leq m,\ x_i \in C\ i = 1,\ldots,s, x_i \neq x_j, \text{ if } i \neq j\}.$$

In other words C_m^* consists of all superpositions of up to m elements from C. We set the emty sum to be the all-zero vector in $H(n,2)$. For the multi-set C_m^* we define its the minimum distance $d(C_m^*)$ as

$$d(C_m^*) \triangleq \min\{d_H(x,y) : x,y \in C_m^*,\ x \neq y\},$$

if C_m^* is genuine set and $d(C_m^*) = 0$ otherwise.

Definition 1. *The set C is called a (n,d,m,T) binary additive superimposed code (BASC) if $C \subseteq H(n,2)$, $|C| = T$ and $d(C_m^*) = d$.*

We denote by $T(n,d,m)$ the maximal T for which (n,d,m,T) superimposed code exists. In order to determine the value of $T(n,d,m)$ we have to find upper and lower bounds on this quantity. A general construction with the aid of binary linear codes is described in [4]. It gives a lower bound on $T(n,d,m)$ which is

$$T(n,d,m) \geq \max\{T : T \leq K(n,d) + K(T,2m+1)\}, \tag{1}$$

where $K(n,d)$ denotes the greatest possible dimension of a binary linear code of length n and minimum distance d. A general implicit upper bound [4, eq. 6] is

$$\sum_{i=0}^{m} \binom{T(n,d,m)}{i} \leq A(n,d), \tag{2}$$

where $A(n,d)$ is the greatest possible cardinality of a binary code of length n and minimum distance d.

2 The classification theorem

One of the first bounds on the size of unrestricted binary codes with prescribed minimum distance is the well known sphere-packing or Hamming bound. It states that

$$A(n,2t+1) \leq A_H(n,2t+1) \triangleq \frac{2^n}{\sum_{i=0}^{t}\binom{n}{i}}.$$

The codes with parameters $(n, A_H(n,d), d)$ are known as *perfect codes*. One reason for this name is the fact that all the spheres of radius t with centers in the codewords constitute a partition of the whole Hamming space $H(n,d)$. Complete classification of the parameters for which binary perfect codes exist is done by Tietäväinen in 1973 [7].

The upper bound (2) on the size of a binary additive superimposed code is based on the same arguments as the Hamming bound. Hence a BASC attaining this bound is called *perfect superimposed code*. If, moreover, the equality $A(n,d) = A_H(n,d)$ occurs the code is referred to as a *doubly perfect superimposed code*. The task of determining the set of all possible parameters for which perfect superimposed codes exist seems to be impossible to solve. The main obstacle is the lack of knowledge of the function $A(n,d)$ for almost all pairs (n,d). However the classification of the doubly perfect superimposed codes seems to be within reach due to Tietäväinen's result. Indeed Theorem 4 in [4] has almost completed this task. The only uncleared situation is the eventual existence of non-systematic binary perfect codes. At the time of writing [4] such codes were not known to exists. Solov'eva and Avgustinovich [1] in 1996 showed their existence for the parameters $(2^k-1, 2^{2^k-k-1}, 3)$ for all $k \geq 8$. Later the existence of such codes in the cases $k = 5,6,7$ was shown by Phelps and Levan [6]. Considering these discoveries the classification of the doubly perfect superimposed codes was not completed in [4]. However the conjecture formulated there that the described cases give all the possibilities turns out to be true.

Here we state the result proving this conjecture.

Theorem 1 (Classification of the non-trivial DPSC). *Let X be a doubly perfect superimposed code with parameters (n,d,m,T) such that $T > m \geq 2$. Then the 4-tuple (n,d,m,T) is equal to one of the following 4-tuples*

1) $(2m, 1, m, 2m+1)$, $m \in \mathbb{N}$;

2) $(11, 1, 3, 23)$;

3) $(2^{2s+1}-1, 3, 2^{2s}-s-1, 2^{2s+1}-2s-1)$, $s \in \mathbb{N}$;

4) $(15, 3, 3, 23)$;

5) $(23, 7, 6, 13)$;

Proof. Binary superimposed codes with parameters described above exist in all cases. All these codes can be obtained by the main construction. The codes used are given in [4, Table II]. We have to prove that these are the only possibilities.

According to the definition if X is a doubly perfect superimposed code we have the equality

$$T_m^* \triangleq |X_m^*| = \sum_{i=0}^{m}\binom{T}{i} = A_H(n,d)$$

and the set X_m^* constitutes a binary perfect code. Such codes exist only for odd distances d. We consider three cases.

I) Let $d = 1$. Then $X_m^* = H(n,2)$ and thus X_m^* is a systematic perfect code. Applying Theorem 4 in [4] we obtain that X is either a trivial DPSC (i.e. $T = m$) or there exists a perfect linear $[T, T-n, 2m+1]_2$-code. When $m \geq 2$ we have two possibilities $(T, T-n, 2m+1) = (2m+1, 1, 2m+1)$ and $(T, T-n, 2m+1) = (23, 12, 7)$ from which we get cases 1) and 2), respectively.

II) Let $d = 3$. We have

$$A_H(n,3) = \frac{2^n}{n+1} = T_m^*.$$

Applying Theorem 2 from [3] we obtain

$$T_m^* = \frac{2^n}{n+1} \geq \frac{2^T}{A(T,2m+1)} \geq \frac{2^T}{A_H(T,2m+1)} = \sum_{i=0}^{m}\binom{T}{i} = T_m^*,$$

which implies $A_2(T,2m+1) = A_H(T,2m+1)$. This means that there exists a binary perfect $(T, A_H(T,2m+1), 2m+1)$-code. Recall the assumption $m \geq 2$. There are two possibilities. Let first $(T, A_H(T,2m+1), 2m+1) = (2m+1, 2, 2m+1)$. Thus $T = 2m+1$ and from the equation for T_m^* we get

$$\frac{2^n}{n+1} = T_m^* = \sum_{i=0}^{m}\binom{2m+1}{i} = 2^{2m},$$

giving the equality $n+1 = 2^{n-2m}$. If $n - 2m = 0$ we have $n = m = 0$. Otherwise n must be an odd number and $n - 2m = 2s+1$ holds for some $s \in \mathbb{N}$. This gives case 3). What is left to consider is $(T, A_H(T, 2m+1), 2m+1) = (23, 2^{12}, 7)$. This directly implies $T = 23$, $m = 3$ and $2^n/(n+1) = 2^{12}$. Since n is a positive number we have the unique solution $n = 15$. These are the parameters given in case 4).

III) The rest of the possibilities for d to consider are $d \geq 5$. For these minimum distances there exist two types of perfect codes. The first are the repetition codes with parameters $(2t+1, 2, 2t+1)$. In this case we have $T_m^* = 2$ which is impossible when $T \geq m \geq 2$. The second type is actually only one code — the well known Golay code with parameters $(23, 2^{12}, 7)$. Thus $T_m^* = \sum_{i=0}^{m} \binom{T}{i} = 2^{12}$ and we have to solve a Diofant equation in natural numbers $T > m \geq 2$. This can be done by exhaustive search since for m we clearly have $2 \leq m \leq 11$. We obtain two solutions $(T, m) \in \{(90, 2), (13, 6)\}$. The second solution provides the parameters in case 5). Since the Golay code is unique and systematic, according [4, Theorem 4] existence of $(23, 7, 2, 90)$-BASC implies the existence of a perfect linear $[90, 78, 5]_2$-code which is a contradiction. Thus the solution $(T, m) = (90, 2)$ does not provide a DPSC. This concludes the proof. □

In the theorem we did not consider the possibilities $T = m$. They correspond to trivial doubly perfect superimposed codes. All these codes are actually bases of perfect binary linear codes and the classification of their parameters comes from [7].

References

[1] Avgustinovich, S. V., and Solov'eva, F. I. On nonsystematic perfect binary codes. *Problemy Peredachi Informatsii 32*, 3 (1996), 47–50.

[2] Chien, R. T., and Frazer, W. D. An Application of Coding Theory to Document Retrieval. *IEEE Transactions on Information Theory 12*, 2 (April 1966), 92–96.

[3] Danev, D. A Construction of Optimal Superimposed Codes in the Binary Hamming Space. In *Proceedings International Workshop on Algebraic and Combinatorial Coding Theory* (Bansko, Bulgaria, June 2000), pp. 107–111.

[4] Ericson, T., and Levenshtein, V. I. Superimposed Codes in the Hamming Space. *IEEE Transactions on Information Theory 40*, 6 (November 1994), 1882–1893.

[5] Kautz, W. H., and Singleton, R. C. Nonrandom Binary Superimposed Codes. *IEEE Transactions on Information Theory 10* (October 1964), 363–377.

[6] Phelps, K. T., and Levan, M. Nonsystematic perfect codes. *SIAM J. Discrete Math. 12*, 1 (1999), 27–34 (electronic).

[7] Tietäväinen, A. On the nonexistence of perfect codes over finite fields. *SIAM J. Appl. Math. 24* (1973), 88–96.

New Quasi-Twisted Degenerate Linear Codes over GF(9) *

Rumen Daskalov and Plamen Hristov
Department of Mathematics
Technical University of Gabrovo
5300 Gabrovo, Bulgaria, daskalov@tugab.bg

Abstract

Let $[n, k, d]_q$-code be a linear code of length n, dimension k and minimum Hamming distance d over $GF(q)$. In this paper, eleven new codes over GF(9) are constructed, which improve the known lower bounds on the minimum distance.

1 Introduction

Let $GF(q)$ denote the Galois field of q elements. A linear code over $GF(q)$ of length n, dimension k and minimum Hamming distance d is called an $[n, k, d]_q$-code.

A code C is said to be quasi-twisted (QT or p-QT) if a constacyclic shift of a codeword by p positions results in another codeword. A constacyclic shift of an m-tuple $(x_0, x_1, \ldots, x_{m-1})$ is the m-tuple $(\alpha x_{m-1}, x_0, \ldots, x_{m-2}), \alpha \in GF(q)\backslash\{0\}$.

The blocklength, n, of a QT code is a multiple of p, so that $n = mp$ for some integer m. A matrix B of the form

$$B = \begin{bmatrix} b_0 & b_1 & b_2 & \cdots & b_{m-2} & b_{m-1} \\ \alpha b_{m-1} & b_0 & b_1 & \cdots & b_{m-3} & b_{m-2} \\ \alpha b_{m-2} & \alpha b_{m-1} & b_0 & \cdots & b_{m-4} & b_{m-3} \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ \alpha b_1 & \alpha b_2 & \alpha b_3 & \cdots & \alpha b_{m-1} & b_0 \end{bmatrix}, \qquad (1)$$

where $\alpha \in GF(q)\backslash\{0\}$ is called a *twistulant matrix*. A class of QT codes can be constructed from $m \times m$ twistulant matrices (with a suitable permutation of coordinates). In this case the generator matrix G can be represented as

$$G = [B_1, B_2, \ldots, B_p], \qquad (2)$$

*This work was partially supported by the Bulgarian National Science Fund.

where B_i is a twistulant matrix.

The algebra of $m \times m$ twistulant matrices over $GF(q)$ is isomorphic to the algebra of polynomials in the ring $GF(q)[x]/(x^m - \alpha)$ if B is mapped onto the polynomial, $d(x) = b_0 + b_1x + b_2x^2 + \cdots + b_{m-1}x^{m-1}$, formed from the entries in the first row of B. The $d_i(x)$ associated with a QT code are called the *defining polynomials* [3, 4]. If $\alpha = 1$, we obtain the algebra of $m \times m$ circulant matrices [6], and a subclass of quasi-cyclic codes (QC). If $p = 1$ then we obtain codes, which we call *twisted* codes (T). If $\alpha = 1$ and $p = 1$ then we obtain a subclass of well-known cyclic codes.

If the defining polynomials $d_i(x)$ contain a common factor which is also a factor of $x^m - \alpha$, then the QT code is called *degenerate* [3, 4]. The dimension k of the QT code is equal to the degree of $h(x)$, where [8]

$$h(x) = \frac{x^m - \alpha}{\gcd\{x^m - \alpha, d_1(x), d_2(x), \cdots, d_p(x)\}}. \qquad (3)$$

If the polynomial $h(x)$ has degree m, the dimension of the code is m, and (2) is a generator matrix. If $\deg(h(x)) = k < m$, a generator matrix for the code can be constructed by deleting $m - k$ rows of (2).

Let the defining polynomials of the code C be in the next form

$$d_1(x) = g(x),\ d_2(x) = f_2(x)g(x),\ \cdots\ ,\ d_p(x) = f_p(x)g(x), \qquad (4)$$

where $g(x)|(x^m - \alpha), g(x), f_i(x) \in GF(q)[x]/(x^m - \alpha)$, $\gcd\{f_i(x), (x^m - \alpha)/g(x)\} = 1$ and $\deg f_i(x) < m - \deg g(x)$ for all $1 \le i \le p$.

Then we obtain a degenerate QT code, which, by analogy with one-generator QC codes, we call one-generator QT code and for this code $n = mp, \quad k = m - \deg g(x)$.

Quasi-twisted (QT) codes [4] form an important class of linear codes which are a natural generalization of the QC codes. QT codes were first defined by Berlekamp [1] as constacyclic codes. In this paper, new one-generator twisted codes are constructed using a nonexhaustive algebraic-combinatorial computers search, similar to that in [7]. The codes presented here improve the respective lower bounds on the minimum distance in [2].

2 The New Codes

We have restricted our search to one-generator QT codes with defining polynomials in the form (4). In all cases we have taken $p = 1$. The main aim in our search is to find good $g(x)$, i.e. $g(x)$ which gives better minimum distance. We illustrate the search method in the following example. Let $m = 61$, $\alpha = 8$ and $q = 9$. Then

$$x^m - \alpha = \prod_{i=1}^{13} h_i(x),$$

where irreducible factors $h_i(x)$ are as follows

$$\begin{array}{ll}
h_1(x) = x^5 + x^4 + 5x^3 + 8x^2 + x + 1 & h_2(x) = x^5 + x^4 + 8x^3 + 5x^2 + x + 1 \\
h_3(x) = x^5 + 6x^4 + 3x^3 + 6x^2 + 3x + 1 & h_4(x) = x^5 + 3x^4 + 6x^3 + 3x^2 + 6x + 1 \\
h_5(x) = x^5 + 6x^4 + x^3 + x^2 + 3x + 1 & h_6(x) = x^5 + 3x^4 + x^3 + x^2 + 6x + 1 \\
h_7(x) = x^5 + 7x^3 + 4x^2 + 1 & h_8(x) = x^5 + 4x^3 + 7x^2 + 1 \\
h_9(x) = x^5 + 8x^4 + 4x^3 + 7x^2 + 5x + 1 & h_{10}(x) = x^5 + 5x^4 + 7x^3 + 4x^2 + 8x + 1 \\
h_{11}(x) = x^5 + 7x^4 + 3x^3 + 6x^2 + 4x + 1 & h_{12}(x) = x^5 + 4x^4 + 6x^3 + 3x^2 + 7x + 1 \\
h_{13}(x) = x + 1 &
\end{array}$$

There are $\binom{12}{2} = 66$ possibilities to obtain polynomial $g(x)$ of degree 51, which is a factor of $x^m - \alpha$. From these polynomials 64 have Hamming weight greater than or equals 38 (The best-known code is a $[61, 10, 38]_4$-code.). Taken one of these 64 polynomials

$$g(x) = \prod_{i=1}^{6} h_i(x) \prod_{i=8}^{10} h_i(x) \prod_{i=12}^{13} h_i(x),$$

we obtain new twisted $[61, 10, 39]_9$ code.

Now, we present the new codes. The parameters of these codes are given in the Table 1. The minimum distances, d_{br} [2], of the previously best known codes are given for comparison. For convenience, the coefficients of the defining polynomials are given as integers - $\alpha = 5$, $\alpha^2 = 3$, $\alpha^3 = 8$, $\alpha^4 = 2$, $\alpha^5 = 7$, $\alpha^6 = 6$, $\alpha^7 = 4$, where α is a root of the ternary primitive polynomial $x^2 + 1$. The defining polynomials are listed with the lowest degree coefficient on the left. The coefficients of the defining polynomials of the new codes are as follows:

A $[37, 9, 23]_9$**-code:** 1306568348061805130576538306100000000;

A $[41, 8, 28]_9$**-code:** 17426514818713875578317818415624710000000;

The dual of the above code is $[41, 33, 6]_9$**-code** with $B_6 = 20992$.

A $[41, 9, 25]_9$**-code:** 18767368620108652568010268637678100000000;

A $[61, 5, 49]_9$**-code:**
1804126677606353820513365865283583661802568630344332170510000;

A $[61, 6, 46]_9$**-code:**
1752206070633051748642450814715087273574180663040302284100000;

A $[61, 10, 39]_9$**-code:**
1883672353624600172826275588423252410037236862436551000000000;

A $[73, 7, 57]_9$**-code:**
1835422825321247454362087375366253526635737802634547421235282245381000000;

The dual of the above code is an **optimal** $[73, 66, 6]_9$**-code** with $B_6 = 11135712$.

Table 1: Minimum distances of the new linear codes over GF(9).

code	d	d_{br}	code	d	d_{br}	code	d	d_{br}
[37,9]	23	22	[61,5]	49	48	[73,66]	6	5
[41,8]	28	26	[61,6]	46	45	[85,10]	59	58
[41,9]	25	24	[61,10]	39	38	[91,10]	64	63
[41,33]	6	5	[73,7]	57	55			

A $[85, 10, 59]_9$-code:
16216285527803035166581481058817068571227444014735481346212835180880083762210000000000;

A $[91, 10, 64]_9$-code:
1448072334108076207600284242240118172710100054382884036206318324626822347356027731000000000;

Remark: The weight distributions of the dual codes are obtained with the aid of QLC [5], using the MacWilliams' identities. By this reason B_i denotes the number of codewords of weight i in the dual code.

References

[1] E.R. Berlekamp, *Algebraic Coding Theory*, McGraw-Hill, New York, 1968.

[2] A.E. Brouwer, Linear code bound [electronic table; online], `http://www.win.tue.nl/~aeb/voorlincod.html`.

[3] P.P. Greenough and R. Hill, "Optimal ternary quasi-cyclic codes," *Designs, Codes and Cryptography*, vol. 2, (1992), 81-91.

[4] R. Hill and P.P. Greenough, "Optimal quasi-twisted codes," Proc. *Int. Workshop Algebraic and Comb. Coding Theory*, Voneshta Voda, Bulgaria, (1992), 92-97.

[5] S.N. Kapralov, P. Hristov, G.T. Bogdanova,"The new version of QLC - a computer program for linear codes studying", Proc. *Int.Workshop on Optimal Codes and Related Topics*, Sozopol, Bulgaria, (1995), 11-14.

[6] F.J. MacWilliams and N.J.A. Sloane, *The Theory of Error-Correcting Codes*, New York, NY: North-Holland Publishing Co., 1977.

[7] I. Siap, N. Aydin and D. Ray-Chaudhury, "New ternary quasi-cyclic codes with better minimum distances", *IEEE Trans. Inform. Theory*, vol. 46, no. 4, (2000) 1554-1558.

[8] G.E. Séguin and G. Drolet, "The theory of 1-generator quasi-cyclic codes," Technical Report, Royal Military College of Canada, Kingston, ON, 1991.

New Ternary Linear Codes in Dimensions 18 and 19 *

Rumen Daskalov

Department of Mathematics
Technical University of Gabrovo
5300 Gabrovo, Bulgaria, daskalov@tugab.bg

Abstract

Let $[n, k, d]_q$-code be a linear code of length n, dimension k and minimum Hamming distance d over $GF(q)$. Twelve new ternary ($q = 3$) linear codes are constructed in this paper.

1 Introduction

Let $GF(q)$ denote the Galois field of q elements. A linear code C over $GF(q)$ of length n, dimension k and minimum Hamming distance d is called an $[n, k, d]_q$-code.

A code C is said to be quasi-cyclic (QC or p-QC) if a cyclic shift of a codeword by p positions results in another codeword. A cyclic shift of an m-tuple $(x_0, x_1, \dots, x_{m-1})$ is the m-tuple $(x_{m-1}, x_0, \dots, x_{m-2})$.

The blocklength, n, of a p-QC code is a multiple of p, so that $n = pm$ for some integer m [4]. A matrix B of the form

$$B = \begin{bmatrix} b_0 & b_1 & b_2 & \cdots & b_{m-2} & b_{m-1} \\ b_{m-1} & b_0 & b_1 & \cdots & b_{m-3} & b_{m-2} \\ b_{m-2} & b_{m-1} & b_0 & \cdots & b_{m-4} & b_{m-3} \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ b_1 & b_2 & b_3 & \cdots & b_{m-1} & b_0 \end{bmatrix} \qquad (1)$$

is called a *circulant matrix*. A class of QC codes can be constructed from $m \times m$ circulant matrices (with a suitable permutation of coordinates [11]). In this case the generator matrix G can be represented as

$$G = [B_1, B_2, \dots, B_p], \qquad (2)$$

*This work was partially supported by the Bulgarian National Science Fund.

where B_i is a circulant matrix.

The algebra of $m \times m$ circulant matrices over $GF(q)$ is isomorphic to the algebra of polynomials in the ring $GF(q)[x]/(x^m - 1)$ if B is mapped onto the polynomial, $d(x) = b_0 + b_1x + b_2x^2 + \cdots + b_{m-1}x^{m-1}$, formed from the entries in the first row of B. The $d_i(x)$ associated with a QC code are called the *defining polynomials* [4, 6].

If the defining polynomials $d_i(x)$ contain a common factor which is also a factor of $x^m - 1$, then the QC code is called *degenerate* [4, 6]. The dimension k of the QC code is equal to the degree of $h(x)$, where [10]

$$h(x) = \frac{x^m - 1}{\gcd\{x^m - 1, d_1(x), d_2(x), \cdots, d_p(x)\}}. \qquad (3)$$

If the polynomial $h(x)$ has degree m, the dimension of the code is m, and (2) is a generator matrix. If $\deg(h(x)) = k < m$, a generator matrix for the code can be constructed by deleting $m - k$ rows of (2).

Quasi-cyclic codes form am important class of linear codes which contains the well-known class of cyclic codes. The investigation of QC codes is motivated by the following facts: QC codes meet a modified version of Gilbert-Varshamov bound [2]; some of the best quadratic residue codes and Pless symmetry codes are QC codes [8]; a large number of record breaking (and optimal codes) are QC codes [1].

In this paper, new QC codes are constructed using a nonexhaustive heuristic combinatorial computers search, similar to that in [3], [5], [9]. The codes presented here improve the respective lower bounds on the minimum distance in [1].

2 The New QC Codes

In this section, we present the new quasi-cyclic codes. The parameters of these codes are given in Table I. The minimum distances, d_{br} [1], of the previously best known codes are given for comparison.

Theorem 1: There exist quasi-cyclic codes with parameters:

$$[90, 18, 38]_3, [95, 18, 41]_3, [126, 18, 58]_3, [133, 18, 62]_3, [144, 18, 68]_3, [152, 18, 73]_3.$$

Prof. The coefficients of the defining polynomials of these codes are as follows:

A $[90, 18, 38]_3$**-code:**
011121111211122101, 000000001011121202, 000000000102201021, 000112210221022222, 000000000000000001;

A $[95, 18, 41]_3$**-code:**
111111111120002012, 121112210021210221, 111221220211120200, 001221121211122211, 000000011212111211;

A $[126, 18, 58]_3$**-code:**
011121111211122101, 011112121112210010, 000112122200200012, 000112210221022222, 000000000102201021, 000000001011121202, 000000000000000001;

A $[133, 18, 62]_3$**-code:**
112212121111112000, 000121121111111120, 111111111120002012, 121112210021210221, 000000011212111211, 111221220211120200, 001221121211122211;

A $[144, 18, 68]_3$**-code:**
000112122200200012, 011121111211122101, 011112121112210010, 001111212121112001, 000000001011121202, 000112210221022222, 000000000102201021, 000000000000000001;

A $[152, 18, 73]_3$**-code:**
111212221211211010, 112212121111112000, 000121121111111120, 111111111120002012, 111221220211120200, 001221121211122211, 121112210021210221, 000000011212111211;

Theorem 2: There exist quasi-cyclic codes with parameters:

$$[57, 19, 20]_3, [76, 19, 30]_3, [95, 19, 40]_3, [114, 19, 51]_3, [133, 19, 61]_3, [152, 19, 72]_3.$$

Prof. The coefficients of the defining polynomials of these codes are as follows:

A $[57, 19, 20]_3$**-code:**
0000102222122101111, 0000001002102111221, 0000000000000000001;

A $[76, 19, 30]_3$**-code:**
0000112011001222100, 0000001002102111221, 0000102222122101111, 0000000000000000001;

A $[95, 19, 40]_3$**-code:**
0001111212112120111, 0000112011001222100, 0000102222122101111, 0000001002102111221, 0000000000000000001;

A $[114, 19, 51]_3$**-code:**
0121112122000220200, 0000112011001222100, 0001111212112120111, 0000102222122101111, 0000001002102111221, 0000000000000000001;

A $[133, 19, 61]_3$**-code:**
0001121211120101221, 0000102222122101111, 0000112011001222100, 0121112122000220200, 0000001002102111221, 0001111212112120111, 0000000000000000001;

A $[152, 19, 72]_3$**-code:**
0000122121112222021, 0001121211120101221, 0121112122000220200, 0001111212112120111, 0000112011001222100, 0000102222122101111, 0000001002102111221, 0000000000000000001;

In the following Table I we compare the minimum distances of the new codes with the previously best known minimum distances d_{br} in [1].

Table I: New codes over GF(3).

code	d	d_{br}	code	d	d_{br}
[90,18]	38	37	[57,19]	20	19
[95,18]	41	40	[76,19]	30	28
[126,18]	58	57	[95,19]	40	38
[133,18]	62	60	[114,19]	51	48
[144,18]	68	67	[133,19]	61	60
[152,18]	73	72	[152,19]	72	71

References

[1] A.E. Brouwer, Linear code bound [electronic table; online], `http://www.win.tue.nl/~aeb/voorlincod.html`.

[2] T. Kasami, "A Gilbert-Varshamov bound for quasi-cyclic codes of rate 1/2", *IEEE Trans. Inform. Theory*, vol. IT-20, (1974), 679–680.

[3] R.N. Daskalov, T.A. Gulliver, "New good quasi-cyclic ternary and quaternary linear codes", *IEEE Trans. Inform. Theory*, vol.43, no.5, (1997), 1647–1650.

[4] P.P. Greenough and R. Hill, "Optimal ternary quasi-cyclic codes," *Designs, Codes and Crypt.*, vol. 2, (1982), 81-91.

[5] T.A.Gulliver and P.R.J.Őstergard, "Improved bounds for ternary linear codes of dimension 7", *IEEE Trans. Inf. Theory*, vol.43, no.4, (1997), 1377–1381.

[6] R. Hill and P.P. Greenough, "Optimal quasi-twisted codes," Proc. *Int. Workshop Algebraic and Comb. Coding Theory*, Voneshta Voda, Bulgaria, (1992), pp.92–97.

[7] K. Lally and P. Fitzpatrick, "Construction and classification of quasi-cyclic codes," Proc. *Int. Workshop on Coding and Cryptography, WCC'99*, Paris, France, (1999), 11–20.

[8] F.J. MacWilliams and N.J.A. Sloane, *The Theory of Error-Correcting Codes*, New York, NY: North-Holland Publishing Co., 1977.

[9] I. Siap, N. Aydin and D. Ray-Chaudhury, "New ternary quasi-cyclic codes with better minimum distances", *IEEE Trans. Inform. Theory*, vol. 46, no. 4, (2000), 1554–1558.

[10] G.E. Séguin and G. Drolet, "The theory of 1-generator quasi-cyclic codes," Technical Report, Royal Military College of Canada, Kingston, ON, 1991.

[11] K. Thomas, "Polynomial approach to quasi-cyclic codes", *Bul. Cal. Math. Soc.*, vol.69, (1977), 51–59.

Binary self-dual codes with an automorphism of non-prime order

R. A. Dontcheva*, A. J. van Zanten,
ITS, TU Delft Mekelweg 4, 2628 CD Delft, The Netherlands
S. M. Dodunekov
IMI, Acad. G. Bonchev Str., Bl. 8, 1113 Sofia, Bulgaria

Abstract

In this note we present some results on a certain decomposition of a binary self-dual code having an automorphism of an order which is the product of two odd prime numbers. These results are applied to construct self-dual $[34, 17, 6]$ codes with an automorphism of order 15.

1 Introduction

Let C be a self-dual $[n, n/2, d]$ code and let *Aut* C be the automorphism group of C.

Lemma 1. *[4, Lemma 3] Let C be a self-dual code with an automorphism ϕ of odd prime order p. Then, the cyclic group $\langle\phi\rangle$ generated by ϕ is a Sylow p-subgroup of Aut C.*

Suppose that the code C has an automorphism σ of order pr, where p and r are odd prime numbers. Due to Lemma 1, the numbers p and r have to be different and we assume $p < r$. Since the automorphism σ has order pr, σ can contain nontrivial cycles of length p, r and pr. Denote by $\Omega_1, \ldots, \Omega_{t_1}$ the cycles of length p, by $\Omega_{t_1+1}, \ldots, \Omega_{t_1+t_2}$ the cycles of length r, by $\Omega_{t_1+t_2+1}, \ldots, \Omega_{t_1+t_2+t_3}$ the cycles of length pr, and by $\Omega_{t_1+t_2+t_3+1}, \ldots, \Omega_{t_1+t_2+t_3+f}$ the fixed points of σ. Hence, for the length of code C we have

$$n = t_1p + t_2r + t_3pr + f. \tag{1}$$

In this note, we shall call a permutation of order N, having f fixed points and t_1 cycles of length a_1, t_2 cycles of length a_2, $\ldots$, t_h cycles of length a_h, with $1 < a_1 < a_2 < \cdots < a_h$, a permutation of type N-$(t_1, t_2, \ldots, t_h;\ f)$. For reasons of convenience, the parameters a_1, $a_2, \ldots, a_h$ have been left out from this notation. Hence, σ is of type pr-$(t_1, t_2, t_3;\ f)$.

*On leave from University of Shumen, Bulgaria. This work was supported in part by the Shumen University under Grant N15/14.03.2002

Proposition 1. *Let $\mathcal{C}$ be a self-dual $[n, n/2, d]$ code and let $\mathcal{C}$ have an automorphism σ of type pr-$(t_1, t_2;\ f)$. Then $\mathcal{C}$ has automorphisms of type p-$(t_1 + t_3r;\ t_2r + f)$ and of type r-$(t_2 + t_3p;\ t_1p + f)$.*

Proof. The cycles of the permutation σ are independent and therefore they commute. Hence $\sigma^p = \Omega_1^p\, \Omega_2^p\, \dots\, \Omega_{t_1+t_2+t_3}^p$. For $1 \leq i \leq t_1$ we have that Ω_i has length p, and so, Ω_i^p consists of fixed points with respect to σ^p.

For the cycle Ω_j, $t_1 + 1 \leq j \leq t_1 + t_2$, of length r, it follows that Ω_j^p is again a cycle of length r, since p is prime to r.

It will also be obvious that Ω_k^p, $t_1 + t_2 + 1 \leq k \leq t_1 + t_2 + t_3$, is the product of p cycles of length r with respect to σ^p. Thus, $\Omega_{t_1+t_2+1}^p \Omega_{t_1+t_2+2}^p \cdots \Omega_{t_1+t_2+t_3}^p$ is a product of t_3p cycles of length r.

So, we may conclude that σ^p consists of $t_2 + t_3p$ cycles of length r and of $t_1p + f$ fixed points, or equivalently, $\mathcal{C}$ has an automorphism of type r-$(t_2 + t_3p;\ t_1p + f)$.

For similar reasons it follows that the permutation σ^r consists of $t_1 + t_3r$ cycles of length p and $t_2r + f$ fixed points. Hence, the code $\mathcal{C}$ has also an automorphism of type p-$(t_1 + t_3r;\ t_2r + f)$. □

We define linear subcodes $F_\sigma(\mathcal{C})$ and $E_\sigma(\mathcal{C})$ as follows

$$F_\sigma(\mathcal{C}) = \{v \in \mathcal{C} |\ \ \sigma(v) = v\}, \tag{2}$$

$$E_\sigma(\mathcal{C}) = \{v \in \mathcal{C} |\ \ wt(v|\Omega_i) = 0\ (mod\ 2),\ i = 1, \dots, t_1 + t_2 + t_3 + f\}, \tag{3}$$

where $v|\Omega_i$ is the restriction of v to Ω_i.

It is clear that $v \in F_\sigma(\mathcal{C})$ if and only if $v \in \mathcal{C}$ is constant on each cycle Ω_j, $j = 1, 2, \dots, t_1 + t_2 + t_3 + f$. Let the projection map π be defined by

$$\pi :\ F_\sigma(\mathcal{C}) \to \mathbb{F}_2^{t_1+t_2+t_3+f}, \qquad \pi(v|\Omega_i) = v_j, \tag{4}$$

for some $j \in \Omega_i$, $i = 1, 2, \dots, t_1 + t_2 + t_3 + f$.

With the above notation the next proposition holds.

Proposition 2. *If $\mathcal{C}$ is binary self-dual code having an automorphism of type pr-$(t_1, t_2, t_3;\ f)$, then $\pi(F_\sigma(\mathcal{C}))$ is a binary self-dual code of length $t_1 + t_2 + t_3 + f$.*

Proof. The code $\mathcal{C}$ is self-dual, and hence, any pair of vectors $u = (u_1, u_2, \dots, u_n)$, $v = (v_1, v_2, \dots, v_n) \in \mathcal{C}$ satisfies $(u, v) = 0$. In particular for $u, v \in F_\sigma(\mathcal{C})$ we can write

$$0 = (u, v) = (\pi(u), \pi(v)) \quad (mod\ 2),$$

since p, r and pr are odd, and since u and v are constant on each cycle Ω_i, $1 \leq i \leq t_1 + t_2 + t_3 + f$. Hence, we have $\pi(F_\sigma(\mathcal{C})) \subseteq \pi(F_\sigma(\mathcal{C}))^\perp$.

In order to prove that the equality sign holds in this last relation, we first show that $\mathcal{C} = F_\sigma(\mathcal{C}) + E_\sigma(\mathcal{C})$. Take an arbitrary vector $v \in \mathcal{C}$. For $v' := \sum\limits_{j=0}^{pr-1} \sigma^j(v)$, we have $\sigma(v') = v'$, and so $v' \in F_\sigma(\mathcal{C})$. If we write $v = v' + v''$, it follows that $v'' = \sum\limits_{j=1}^{pr-1} \sigma^j(v)$. The automorphism σ permutes the coordinates of $v|\Omega_i$, and hence, $wt(\sigma^j(v|\Omega_i)) = wt(v|\Omega_i)$ for $0 \leq j < pr$ and $1 \leq i \leq t_1 + t_2 + t_3 + f$. Therefore, we have $wt(\sigma(v''|\Omega_i)) = (pr - 1)\ wt(v|\Omega_i) = 0\ (mod\ 2)$, and $v'' \in E_\sigma(\mathcal{C})$.

Now, let $x \in \pi(F_\sigma(\mathcal{C}))^\perp$. Let furthermore w be a vector of length n, such that w is constant on Ω_i with value x_i for $1 \leq i \leq t_1 + t_2 + t_3 + f$.

For any $v \in \mathcal{C}$ we now have

$$(w, v) = (w, v' + v'') = (w, v') = \sum_{i=1}^{n} w_i v_i' = \sum_{j=1}^{t_1+t_2+t_3+f} x_j y_j = 0 \quad (mod\ 2),$$

where $y = \pi(v')$. So, $w \in \mathcal{C}^\perp$, but then also $w \in \mathcal{C}$, since $\mathcal{C}$ is self-dual. Hence, $x = \pi(w) \in \pi(F_\sigma(\mathcal{C})$, or $\pi(F_\sigma(\mathcal{C}))^\perp \subseteq \pi(F_\sigma(\mathcal{C}))$. □

Proposition 3. *A self-dual code $\mathcal{C}$ as described in Proposition 2 can be decomposed according to $\mathcal{C} = F_\sigma(\mathcal{C}) \oplus E_\sigma(\mathcal{C})$, where $\oplus$ stands for the direct sum of linear subspaces, and $dim\ E_\sigma(\mathcal{C}) = \frac{1}{2}(t_1(p-1) + t_2(r-1)) + t_3(pr-1))$.*

Proof. In the proof of Proposition 2 we derived $\mathcal{C} = F_\sigma(\mathcal{C}) + E_\sigma(\mathcal{C})$. Let $v \in F_\sigma(\mathcal{C}) \bigcap E_\sigma(\mathcal{C})$. Since $wt(v|\Omega_i)$ is even and since p, r and pr are odd, it immediately follows that $v = 0$, and so $\mathcal{C}$ is the direct sum of the two subspaces. The remaining part of the proposition is a consequence of the self-duality of $F_\sigma(\mathcal{C})$. □

Proposition 4. *If $\mathcal{C}$ is a doubly-even self-dual code, and if $p = 1\ (mod\ 4)$ and $r = 1\ (mod\ 4)$, then $\pi(F_\sigma(\mathcal{C}))$ is a doubly-even self-dual code.*

Proof. Since $p = 1\ (mod\ 4)$, $r = 1\ (mod\ 4)$, and $\mathcal{C}$ is a doubly-even code, it follows immediately that $0 = wt(v) = wt(\pi(v))\ \ (mod\ 4)$. This implies, by definition, that $\pi(F_\sigma(\mathcal{C}))$ is doubly-even. □

Let $\mathcal{P}_1$, $\mathcal{P}_2$ and $\mathcal{P}$ be the sets of even-weight polynomials respectively in $\mathbb{F}_2[x]/(x^p - 1)$, $\mathbb{F}_2[x]/(x^r - 1)$ and $\mathbb{F}_2[x]/(x^{pr} - 1)$. It is well known that $\mathcal{P}_1$, $\mathcal{P}_2$ and $\mathcal{P}$ are cyclic codes of lengths p, r and pr.

Any vector of length p, r or pr and with even weight can be identified with a polynomial in $\mathcal{P}_1$, $\mathcal{P}_2$ and $\mathcal{P}$, respectively.

Let $E_\sigma(\mathcal{C})^*$ be the code $E_\sigma(\mathcal{C})$ with the last f coordinates deleted. Remember that the coordinates of any vector of $E_\sigma(\mathcal{C})$ in the fixed points of σ are equal to zero.

We define the map $\varphi :\ \ E_\sigma(\mathcal{C})^* \to \mathcal{P}_1 \times \mathcal{P}_2 \times \mathcal{P}$ by identifying a restricted vector $v|\Omega_i = (v_0, v_1, \dots, v_{p-1})$ with the polynomial $\varphi(v|\Omega_i)(x) = v_0 + v_1x + \cdots + v_{p_1-1}x^{p-1}$ in $\mathcal{P}_1$ for $1 \leq i \leq t_1$, and similarly for $t_1 < i \leq t_1 + t_2$ and $t_1 + t_2 < i \leq t_1 + t_2 + t_3$.

2 An application

Let $\mathcal{C}$ be an extremal self-dual $[34, 17, 6]$ code. The possible weight enumerators for such a code are derived in [1]:

$$W_{34,1}(y) = 1 + (34 - 4\beta)y^6 + (255 + 4\beta)y^8 + (1921 + 20\beta)y^{10} + (8466 - 20\beta)y^{12} + \cdots, \tag{5}$$

$$W_{34,2}(y) = 1 + 6y^6 + 411y^8 + 1165y^{10} + 10886y^{12} + \cdots, \tag{6}$$

where β is an undetermined integer parameter.

We suppose that $\mathcal{C}$ has an automorphism σ of type $3 \cdot 5$-$(t_1, t_2, t_3;\ f)$. Hence,

$$34 = 3t_1 + 5t_2 + 15t_3 + f. \tag{7}$$

From Proposition 1 it follows that $\mathcal{C}$ has also an automorphism of type 3-$(t_1 + 5t_3;\ 5t_2 + f)$ and of type 5-$(t_2 + 3t_3;\ 3t_1 + f)$.

From [2, Theorem 1] we know that the code $\mathcal{C}$ can have automorphisms of odd prime order of type 5-(6; 4), 3-(10; 4), 3-(8; 10), 3-(6; 16) or 3-(4; 22). Therefore t_1, t_2, t_3 and f satisfy

$$4 \le t_1 + 5t_3 \le 10, \quad t_2 + 3t_3 = 6, \quad 4 \le 5t_2 + f \le 22, \quad 3t_1 + f = 4. \tag{8}$$

Combining (8) and (7) leaves only two possible types of automorphisms, i.e. $3 \cdot 5$-(1, 3, 1; 1) and $3 \cdot 5$-(0, 0, 2; 4).

Suppose that the code $\mathcal{C}$ has an automorphism σ of type $3 \cdot 5$-(1, 3, 1; 1). We may denote the automorphism σ as

$$\sigma = \Omega_1\ \Omega_2\Omega_3\Omega_4\ \Omega_5\ \Omega_6, \tag{9}$$

where Ω_1 is a cycle of length 3, Ω_2, Ω_3, Ω_4 are cycles of length 5, Ω_5 is a cycle of length 15 and Ω_6 is a fixed point.

Let the subcode $F_\sigma(\mathcal{C})$ and the map $\pi : F_\sigma(\mathcal{C}) \to \mathbb{F}_2^6$ be defined as in (2) and (4), respectively. From Proposition 2 it follows that $\pi(F_\sigma(\mathcal{C}))$ has to be a self-dual [6, 3] code. From [3, Table 2] it follows that, up to equivalence, there exists only one such code, which is C_2^3. Hence, up to permutations of the first 5 coordinates, the generator matrix of $\pi(F_\sigma(\mathcal{C}))$ is

$$X' = \left(\begin{array}{ccccc|c} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{array}\right).$$

So, the generator matrix of $F_\sigma(\mathcal{C})$, which correspond to X', has the form

$$X = \left(\begin{array}{ccccc|c} a & b & & & & 0 \\ & & b & b & & 0 \\ & & & & c & 1 \end{array}\right), \tag{10}$$

where a, b, c are all-one vectors of lengths 3, 5 and 15, respectively, whereas non-indicated entries are equal to zero vectors.

Denote by $\mathcal{P}_1$, $\mathcal{P}_2$ and $\mathcal{P}$ the sets of even-weight polynomials in the factor-rings $\mathbb{F}_2[x]/(x^3 - 1)$, $\mathbb{F}_2[x]/(x^5 - 1)$ and $\mathbb{F}_2[x]/(x^{15} - 1)$, respectively. Since the polynomials $x^2 + x + 1$ and $x^4 + x^3 + x^2 + x + 1$ are irreducible over $\mathbb{F}_2$, $\mathcal{P}_1$ and $\mathcal{P}_2$ are fields with 2^2 and 2^4 elements. These fields have idempotents $l_1(x) = x^2 + x + 1$ and $l_2(x) = x^4 + x^3 + x^2 + x + 1$, whereas $\alpha_1(x) = \alpha_2(x) = x + 1$ is a primitive element of $\mathcal{P}_1$ as well as of $\mathcal{P}_2$.

The decomposition of the polynomial $x^{15} - 1$ over the binary field is $x^{15} - 1 = (x - 1)h_1(x)h_2(x)h_3(x)h_4(x)$, where $h_1(x) = 1 + x + x^2 + x^3 + x^4$, $h_2(x) = 1 + x^3 + x^4$, $h_3(x) = 1 + x + x^4$ and $h_4(x) = 1 + x + x^2$ are irreducible polynomials over $\mathbb{F}_2$. Let I_j be the ideal of $\mathcal{P}$ generated by the polynomial $\frac{x^{15}-1}{h_j(x)}$. Then $\mathcal{P} = I_1 \oplus I_2 \oplus I_3 \oplus I_4$. The idempotents are: $e_1(x) = x^{14} + x^{13} + x^{12} + x^{11} + x^9 + x^8 + x^7 + x^6 + x^4 + x^3 + x^2 + x$, $e_2(x) = x^{14} + x^{13} + x^{12} + x^{11} + x^9 + x^7 + x^6 + x^3$, $e_3(x) = x^{12} + x^9 + x^8 + x^6 + x^4 + x^3 + x^2 + x$ and $e_4(x) = x^{14} + x^{13} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x$. As primitive elements we take $\mu_1(x) = x^{11} + x^{10} + x^6 + x^5 + x + 1$, $\mu_2(x) = x^{14} + x^{13} + x^{11} + x^9 + x^8 + x^5 + x + 1$, $\mu_3(x) = x^{11} + x^8 + x^7 + x^5 + x^3 + x^2 + x + 1$ and $\mu_4(x) = x^{13} + x^{12} + x^{10} + x^9 + x^7 + x^6 + x^4 + x^3 + x + 1$.

From Proposition 3 we have that the dimension of the subcode $E_\sigma(\mathcal{C})$ is 14. Denote by $E_\sigma(\mathcal{C})^*$ the code $E_\sigma(\mathcal{C})$ with the last coordinate deleted. Corresponding to the map $\varphi : E_\sigma(\mathcal{C})^* \to \mathcal{P}_1 \times \mathcal{P}_2 \times \mathcal{P}$, we have a generator matrix for the subcode $E_\sigma(C)^*$ of the form

$$Y = \begin{array}{cc} \begin{matrix} 3 & 5 & 5 & 5 & 15 \end{matrix} & \\ \begin{pmatrix} r_1 & & & & s_1 \\ & u_1 & u_2 & u_3 & s_2 \\ & u_4 & u_5 & u_6 & s_3 \\ & & & & s_4 \end{pmatrix} & \begin{matrix} \}2 \\ \}4 \\ \}4 \\ \}4 \end{matrix} \end{array}, \tag{11}$$

where r_1, s_1, u_i, for $i = 1, \ldots, 6$, and s_j, for $j = 2, 3, 4$, are right-circulant matrices of size 2×3, 2×15, 4×5 and 4×15, respectively, and where non-indicated entries are equal to zero. The first rows of the circulant matrices r_1, u_i and s_j correspond to polynomials of $\mathcal{P}_1$, $\mathcal{P}_2$ and $\mathcal{P}$.

We are able now to present some possible generator matrices of $\mathcal{D}_{34}$.

Proposition 5. *Let an extremal self-dual* [34, 17, 6] *code* $\mathcal{C}$ *have an automorphism of type* $3 \cdot 5$-(1, 3, 1; 1). *Then a possible generator matrix of* $\mathcal{C}$ *can be written as*

$$G = \left(\begin{array}{c|c} \multicolumn{2}{c}{X} \\ \hline Y & \begin{matrix} 0 \\ \vdots \\ 0 \end{matrix} \end{array}\right), \tag{12}$$

where X and Y are defined in (10) and (11).

By computer check it appears that many matrices G from (12) really generate an extremal self-dual code of length 34. Here, we present three examples $\mathcal{C}_i$, $i = 1, 2, 3$, of an extremal self-dual [34, 17, 6] code, with generator matrices

$$\left(\begin{array}{ccccc|c} a & b & & & & 0 \\ & & b & b & & 0 \\ & & & & c & 1 \\ \hline r_1 & & & & s_1 & \\ & u_1 & & u_2 & s_2 & \\ & & u_1 & u_3 & s_3 & \\ & & & & s_4 & \end{array}\right), \left(\begin{array}{ccccc|c} a & b & & & & 0 \\ & & b & & c & 0 \\ & & & b & & 1 \\ \hline r_1 & & & & s_1 & \\ & u_1 & & u_2 & s_2 & \\ & & u_1 & u_3 & s_3 & \\ & & & & s_4 & \end{array}\right), \left(\begin{array}{ccccc|c} a & & & & c & 0 \\ & b & & b & & 0 \\ & & b & & & 1 \\ \hline r_1 & & & & s_1 & \\ & u_1 & & u_2 & s_2 & \\ & & u_1 & u_3 & s_3 & \\ & & & & s_4 & \end{array}\right),$$

where a, b, c defined as right after (10) and where r_1, s_1, u_i, for $i = 1, \ldots, 3$, and s_j, for $j = 2, 3, 4$, are matrices as described right after (11).

The first rows of the circulant matrices r_1, u_i and s_j correspond to polynomials of $\mathcal{P}_1$, $\mathcal{P}_2$ and $\mathcal{P}$, respectively. The polynomials and the corresponding first rows of r_1, u_i and s_j are listed in the table at the end of this section. The codes $\mathcal{C}_1$ and $\mathcal{C}_3$ have weight enumerators of type $W_{34,1}$ (5) with β=6 and β=0, whereas the code $\mathcal{C}_2$ has weight enumerator $W_{34,2}$ (6). This was established by counting the numbers of codewords of weight 6 and of weight 8.

We may conclude that there exist self-dual $[34, 17, 6]$ codes with an automorphism of type $3 \cdot 5$-$(1,\ 3,\ 1;\ 1)$.

element	*polynomial*	*first row*	*element*	*polynomial*	*first row*
r_1	$l_1(x)$	011	s_1	$\mu_4(x)$	110110110110110
u_1	$l_2(x)$	01111	s_2	$\mu_1^2(x)$	101001010010100
u_2	$\alpha_2(x)$	11000	s_3	$\mu_1^{13}(x)$	011000110001100
u_3	$\alpha_2^2(x)$	10100	s_4	$e_3(x)$	111101011001000

References

[1] J. H. Conway and N. J. Sloane, A new upper bound of the minimal distance of self-dual codes, *IEEE Trans. Inform. Theory* **36** (1990), 1319-1333.

[2] R. Dontcheva and V. Y. Yorgov, The binary self-dual [34,17,6] codes with automorphism of odd prime order greater than 3, *Mathematics and Education in Mathematics*, Proc.27-th Conf. of UBM, Pleven, Bulgaria (1998), 161-166.

[3] V.Pless, A classification of self-orthogonal codes over GF(2), *Discrete Math.*, vol.3, pp. 209-246, 1972.

[4] V. Y. Yorgov, A method for constructing inequivalent self-dual codes with applications to length 56, *IEEE Trans. Inform. Theory* **33** (1987), 77-82.

Asymptotic analysis of recursive decoding techniques for Reed-Muller codes

Ilya Dumer and Kirill Shabunov*

University of California, Riverside, CA 92521

dumer@ee.ucr.edu shabunov@ee.ucr.edu

Abstract: Two new recursive decoding techniques are described for Reed-Muller (RM) codes and their subcodes. We analyze asymptotic properties of these algorithms and show that they substantially outperform other nonexponential algorithms known for RM codes. Decoding performance is further enhanced by using intermediate code lists and permutation procedures.

1 Introduction

Below we design new recursive decoding algorithms for general Reed-Muller (RM) codes $\left\{ {m \atop r} \right\}$ of length $n = 2^m$ and distance $d = 2^{m-r}$. To do so, we use the *Plotkin construction* $\mathbf{c} = (\mathbf{u}, \mathbf{u} + \mathbf{v})$, that builds the code $\left\{ {m \atop r} \right\}$ by taking two subblocks $\mathbf{u}$ and $\mathbf{v}$ from codes $\left\{ {m-1 \atop r} \right\}$ and $\left\{ {m-1 \atop r-1} \right\}$. This recursion can be continued further on the two descendant codes of length $n/2$ until we finally arrive at the end codes. These are either repetition codes $\left\{ {g \atop 0} \right\}$ for any $g = 1, \ldots, m - r$ or full spaces $\left\{ {h \atop h} \right\}$ for any $h = 1, \ldots, r$. The design is shown on Fig. 1, while Fig. 2 depicts a partial splitting ending at $\left\{ {g \atop 1} \right\}$ nodes.

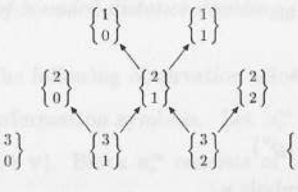


Figure 1: Full decomposition.

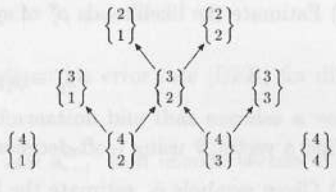


Figure 2: Partial decomposition.

In [1] and [2], the Plotkin construction was recursively used in decoding design to execute bounded distance decoding with the lowest complexity order of $n \min(r, m - r)$ known for

*This research was supported by the NSF grant CCR-0097125.

RM codes. Our main goal below is to show that recursive techniques can correct many error patterns of heavier weights. For this reason, we consider a different setting. Given an infinite sequence of codes $A_i(n_i, d_i)$, we say that a decoding algorithm has a *threshold sequence* δ_i with a *residual sequence* ε_i if

- $\varepsilon_i > 0$, $\varepsilon_i \to 0$ as $n_i \to \infty$;
- only a vanishing fraction of error patterns of weight $(1-\varepsilon_i)\delta_i$ is left uncorrected;
- only a vanishing fraction of error patterns of weight $(1+\varepsilon_i)\delta_i$ can be corrected[1].

A similar setting was considered in [3] for majority decoding. Namely, for low-rate RM codes of fixed order r, majority decoding has threshold $\delta = n/2$ and residuals $\varepsilon \sim (m/d)^{1/2^{r+1}}$. Note that δ exceeds 2^r times the bounded distance threshold $d/2$. Another efficient algorithm based on permutation decoding was designed in [4] for RM codes $\left\{{m \atop 2}\right\}$. This algorithm reduces the ε-term to the order of $(m/d)^{1/4}$. For long RM codes of fixed rate R, it is proven [5] that majority algorithm achieves a threshold $(d \ln d)/4$ with vanishing ε-terms. However, the two latter algorithms increase complexity of recursive decoding to the order of $n^2 m$ and n^2, respectively. Below we consider two recursive decoding algorithms that have low complexity $O(n \log n)$ and increase the thresholds known for polynomial-type decoding algorithms of general RM codes.

2 New decoding techniques

Let $\mathbf{y} = (\mathbf{y}', \mathbf{y}'') = (\widetilde{\mathbf{u}, \mathbf{u}+\mathbf{v}})$ be the received block. For any output $y \in \mathbb{R}$ define the likelihood $\rho = \ln \Pr\{u = 0 \mid y\} - \ln \Pr\{u = 1 \mid y\}$ that 0 is transmitted. Correspondingly, define the likelihoods ρ_i', ρ_i'', $i = 1, \dots, n/2$ on both parts $\mathbf{y}', \mathbf{y}''$. The following algorithm reduces decoding on length $n = 2^m$ to two decodings on length $n/2$:

a) Estimate the likelihoods ρ_i^{v} of symbols $v_i = c_i' + c_i''$ as

$$\rho_i^{\mathrm{v}} = \rho_i' \rho_i''/2, \quad i = 1, \dots, 2^{m-1}. \tag{1}$$

Find a vector $\hat{\mathbf{v}}$ using (soft-decision) decoding $\hat{\mathbf{v}} = \Psi_{\mathbf{v}}(\boldsymbol{\rho}^{\mathrm{v}})$.

b) Given symbols $\hat{v}_i$, estimate the likelihoods ρ_i^{u} of symbols u_i:

$$\rho_i^{\mathrm{u}} = \rho_i' + (-1)^{\hat{v}_i} \rho_i'', \quad i = 1, \dots, 2^{m-1}. \tag{2}$$

Find a vector $\hat{\mathbf{u}}$ using (soft-decision) decoding $\hat{\mathbf{u}} = \Psi_{\mathbf{u}}(\boldsymbol{\rho}^{\mathrm{u}})$.

c) Combine $\hat{\mathbf{v}}$ and $\hat{\mathbf{u}}$ to get a codeword $\hat{\mathbf{c}} = (\hat{\mathbf{u}}, \hat{\mathbf{u}} + \hat{\mathbf{v}})$.

If we recursively apply (1) and (2) to the inputs $\boldsymbol{\rho}^{\mathrm{v}}$ and $\boldsymbol{\rho}^{\mathrm{u}}$ we can reach the end codes $\left\{{g \atop 0}\right\}$ and $\left\{{h \atop h}\right\}$. Here we can perform ML decoding with linear complexity. Formally, recursive decoding $\hat{\mathbf{c}} = \Psi_r^m(\boldsymbol{\rho})$ is defined as follows:

```
1. If $r = 0$ decode the repetition code $\left\{{m \atop 0}\right\}$.
2. If $r = m$ decode the full space code $\left\{{m \atop m}\right\}$.
3. Else recursively decode $\left\{{m \atop r}\right\}$:
    a) Calculate $\rho_i^{\mathrm{v}}$ from (1). Find $\hat{\mathbf{v}} := \Psi_{r-1}^{m-1}(\boldsymbol{\rho}^{\mathrm{v}})$.
    b) Calculate $\rho_i^{\mathrm{u}}$ from (2). Find $\hat{\mathbf{u}} := \Psi_{r-1}^{m-1}(\boldsymbol{\rho}^{\mathrm{u}})$.
    c) $\hat{\mathbf{c}} = (\hat{\mathbf{u}}, \hat{\mathbf{u}} + \hat{\mathbf{v}})$.
```

Another decoding algorithm Φ_r^m is obtained if Ψ_r^m is terminated at the biorthogonal codes $\left\{{g \atop 1}\right\}$. In this case, ML decoding of end codes $\left\{{g \atop 1}\right\}$ has complexity $O(n \log n)$. More detailed analysis also shows that the overall complexity of both algorithms Φ_r^m and Ψ_r^m has the order of $O(n \log n)$.

3 Asymptotic analysis of algorithms Ψ_r^m and Φ_r^m.

Consider a binary symmetric channel with error probability p.

Theorem 1 *For RM codes with $m \to \infty$ and fixed r, algorithms Ψ_r^m and Φ_r^m have a decoding threshold $\delta = n/2$ with $\varepsilon \sim (m/d)^{1/2^{r+1}}$ for Ψ_r^m and $\varepsilon' \sim (m/d)^{1/2^r}$ for Φ_r^m.*

Theorem 2 *For RM codes with $m \to \infty$ and fixed rate R, algorithms Ψ_r^m and Φ_r^m have decoding thresholds $(d \ln d)/4$ and $(d \ln d)/2$, which increase $(\ln d)/2$ and $\ln d$ times the threshold $d/2$ of bounded-distance decoding.*

The following observation allows to evaluate the output bit error rate (BER) for different information symbols. Let $\mathbf{a}_r^m$ be a block of k information bits that encodes a vector $(\mathbf{u}, \mathbf{u} + \mathbf{v})$. Block $\mathbf{a}_r^m$ consists of two subblocks $\mathbf{a}_r^{m-1}$ and $\mathbf{a}_{r-1}^{m-1}$ that encode vectors $\mathbf{u}$ and $\mathbf{v}$, respectively. These subblocks are decomposed further until we arrive at the end codes. Thus, any information bit can be mapped onto a specific path j leading from the initial code $\left\{{m \atop r}\right\}$ to one of the end codes $\left\{{g \atop 0}\right\}$ or $\left\{{h \atop h}\right\}$. For example, the leftmost code $\left\{{m-r \atop 0}\right\}$ is mapped on the path $(1, \dots, 1)$ of length $m - r$, while the rightmost code $\left\{{r \atop r}\right\}$ corresponds to 2^r information bits with the common path $(0, \dots, 0)$ of length r. Below, $j = j_0^g$ (or j_h^h)

[1] Note that thresholds δ_i are defined up to the terms of order $\delta_i \varepsilon_i$

denotes a specific path leading to the code $\left\{ {g \atop 0} \right\}$ (or $\left\{ {h \atop h} \right\}$), while a_j is a bit (or a subset of bits) associated with this path. For $c = 0, 1$ and $x > 0$, we use notation $x \diamond c = 2^{c-1} x^{c+1}$ and define the product $x \diamond j_0^g = (\cdots((x \diamond j_1) \diamond j_2) \cdots \diamond j_{m-g})$.

Now let algorithms Ψ_r^m and Φ_r^m be used on an AWGN channel with a noise power σ^2 (equivalently, the results carry over to a BSC with error probability $p = \int_{1/\sigma}^{\infty} e^{-u^2/2}\, du / \sqrt{2\pi}$). These algorithms enter each end code by taking all paths leading to this code. It turns out that the output BER significantly varies on different codes and even on the different paths leading to the same code. In particular, the leftmost code $\left\{ {m-r \atop 0} \right\}$ has the lowest error protection, and can sustain a noise power $\sigma^2 = o(2^{(m-r)/2^r})$ and $o(2^{(m-r)/2^{r-1}})$ for algorithms Φ_r^m and Φ_r^m, respectively. More detailed statements are given below.

Theorem 3 *For RM codes with $m \to \infty$ and fixed r, algorithm Ψ_r^m gives a vanishing bit error rate on a specific path j_0^g if the noise power σ^2 satisfies the following condition*

$$\sigma^2 \diamond j_0^g = o(2^g).$$

Theorem 4 *For RM codes with $m \to \infty$ and fixed r, algorithm Φ_r^m gives a vanishing bit error rate on a specific path j_1^g if the noise power σ^2 satisfies the following condition*

$$\sigma^2 \diamond j_1^g = o(2^g).$$

4 Improvements and applications

We use three different techniques to improve the word error rates (WER) obtained for algorithm Φ_r^m. These techniques are:

- eliminating a few least protected information bits and using remaining subcodes of RM codes,
- list decoding techniques using L most probable candidates,
- using permutations of original RM codes.

These techniques give substantial improvements of 2 to 4 dB even on moderate lengths of 256 and 512 (see Fig. 3) and show that nearly ML decoding is obtained on the lengths up to 512 with the lists of moderate size $L \leq 512$. The conclusion is that for moderate blocklengths, the recursive decoding of RM codes combines nearly optimal performance with low complexity of order $n \log n$.

Upper Bounds on the Rate of Superimposed (s, ℓ)-Codes Based on Engel's Inequality [1]

A.G. D'yachkov, P.A. Vilenkin, S.M. Yekhanin
Moscow State University, Faculty of Mechanics and Mathematics,
Department of Probability Theory, Moscow, 119992, Russia
dyachkov@mech.math.msu.su, paul@vilenkin.dnttm.ru, yekhanin@cityline.ru

Abstract

Applying an important combinatorial result of K. Engel [2], we improve upper bounds on the rate of superimposed (s, ℓ) - codes obtained in [3, 4].

1 Definitions and Formulations of Results

We use the symbol $\triangleq$ to denote definitional equalities.

Let $N \geq 1$, $t \geq 1$, $s \geq 1$ and $\ell \geq 1$, where $s + \ell \leq t$, be arbitrary integers. A family of t binary codewords of length N is called a *superimposed (s, ℓ)-code* [3, 4] of size t and length N if for any two non-intersecting subsets of codewords $\mathcal{S}$ and $\mathcal{L}$, $|\mathcal{S}| = s$, $|\mathcal{L}| = \ell$, there exists a coordinate $k \in \{1, 2, \ldots, N\}$, in which all codewords from set $\mathcal{S}$ have 0's and all codewords from set $\mathcal{L}$ have 1's.

Let $N(t, \ell, s) = N(t, s, \ell)$ denote the minimal possible length of superimposed (s, ℓ)-code of size t. For fixed s and ℓ, the number

$$R(\ell, s) = R(s, \ell) \triangleq \overline{\lim_{t \to \infty}} \frac{\log_2 t}{N(t, \ell, s)}$$

is called [3, 4] a *rate* of superimposed (s, ℓ) - code.

Let $h(u) \triangleq -u \log_2 u - (1-u)\log_2(1-u)$, $0 < u < 1$, be the binary entropy. To formulate the upper bound on the rate $R(s, \ell)$, $s \geq \ell \geq 1$, we introduce the function [1]

$$\mathsf{f}_s(v) \triangleq h(v/s) - v \cdot h(1/s), \quad s = 1, 2, \ldots,$$

of argument v, $0 < v < 1$. The following three statements are true.

Theorem 1. 1. *If $s = 1, 2, \ldots$, then the rate $R(s, 1) \leq \overline{R}(s, 1)$, where*

$$\overline{R}(1,1) = R(1,1) = 1, \quad \overline{R}(2,1) \triangleq \max_{0<v<1} \mathsf{f}_2(v) = 0.321928 \tag{1}$$

and sequence $\overline{R}(s, 1)$, $s = 3, 4, \ldots$, is defined recurrently as the unique solution of the equation

$$\overline{R}(s,1) = \mathsf{f}_s\left(1 - \frac{\overline{R}(s,1)}{\overline{R}(s-1,1)}\right). \tag{2}$$

2. *The rate*

$$R(2,2) \leq \overline{R}(2,2) \triangleq \overline{R}(2,1)/2 = 0.160964. \tag{3}$$

[1] Work was supported by the Russian Foundation of Basic Research, grant N. 01-01-00495 and the INTAS, grant N. 00-738

3. *If $s > \ell \geq 2$ or $s \geq \ell \geq 3$, then the rate $R(s,\ell)$ satisfies the inequality*

$$R(s,\ell) \leq \overline{R}(s,\ell) \triangleq \min_{x=0,1,\ldots,s-1} \min_{y=0,1,\ldots,\ell-1} \left\{ \overline{R}(s-x,\ell-y) \cdot \frac{x^x \cdot y^y}{(x+y)^{x+y}} \right\}, \tag{4}$$

where sequence $\overline{R}(s,1)$, $s = 1,2,\ldots$, and the number $\overline{R}(2,2)$ are defined by (1)-(3).

The first statement was proved in [1] (see, also [4]). The second statement was proved in [4]. The third statement is an evident consequence of the following result obtained by K. Engel [2].

Theorem 2. (Engel's inequality [2].) *If $s \geq \ell \geq 2$, then for any $x = 0,1,\ldots,s-1$ and any $y = 0,1,\ldots,\ell-1$, the rate*

$$R(s,\ell) \leq R(s-x,\ell-y) \cdot \frac{x^x \cdot y^y}{(x+y)^{x+y}}. \tag{5}$$

In section 3, we briefly present the proof of Theorem 2 from paper [2]. The numerical values of upper bound $\overline{R}(s,\ell)$, $1 \leq \ell \leq s \leq 4$, are:

$$\overline{R}(2,1) = .32193, \quad \overline{R}(3,1) = .19928, \quad \overline{R}(4,1) = .14046, \quad \overline{R}(2,2) = .16096,$$

$$\overline{R}(3,2) = .08048, \quad \overline{R}(4,2) = .04769, \quad \overline{R}(3,3) = .04024, \quad \overline{R}(4,3) = .02012$$

and $\overline{R}(4,4) = .01006$.

2 Asymptotics of $\overline{R}(s,\ell)$

If $s \to \infty$ and $\ell \geq 2$ is fixed, then the optimal values of x and y in definition (4) of $\overline{R}(s,\ell)$ have the form $y = \ell - 1$, $x \sim ps$, $0 < p < 1$, and

$$\overline{R}(s,\ell) \sim \min_{0<p<1} \left\{ \overline{R}(s(1-p),1) \cdot \frac{(ps)^{ps} \cdot (\ell-1)^{\ell-1}}{(ps+\ell-1)^{ps+\ell-1}} \right\}.$$

Using the asymptotic ($s \to \infty$) form [1, 4] of upper bound $\overline{R}(s,1) \sim 2\log s/s^2$, we get

$$\overline{R}(s,\ell) \sim \min_{0<p<1} \left\{ \frac{2\log[s(1-p)]}{s^2(1-p)^2} \cdot \frac{(ps)^{ps} \cdot (\ell-1)^{\ell-1}}{(ps+\ell-1)^{ps+\ell-1}} \right\} \sim \frac{(\ell+1)^{\ell+1}}{2\mathrm{e}^{\ell-1}} \cdot \frac{\log s}{s^{\ell+1}}, \tag{6}$$

where $\mathrm{e} = 2.71828$ is the base of natural logarithm and we took into account that

$$\max_{0<p<1} \{(1-p)^2 p^{\ell-1}\} = (\ell-1)^{\ell-1} \frac{4}{(\ell+1)^{\ell+1}}$$

with the optimal value $p = \frac{\ell-1}{\ell+1}$. For $\ell \geq 2$, upper bound (6) is better than the similar upper bound

$$\overline{R}_{old}(s,\ell) \sim (\ell+1)! \cdot \frac{\log s}{s^{\ell+1}},$$

which was obtained in [4].

3 Proof of Engel's inequality

Let $s \geq 1$, $\ell \geq 1$, and $t \geq s + \ell$, be arbitrary integers, $[t] \triangleq \{1,2,\ldots,t\}$ and the set B_t of size $|B_t| = 2^t$ be the *Boolean lattice* constituted of all subsets of $[t]$.

Introduce the set $P = P(t,\ell,t-s) \subseteq B_t$, $|P| = |P(t,\ell,t-s)| = \sum\limits_{n=\ell}^{t-s} \binom{t}{n}$, whose elements are n-subsets of $[t]$, where $\ell \leq n \leq t - s$. Let $Z \subseteq Y \subseteq [t]$ be arbitrary subsets of $[t]$. Denote by $J = J(t,\ell,t-s)$ the set of all *intervals* $I = I(Z,Y)$:

$$I = I(Z,Y) \triangleq \{X : X \in P,\ Z \subseteq X \subseteq Y\}, \quad \text{where} \quad |Z| = \ell,\ |Y| = t-s,\ |[t] \setminus Y| = s.$$

Obviously, each interval $I \in J$ is isomorphic to $B_{t-s-\ell}$ and $|I| = 2^{t-s-\ell}$. In addition, any element $X \in P$ is contained in $\binom{|X|}{\ell}\binom{t-|X|}{t-s-|X|}$ intervals of J. Taking all X with $|X| = \ell$ (resp. all X with $|X| = t - s$) we obtain

$$|J| = \binom{t}{\ell}\binom{t-\ell}{t-s-\ell} = \binom{t}{t-s}\binom{t-s}{\ell}. \tag{7}$$

A set $T \subseteq P$ is called a *point cover* of J if for any interval $I \in J$, the intersection $T \cap I \neq \varnothing$. The minimal size of point cover T is denoted by $\tau(t,\ell,t-s)$.

Lemma 1. *The minimal length of superimposed (s,ℓ)-code $N(t,\ell,s) = \tau(t,\ell,t-s)$.*

Proof of Lemma 1. Let C be a superimposed (s,ℓ)- code of length N and size t. Fix an order over codewords of $C = \{c_1, c_2, \ldots, c_t\}$. Introduce the following *correspondence* between coordinates of codewords $c_1, c_2, \ldots, c_t$ and subsets of $[t]$: *a set $X_k \subseteq [t]$ corresponding to a coordinate k, $k = 1,2,\ldots,N$, contains the numbers i of codewords c_i having 1's in the k-th coordinate.* Without loss of generality, $\ell \leq |X_k| \leq t - s$. Consider the set $T \triangleq \{X_1, X_2, \ldots, X_N\} \subseteq P = P(t,\ell,t-s)$. Take an arbitrary interval $I = I(Z,Y) \in J = J(t,\ell,t-s)$. By definition of the superimposed (s,ℓ) - code C, there exists a coordinate k such that all codewords with numbers in Z have 1's in the k-th coordinate and all codewords with numbers in $[t] \setminus Y$ have 0's in the k-th coordinate, i.e., $Z \subseteq X_k \subseteq Y$. Hence, $X_k \in I$ and $T \cap I \neq \varnothing$. Therefore, T is a point cover of J. Thus, we have proved that $N \geq \tau(t,\ell,t-s)$, i.e., $N(t,\ell,s) \geq \tau(t,\ell,t-s)$. To prove $N(t,\ell,s) \leq \tau(t,\ell,t-s)$ one needs to check that superimposed (s,ℓ)-code can be constructed from a point cover using the correspondence described above.

We introduce several additional definitions. A *fractional matching* of $P = P(t,\ell,t-s)$ is a function $\quad f = f(I) \geq 0, \quad I \in J = J(t,\ell,t-s) \quad$ such that

$$\forall X \in P: \qquad \sum_{I \ni X} f(I) \leq 1.$$

A *fractional point cover* of J is a function $\quad g = g(X) \geq 0, \quad X \in P \quad$ such that

$$\forall I \in J: \qquad \sum_{X \in I} g(X) \geq 1.$$

The *fractional matching number* $\nu^*(t,\ell,t-s)$ and *fractional covering number* $\tau^*(t,\ell,t-s)$ are defined by

$$\nu^*(t,\ell,t-s) \triangleq \max \left\{ \sum_{I \in J} f(I) : f \text{ is a fractional matching of } P \right\},$$

$$\tau^*(t,\ell,t-s) \triangleq \min\left\{\sum_{X\in P} g(X) : g \text{ is a fractional point cover of } J\right\}.$$

Lemma 2. *We have* $\nu^*(t,\ell,t-s)=\tau^*(t,\ell,t-s)=\min\limits_{\ell\le m\le t-s}\binom{t}{m}/\binom{t-s-\ell}{m-\ell}$.

Proof of Lemma 2. The first equality follows from the Duality Theorem of linear programming. Suppose that the minimum in the right-hand side is attained at $m=m_0$. To prove the second equality, it is enough to find a fractional matching f and a fractional point cover g such that

$$\sum_{I\in J} f(I) = \frac{\binom{t}{m_0}}{\binom{t-s-\ell}{m_0-\ell}} = \sum_{X\in P} g(X). \tag{8}$$

We choose

$$f(I) \triangleq \frac{1}{\binom{m_0}{\ell}\binom{t-m_0}{t-s-m_0}} \qquad \text{for all } I\in J$$

and

$$g(X) \triangleq \begin{cases} 0, & \text{if } |X|\ne m_0; \\ \frac{1}{\binom{t-s-\ell}{m_0-\ell}}, & \text{if } |X|=m_0. \end{cases}$$

The function f is a fractional matching since

$$\sum_{I\ni X} f(I) = \frac{\binom{|X|}{\ell}\binom{t-|X|}{t-s-|X|}}{\binom{m_0}{\ell}\binom{t-m_0}{t-s-m_0}} = \frac{\binom{t}{m_0}/\binom{t-s-\ell}{m_0-\ell}}{\binom{t}{|X|}/\binom{t-s-\ell}{|X|-\ell}} \le 1 \qquad \text{for all } X\in P,$$

and g is a fractional point cover since

$$\sum_{X\in I} g(X) = \frac{\binom{t-s-\ell}{m_0-\ell}}{\binom{t-s-\ell}{m_0-\ell}} = 1 \quad \text{for all} \quad I\in J.$$

The equality (8) can be verified by straightforward computation using equality (7).

Lemma 3. *For fixed* ℓ, s *and* $t\to\infty$, *the number* $\tau^*(t,\ell,t-s)\sim\frac{(s+l)^{s+l}}{s^s l^l}$.

Proof of Lemma 3. Let ℓ, s and u, $0<u<1$, be fixed. If $t\to\infty$ and $m\sim ut$, then

$$\frac{\binom{t}{m}}{\binom{t-s-\ell}{m-\ell}} = \frac{t(t-1)\cdots[t-(s+\ell-1)]}{\{[m-(\ell-1)]\cdots m\}\cdot\{[(t-m)-(s-1)]\cdots(t-m)\}} \sim \left[u^\ell\cdot(1-u)^s\right]^{-1}.$$

Using the definition of $\tau^*(t,\ell,t-s)$ in Lemma 2, we have

$$\tau^*(t,\ell,t-s) \sim \left\{\max_{0<u<1}\left[u^\ell\cdot(1-u)^s\right]\right\}^{-1} = \left\{\frac{s^s l^\ell}{(s+\ell)^{s+\ell}}\right\}^{-1},$$

where the maximum is achieved at $u=\frac{\ell}{\ell+s}$.

Lemma 4. *For any* $x=0,1,\ldots,s-1$ *and* $y=0,1,\ldots,\ell-1$,

$$\frac{\tau(t,\ell,t-s)}{\tau(t-x-y,\ell-y,t-s-y)} \ge \tau^*(t,y,t-x).$$

Proof of Lemma 4. Let T, $|T|=\tau(t,\ell,t-s)$ be an optimal point cover of $J(t,\ell,t-s)$. We have $0\le y<\ell<t-s<t-x$ and $T\subset P(t,\ell,t-s)\subset P(t,y,t-x)$. For $X\in P(t,y,t-x)$, we define the function

$$g(X) \triangleq \begin{cases} 1/\tau(t-x-y,\ell-y,t-s-y), & \text{if } X\in T, \\ 0, & \text{otherwise.} \end{cases}$$

It is enough to show that g is a fractional point cover of $J(t,y,t-x)$. Consider an arbitrary interval $I\in J(t,y,t-x)$ which is isomorphic to the Boolean lattice B_{t-x-y}. Moreover, the part of I which lies between levels ℓ and $t-s$ is isomorphic to $P(t-x-y,\ell-y,t-s-y)$. Since the considered set T is a point cover of $J(t,\ell,t-s)$ the intersection $T\cap I$ must be a point cover of the corresponding set of intervals $J(t-x-y,\ell-y,t-s-y)$. Thus,

$$\sum_{X\in I} g(X) \ge \frac{|T\cap I|}{\tau(t-x-y,\ell-y,t-s-y)} \ge 1.$$

Proof of Theorem 2. If $t'\triangleq t-x-y$, then $t-s-y=t'-(s-x)$. Using Lemma 1, we have $\tau(t-x-y,\ell-y,t-s-y)=N(t-x-y,\ell-y,s-x)$. Therefore, we can rewrite the inequality from Lemma 4 in the form

$$N(t,\ell,s)\ge\tau^*(t,y,t-x)\cdot N(t-x-y,\ell-y,s-x).$$

For s,ℓ,x,y fixed and $t\to\infty$, the application of Lemma 3 yields

$$N(t,\ell,s) \ge \frac{(x+y)^{x+y}}{x^x y^y}\cdot N(t,\ell-y,s-x)(1+o(1)). \tag{9}$$

If we multiply by $\log_2 t$ the opposite inequality for reciprocals in (9) and pass to the limit, then we obtain inequality (5).

Theorem 2 is proved.

References

[1] A.G. D'yachkov, V.V. Rykov, "Bounds on the Length of Disjunct Codes", *Problemy Peredachi Informatsii*, **17**, 1982, n. 3, pp. 7-13, (in Russian).

[2] K. Engel, "Interval Packing and Covering in the Boolean Lattice", *Combinatorics Prob. and Computing*, **5**, 1996, pp. 373-384.

[3] A.G. D'yachkov, A.J. Macula, D.C. Torney, P.A. Vilenkin, S.M. Yekhanin, "New Results in the Theory of Superimposed Codes", *Proc. of the 7-th Inter. Workshop "Algebraic and Combinatorial Coding Theory"*, *ACCT-7*, Bansko, Bulgaria, June 17-24, 2000, pp. 126-136.

[4] A.G. D'yachkov, A.J. Macula, D.C. Torney, P.A. Vilenkin, "Families of Finite Sets in which No Intersection of ℓ Sets Is Covered by the Union of s Others", *submitted to J. Comb. Theory, ser. A.*

CODES FOR A LONG SILENCE

Emanuela Fachini and János Körner
"La Sapienza" University of Rome
via Salaria 113, 00198 Roma, Italy
e-mail: fachini, korner@dsi.uniroma1.it
ITALY

Abstract

We determine the exact exponential asymptotics of the maximum number of n-length binary strings every pair of which differ in the following strong sense: there must be a coordinate in which one of them has a 1 in correspondence with a predetermined position within a "long run" of zeroes in the other string.

1 Introduction

A central question of zero-error information theory in the sense of [6] is of the following type: given a (not necessarily binary) alphabet $\mathcal{X}$ and a natural number n, how large can a set $\mathcal{D} \subseteq \mathcal{X}^n$ be if any two of its members must differ in some specific sense. The sense in which the sequences must differ is determined by a model of communication through a noisy transmission device. If this channel is *memoryless*, then the corresponding mathematical problems include the code distance problem in which every two elements of $\mathcal{D}$ must differ in a fixed proportion of their coordinates and Shannon's graph capacity problem in which any two members of $\mathcal{D}$ must differ in a pair of letters from the alphabet that form an edge in a fixed graph whose vertex set is $\mathcal{X}$. In information theory it is customary to determine just the exponential asymptotics of the maximum cardinality of the set of n-length sequences in question. The asymptotic exponent is often called the capacity or the maximum achievable coding rate of the channel under consideration. Multi-user generalizations of these models furnish a framework allowing to tackle with sometimes surprising success many classical problems of Extremal Set Theory (extremal hypergraph theory) in the sense of Erdős, cf. Tolhuizen's breakthrough result [8] and [6].

In this paper we introduce similar questions for certain channels with memory. In order to keep this paper short we will not introduce a general model, rather, we will state the special cases we want to treat in the form of simple purely combinatorial problems about families of binary strings. We interpret the two elements, 0 and 1 of a binary alphabet as *silence*, (i. e., absence of signals) and *signal*, respectively. Further, as it is usually done, we interpret the coordinate indices as subsequent discrete instants of time. We suppose that the presence of a bit equal to 1 somewhere in a transmitted string can effectively serve to distinguish this string from an other one, having a zero in the same coordinate, only in case the latter string has, in addition, a predetermined number of zeroes both preceding and following the coordinate in question. This gives us two additional parameters, the number k of "zeroes before" and the number l of "zeroes after". These runs of zeroes represent a long enough period of silence necessary in order that the presence of a signal be perceived as such.

More formally, let us be given two non–negative integers, k and l. We shall say that the binary sequences $\mathbf{x} \in \{0,1\}^n$ and $\mathbf{y} \in \{0,1\}^n$ (with $n \geq k+l+1$) are (k,l)–different if there is a coordinate i with $k+1 \leq i \leq n-l$ such that either

$$x_i = 1 \quad \text{and} \quad y_j = 0 \quad \text{for every} \quad j \in \{i-k, i+l\}$$

or, vice versa,

$$y_i = 1 \quad \text{and} \quad x_j = 0 \quad \text{for every} \quad j \in \{i-k, i+l\}$$

Let us denote by $M(k,l,n)$ the largest cardinality of a set $D \subseteq \{0,1\}^n$ every pair $\{\mathbf{x},\mathbf{y}\}$ of distinct elements of which is (k,l)–different in our previous sense. We will call such a set a *code for a* (k,l)*–long silence*. Our aim in this paper is to determine the asymptotics of $M(k,l,n)$ in n, for every fixed k and l.

Throughout the paper exponentials and logarithms are to the base 2.

2 Codes for a long silence

Obviously, $M(0,0,n) = 2^n$ and $M(k,l,n) = M(l,k,n)$. It is an easy exercise to show that

Proposition 1

$$M(0,1,n) = f_{n+1}$$

where $f_1 = 1$, $f_2 = 1$ and, in general, $f_n = f_{n-2} + f_{n-1}$ is the standard Fibonacci sequence. Thus, at once, we obtain

$$\lim_{n\to\infty} \frac{1}{n} \log M(0,1,n) = \log \frac{1+\sqrt{5}}{2}$$

It is just a little bit more complicated to realize that

Proposition 2

$$\lim_{n\to\infty}\frac{1}{n}\log M(1,1,n)=1/2$$

A more careful analysis of the previous proof gives our general result right away. The details will be given in an extended journal version [3].

Theorem 1 *Let* $q=q(k,l)$ *be the unique positive root of the equation* $2^{-q(k+1)}+2^{-q(l+1)}=1$. *Then*

$$\lim_{n\to\infty}\frac{1}{n}\log M(k,l\,n)=q(k,l).$$

3 Generalizations

Our problem opens up the way to numerous generalizations. One of these is fairly obvious and has been already mentioned. In fact, our problem can be viewed as that of determining the zero–error capacity of discrete stationary channels with memory, a problem hopelessly general in this form, but probably reasonable to tackle in case of a three–letter alphabet, in which case one would ask for the determination of the asymptotic growth (with n) of the largest size of a set $D_n\subseteq\{a,b,0\}^n$, where, once again, interpreting a and b as signals and 0 as silence, one would require for any two strings $\mathbf{x}$ and $\mathbf{y}$ from D_n to differ in a coordinate i so that $\{x_i,y_i\}=\{a,b\}$, while this occurence of b is preceded by k and followed by l consecutive zeroes. This problem is easy in itself, but the compound channel extension of it in the sense of [4] might be quite interesting. However, undoubtedly, the most interesting and practically relevant immediate generalization is for superimposed codes, [5], [2], [1].

We will say that a set of binary sequences, $D_n\subseteq\{0,1\}^n$ is a superimposed code for a (k,l)–long silence if for any ordered triple of three distinct strings $(\mathbf{x},\mathbf{y},\mathbf{z})$ from D_n there is a coordinate in which $(x_i,y_i,z_i)=(1,0,0)$ and the k preceding and the l consecutive coordinates of both $\mathbf{y}$ and $\mathbf{z}$ are equal to 0. The determination of the asymptotic exponent of the largest size of such a set is a well–known open problem even if $k=l=0$ ([2]), yet as in that case, non–trivial bounds of great practical relevance exist in the literature, cf. [1] and [7].

References

[1] A. G. Dyachkov and V.V. Rykov, Bounds on the length of disjunctive codes, *Problems Per. Inform.*, vol. 18(1982) no. 3, pp.7–13,

[2] P. Erdős, P. Frankl and Z. Füredi, Families of finite sets in which no set is covered by the union of two others, *J. Combinatorial Theory, Ser.* A 33(1982), pp. 158–166,

[3] E. Fachini, J. Körner, Coding for a long silence, paper in preparation

[4] L. Gargano, J. Körner, U. Vaccaro, Capacities: from information theory to extremal set theory, *J. Comb. Theory Ser.* A, vol. 68(1994), no. 2, pp. 296–316,

[5] W. H. Kautz and R. C. Singleton, Nonrandom binary superimposed codes, *IEEE Trans. Inform. Theory*, **10**(1964), pp. 363–377,

[6] J. Körner and A. Orlitsky, Zero–error information theory, *IEEE Trans. Inform. Theory*, **44** no. 6, October 1998, pp. 2207–2229,

[7] M. Ruszinkó, On the upper bound of the size of the r-cover-free families, *J. Combin. Theory Ser.* A, 66 (1994), no. 2, 302–310,

[8] L. M. Tolhuizen, New rate pairs in the zero–error capacity region of the binary multiplying channel without feedback, *IEEE Trans. Inform. Theory*, **46** no. 3, May 2000, pp. 1043–1046

New Results On Impossibility of Uniform Distribution of Codewords over Spheres

Maria Fedorova

Mech. & Math. Department
Moscow State University
119992 Moscow, Russia
e-mail: maria_fedorova@yahoo.com

Abstract

In this paper we continue to investigate the codes uniformly distributed over spheres. We prove that for the fixed s and other parameters in some ranges it is impossible to distribute codewords uniformly over spheres of radii s and $2s$ simultaneously.

The binary codes with codewords uniformly distributed over subcubes (and also its characteristic Boolean functions and arrays with codewords written in rows) are studied extensively in differents fields of mathematics and its applications. Such structures are known as codes with high dual distance, correlation-immune, resilient and ε-biased Boolean functions, orthogonal arrays and so on. Such structures are important in statistics to design an experiment, in cryptology to hide a secret and to generate pseudorandom sequences. The uniform distribution of codewords over spheres was not studied extensively before (we can mention only the papers [1], [2]) although codes with codewords uniformly distributed over spheres can have some helpful applications, for example, when the code plays a hash-like function or when we want to have for all possible words at the output of the channel approximately the equal numbers of proper decodings. The characteristic Boolean functions of such codes has a good resistance as a combiner in stream ciphers against statistical attacks when an opponent has the possibility to change some (restricted) number of inputs of the function.

The paper [1] introduces the general concept of UDS-codes (uniformly distributed over spheres, a generalization of the concept introduced in [2] for Boolean functions). In [1] two nonexistence results on UDS-codes are proved. In this paper we generalize the theorem on UDS-codes of high cardinality.

We consider V^n, the vector space of n-tuples of elements from $GF(2)$. An arbitrary set of vectors $C \subseteq V^n$ is called *a (binary) code*. Each vector from C is called *a codeword* of C. The number of codewords in C is called *a cardinality* of C and is denoted by $|C|$. For a code C the code $\overline{C} = V^n \setminus C$ is called *a complementary code*. A *Boolean function* is a function from V^n to $GF(2)$. The *weight* $wt(f)$ of a function f on V^n is the number of vectors x on V^n such that $f(x) = 1$. The concepts of a Boolean function and a code are closely connected. An arbitrary Boolean function f on V^n is associated with its *characteristic set* — the code C: $\{x \in V^n \mid f(x) = 1\}$. Conversely, an arbitrary code $C \subseteq V^n$ is associated with its *characteristic function* — the function f: $f(x) = \{1 \text{ if } x \in C, 0 \text{ if } x \notin C$.

The *Hamming distance* $d(x, y)$ between two vectors x and y is the number of components where vectors x and y differ. The sphere $S_r(x)$ with center $x \in V^n$ and radius r is the set of all vectors y in V^n such that $d(x, y) \leq r$. Let C be a code in V^n. The weight $wt(S_r(x), C)$ of the sphere $S_r(x)$ regarding C (or simply the weight of the sphere $S_r(x)$) is the cardinality of the intersection $S_r(x) \cap C$.

Definition 1 *Let l_i and r_i be nonnegative integers, all r_i are different, $i = 1, \ldots, h$. We say that a code $C \subseteq V^n$ is an* UDS$(l_1, r_1; \ldots; l_h, r_h)$*-code (uniformly distributed over spheres) if for any $x, y \in V^n$ we have $|wt((S_{r_i}(x), C) - wt((S_{r_i}(y), C)| \leq l_i$, $i = 1, \ldots, h$. If $r_1, \ldots, r_h$ take all nonnegative integer values from 0 until n and $l_1 = \cdots = l_h$ then we say that C is an l-UDS code.*

If a code C is an UDS$(l_1, r_1; \ldots; l_h, r_h)$-code then, obviously, the complementary code $\overline{C}$ is also an UDS$(l_1, r_1; \ldots; l_h, r_h)$-code. Therefore it's sufficient to consider only codes C of cardinality $|C| \leq 2^{n-1}$.

Some examples of UDS-codes with good parameters were given in [1]. All 1-UDS codes were described in [2].

Theorem 1 *[1] Let l, m, n be positive integers and $u > 1$ provide $(u-1)n > 3ul + u + \frac{u^2l^2}{4}$, $n \geq 6l + 3 + \frac{ul^2}{2}$, $\frac{ul^2}{4}\frac{n+1}{2^n} \leq m \leq 2^{n-1}$. Then there does not exist an* UDS$(l, 1; l, 2)$*-code in V^n of cardinality m.*

In this paper we generalize Theorem 1 to prove the nonexistence of UDS$(l, s; l, 2s)$-codes.

Theorem 2 *Let $l, s \in$, $u > 1$, $a(s)$ is some constant of s and $m = m(n)$ provide*

$$\frac{ul^2}{4}\frac{2^n}{\sum\limits_{i=0}^{s}\binom{n}{i}} \leq m(n) \leq \left(\frac{n}{s^2} + a(s)\right)$$

Then there does not exist an UDS$(l, s; l, 2s)$*-code in V^n of cardinality m beginning with some sufficiently large n.*

Proof. Let C be a code in V^n, $|C| = m = \mathrm{P_s}\frac{2^n}{\sum\limits_{i=0}^{s}\binom{n}{i}} \leq 2^{n-1}$, where P_s is an average weight of spheres of radius s. Suppose that C is an UDS$(l, s; l, 2s)$-code. Denote the minimum weight of a sphere of radius s by h. Then the maximum weight of a sphere of radius s does not exceed $h + l$. Denote by t_i, $i = 0, \ldots, l$ the numbers of spheres of radius s with weight $h + i$. Obviously, $\sum\limits_{i=0}^{l} t_i = 2^n$, $\sum\limits_{i=0}^{l} t_i(h+i) = m \cdot \sum\limits_{i=0}^{s}\binom{n}{i}$.

We define K as the number of pairs (x, y), $x, y \in C$, $x \neq y$, such that $d(x, y) \leq 2s$. First, consider for any vector $x \in C$ the sphere $S_{2s}(x)$. The weight of this sphere is at least $(\lfloor P_{2s} \rfloor - l)$. It follows that the number of desired pairs with x is at least $(\lfloor P_{2s} \rfloor - l)$. Any desired pair is counted twice. As a result we have

$$2K \;\geq\; m \cdot (\lfloor P_{2s} \rfloor - l) = m\left(\left\lfloor \frac{m}{2^n}\sum_{i=0}^{2s}\binom{n}{i}\right\rfloor - l\right) \geq m\left(\frac{m}{2^n} \cdot \sum_{i=0}^{2s}\binom{n}{i} - l - 1\right).$$

On the other hand, any pair (x, y) in the code C such that $d(x, y) = 2s$ belongs to exactly $\binom{2s}{s}$ spheres of radius s, any pair (x, y) such that $d(x, y) = 2s - 1$ belongs to exactly $2\binom{2s-1}{s} = \binom{2s}{s}$ spheres of radius s, any pair (x, y) such that $d(x, y) < 2s - 1$ belongs to $O(n)$ spheres of radius s. Thus, any desired pair belongs to at least $\binom{2s}{s}$ spheres of radius s. Each sphere of radius s with weight $h + i$ contains $\frac{(h+i)(h+i-1)}{2}$ desired pairs. It follows

$$\binom{2s}{s}K \leq \sum_{i=0}^{l} t_i \frac{(h+i)(h+i-1)}{2}.$$

It is easy to check that for any $0 < i < j < l$ we have

$$\frac{(h+i)(h+i-1)}{2} + \frac{(h+j)(h+j-1)}{2} \leq \frac{(h+i-1)(h+i-2)}{2} + \frac{(h+j+1)(h+j)}{2}.$$

Therefore,

$$\begin{aligned}\binom{2s}{s}K &\le \sum_{i=0}^{l} t_i\frac{(h+i)(h+i-1)}{2} \le t_0'\frac{h(h-1)}{2}+t_l'\frac{(h+l)(h+l-1)}{2} = \\ &= (t_0'+t_l')\frac{h(h-1)}{2}+t_l'\frac{2hl+l^2-l}{2}\end{aligned}$$

where $t_0'+t_l'=2^n$ and $t_0'h+t_l'(h+l)=m\sum_{i=0}^{s}\binom{n}{i}$, it follows $t_l'=\frac{m\sum_{i=0}^{s}\binom{n}{i}-2^nh}{l}$. Thus,

$$\binom{2s}{s}K \;\le\; 2^n\frac{h(h-1)}{2}+\left(m\sum_{i=0}^{s}\binom{n}{i}-2^nh\right)\frac{2h+l-1}{2}.$$

Using the lower and upper bounds for K we have

$$\binom{2s}{s}m\left(\frac{m}{2^n}\cdot\sum_{i=0}^{2s}\binom{n}{i}-l-1\right)\le 2\left(2^n\frac{h(h-1)}{2}+\left(m\sum_{i=0}^{s}\binom{n}{i}-2^nh\right)\frac{2h+l-1}{2}\right).$$

Next, substitute $m=P_s\frac{2^n}{\sum_{i=0}^{s}\binom{n}{i}}$. We obtain

$$\binom{2s}{s}P_s^2\cdot\frac{2^n\sum_{i=0}^{2s}\binom{n}{i}}{\left(\sum_{i=0}^{s}\binom{n}{i}\right)^2}-\binom{2s}{s}P_s\cdot\frac{2^n}{\sum_{i=0}^{s}\binom{n}{i}}(l+1)\le 2^nh(h-1)+(P_s\cdot 2^n-2^nh)\,(2h+l-1).$$

Multiply both sides of inequality by $\frac{\sum_{i=0}^{s}\binom{n}{i}}{2^n}$. We have

$$\binom{2s}{s}P_s^2\cdot\frac{\sum_{i=0}^{2s}\binom{n}{i}}{\sum_{i=0}^{s}\binom{n}{i}}-\binom{2s}{s}P_s(l+1)\le(h(h-1)+(P_s-h)(2h+l-1))\cdot\sum_{i=0}^{s}\binom{n}{i}. \quad (1)$$

Decompose the sums of binomial coefficients into powers of n. We can assume that $\frac{n}{2}>2s$.

$$\begin{aligned}\sum_{i=0}^{s}\binom{n}{i} &= \frac{n^s}{s!}+\left(-\frac{1+2+\cdots+(s-1)}{s!}+\frac{1}{(s-1)!}\right)n^{s-1}+O(n^{s-2})= \\ &= \frac{n^s+\frac{3s-s^2}{2}n^{s-1}+O(n^{s-2})}{s!}, \\ \sum_{i=0}^{2s}\binom{n}{i} &= \frac{n^{2s}+(3s-2s^2)n^{2s-1}+O(n^{2s-2})}{(2s)!}.\end{aligned}$$

The last two equalities follow

$$\binom{2s}{s}\cdot\frac{\sum_{i=0}^{2s}\binom{n}{i}}{\sum_{i=0}^{s}\binom{n}{i}}=\frac{1}{s!}\left(n^s+\frac{3s-3s^2}{2}n^{s-1}+O(n^{s-2})\right).$$

Next, substitute these two equalities to (1):

$$\begin{aligned}&\frac{1}{s!}\left(n^s+\frac{3s-3s^2}{2}n^{s-1}+O(n^{s-2})\right)P_s^2-\binom{2s}{s}P_s(l+1)\le \\ &\le(h(h-1)+(P_s-h)(2h+l-1))\frac{1}{s!}\left(n^s+\frac{3s-s^2}{2}n^{s-1}+O(n^{s-2})\right),\end{aligned}$$

Divide both sides of the last inequality by $\frac{n^{s-1}}{s!}$:

$$\begin{aligned}&\left(n+\frac{3s-3s^2}{2}+O(\frac{1}{n})\right)P_s^2-\frac{(2s)!}{s!}\frac{P_s}{n^{s-1}}(l+1)\le \\ &\le(h(h-1)+(P_s-h)(2h+l-1))\left(n+\frac{3s-s^2}{2}+O\left(\frac{1}{n}\right)\right),\end{aligned}$$

The hypothesis of Theorem follows $\frac{P_s}{n^{s-1}}\le P_s^2\cdot O(\frac{1}{n})$ for $s>1$. Therefore the term $\frac{(2s)!}{s!}\frac{P_s}{n^{s-1}}(l+1)$ can be considered as negligible. Grouping the terms we have

$$\begin{aligned}&A=\left(n+\frac{3s-3s^2}{2}+O\left(\frac{1}{n}\right)\right)(P_s-h)(P_s-h-l)+ \\ &P_s\left(n+\frac{3s-s^2}{2}+O\left(\frac{1}{n}\right)-(h+l)\left(s^2+O\left(\frac{1}{n}\right)\right)\right)+h(h+l-P_s)\left(s^2+O\left(\frac{1}{n}\right)\right)\le 0.\end{aligned}$$

In order to prove the nonexistence of UDS $(l,s;l,2s)$-codes we demonstrate that under the hypothesis of Theorem the left side of the last inequality is nonnegative. The inequality $P_s^2-2P_sh+h^2+hl\ge 0$ follows

$$A\ge\left(n+\frac{3s-3s^2}{2}+O\left(\frac{1}{n}\right)\right)(P_s-h)(P_s-h-l)+s^2P_s\left(\frac{n}{s^2}+\frac{3-s}{2s}+O\left(\frac{1}{n}\right)-l-P_s+O\left(\frac{P_s}{n}\right)\right).$$

Properties of parabola follow the next inequalities.

First, for any P_s it holds $(P_s-h)(P_s-h-l)\ge-\frac{l^2}{4}$,

second, if $\quad \frac{ul^2}{4}\le P_s\le\frac{n}{s^2}+\mathrm{const}(s)-\frac{ul^2}{4}\le\frac{n}{s^2}+a(s)$

holds then $\quad s^2P_s\left(\frac{n}{s^2}+\frac{3-s}{2s}+O\left(\frac{1}{n}\right)-l-P_s+O\left(\frac{P_s}{n}\right)\right)\ge s^2\frac{ul^2}{4}\left(\frac{n}{s^2}+O(1)\right).$

Thus, it is sufficient to prove that $-\frac{l^2}{4}(n+O(1))+\frac{ul^2}{4}(n+O(1))\ge 0$. If $u>1$ this inequiality holds for sufficiently large n. □

Theorem 2 proves the nonexistence of l-UDS codes for the next cardinalities m:

$$\frac{ul^2}{4}\cdot\frac{2^n}{n^s+O(n^{s-1})}\;\le m(n)\le\;\frac{2^n}{s^2n^{s-1}+O(n^{s-2})}\quad\text{for}\quad s=2,3,\ldots$$

References

[1] M.Fedorova, Yu. Tarannikov. On impossibility of uniform distribution of codewords over spheres in some cases. Proceedings of 2002 IEEE International Symposium on Information Theory (Lausanne, June 30 – July 5, 2002), to appear.

[2] Yu. Tarannikov. A class of Boolean functions homogeneously distributed over balls with degree 1, Moscow University Mathematics Bulletin, 1997, Vol. 52, No 5, pp. 18–22.

On Computing the Fast Fourier Transform over Finite Fields

Sergei Fedorenko[1] and Peter Trifonov

St.Petersburg State Polytechnical University,
Distributed Computing and Networking Department,
Politekhnitcheskaya st., 21, Room 9–104,
St.Petersburg, 194021, Russia.
sfedorenko@icee.org petert@dcn.nord.nw.ru

In this paper we consider the problem of computing the Fast Fourier Transform of a polynomial over finite fields. The polynomial is decomposed into a sum of linearized polynomials allowing one to use fast evaluation algorithms. An example of the FFT algorithm with the complexity lower than the best one known to the authors is provided.

1 Introduction

Currently there exist a lot of algorithms for computing the Fast Fourier Transform (FFT) over the field of complex numbers. Many of these algorithms can be used in the case of finite fields, but in practice the problem of construction of FFT for a finite field remains hard and poorly formalized [3].

In this paper we suggest an universal approach for the construction of FFT algorithms over the fields of characteristics 2. The algorithm is based on the decomposition of an arbitrary polynomial into a sum of linearized polynomials allowing thus usage of the effective evaluation algorithms [2].

2 Basic definitions

Definition 1. The polynomial over $GF(2^m)$ is called linearized if

$$L(x)=\sum_i l_i x^{2^i}, \quad l_i \in GF(2^m).$$

It can be easily proved that for linearized polynomials $L(a+b)=L(a)+L(b)$ holds. This property leads to the following theorem presented here in a slightly modified form:

[1] This work was supported by the Alexander von Humboldt Foundation.

Theorem 1 ([1]). *Let $x \in GF(2^m)$ and let $\beta_0, \beta_1, \ldots, \beta_{m-1}$ be a basis of the field.*

$$\textit{If} \quad x=\sum_{i=0}^{m-1} x_i\beta_i, \quad x_i \in GF(2), \qquad \textit{then} \quad L(x)=\sum_{i=0}^{m-1} x_i L(\beta_i).$$

Let us consider cyclotomic cosets modulo $n=2^m-1$ over $GF(2)$: $\{0\}, \{k_1, k_1 2, k_1 2^2, \ldots, k_1 2^{m_1-1}\}, \ldots, \{k_l, k_l 2, k_l 2^2, \ldots, k_l 2^{m_l-1}\}$, where $k_i \equiv k_i 2^{m_i} \bmod n$.

Then any polynomial $f(x)=\sum_{i=0}^{n-1} f_i x^i, f_i \in GF(2^m)$ can be decomposed as

$$f(x)=\sum_{i=0}^{l} L_i(x^{k_i}),\, L_i(y)=\sum_{j=0}^{m_i-1} f_{k_i 2^j \bmod n} y^{2^j}. \tag{1}$$

In fact (1) represents a way of grouping numbers $0 \le s < n$ into cyclotomic cosets: $s \equiv k_i 2^j \bmod n$. Obviously, this decomposition is always possible. Note, that term f_0 can be represented as $L_0(x^0)$, where $L_0(y)=f_0 y$.

3 Fast Fourier Transform

Let us consider the problem of computing the FFT of a polynomial $f(x)$, i.e. computing values $f(\alpha^j)=\sum_{i=0}^{n-1} f_i \alpha^{ij}$, where α is a primitive element of $GF(2^m)$. According to (1), $f(\alpha^j)$ can be represented as $f(\alpha^j)=\sum_{i=0}^{l} L_i(\alpha^{jk_i})$. It is known [1] that α^{k_i} is a root of a minimal polynomial of degree m_i and thus belongs to a subfield $GF(2^{m_i})$, $m_i \mid m$. Thus all the values $(\alpha^{k_i})^j$ lie in $GF(2^{m_i})$ and so they can be decomposed in some basis $(\beta_{i,0}, \ldots, \beta_{i,m_i-1})$ of the subfield: $\alpha^{jk_i}=\sum_{s=0}^{m_i-1} a_{ijs}\beta_{i,s}, a_{ijs} \in GF(2)$. Then, according to the theorem 1,

$$F_j=f(\alpha^j)=\sum_{i=0}^{l}\sum_{s=0}^{m_i-1} a_{ijs}L_i(\beta_{i,s})=\sum_{i=0}^{l}\sum_{s=0}^{m_i-1} a_{ijs}\left(\sum_{p=0}^{m_i-1}\beta_{i,s}^{2^p} f_{k_i 2^p}\right). \tag{2}$$

This equation can be represented in matrix form as $F=ALf$, where $F=\|F_j\|$, $f=\|f_j\|$, A is a matrix with elements $a_{ijs} \in GF(2)$, L is a block diagonal matrix with elements $\beta_{i,s}^{2^p}$.

It is possible to choose the same basis for all the linearized polynomials of the same degree m_i in (1) and obtain very small amount of different blocks in matrix L. This can simplify the problem of construction of a fast algorithm for multiplication of a matrix L by a vector f over $GF(2^m)$.

The described transforms are similar to the ones presented in [4]. The main differences are:

1. Matrix L has regular structure which can be used for a further optimization.
2. There is a single multiplication of a binary matrix by a vector. This can be used for a better optimization.

Example 1. A polynomial $f(x) = \sum_{i=0}^{6} f_i x^i, \quad f_i \in GF(2^3)$ can be represented as

$$\begin{aligned} f(x) &= L_0(x^0) + L_1(x) + L_2(x^3) \\ L_0(y) &= f_0 y \\ L_1(y) &= f_1 y + f_2 y^2 + f_4 y^4 \\ L_2(y) &= f_3 y + f_6 y^2 + f_5 y^4. \end{aligned}$$

Let us choose as basis elements of $GF(2^3)$ the standard basis and represent the components of Fourier transform as

$$\begin{aligned} f(\alpha^0) &= L_0(\alpha^0) + L_1(\alpha^0) + L_2(\alpha^0) \\ f(\alpha^1) &= L_0(\alpha^0) + L_1(\alpha) + L_2(\alpha^3) = L_0(1) + L_1(\alpha) + L_2(1) + L_2(\alpha) \\ f(\alpha^2) &= L_0(\alpha^0) + L_1(\alpha^2) + L_2(\alpha^6) = L_0(1) + L_1(\alpha^2) + L_2(1) + L_2(\alpha^2) \\ f(\alpha^3) &= L_0(\alpha^0) + L_1(\alpha^3) + L_2(\alpha^2) = L_0(1) + L_1(1) + L_1(\alpha) + L_2(\alpha^2) \\ f(\alpha^4) &= L_0(\alpha^0) + L_1(\alpha^4) + L_2(\alpha^5) = L_0(1) + L_1(\alpha) + L_1(\alpha^2) + L_2(1) + L_2(\alpha) + L_2(\alpha^2) \\ f(\alpha^5) &= L_0(\alpha^0) + L_1(\alpha^5) + L_2(\alpha) = L_0(1) + L_1(1) + L_1(\alpha) + L_1(\alpha^2) + L_2(\alpha) \\ f(\alpha^6) &= L_0(\alpha^0) + L_1(\alpha^6) + L_2(\alpha^4) = L_0(1) + L_1(1) + L_1(\alpha^2) + L_2(\alpha) + L_2(\alpha^2), \end{aligned}$$

where α is a root of the primitive polynomial x^3+x+1. These equations can be represented in a matrix form as

$$F = \begin{pmatrix} F_0 \\ F_1 \\ F_2 \\ F_3 \\ F_4 \\ F_5 \\ F_6 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} L_1(1) \\ L_1(\alpha) \\ L_1(\alpha^2) \\ L_2(1) \\ L_2(\alpha) \\ L_2(\alpha^2) \\ f_0 \end{pmatrix} = AS.$$

Then the problem of computing the FFT of a polynomial $f(x)$ can be represented as

$$F = A \begin{pmatrix} W & \mathbf{0} & 0 \\ \mathbf{0} & W & 0 \\ 0 & 0 & 1 \end{pmatrix} (f_1, f_2, f_4, f_3, f_6, f_5, f_0)^T, \quad W = \begin{pmatrix} 1 & 1 & 1 \\ \alpha & \alpha^2 & \alpha^4 \\ \alpha^2 & \alpha^4 & \alpha^8 \end{pmatrix}. \qquad (3)$$

The first stage of the algorithm is computing $\begin{pmatrix} b_{i1} \\ b_{i2} \\ b_{i3} \end{pmatrix} = W \begin{pmatrix} a_{i1} \\ a_{i2} \\ a_{i3} \end{pmatrix}$ for $i = 1, 2$, where $a_{11} = f_1$, $a_{12} = f_2$, $\ldots$, $a_{23} = f_5$ (see (3)). This can be implemented using the following algorithm:

$$\begin{aligned} b_{i1} &= a_{i1} + a_{i2} + a_{i3} \\ b_{i2} &= \alpha(a_{i1} + a_{i2}) + \alpha^4(a_{i2} + a_{i3}) \\ b_{i3} &= \alpha^2(a_{i1} + a_{i3}) + \alpha^4(a_{i2} + a_{i3}), \end{aligned}$$

which requires 3 multiplications and 6 additions. At the end of the first stage one obtains the vector $S = (S_0, \ldots, S_6) = (b_{11}, b_{12}, b_{13}, b_{21}, b_{22}, b_{23}, f_0)$. The following algorithm computes the product of a binary matrix A with vector S:

$$\begin{array}{ll} T_7 = S_3 + S_6 & F_2 = T_2 = T_7 + T_{10} \\ T_8 = S_1 + S_5 & F_3 = T_3 = S_6 + T_{12} \\ T_9 = S_1 + S_4 & F_4 = T_4 = T_1 + T_{10} \\ T_{10} = S_2 + S_5 & T_{11} = S_2 + T_3 \\ T_{12} = S_0 + T_8 & F_6 = T_6 = T_9 + T_{11} \\ F_0 = T_0 = S_0 + T_7 & F_5 = T_5 = T_6 + T_8. \\ F_1 = T_1 = T_7 + T_9 & \end{array}$$

Thus the FFT of length 7 can be computed with $2 \times 3 = 6$ multiplications and $2 \times 6 + 13 = 25$ additions. This is smaller by one addition than in the algorithm presented in [4].

If one chooses the normal basis in (2) then all the blocks of the matrix L are circulant matrices. Thus the problem of the multiplication by this matrix can be considered as a problem of the computing a set of circular convolutions of degree $m_i \mid m$. Application of these techniques allowed us to construct the FFT algorithm of length 15 with $3 \times 5 + 1 \times 1 = 16$ multiplications and $3 \times 10 + 1 \times 2 + 45 = 77$ additions which is better than the ones presented in [4] (16 multiplications and 100 additions) and [3] (20 multiplications and 70 additions).

4 Conclusions

In this paper we suggested an algorithm for computing the FFT of a polynomial over $GF(2^m)$. The task of computing the FFT of length $n = 2^m -$ can be reduced to computing the circular convolutions of length $m_i \mid m$ and multiplication of a binary matrix by a vector.

References

[1] E.R. Berlekamp. *Algebraic coding theory.* New York: McGraw-Hill. 1968.

[2] S.V. Fedorenko and P.V. Trifonov. Finding roots of polynomials over finite fields. *Accepted for publication in IEEE Transactions on Communications*, 2002.

[3] E.M. Gabidulin and V.B. Afanasyev. *Coding in radioelectronics.* Moscow, Radio i Svyaz, 1986 (in Russian).

[4] T.G. Zakharova. Fourier transform evaluation in fields of characteristic 2. *Problems of Information Transmission*, 28(2): 154–167, 1992.

New characterizations of $\{\delta v_{\mu+1}, \delta v_{\mu}; N, q\}$-minihypers

S. Ferret[1]

(joint work with **L. Storme**)

Ghent University, Dept. of Pure Maths and Computer Algebra, Krijgslaan 281, 9000 Gent, Belgium

(S. Ferret: saferret@cage.rug.ac.be, http://cage.rug.ac.be/~saferret)

(L. Storme: ls@cage.rug.ac.be, http://cage.rug.ac.be/~ls)

Abstract

We classify all $\{\delta v_{\mu+1}, \delta v_{\mu}; t, q\}$-minihypers, δ small, $q = p_0^h$, $h \geq 1$, for a prime number $p_0 \geq 7$. When q is a third power, the minihyper is the disjoint union of $PG(\mu, q)$'s and of projected $PG(3\mu + 2, \sqrt[3]{q})$'s; when q is a square, also Baer subgeometries $PG(2\mu + 1, \sqrt{q})$ can occur.

1 Introduction

We continue the study of the minihypers considered by P. Govaerts and L. Storme [2].

For simplicity, in this abstract, we will describe how to classify $\{\delta(q+1), \delta; 3, q\}$-minihypers

F.

At the end, we will also state the general result.

For references and the proofs of the general result, we refer to the article [1].

Definition 1.1 *An $\{f, m; N, q\}$-minihyper is a pair (F, w), where F is a subset of the point set of $PG(N, q)$ and w is a weight function $w : PG(N, q) \rightarrow \mathbb{N} : x \mapsto w(x)$, satisfying*

(1) $w(x) > 0 \Leftrightarrow x \in F$,

(2) $\sum_{x \in F} w(x) = f$, and

(3) $\min\{\sum_{x \in H} w(x) | H \in \mathcal{H}\} = m$; where $\mathcal{H}$ denotes the set of hyperplanes.

[1] This author's research was supported by the Flemish Institute for the Promotion of Scientific and Technological Research in Industry (IWT), grant No. IWT/SB/991011/Ferret

Definition 1.2 *A blocking set of $PG(2, q)$ is a set of points intersecting every line of $PG(2, q)$ in at least one point.*

A blocking set is called minimal *when no proper subset of it is still a blocking set; and we call a blocking set* non-trivial *when it contains no line.*

Theorem 1.3 (Polverino and Storme, [4]) *The smallest minimal blocking sets in $PG(2, p^3)$, $p = p_0^h$, p_0 prime, $p_0 \geq 7$, with exponent $e \geq h$, are:*

(1) a line,

(2) a Baer subplane of cardinality $p^3 + p^{3/2} + 1$, when p is a square,

(3) a set B of cardinality $p^3 + p^2 + 1$, equivalent to

$$\{(x, T(x), 1) || x \in GF(p^3)\} \cup \{(x, T(x), 0) || x \in GF(p^3) \setminus \{0\}\},$$

with T the trace function from $GF(p^3)$ to $GF(p)$,

A line intersects B in $1, p + 1$ or $p^2 + 1$ points.

The last intersection will be called a $(p^2 + 1)$-set.

(4) a set B of cardinality $p^3 + p^2 + p + 1$, equivalent to

$$\{(x, x^p, 1) || x \in GF(p^3)\} \cup \{(x, x^p, 0) || x \in GF(p^3) \setminus \{0\}\}.$$

A line intersects B in $1, p + 1$ or $p^2 + p + 1$ points.

The last intersection will be called a $(p^2 + p + 1)$-set.

Remark 1.4 These two latter blocking sets (3) and (4) are also characterized as being a projected $PG(3, p)$ in the plane $PG(2, p^3)$. Namely, embed $PG(2, p^3)$ in a 3-dimensional space $PG(3, p^3)$. Consider a subgeometry $PG(3, p)$ of $PG(3, p^3)$ and a point r not belonging to this subgeometry $PG(3, p)$ and not belonging to the plane $PG(2, p^3)$.

Project $PG(3, p)$ from r onto $PG(2, p^3)$.

If the point r belongs to a line of the subgeometry $PG(3, p)$, then this $PG(3, p)$ is projected onto the blocking set of size $p^3 + p^2 + 1$; otherwise we obtain the blocking set of size $p^3 + p^2 + p + 1$.

2 The classification

Theorem 2.1 (Hamada and Helleseth, [3]) *Let F be a $\{\delta(q + 1), \delta; 3, q\}$-minihyper where $t \geq 3$, $\delta \leq 2p^2$.*

between $x=(x_1,\dots,x_n)$, $y=(y_1,\dots,y_n)\in Q^n=\{0,1,\dots,q-1\}^n$.

$$d_a(x,y)=\max\{|x_i-y_i| : i=1,\dots,n\}$$
$$d_u(x,y)=\begin{cases} d_a(x,y), & \text{if } x\geq y \text{ or } x\leq y\\ 2d_a(x,y), & \text{if } x \text{ and } y \text{ are incomparable}\end{cases}$$

where $x\geq y$ means that $x_i-y_i\geq 0$, for $i=1,\dots,n$.

Proposition 1. Let $\mathcal{C}\subset\{0,\dots,q-1\}^n$. Then

(i) $\mathcal{C}$ is an ℓ–AAEC–code iff for every $x,y\in\mathcal{C}$ holds $d_a(x,y)\geq \ell+1$

(ii) $\mathcal{C}$ is an ℓ–AUEC–code iff for every $x,y\in\mathcal{C}$ holds $d_u(x,y)\geq 2\ell+1$.

It turns out that it is very easy to determine $A_a(n,\ell)_q$ for any given parameters $1\leq\ell\leq q-2$ and n. However this is not the case for unidirectional codes.

Theorem 1. *For $1\leq\ell\leq q-2$ one has $A_a(n,\ell)_q=\lceil\frac{q}{\ell+1}\rceil^n$.*

Theorem 2. *Given integers $\ell\geq 1$, $q>2(\ell+1)$ we have $c\left(\frac{q}{\ell+1}\right)^n\leq A_u(n,\ell)_q\leq\lceil\frac{q}{\ell+1}\rceil^n$ for some constant c.*

Write $q=2m+\varepsilon$, where $\varepsilon\in\{0,1\}$, and let $Q=\{-m,-m+1,\dots,m+\varepsilon\}$. Let us define X to be the set of solutions $x\in Q^n$ of the equation

$$\sum_{i=0}^{n-1}(\ell+1)^i x_i=a. \tag{2.1}$$

It is easy to see that X is a l–AUEC–code.In a special case when $\ell+1|q$ we can maximize $|X|$ over all choices of a.

Theorem 3. *For $\ell+1|q$ $(q=|Q|)$ $\max_a|X|=\left(\frac{q}{\ell+1}\right)^{n-1}$.* The maximum assumed for any $a\in Q=[-m,m+\varepsilon]$ in (2.1).

What can we say about $A_u(n,\ell)_q$, when $\ell+2\leq q\leq 2(\ell+1)$?

The simplest case is $q=2(\ell+1)$.In this case $A_u(n,\ell)_q=2^n$. However, we have no "good" lower bounds for other cases. A simple lower bound is $A_u(n,\ell)_q\geq\binom{n}{\lfloor\frac{n}{2}\rfloor}$.

Can we do it better?

The Case: $\ell=1$

For $q=3$ we have $A_u(n,1)_3\geq\binom{n}{\lfloor\frac{n}{2}\rfloor}$.

We believe that one has equality in this case.

For $q=4$ $A_u(n,1)_4=2^n$.

$q=5$. Simple bounds observed above give us $c(2,5)^n\leq A_u(n,1)_5\leq 3^n$. However the lower bound can be improved. To this end we look for good constructions of 1–AUEC codes given by means of some equation. Let $Q=\{0,\pm1,\pm2\}$. Given integers $a_0,\dots,a_{n-1},\lambda$ let X be the set of all solutions $x=(x_0,\dots,x_{n-1})\in Q^n$ of an equation

$$\sum_{i=0}^{n-1}a_i x_i=\lambda. \tag{2.2}$$

Proposition 2. The set X is a 1–AUEC code if all subset sums of $a_0,\dots,a_{n-1}$ are distinct.

Note that for $\lambda=0$ this is also a necessary condition. Let $\{a_0,\dots,a_n\}\subset\mathbb{N}$ has distinct subset sums. Denote by $LA_u(n)_5$ the maximum possible number of solutions $x\in Q^n$ of the (2.2) over all choices of $a_0,\dots,a_n$ and integer λ. A slightly modified version of this problem was raised by Bohman (see [1]) in connection with a sum packing problem of Erdős [3].

Theorem 4. *For some constants c_1,c_2 one has $c_1(2,538)^n<LA_u(n)_5<c_2(2,723)^n$.*

Error Detection Problem The detection problems for asymmetric and unidirectional errors are equivalent, i.e. any t–error detecting asymmetric code is also a t–error detecting unidirectional code. In fact the detection problem for unidirectional errors is much easier than the error correction problem. This problem is completely solved for binary channels (see Borden in [2]). That is for any $1\leq t\leq n$; $t,n\in\mathbb{N}$; an optimal code of length n that can detect up to t errors is constructed. For $t<n$ observe that a code C detects all patterns of t or fewer unidirectional errors, iff whenever a codeword x covers a codeword y then for the Hamming distance $d(x,y)>t+1$. In this case as an optimal code one has to take as codewords all vectors with Hamming weight $w=\lfloor\frac{n}{2}\rfloor \mod (t+1)$. This follows from a result of Katona [4].The problem is also solved for the Varshamov's channel, however for the channel we described above the problem is open.

References

[1] R. Ahlswede, H. Aydinian and L.H. Khachatrian, On Bohman's conjecture related to a sum packing problem of Erdős, submitted to Proceedings of the Amer. Math. Aoc.

[2] M. Blaum, Codes for detecting and correcting unidirectional errors. Edited by Mario Blaum. IEEE Computer Society Press Reprint Collections. IEEE Computer Society Press, Los Alamitos, CA, 1993.

[3] P. Erdős, Problems and results from additive number theory, Colloq. Theoretic des Nombres, Bruxelles, 1955, Liege&Paris, 1956.

[4] G. Katona, Families of subsets having no subset containing another with small difference, Niew. Arch. Wisk. (3) 20, 54–67, 1972.

[5] P.R. Varshamov and G.M. Tennengolts, A code which corrects single asymmetric errors (Russian) Avtomat. Telemeh. 26, 282–292, 1965.

[6] P.R. Varshamov, A class of codes for asymmetric channels and a problem from the additive theory of numbers, IEEE Trans. Inform. Theory, IT–19, No. 1, 92–95, 1973.

New families of unimodular perfect sequences of non-prime length

Ernst M. Gabidulin, Vitaly V. Shorin
Moscow Institute of Physics and Technology
gab@pop3.mipt.ru, shorin@dgap.mipt.ru

Abstract

New classes of unimodular perfect sequences are proposed. The lengths of the sequences are the product of two primes.

1 Introduction

A complex valued sequence $\mathbf{x} = (x_0, x_1, \ldots, x_{n-1})$ of length n containing at least one non-zero component is called a **perfect** sequence provided all the out-of-phase autocorrelation coefficients are equal to 0, i.e.

$$R_{\mathbf{x}}(\tau) = \sum_{s=0}^{n-1} x_s x^*_{(s+\tau) \bmod n} = 0, \ \tau = 1, 2, \ldots, n-1, \qquad (1)$$

where x^* denotes the complex conjugation of x. The sequence $\mathbf{x}$ is called **unimodular** if all components of the sequence are unimodular i.e. lying on the unit circle.

Perfect sequences have the following properties [1]. Let $\mathbf{x}$ be a perfect sequence of length n. Then sequences $\mathbf{y} = (y_0, \ldots, y_{n-1})$, where

- $\mathbf{y} = \{y_i\}, y_i := ax_i, |a| = 1$
- $\mathbf{y} = \{y_i\}, y_i := x_{(i+j)\, mod\, n}, j = 1, \ldots, n-1$
- $\mathbf{y} = \{y_i\}, y_i := x_{(ji)\, mod\, n}, gcd(j, n) = 1$
- $\mathbf{y} = \{y_i\}, y_i := x_i^*$
- $\mathbf{y} = \{y_i\}, y_i := x_i \zeta^{is}, s = 1, \ldots, n-1, \zeta^n = 1, \zeta^i \neq 1, 0 < i < n$

for $i = 0, \ldots, n-1$, are perfect as well. In addition, the Discrete Fourier Transform of a unimodular perfect sequence is the unimodular perfect sequence.

These transformations induce the following equivalence relation: perfect sequences $\mathbf{x}$ and $\mathbf{y}$ are *equivalent*, $\mathbf{x} \sim \mathbf{y}$, if and only if they can be obtained one from another by using several above transformations.

Thus the set of all unimodular perfect sequences is disjointed into equivalence classes.

The general problem is to classify unimodular perfect sequences up to equivalence and to construct unimodular perfect sequences.

From now on we consider only the case $n = p_1 p_2$, where p_1 and p_2 are primes.

It is known that there are only finitely many of equivalence classes ([1], [2]). For any $n = p_1 p_2$, construction is known for single equivalence class.

We propose a number of new equivalence classes. In addition, some sequences are obtained numerically.

2 Known results

There is a known construction of unimodular perfect sequence of length $p_1 p_2$ in the case when two unimodular perfect sequences of prime lengths p_1 and p_2 are known([4]).

Let $\mathbf{x} = (x_0, \ldots, x_{p_1 - 1})$ and $\mathbf{y} = (y_0, \ldots, y_{p_2 - 1})$ are unimodular perfect sequences. Then a sequence $\mathbf{z} = (z_0, \ldots, z_{p_1 p_2 - 1})$, where

$$z_i = x_{i \bmod p_1} y_{i \bmod p_2}, \qquad (2)$$

is the unimodular perfect sequence. This procedure is referred to as Chinese Remainder Theorem (or CRT) construction.

To the best our knowledge, no other unimodular perfect sequences are known.

3 Sequences of length 6

It is convinient to represent unimodular perfect sequence $\mathbf{x} = (x_0, x_1, x_2, x_3, x_4, x_5)$ as a 2×3 matrix

$$\mathbf{x} = \begin{bmatrix} 1 & x_4 & x_2 \\ x_3 & x_1 & x_5 \end{bmatrix}. \qquad (3)$$

We can rewrite the set of equations (1) as follows:

$$\sum_{i=0}^{n-1} \frac{x_i}{x_{(i+\tau) \bmod n}} = 0. \tag{4}$$

Then we can apply theory of exclusion consecutively for each variable x_i. Thus, equivalence classes may exist only in the next cases:

- $x_3^2 = -1$;
- $x_2 = 1$;
- $x_1^2 + x_3 x_1 + x_3^2 = 0$;
- $x_2^2 + x_2 + 1 = 0$;
- $x_2^2 + x_1 x_3 = 0$;
- $x_1^2 + x_2 = 0$;

The above cases led us to 2 non-equivalent solutions. Their representatives are:

1.
$$\mathbf{x}_1 = \begin{bmatrix} 1 & \zeta & 1 \\ \mathrm{i} & \mathrm{i}\zeta & \mathrm{i} \end{bmatrix}, \tag{5}$$
where ζ is a primitive root of degree 3 of identity. This solution was known previously [4].

2.
$$\mathbf{x}_3 = \begin{bmatrix} 1 & C & C^* \\ \mathrm{i} & \mathrm{i}\,C^* & \mathrm{i}C \end{bmatrix}, \tag{6}$$
where $C = .3660254038 + .9306048591\mathrm{i}$ is a unimodular root of polynomial $k(x) = x^4 + 2x^3 + 2x + 1$.

4 Sequences of length 15

We have found that all sequences of length 15 can be disjointed into 71 equivalence classes. These classes can be described as follows:

- 1 previously known sequence, CRT-constructed from sequences $(1, 1, \zeta_3)$ and $(1, \zeta_5, \zeta_5^4, \zeta_5^4, \zeta_5)$, where $\zeta_3^3 = 1$ and $\zeta_5^5 = 1$

- 1 previously known sequence, CRT-constructed from sequences $(1, 1, \zeta_3)$ and $(1, \zeta_5^2, \zeta_5^3, \zeta_5^3, \zeta_5^2)$, where $\zeta_3^3 = 1$ and $\zeta_5^5 = 1$
- 1 2-phase sequence $(1, 1, a, a, 1, a, 1, a, a, a, a, 1, 1, 1, a)$, where $Re\, a = -\frac{7}{8}$
- 1 3-phase sequence $(1, a, b, b, 1, b, a, b, b, b, b, a, 1, 1, b)$, where $Re\, a = -\frac{1}{4}$, $Re\, b = -\frac{7}{8}$
- also 67 sequences not listed here.

5 Results

We propose several new equivalence classes for unimodular perfect sequences of non-prime length, namely 1 new class for length $n = 6$ and 69 new classes for $n = 15$. 3 classes are presented in the paper.

References

[1] E.M. Gabidulin, "On Classification of Sequences with the Perfect Periodic Auto-Correlation Function," *Proceedings of the third International Colloquium on Coding Theory*, Sept. 25 - Oct. 2, 1990, Dilijan, pp. 24-30, Yerevan, 1991.

[2] E.M. Gabidulin "Further Results on PSK-Sequences with the Perfect Periodic Autocorrelation Function," B. Honary, M. Darnell, P. Farrell (Eds), COMMUNICATION THEORY AND APPLICATIONS I, pp. 171-176, HW Communications Ltd. 1993.

[3] R.L. Frank, "Polyphase Codes with Good Nonperiodic Correlation Properties," *IEEE Trans. Inform. Theory*, vol. IT-9, pp. 43-45, January 1963.

[4] P. Fan, M. Darnell, "Sequences Design for Communicationa Applications," RSP Ltd. 1996.

Representation of a finite field by symmetric matrices and applications

Ernst M. Gabidulin Nina I. Pilipchuk
gab@pop3.mipt.ru, nina.pilipchuk@pop3.mipt.ru

Abstract

It is well known [1] that a field $GF(q^n)$ can be described in terms of an $n \times n$ matrix A such that all the powers A^i, $i = 1, 2, \ldots, q^n - 1$, corresponds to field elements α^i, where α is a primitive element of a field. We show that for binary fields $GF(2^n)$ a matrix A can be chosen as a symmetric one. This representation is useful in theory of rank codes. The standard fast decoding algorithm can be applied to a corrupted received code matrix as well as to the transposed version of this matrix. It allows to correct rank erasures in more easy way.

1 Matrix representations of a finite field

Let $GF(q^n)$ be a finite field with a primitive element α. Let α be a root of an irreducible primitive monic polynomial

$$f(\lambda) = \lambda^n + a_{n-1}\lambda^{n-1} + a_{n-2}\lambda^{n-2} + \ldots + a_1\lambda^1 + a_0. \qquad (1)$$

The elements α^j, $j = 1, 2, \ldots, q^n - 1$ are all non zero elements of $GF(q^n)$. Moreover, for $i \neq j$, we have $\alpha^i - \alpha^j = \alpha^k$.

Let A be an $n \times n$ matrix over the base field $GF(q)$. We say that the matrix A represents the field $GF(q^n)$ if and only if all the powers A^j, $j = 1, 2, \ldots, q^n - 1$ are distinct, $A^{q^n-1} = I_n$, where I_n is the identity matrix of order n, and $A^i - A^j = A^k$, $i \neq j$.

It is known [1] that the companion matrix C of the polynomial (1)

$$C = \begin{pmatrix} 0 & 0 & \ldots & 0 & a_0 \\ 1 & 0 & \ldots & 0 & a_1 \\ \vdots & \vdots & \ldots & \vdots & \vdots \\ 0 & 0 & \ldots & 0 & a_{n-2} \\ 0 & 0 & \ldots & 1 & a_{n-1} \end{pmatrix} \qquad (2)$$

represents the field $GF(q^n)$. The characteristic polynomial of this matrix is $f(x)$ from (1).

All the other matrices representing the same field are of the form $A = QCQ^{-1}$, where Q is a square nonsingular matrix of order n over the base field $GF(q)$.

2 Symmetric matrices representing a field

From now on we consider binary fields only, i.e., $q = 2$. We find a symmetric representing matrix A in the form

$$A = \begin{pmatrix} y_1 & 1 & 0 & \ldots & 0 & 0 & x_1 \\ 1 & y_2 & 1 & \ldots & 0 & 0 & x_2 \\ 0 & 1 & y_3 & \ldots & 0 & 0 & x_3 \\ \vdots & \vdots & \vdots & \ldots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \ldots & y_{n-2} & 1 & x_{n-2} \\ 0 & 0 & 0 & \ldots & 1 & y_{n-1} & x_{n-1} \\ x_1 & x_2 & x_3 & \ldots & x_{n-2} & x_{n-1} & y_n \end{pmatrix} \qquad (3)$$

Let $\Delta_n(z_1, z_2, \ldots, z_n) = \det J_n$, where J_n means a three-diagonal Jacobi matrix

$$J_n = \begin{pmatrix} z_1 & 1 & 0 & \ldots & 0 & 0 & 0 \\ 1 & z_2 & 1 & \ldots & 0 & 0 & 0 \\ 0 & 1 & z_3 & \ldots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ldots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \ldots & z_{n-2} & 1 & 0 \\ 0 & 0 & 0 & \ldots & 1 & z_{n-1} & 1 \\ 0 & 0 & 0 & \ldots & 0 & 1 & z_n \end{pmatrix}.$$

Let $F(\lambda) = \det(\lambda I_n + A)$ be the characteristic polynomial of the matrix (3). For $i = 1, 2, \ldots, n$, let $z_i = \lambda + y_i$.

Lemma 1 *The characteristic polynomial of the matrix (3) is given by*

$$\begin{aligned} F(\lambda) = \; & z_n\Delta_{n-1}(z_1, z_2, \ldots, z_{n-1}) + \\ & x_1\Delta_{n-2}(z_2, z_3, \ldots, z_{n-1}) + \\ & x_2\Delta_1(z_1)\Delta_{n-3}(z_3, z_4, \ldots, z_{n-1}) + \\ & \ldots + \\ & x_{n-2}\Delta_{n-3}(z_1, z_2, \ldots, z_{n-3})\Delta_1(z_{n-1}) + \\ & x_{n-1}\Delta_{n-2}(z_1, z_2, \ldots, z_{n-2}). \end{aligned}$$

We have to prove that there exist binary entries $y_1, y_2, \ldots, y_n$; $x_1, x_2, \ldots, x_{n-1}$ such that the polynomial $F(\lambda)$ coincides with the irreducible polynomial (1).

The leading coefficient of $F(\lambda)$ is equal to 1. All the other coefficients are rather complicated functions of y_j, x_k, $j = 1, \ldots, n$; $k = 1, \ldots, n-1$. We will consider these entries as i.i.d. binary variables with the uniform distribution.

Theorem 1 *All the coefficients of $F(\lambda)$ except the leading coefficient 1 of λ^n are i.i.d. binary variables with the uniform distribution.*

This means that, in fact, there exists a matrix of the form (3) with **any** characteristic polynomial of degree n. In particular, there exists a matrix of the form (3) with the characteristic polynomial coinciding with the irreducible polynomial (1).

Proof of Theorem 1 is based on a few lemmata. The idea is that each coefficient of $F(\lambda)$ contains some y and x as a linear part and these linear parts are linearly independent.

Consider a special case of Jacobi matrices when $z_1 = z_2 = z_3 = \ldots = z_n = \lambda$.

Lemma 2 *The polynomial $\Delta_n(\lambda, \lambda, \ldots, \lambda)$ contains only* ***odd*** *powers of λ if n is* ***odd****, and only* ***even*** *powers of λ if n is* ***even****.*

Lemma 3 *The variables $y_1, y_2, \ldots, y_n$ are linear parts of coefficients for powers of λ defined by polynomials*

$$\begin{array}{ll} y_1: & \lambda\Delta_{n-2}(\lambda,\lambda,\ldots,\lambda) \\ y_2: & \lambda\Delta_{n-3}(\lambda,\lambda,\ldots,\lambda)\Delta_1(\lambda) \\ \vdots: & \vdots \\ y_{n-1}: & \lambda\Delta_{n-2}(\lambda,\lambda,\ldots,\lambda) \\ y_n: & \Delta_{n-1}(\lambda,\lambda,\ldots,\lambda). \end{array}$$

Corollary 1 *Variables $\{y\}$ are linear parts for the next powers of λ: $\{n-1, n-3, n-5, \ldots\}$*

Lemma 4 *Polynomials corresponding to variables $y_n, y_{n-1}, \ldots, y_{[\frac{n}{2}]+1}$ are linearly independent.*

Lemma 5 *The variables $x_1, x_2, \ldots, x_n$ are linear parts of coefficients for powers of λ defined by polynomials*

$$\begin{array}{ll} x_1: & \Delta_{n-2}(\lambda,\lambda,\ldots,\lambda) \\ x_2: & \Delta_{n-3}(\lambda,\lambda,\ldots,\lambda)\Delta_1(\lambda) \\ \vdots: & \vdots \\ x_{n-2}: & \Delta_{n-3}(\lambda,\lambda,\ldots,\lambda)\Delta_1(\lambda) \\ x_{n-1}: & \Delta_{n-2}(\lambda,\lambda,\ldots,\lambda). \end{array}$$

Corollary 2 *Variables $\{x\}$ are linear parts for the next powers λ: $\{n-2, n-4, n-6, \ldots\}$*

Corollary 3 *Couples of variables $\{x_1, x_{n-1}\}, \{x_2, x_{n-2}\}, \ldots$ are in the same linear parts.*

Lemma 6 *Polynomials corresponding to variables $x_{n-1}, x_{n-2}, \ldots, y_{[\frac{n}{2}]}$ are linearly independent.*

Corollary 4 *Linear parts for powers of λ $n-1, n-2, n-3, \ldots, 2, 1, 0$ are linearly independent.*

2.1 Examples

Next symmetric matrices represent fields $GF(2^2)$, $GF(2^3)\, GF(2^4)$, $GF(2^5)$, $GF(2^6)$:

$$A_2 = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}; \quad A_3 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix}; \quad A_4 = \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix};$$

$$A_5 = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \end{pmatrix}; \quad A_6 = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \end{pmatrix},$$

with the corresponding irreducible primitive polynomials $f_2(\lambda) = \lambda^2 + \lambda + 1$, $f_3(\lambda) = \lambda^3 + \lambda + 1$, $f_4(\lambda) = \lambda^4 + \lambda^3 + 1$, $f_5(\lambda) = \lambda^5 + \lambda^2 + 1$, $f_6(\lambda) = \lambda^6 + \lambda + 1$.

3 Applications to rank codes

Consider a Rank code consisting of the set of binary $n \times n$ matrices $\left\{0_n \bigcup_{j=1}^{2^{n-1}} (C^j), \right\}$, where C is the companion matrix (2) of an irreducible primitive polynomial. This code is a maximal one and can correct up to $\lfloor \frac{(n-1)}{2} \rfloor$ rank errors, or, up to $n-1$ rank erasures.

If we represent each **column** of all the code matrices as an element of the extended field $GF(2^n)$ then we obtain a $GF(2^n)$-linear $[n, 1, n]$ Rank code of length n, number of information symbols 1 and rank distance n (see, [2] for details). There exist fast algorithms for correcting rank errors and similar algorithms for correcting **column** erasures. Correcting **row** erasures is still more complicated.

On the other hand, if we represent each **row** of all the code matrices as an element of the extended field $GF(2^n)$ then we obtain a $GF(2)$-linear $[n, 1, n]$ Rank code. To the best our knowledge fast decoding algorithms do not exists for this representation.

Use a symmetric representing matrix A instead of the companion matrix C allows to overcome the difference between columns and rows. Lifting to the extended field $GF(2^n)$ by columns is equivalent to lifting by rows. Hence we have dual symmetric algorithms to correct both row erasures and column erasures.

References

[1] F.J. MacWilliams, N.J.A. Sloane, "The Theory of Error Correcting Codes," 8th ed, North Holland Press, Amsterdam, 1993.

[2] E.M.Gabidulin, "Theory of Codes with Maximum Rank Distance," *Problems of Information Transmission*, v. 21, No. 1, pp. 3-14, 1985.

Vandermonde and $\mathcal{F}$-metrics

Gabidulin E.M.,[*] Obernikhin V.A. [†]

May 2002

Abstract

Metrics based on projective sets ($\mathcal{F}$-metrics) first have been introduced in [1]. Special $\mathcal{F}$-metrics associated with generalized Vandermonde matrixes is used in this paper to construct a linear code. Weight distribution of the code is found. Algorithms of coding and fast decoding are described.

1 Introduction

A lot of paper in algebraic coding theory are devoted to codes in Hamming metrics, rank metrics as well as to burst-correcting codes. Other metrics, e.g. $\mathcal{F}$-metrics suggested in [2], [3], [4], are not very well examined. One can mention for example, papers [5], [6] and others. However, these metrics can open possibilities not only for correcting new types of errors but also for applications in other fields, for example, in cryptography. In this work we use a class of $\mathcal{F}$-metrics associated with generalized Vandermonde matrix. It happened possible to construct a meaningfull theory for the class of metrics.

The paper is organized as follows: in Section II $\mathcal{F}$-metrics associated with generalized Vandermonde matrix are introduced. Code properties in the metrics are examined. Weight distribution and fast decoding algorithm are found.

Some future directions are mentioned in conclusion.

A reader interested in definitions and general properties of $\mathcal{F}$-metrics as well as in that of the parent code should refer to [1].

2 Codes in Vandermonde $\mathcal{F}$-metrics

2.1 Vandermonde $\mathcal{F}$-metrics

Let Ω be an n-dimensional vector space over a finite field $\mathbb{F}_q = GF(q)$. Usually Ω will be a vector space $\mathbb{F}_q^n$.

Let the vectors $\vec{f}_1, \vec{f}_2, \dots, \vec{f}_N$ defining a projective $\mathcal{F}$-metrics be columns of a generalized Vandermonde matrix.

$$\mathbf{F} = \begin{pmatrix} u_1 & u_2 & \dots & u_N \\ u_1x_1 & u_2x_2 & \dots & u_Nx_N \\ u_1x_1^2 & u_2x_2^2 & \dots & u_Nx_N^2 \\ \dots & \dots & \dots & \dots \\ u_1x_1^{n-1} & u_2x_2^{n-1} & \dots & u_Nx_N^{n-1} \end{pmatrix} \quad (1)$$

where $n \le N$, $x_i \in \mathbb{F}_q$ are different from each other and $u_i \in \mathbb{F}_q$ do not equal zero, $i \in \{1, \dots, N\}$.

We will take the liberty to refer to the above $\mathcal{F}$-metrics as "Vandermonde $\mathcal{F}$-metrics".

Parent code (see [1]) for the $\mathcal{F}$-metrics will be GRS-code. Coset weight distribution for the code is well known.

2.2 Codes

Linear $[n, k]$ code C is defined by its transposed generator matrix

$$\mathbf{G}^T = \begin{pmatrix} g_{11} & g_{21} & \dots & g_{k1} \\ g_{12} & g_{22} & \dots & g_{k2} \\ \dots & \dots & \dots & \dots \\ g_{1n} & g_{2n} & \dots & g_{kn} \end{pmatrix}. \quad (2)$$

If $\vec{a} = (a_1, a_2, \dots, a_k)^T$ is a message, the corresponding code vector is calculated as $\vec{g} = \mathbf{G}^T \cdot \vec{a}$. Let us define $\mathbf{G}^T$ matrix:

$$\mathbf{G}^T = \begin{pmatrix} v_1 & v_2 & \dots & v_k \\ v_1y_1 & v_2y_2 & \dots & v_ky_k \\ v_1y_1^2 & v_2y_2^2 & \dots & v_ky_k^2 \\ \dots & \dots & \dots & \dots \\ v_1y_1^{n-1} & v_2y_2^{n-1} & \dots & v_ky_k^{n-1} \end{pmatrix}, \quad (3)$$

where $v_i \in \mathbb{F}_q$ are not equal to zero and $y_i \in \mathbb{F}_q$ differ from each other.

Besides, let us choose y_i in a way that no y_i equals x_j, $i \in \{1, \dots, k\}$, $j \in \{1, \dots, N\}$. Code dimension k must satisfy an inequality $k + N \le q + 1$. We impose the last condition because maximum number of columns in a generalized Vandermonde matrix over $GF(q)$ equals $q + 1$ and we would like for a concatenation of the two matrixes: $\mathbf{F}$ and $\mathbf{G}^T$to be the generalized Vandermonde matrix.

Lemma 1. *If $\vec{g}$ is a non-zero code vector*

$$\vec{g} = \mathbf{G}^T \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_k \end{pmatrix}$$

and its Hamming weight is $d_H(\vec{a}) = m$, then $N_{\mathcal{F}}(\vec{g}) = n - m + 1$.

Proof. Proof follows directly from the fact that any n columns of the Vandermonde matrix are linearly independent. □

Corollary 1. *Maximum $\mathcal{F}$-distance $d_{\mathcal{F}}$ of the code C equals $n - k + 1$. Hence, the code can correct up to $t_k = \left[\frac{n-k}{2}\right]$ $\mathcal{F}$-errors.*

Corollary 2. *($\mathcal{F}$-weight distribution of the code C). The number $A(i)$ of code vectors that have $\mathcal{F}$-weight equal to i is defined by the formula:*

$$A(i) = \left[\begin{array}{l} 1, \ i = 0; \\ A(i) = C_k^{n-i+1}(q-1)^{n-i+1}, \ i \in \{n-k+1, \dots, n\}. \end{array} \right.$$

[*] gab@pop3.mipt.ru

[†] obernikhin@8ka.mipt.ru

2.3 Fast decoding algorithm

We reduce decoding in the $\mathcal{F}$-metrics to the decoding of GRS codes.

Let $\vec{g}$ be a code vector and $\vec{e}$ be an error. We will show there is a fast decoding algorithm if the $\mathcal{F}$-weight $\mathcal{N}_{\mathcal{F}}(\vec{e})$ is not greater than $\left[\frac{d_{\mathcal{F}}-1}{2}\right]$.

Let us consider concatenation $(\mathbf{F}|\mathbf{G}^T)$ of the matrixes $\mathbf{F}$ and $\mathbf{G}^T$. If $\mathbf{R}$ is a non-singular square matrix formed by last $n-k$ columns of $\mathbf{F}$ and k (all) columns of $\mathbf{G}^T$, then

$$\mathbf{R}^{-1}\left(\mathbf{F}|\mathbf{G}^T\right) = \left(\ \widetilde{\mathbf{F}} \mid \widetilde{\mathbf{G}}^T\ \right) = \left(\mathbf{B}\ \begin{matrix}\mathbf{E}_{n-k}\\ 0\end{matrix}\ \middle|\ \begin{matrix}0\\ \mathbf{E}_k\end{matrix}\right) \tag{4}$$

where $\mathbf{E}_l$ is an identity $l \times l$-matrix. Matrix $\mathbf{B}_{n\times(N+k-n)}$ is a generalized Cauchy matrix with elements $b_{ij} = \dfrac{\alpha_i \cdot \beta_j}{\mu_i - \nu_j}$ that can be obtained explicitly. If we represent the error vector $\vec{e}$ by a linear combination: $m_1 \cdot \vec{f_1} + m_2 \cdot \vec{f_2} + \ldots + m_N \cdot \vec{f_N}$, the Hamming weight of the vector $\vec{m}$ is less or equal to t_k. To derive the original code vector $\vec{g}$ we will multiply the resulting vector $\vec{c} = \vec{g} + \vec{e}$ by $\mathbf{R}^{-1}$:

$$\mathbf{R}^{-1} \cdot (\vec{g} + \vec{e}) = \mathbf{R}^{-1} \cdot (\vec{g} + \mathbf{F} \cdot \vec{m}) = \widetilde{\vec{g}} + \widetilde{\mathbf{F}} \cdot \vec{m} = \widetilde{\vec{g}} + \widetilde{\vec{e}}.$$

The first $n-k$ components of the vector $\widetilde{\vec{g}}$ will be zeroes:

$$\widetilde{\vec{g}} = \begin{pmatrix} 0\\ 0\\ \vdots\\ 0\\ \widetilde{g}_{n-k+1}\\ \widetilde{g}_{n-k+2}\\ \vdots\\ \widetilde{g}_n \end{pmatrix},$$

So, we will know the first $n-k$ components of the vector $\widetilde{\vec{e}}$. This knowledge allows us to reconstruct vector $\vec{m}$ which weight is

$$d_H(\vec{m}) = t \le t_k = \left[\frac{d_{\mathcal{F}}(C) - 1}{2}\right]$$

We need to solve the system of equations $\widetilde{\mathbf{F}} \cdot \vec{m} = \widetilde{\vec{e}}$:

$$\left(\mathbf{B}\ \begin{matrix}\mathbf{E}_{n-k}\\ 0\end{matrix}\right)\begin{pmatrix} m_1\\ m_2\\ \vdots\\ m_N\end{pmatrix} = \begin{pmatrix} \widetilde{e}_1\\ \widetilde{e}_2\\ \vdots\\ \widetilde{e}_{n-k}\\ *\\ \vdots\\ * \end{pmatrix} \tag{5}$$

Let us consider the first $n-k$ rows of the system (5): a matrix $\mathbf{H}$ containing the first $n-k$ rows of the matrix $\widetilde{\mathbf{F}}$ is a concatenation of a generalized Cauchy matrix and identity matrix. Therefore, it can be converted to a generalized Vandermonde matrix $\mathbf{H}'$. We will refer to the mapping from $\mathbf{H}$ to $\mathbf{H}'$ as $\mathbf{\Psi}$.

So, we have a system of linear equations:

$$\mathbf{H}' \cdot \mathbf{m} \ = \ \mathbf{\Psi} \cdot \begin{pmatrix} \widetilde{e}_1\\ \widetilde{e}_2\\ \vdots\\ \widetilde{e}_{n-k} \end{pmatrix}.$$

This is a problem of decoding a GRS code which has a solution when $d_H(\vec{m}) \le \left[\frac{n-k}{2}\right]$. This is true in our case.

3 Conclusion

In this paper codes were constructed for a projective $\mathcal{F}$-metrics associated with a generalized Vandermonde matrix. Fast decoding algorithm was presented. It seems quite interesting to find useful $\mathcal{F}$-metrics and fast decoding algorithms for other codes. It is also interesting to consider applications of codes in $\mathcal{F}$-metrics in other fields, e.g. cryptography.

References

[1] E.M. Gabidulin and J. Simonis, "Metrics Generated by Families of Subspaces", *IEEE Trans. Inform. Theory*, vol. 44, no. 5, May 1998, pp. 1336-1341.

[2] B.D.Sharma, M. Lal "On Algebra of Sharma and Kaushik's Metric Inducing Partitions of $\mathbb{Z}_q$," J. Combinatorics, Inform Sys. Sci. 11 (1986) 1 - 14.

[3] E.M. Gabidulin "Combinatorial Metrics in Coding Theory," Proceedings of the 2nd International Symposium on Information Theory, pp. 39-43, 1971, Moscow-Yerevan.

[4] E.M. Gabidulin, M. Bossert " Codes Resistant to the Phase Rotation," Proceedings of the 4-th Symposium on Communication and Applications, pp. 253-257, July 1997, pp. 65-84, Charlotte Mason College, Lake District, UK.

[5] E.M. Gabidulin, M. Bossert " Hard and Soft Decision Decoding of Phase Rotation Invariant Block Codes," Proceedings of the 1998 International Zurich Seminar on Broadband Communications - Accessing, Transmission, Networking - 17-19 February, 1998, ETH Zurich, Switzerland.

[6] Gabidulin E.M., Simonis J. "Perfect codes for metrics generated by primitive binary BCH codes correcting double errors", *Problems of information transmittion*, **35**, No.3, pp. 40-47, 1999.

[7] E.M Gabidulin, "Metrics Generated by Linear Codes in Cryptography", *Proceedings on Coding and Cryptography - WCC'99*, 11-14 Jan. 1999, Paris, France, p.9

Experimental constructions of codes over rings and construction of an optimal unimodular lattice in dimension 43

P. Gaborit and A. Otmani
LACO, Université de Limoges,
87060 Limoges, France
e-mail: gaborit@unilim.fr, otmani@unilim.fr

Abstract

In this paper we consider two methods to construct codes over rings. The first method generalizes to rings the constructions of [GO], the second method uses the quadratic double circulant codes of [G]. As an application we extend the tables of best known self-dual codes over the ring $\mathbb{Z}_4 = \mathbb{Z}/(4)$ to length 42 for the Lee and Euclidean distances, we construct a self-dual code over the ring $\mathbb{Z}/(25)$ which gives the first unimodular lattice of dimension 43 and norm 4, and we construct codes over $\mathbb{Z}_4$ of rate 1/2 which realize some of the best Lee and Hamming minimum distances for lengths 8 to 22.

Keywords : self-dual codes, lattices, codes over rings.

1 General construction

For general definitions and notation related to codes over rings and the different distances we refer to [RS]. In this paper we only consider the Euclidean scalar product.

Let R be a ring. Let C be a self-dual code over R of length n. Let G be a generator matrix of C. Now define $n \times n$ square matrices M_r satisfying for λ_r invertible in R: $M_r \cdot M_r^t = \lambda_r I_n$. For $\Pi_1, \ldots, \Pi_r$ permutations of the symmetric group S_n $(r \geq 1)$, we consider the codes C_r with generator matrices G_r: $G_r = GM_1\Pi_1 \cdots M_r\Pi_r$. In fact this action keeps the self-duality of the code as we show it in the next proposition.

Proposition 1 *[GO] If the code C is self-dual then so is C_r.*

Since we are interested in simple constructions of codes, we take all matrices M_i to be equal to a simple matrix M, constructed from smaller square $b_i \times b_i$ matrices B_i $(i = 1, \ldots, p)$ satisfying: $B_i \cdot B_i^T = \lambda I_{b_i}$, and from a β in R such that $\beta^2 = \lambda$. Then the block matrix M consists of k_i times the matrix B_i on the diagonal and k_β times β on the diagonal, with $k_\beta + \sum b_i k_i = n$.

Now, we identify the set $\{1, \ldots, n\}$ with the quotient ring $\mathbb{Z}_n$ and we take all the permutations equal to the same permutation π acting on $\mathbb{Z}_n$ as:

$$\forall z \in \mathbb{Z}_n, \quad \pi(z) = a \cdot (z + 1).$$

with a and n coprimes. We say that a code is obtained from construction $(a; r)$ when it is obtained after applying r times the matrice M and the permutation π to the matrix G.

2 Self-dual codes over $\mathbb{Z}_4$

We now consider the particular case of self-dual codes over the ring $\mathbb{Z}_4$. These codes have been classified up to length 15 and bounds for the Lee distance are given in [R], these bounds are tight up to length 24 [RS]. For higher lengths few is known except for lengths a multiple of 8, corresponding to Type II codes. In the following we extend tables of the best Lee and Euclidean distance for self-dual codes of lengths up to length 42.

We denote by O_8, K_4 and A_2 the following matrices corresponding to matrices of self-dual codes:

$$O_8 = \begin{pmatrix} 1 & 0 & 0 & 0 & 2 & 1 & 3 & 1 \\ 0 & 1 & 0 & 0 & 3 & 2 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 3 & 2 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 2 \end{pmatrix}, \; K_4 = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & 2 & 0 & 2 \\ 0 & 0 & 2 & 2 \end{pmatrix} \text{ and } A_2 = (2).$$

We take for G the generator matrix obtained by the direct sum of $n_3 A_2$, $n_2 K_4$ and $n_1 O_8$, with $n_1 = n$ div $8, n_2 = (n \bmod 8)$ div 4 and $n_3 = n \bmod 4$.

For the square $n \times n$ block matrix M, we consider the two matrices B_4 and B_6 defined respectively by:

$$B_4 = \begin{pmatrix} 2 & 1 & 3 & 1 \\ 3 & 2 & 1 & 1 \\ 1 & 3 & 1 & 1 \\ 1 & 1 & 1 & 2 \end{pmatrix}, \; B_6 = \begin{pmatrix} 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 \end{pmatrix}.$$

If n is a multiple of 4, then M is obtained by a sequence of matrices B_4 on the diagonal and else M is built by assuming that each B_i equals to the same matrix B_6 and that $\beta = 1$ with $k_\beta = n \bmod 6$ on the rest of the diagonal.

We give in Table 1 the best Lee distance (d_L) known for self-dual codes of lengths 25 to 42 and we also give their Euclidean distance (d_E). A reference $(a; r)$ in the table precises the construction used to find the code.

n	d_L	d_E	reference
25	8	8	(1;5)
26	8	8	(1;5)
27	8	8	(1;5)
28	10	12	(15;11)
29	10	12	(2;180)
30	10	12	(11;5)
31	12	12	[RS]
32	14	16	[RS]
33	10	12	(1;10)

n	d_L	d_E	reference
34	10	12	(1;10)
35	10	12	(1;10)
36	12	12	(19;44)
37	12	12	(2;1059)
38	12	12	(3;22)
39	14	16	[GH]
40	14	16	(21;223),[RS]
41	12	12	(1;23)
42	12	16	(1;58)

Table 1: Best known Lee and Euclidean distances for self-dual codes over $\mathbb{Z}_4$

3 Self-dual codes over $\mathbb{Z}_{25}$

Self-dual codes over the ring $\mathbb{Z}_{25}$ have not really been studied, in that case no efficient Gray map is known but it is still possible to apply Construction A of [CS] to build unimodular lattices. One obtains for $x = (x_1, \cdots, x_n)$:

$$A_{25}(C) := \frac{1}{5}\{x \in \mathbb{Z}^n | (x_1 \text{mod } 25, \cdots, x_n \text{mod } 25) \in C\},$$

and the minimum norm of $A_{25}(C)$ is $min(5, \frac{d_E(C)}{25})$.

We take for generator matrix G of length n the direct sum of $[2,1,2]$ codes with generator matrix $(12, 9)$ to which we add the code generated by (5) at the beginning of the diagonal if n is odd. We let:

$$B_3 = \begin{pmatrix} 8 & 9 & 9 \\ 9 & 8 & 9 \\ 9 & 9 & 8 \end{pmatrix}.$$

Then applying different permutations as in the previous section we obtained a code with minimum Euclidean weight 100 with generator matrix $(A_1 A_2)$, where A_1 is a 22×21 matrix composed from the identity matrix of size 21 and a last line of zeros and :

$$A_2 = \begin{pmatrix}
2 & 24 & 1 & 7 & 15 & 2 & 10 & 12 & 4 & 2 & 21 & 21 & 8 & 23 & 15 & 6 & 13 & 22 & 11 & 19 & 22 & 11 \\
0 & 16 & 0 & 18 & 21 & 2 & 8 & 19 & 23 & 4 & 4 & 5 & 21 & 3 & 7 & 7 & 8 & 2 & 18 & 8 & 0 & 22 \\
3 & 7 & 18 & 21 & 21 & 16 & 20 & 22 & 14 & 1 & 15 & 13 & 17 & 8 & 7 & 17 & 9 & 14 & 11 & 5 & 4 & 18 \\
2 & 24 & 16 & 6 & 20 & 7 & 19 & 11 & 22 & 19 & 10 & 7 & 22 & 20 & 18 & 21 & 23 & 0 & 18 & 7 & 10 & 1 \\
1 & 4 & 20 & 17 & 4 & 20 & 16 & 11 & 19 & 7 & 11 & 3 & 2 & 7 & 14 & 12 & 4 & 4 & 21 & 13 & 18 & 24 \\
1 & 21 & 12 & 13 & 17 & 2 & 17 & 12 & 23 & 21 & 12 & 0 & 24 & 2 & 10 & 12 & 14 & 1 & 18 & 16 & 22 & 12 \\
1 & 15 & 21 & 9 & 20 & 11 & 3 & 15 & 13 & 13 & 23 & 2 & 23 & 5 & 0 & 7 & 24 & 23 & 8 & 1 & 21 & 6 \\
4 & 6 & 4 & 22 & 24 & 16 & 8 & 0 & 21 & 4 & 18 & 11 & 19 & 7 & 7 & 16 & 15 & 9 & 7 & 12 & 1 & 3 \\
1 & 20 & 23 & 0 & 10 & 9 & 8 & 12 & 4 & 7 & 8 & 7 & 16 & 19 & 18 & 8 & 1 & 24 & 16 & 17 & 8 & 19 \\
4 & 16 & 18 & 17 & 17 & 5 & 21 & 0 & 14 & 9 & 22 & 14 & 12 & 19 & 12 & 18 & 4 & 0 & 11 & 0 & 24 & 4 \\
0 & 12 & 14 & 6 & 11 & 14 & 20 & 20 & 24 & 20 & 6 & 9 & 8 & 24 & 15 & 0 & 3 & 21 & 8 & 8 & 21 & 23 \\
4 & 2 & 2 & 20 & 3 & 1 & 0 & 22 & 16 & 9 & 23 & 2 & 19 & 7 & 0 & 19 & 10 & 8 & 1 & 0 & 12 & 16 \\
4 & 14 & 6 & 15 & 7 & 13 & 21 & 16 & 16 & 2 & 15 & 4 & 20 & 15 & 2 & 5 & 1 & 19 & 18 & 20 & 24 & 13 \\
1 & 19 & 5 & 22 & 23 & 5 & 15 & 16 & 21 & 0 & 12 & 23 & 23 & 10 & 0 & 10 & 24 & 11 & 17 & 20 & 5 & 17 \\
3 & 23 & 3 & 9 & 21 & 15 & 17 & 7 & 22 & 0 & 14 & 24 & 4 & 11 & 4 & 15 & 21 & 15 & 18 & 21 & 16 & 14 \\
1 & 12 & 11 & 19 & 12 & 19 & 11 & 18 & 20 & 9 & 0 & 14 & 23 & 18 & 3 & 11 & 3 & 8 & 8 & 3 & 15 & 21 \\
2 & 5 & 8 & 20 & 14 & 8 & 14 & 10 & 2 & 19 & 2 & 0 & 21 & 7 & 0 & 20 & 1 & 15 & 4 & 7 & 5 & 0 \\
3 & 24 & 12 & 6 & 13 & 21 & 12 & 6 & 4 & 19 & 22 & 1 & 6 & 0 & 2 & 2 & 20 & 4 & 3 & 7 & 17 & 20 \\
2 & 6 & 23 & 20 & 17 & 23 & 11 & 15 & 2 & 20 & 3 & 13 & 24 & 6 & 14 & 14 & 16 & 13 & 4 & 4 & 2 & 12 \\
0 & 17 & 9 & 17 & 14 & 17 & 5 & 20 & 15 & 7 & 14 & 23 & 2 & 15 & 9 & 6 & 16 & 23 & 11 & 4 & 2 & 22 \\
0 & 0 & 14 & 20 & 15 & 0 & 17 & 14 & 6 & 1 & 3 & 22 & 1 & 7 & 0 & 17 & 11 & 6 & 0 & 21 & 1 & 15 \\
5 & 10 & 15 & 0 & 15 & 15 & 0 & 15 & 10 & 10 & 10 & 5 & 15 & 15 & 15 & 20 & 20 & 5 & 20 & 20 & 15 & 15
\end{pmatrix}.$$

By construction A this code leads to the first extremal unimodular lattice of dimension 43. The only remaining open cases for potential optimal unimodular lattices of norm 4 are dimensions 37 and 41 [NS].

4 Construction of optimal $\mathbb{Z}_4$ codes of rate 1/2

In [G], quadratic double circulant (QDC) codes were introduced over fields. Although these codes can be generalized over any ring we only consider in this section the case of the ring $\mathbb{Z}_4$. Over $\mathbb{Z}_4$, self-dual QDC codes for lengths a multiple of $2p+2$ with $p = 8k+3$ a prime were considered in [CalS] but it is also possible to consider these codes for other primes, in that case one loses self-duality but one still obtains codes with good parameters, often better or equal than those of self-dual codes for lengths inferior to 23.

For q a power of an odd prime and a, b and c in $\mathbb{Z}_4$ we consider the $q \times q$ quadratic matrix $Q_q(a,b,c)$ and the $(q+1)\times(q+1)$ matrix $S_q(a,b,c)$ (with $\alpha = a, \beta = \gamma = 1$) defined in [G]. We recall that the QDC codes are the codes with generator matrices $P_q(a,b,c) = (I|Q_q(a,b,c))$ and $B_q = (I|S_q(a,b,c))$.

We give in Table 2 the best known Lee distances (d_L), and Hamming distances (d_H) for lengths 8 to 24 of $\mathbb{Z}_4$ codes of rate 1/2. The notation $S(C)$ corresponds to the code obtained after shortening the code C in the first column and adding the all 2 word to the shortened code. The code C_{17} is constructed by taking the extension of the quadratic residue code over $\mathbb{Z}_4$ of length 17 (XQ_{17}) in a cyclic form. The code can then be written in the form (IA) and C_{17} is obtained by multiplying the first row by 2 and then deleting the first column. For length 22 two different codes are given.

n	d_L	d_H	Code	n	d_L	d_H	Code
8	6	4	[RS]	17	8	5	C_{17}
9	4	3	$P_5(3,2,1)$	18	8	6	XQ_{17}
10	5	4	$P_5(0,3,1)$	19	6	3	[RS]
11	6	3	$S(B_5(0,3,1))$	20	8	4	[RS]
12	6	4	$B_5(0,3,1)$	21	8	5	[RS]
13	6	3	$S(P_7(2,3,2))$	22	10	6	$P_{11}(0,1,2)$
14	8	4	$P_7(2,1,3)$		9	7	$P_{11}(3,2,1)$
15	6	4	[RS]	23	10	7	[RS]
16	8	4	[RS]	24	12	8	[RS]

Table 2: Best known codes of rate 1/2 over $\mathbb{Z}_4$ of length 8 to 24

The support of words of fixed Hamming weight of these codes contain in general 1-designs. As in the $GF(2)$ case the supports of words of XQ_{17} with a given Hamming weight hold 2-designs and 3-designs if one considers the reunion with the words of the dual. Also the code of length 10 holds a 3-(10, 5, 3) design in its words of Hamming weight 5. Although there is a unique linear binary [28, 14, 8] code, the binary image of the code of length 14 gives a new non-linear $(28, 2^{14}, 8)$ code. It is also worth noticing that the Lee weight distribution of the code of length 11 corresponds to the weight distribution of the unique [22, 11, 6] self-dual binary code.

REFERENCES

[CalS] A. R. Calderbank and N.J.A. Sloane, "Double circulant codes over $\mathbb{Z}_4$ and unimodular lattices", *J. Alg. Comb.*, **6**, p. 119-131,(1997).

[CS] J.H. Conway and N.J.A. Sloane, *Sphere Packings, Lattices and Groups*, Springer (1993).

[G] P. Gaborit, "Quadratic Double Circulant Codes over Fields" *J. Combin. Theory Ser. A*, **97**, p. 85-107, (2002).

[GH] A. Gulliver and M. Harada, "An optimal unimodular lattice in dimension 39", *J. Combin. Theory Ser. A*, **88**, p. 158-161, (1999).

[GO] P. Gaborit and A. Otmani, "Experimental constructions of self-dual codes", *submitted to Finite Fields and Th. Appl.*

[NS] G. Nebe and N.J.A. Sloane, "The Catalogue of Lattices", *http://akpublic.research.att.com/ njas/lattices*

[R] E. M. Rains, "Bounds for self-dual codes over $\mathbb{Z}_4$", *Finite Fields and Th. Appl.*, **6**, p. 146-163, (2000).

[RS] E. M. Rains and N. J. A. Sloane, "Self-dual codes", in *Handbook of Coding Theory*, ed. V. S. Pless and W. C. Huffman. Amsterdam: Elsevier, 1998, pp. 177–294.

Applications of minihypers in finite geometry

P. Govaerts

(joint work with **L. Storme** and **H. Van Maldeghem**)

Ghent University, Dept. of Pure Maths and Computer Algebra
Krijgslaan 281, 9000 Gent, Belgium
pg@cage.rug.ac.be, http://cage.rug.ac.be/~pg

Abstract

Applications of minihypers in finite geometry are discussed. In particular, it is explained how minihypers can be used in the study of (partial) spreads and covers of finite projective and polar spaces.

1 A particular class of minihypers

An $\{f, m; n, q\}$*-minihyper* is a pair (F, w), where F is a subset of the point set of $\mathrm{PG}(n, q)$ and w is a weight function $w : \mathrm{PG}(n,q) \to \mathbb{N} : x \mapsto w(x)$, satisfying: (1) $w(x) > 0 \Leftrightarrow x \in F$, (2) $\sum_{x\in F} w(x) = f$, and (3) $\min\{\sum_{x\in H} w(x) | H \in \mathcal{H}\} = m$, where $\mathcal{H}$ is the set of hyperplanes of $\mathrm{PG}(n, q)$. In the case that w is a mapping onto $\{0, 1\}$, the minihyper (F, w) can be identified with the set F and is simply denoted by F.

Minihypers were introduced in [5] and are usually studied because of their relation to linear codes. But minihypers are also important in finite geometry, especially those with parameters $\{\delta v_{t+1}, \delta v_t; n, q\}$, where $v_i = (q^i - 1)/(q - 1)$. For small values of δ, these have been classified.

Theorem 1 ([3, 2, 1]) 1. *Let $q = p^h$, p prime, and let ϵ equal $\sqrt{q} + 1$ when q is a square, $(q + 3)/2$ when q is a prime, and $c_p q^{2/3} + 1$ otherwise. Here c_p equals $1/\sqrt[3]{2}$ if $p \in \{2, 3\}$ and 1 otherwise. If (F, w) is a $\{\delta v_{t+1}, \delta v_t; n, q\}$-minihyper, $q > 2$, satisfying $0 \le \delta < \epsilon$ and $t \le n - 1$, then w is the weight function induced on the points of $\mathrm{PG}(n, q)$ by a sum of δ t-spaces.*

2. *Let $q > 16$ be a square. A $\{\delta v_{t+1}, \delta v_t; n, q\}$-minihyper F, $\delta < q^{5/8}/\sqrt{2} + 1$, is a unique disjoint union of t-spaces and subgeometries $\mathrm{PG}(2t + 1, \sqrt{q})$.*

3. *A $\{\delta v_{t+1}, \delta v_t; n, q\}$-minihyper F, $q = p^{3h}$, $h \ge 1$, $p \ge 7$ prime, $\delta \le 2p^{2h} - 6p^h$, is the disjoint union of $\mathrm{PG}(t, q)$'s, (projected) $\mathrm{PG}(3t + 2, \sqrt[3]{q})$'s, and $\mathrm{PG}(2t + 1, \sqrt{q})$'s.*

Remark that in part 1 of this theorem weights are allowed, while this is not the case for parts 2 and 3.

2 Partial spreads and covers

A *t-spread* (a *partial t-spread*, a *t-cover*) of $\mathrm{PG}(n, q)$ is a set of t-spaces such that every point of $\mathrm{PG}(n, q)$ is contained in exactly (at most, at least) one of these t-spaces. A partial t-spread (a t-cover) is called *maximal* (*minimal*) if no t-space can be added (deleted) to obtain a larger (smaller) partial t-spread (t-cover). The *deficiency* (*excess*) is the number of t-spaces it lacks (has too many) to be a t-spread. A *hole* (*multiple point*) is a point of $\mathrm{PG}(n, q)$ that is not contained in an (contained in more than one) element of the partial t-spread (t-cover). The *surplus* of a point of $\mathrm{PG}(n, q)$ with respect to a t-cover $\mathcal{C}$ is the number of elements of $\mathcal{C}$ that pass through this point minus one.

It is known that $\mathrm{PG}(n, q)$ has a t-spread iff $(t + 1)|(n + 1)$.

Theorem 2 ([3]) 1. *Let $\mathcal{S}$ be a partial t-spread of $\mathrm{PG}(n, q)$, $(t+1)|(n+1)$, with deficiency $\delta < q$, and let F be the set of holes of $\mathcal{S}$. Then F is a $\{\delta v_{t+1}, \delta v_t; n, q\}$-minihyper.*

2. *Let $\mathcal{C}$ be a t-cover of $\mathrm{PG}(n, q)$, $(t + 1)|(n + 1)$, with excess $\delta < q$. Let F be the set of multiple points of $\mathcal{C}$ and let $w(p) = surplus(p)$ for $p \in \mathrm{PG}(n, q)$. Then (F, w) is a $\{\delta v_{t+1}, \delta v_t; n, q\}$-minihyper.*

The finite classical polar spaces are: (1) $\mathrm{W}_{2n+1}(q)$, which arises from a symplectic polarity of $\mathrm{PG}(2n + 1, q)$, $n \ge 1$; (2) $\mathrm{Q}^-(2n + 1, q)$, which arises from a nonsingular elliptic quadric of $\mathrm{PG}(2n + 1, q)$, $n \ge 2$; (3) $\mathrm{Q}(2n, q)$, which arises from a nonsingular quadric of $\mathrm{PG}(2n, q)$, $n \ge 2$; (4) $\mathrm{Q}^+(2n+1, q)$, which arises from a nonsingular hyperbolic quadric of $\mathrm{PG}(2n+1, q)$, $n \ge 1$; (5) $\mathrm{H}(n, q^2)$, which arises from a nonsingular Hermitian variety in $\mathrm{PG}(n, q^2)$, $n \ge 3$.

Let $\mathcal{P}$ be a finite classical polar space. The definitions of (partial) t-spreads and t-covers of finite classical polar spaces are very similar to these of (partial) t-spreads and t-covers of finite projective spaces: it suffices to replace "t-space" by "totally isotropic or singular t-space".

For $\mathcal{P}$ to have a t-spread, it is necessary that $|\mathrm{PG}(t, q)|$ divides $|\mathcal{P}|$. If this condition is satisfied, $\mathcal{P}$'s size is said to admit a t-spread.

Theorem 3 ([4]) *Let $\mathcal{P}$ be a classical polar space in $\mathrm{PG}(n, q)$ whose size admits a t-spread.*

1. *If $\mathcal{S}$ is a partial t-spread of $\mathcal{P}$ with deficiency $\delta < q$, then the set F of holes forms a $\{\delta v_{t+1}, \delta v_t; n, q\}$-minihyper.*

2. *If $\mathcal{C}$ is a t-cover of $\mathcal{P}$ with excess $\delta < q$, then the weight function $w(p) = surplus(p)$ for $p \in \mathcal{P}$ defines a $\{\delta v_{t+1}, \delta v_t; n, q\}$-minihyper (F, w), where F is the set of points of $\mathcal{P}$ that are covered at least twice by elements of $\mathcal{C}$.*

3 On the structure of partial spreads and covers

Theorems 2 and 3 provide enough information to apply the results on minihypers from Theorem 1 to obtain results on the existence of maximal partial t-spreads.

Theorem 4 *Let $\mathcal{P}$ be either a space* $\mathrm{PG}(n,q)$, $q \neq 2$, *or a classical polar space in* $\mathrm{PG}(n,q)$, $q \neq 2$. *Suppose that $\mathcal{P}$'s size admits a t-spread. If $\mathcal{P} = \mathrm{W}_n(q)$, then suppose that q is even. Suppose that $\mathcal{S}$ is a maximal partial t-spread of $\mathcal{P}$ with deficiency δ and that the conditions for q, n, t, and δ of Theorem 1 are met. Then, either*

1. *the set of holes forms a disjoint union of subgeometries* $\mathrm{PG}(2t+1, \sqrt{q})$, *implying* $\delta \equiv 0 \pmod{\sqrt{q}+1}$; *or*
2. *the set of holes forms a disjoint union of* $\mathrm{PG}(2t+1, \sqrt{q})$*'s and (projected)* $\mathrm{PG}(3t+2, \sqrt[3]{q})$*'s.*

Similarly, restrictions on the existence of t-covers can be obtained. However, in this case only part 1 of Theorem 1 can be applied, since weights are allowed in the minihypers.

Remark 5 *For a more detailed discussion of these applications and for further applications (partial ovoids and blocking sets in finite classical polar spaces and in the split Cayley hexagon), we refer to* [4].

References

[1] S. Ferret and L. Storme. The classification of $\{\delta v_{\mu+1}, \delta v_{\mu}; t, p^{3h}\}$-minihypers. In preparation.

[2] P. Govaerts and L. Storme. On a particular class of minihypers and its applications. II. Improvements for q square. *J. Combin. Theory Ser. A*, 97(2):369–393, 2002.

[3] P. Govaerts and L. Storme. On a particular class of minihypers and its applications. I. The result for general q. *Des. Codes Cryptogr.*, accepted.

On An Algebraic Generalization of Coding Theory and Decoding Methods based on Group Cohomology Theory

Jun IMAI

NTT Communication Science Laboratories *

Abstract

We propose an abstract model of error-correcting codes over fields, and also generalize several notions concerning codes. Then we demonstrate the solution of a problem, which shows how to correct errors on our abstract model. Our model is obtained by abstracting the quotient structure from linear codes and has the form of ashort exact sequence of general groups, modules, and similar algebraic objects.

1 Introduction

The main purpose of this paper is to generalize or extend the coding theory to the level of a more abstract algebraic structure (group extensions). More specifically, it aims

1. to extend the notion of errors to the abstract group extension structure,
2. to introduce the notion of distance which measures the size of errors,
3. to establish an error correcting method (decoding algorithm) for our abstract model (group extensions) using homological algebras and the technique of spectral sequences,
4. to reveal and abstract the essence of coding theory and clarify what kind of structure is important and essential in this theory, and
5. to prove that group extensions are a fundamental model of coding theory using our error correction method.

Note that prior this paper, none of the above items have previously been proposed or achieved yet.

First, we must draw the reader's attention to the following points:

1. Our model for coding theory (group extensions) does not have any basis general.
2. Our result for the decoding algorithm of group extensions are a natural generalization of the existing decoding algorithms which can correct errors of linear codes that have a global basis.
3. None of the existing decoding algorithms ([4], [5], [9]), except for the algorithme proposed in this paper, can correct errors of our models (group extensions).

Next, we state the motivation of our research. In a simple sense, a code C is a subset of $\mathbb{F}_q^n$, where $\mathbb{F}_q$ is a finite field of q elements or some kind of finite set of alphabet. For various reasons one often restricts consideration to linear codes or linear subspaces of $\mathbb{F}_q^n$ over $\mathbb{F}_q$. However,there are some types of nonlinear codes, such as the Nordstrom-Robinson, Kerdoc, and Preparata codes, which are in some ways superior to linear codes. Therefore, the notion of linear subspace is not essential for the coding theory. On the other hand, these nonlinear codes have been proved to be the image of some mapping (Gray map) of submodules of R^n, where R is a ring ([6]). Thus, in this paper, we plan to prove that the essense of error correcting codes is the subquotient structure of some more abstract algebraic objects. We selected group extensions as the abstract algebraic objects with a subquotient structure because they need not have any kind of general basis, and thus they can be considered here as the most general objects.

The significance of our contribution is comprised of two parts. One part relates to practical application; that is, various types of subquotient algebraic structure can be considered as information transmission methods with error correcting procedures. The other part relates to the mathematical solution of the problem; that is, what the error correcting method becomes in the more abstract algebraic framework without basis and matrix calculation. In this problem, we are allowed only to use the restricted data of Cokernel (or syndrome) and derive the corrected data. In this paper the derivation of errors under this restricted condition is proved by using Lyndon-Hochschild-Serre type spectral sequences .

Now let us state a problem in order to generalize a decoding problem of error correcting codes into a more abstract level where we cannot utilize the power of ordinary linear algebra. This abstraction reflects what is essential in decoding methods as well as enlarging the notion of error correction, and this enables us to apply it to more general abstract algebraic objects.

Definition 1.1. *Let G and A be groups (assume A is abelian). An extension of A by G (or G by A) is an exact sequence*

$$0 \longrightarrow A \xrightarrow{\iota} E \xrightarrow{\pi} G \longrightarrow 1.$$

*2-4, Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0237, Japan. *E-mail address:* imai@cslab.kecl.ntt.co.jp

Here A is a G-module with the G-action defining the conjugation action in E. That is, for $a \in A$ and $s \in E$ with $\pi(s) = g$, $si(a)s^{-1} = i(ga)$. Here G can be any group.

Problem 1.1. *(Group extension version) Let $0 \longrightarrow A \xrightarrow{i} E \xrightarrow{\pi} G \longrightarrow 1$ be an extension of A by G. Assume that some information can be represented as each element of group A. Then we have an open problem of how to detect or correct errors which appear in group extensions by utilizing their redundant structure.*

In this research we discuss this problem and try to establish a decoding method for a group extension of G by A by utilizing its redundant structure and try to abstract the substance of error correction methods.

2 Summary of cohomology theory of groups

Let E be a possibly non-abelian group, A an abelian normal subgroup, and $G := E/A$. We write E and its subgroup A additively , but we write the quotient group G multiplicatively. Therefore 0 refers to a unit element of E and A, but 1 is used as a unit element of G.

Definition 2.1. *1. An extension of A by G, $0 \longrightarrow A \longrightarrow E \longrightarrow G \longrightarrow 1$, determines a homomorphism $\theta : G \longrightarrow \mathrm{Aut}(A)$, where $\mathrm{Aut}(A)$ means the group of all automorphisms of A.*

2. An extension $0 \longrightarrow A \longrightarrow E \longrightarrow G \longrightarrow 1$ is called a central extension if $A \subset Z(E)$, where $Z(E)$ denotes the center of E.

3. A stem extension of G is a central extension satisfying $A \subset [E, E]$, where $[E, E]$ means the commutator subgroup of E.

Proposition 2.1. *If $0 \longrightarrow A \longrightarrow E \longrightarrow G \longrightarrow 1$ is an extension, then A is a left $\mathbb{Z}G$-module. Specifically, the action of $x \in G$ on $a \in A$ is defined by*

$$xa := \theta_x(a) = \lambda x + a - \lambda x \quad \in A,$$

where $\lambda x \in E$ is a lifting of x (i.e., $\lambda x \longmapsto x$). Then the action of an arbitrary element of $\mathbb{Z}G$ is defined by $(\sum m_x x)a := \sum m_x(xa)$.

Conversely, every extension arises from some data (G, A, θ); however, as we will see later, the correspondence between extensions and G-module structures is not one to one. The theory of group cohomology makes this relation clearer.

Definition 2.2. *Let $0 \longrightarrow A \longrightarrow E \longrightarrow G \longrightarrow 1$ be an extension. If $\lambda : G \longrightarrow E$ is a lifting (need not be a homomorphism) , then $\mathrm{Im}\lambda$ is a complete set of coset representatives of A in E , which is called a transversal of A in E, so that $A + \mathrm{Im}\lambda = E$. If $a \in A \cap \mathrm{Im}\lambda$, then $a = \lambda 1 = 0$. Therefore every element of E can be uniquely given by $a + \lambda x$ for some $x \in G$. Since the coset of $\lambda(xy)$ and that of $\lambda x + \lambda y$ are the same, we have some element $[x, y] \in A$ such that*

$$\lambda x + \lambda y = \lambda(xy) + [x, y] \quad \ldots (I)$$

and

$$xa = \lambda x + a - \lambda x \quad \ldots (II).$$

This defines the new function $[*, *] : G \times G \longrightarrow A$, which is called a factor set.

Definition 2.3. *$Z^2(G, A)$ is the abelian group of all factor sets under pointwise addition . Note the zero of $Z^2(G, A)$ is the factor set which is identically zero and corresponds to the semidirect product. The definition of factor sets depends on a choice of lifting λ, and representatives of factor sets can classify the class of extensions through liftings.*

Proposition 2.2. *Let $0 \longrightarrow A \longrightarrow E \longrightarrow G \longrightarrow 1$ be an extension, and let λ and μ be liftings. If $[*, *]$ and $(*, *)$ are the corresponding factor sets, then there is a function $\langle * \rangle : G \longrightarrow A$ satisfying*

1. $\langle 1 \rangle = 0$

2. $(x, y) - [x, y] = x\langle y \rangle - \langle xy \rangle + \langle x \rangle, \forall x, y \in G$.

Definition 2.4. *We define a subgroup $B^2(G, A)$ of $Z^2(G, A)$ as follows:*

$$B^2(G, A) := \{f : G \times G \longrightarrow A; \exists\langle\ \rangle : G \longrightarrow A, \text{ with } \langle 1 \rangle = 1, \\ satisfying\ f(x, y) = x\langle y \rangle - \langle xy \rangle + \langle x \rangle\}.$$

We call each element of $B^2(G, A)$ a normalized coboundary.

Definition 2.5. $e(G, A) := Z^2(G, A)/B^2(G, A)$.
That is, e(G,A)= (normalized factor sets)/(normalized coboundaries). Two factor sets stemming from an extension through two choices of liftings determine the same element of the above $e(G, A)$.

Definition 2.6. *Two extensions,*

$$0 \longrightarrow A \longrightarrow E \longrightarrow G \longrightarrow 1 and\ 0 \longrightarrow A \longrightarrow E' \longrightarrow G \longrightarrow 1$$

, are said to be equivalent if there exist factor sets $[*, *]$ and $(*, *)$ each of which corresponds to the above extensions, respectively, satisfying $[*, *] - (*, *) \in B^2(G, A)$.

Definition 2.7. *Let G be a group, A a left G-module, and suppose that $\mathbb{Z}$ (i.e. the ring of integers) is considered as a trivial G-module. Then we define*

$$H^n(G, A) := \mathrm{Ext}^n_{\mathbb{Z}G}(\mathbb{Z}, A), \quad H_n(G, A) := \mathrm{Tor}^{\mathbb{Z}G}_n(\mathbb{Z}, A).$$

Groups H^n are called cohomology groups of G with coefficient A and groups H_n are called homology groups of G.

We can establish an isomorphism $H^2(G, A) \cong e(G, A)$ by constructing some good normalized G-free resolutions of $\mathbb{Z}$.

3 Spectral Sequences

Spectral sequences are certain types of sequences of homology or cohomology modules. One can divide and weaken the complexity and difficulty of computing some kinds of homology or cohomology modules. Suppose one has a short exact sequence of groups

$$0 \longrightarrow N \longrightarrow G \longrightarrow Q \longrightarrow 1$$

and suppose A is a G-module. Then we have a spectral sequence which relates the cohomology of group G with the cohomology of N and Q. This spectral sequence, which is called the Lyndon-Hochschild-Serre spectral sequence, is used for correcting the errors which occur in extensions of groups.

Theorem 3.1. *Let G be a group with normal subgroup N. For each G-module A, there is a spectral sequence satisfying*

$$E_2^{p,q} = H^p(G/N, H^q(N, A)) \Longrightarrow H^n(G, A), \tag{3.1}$$

and there is a spectral sequence satisfying

$$E^2_{p,q} = H_p(G/N, H_q(N, A)) \Longrightarrow H_n(G, A). \tag{3.2}$$

4 Main Theorem

4.1 Statement of the main problem and our contribution

Suppose G is a group with a normal subgroup N. Let $N \xrightarrow{i} G \xrightarrow{\pi} G/N$ be an extension of the group and A be a G-module. Considering the extension which takes the above form to be sufficient to solve the problem proposed in the introduction, we restate the main problem that we should consider below.

Problem 4.1. *Let $N \xrightarrow{i} G \xrightarrow{\pi} G/N$ be an extension. Assume $i(N)$ to be a kind of encoding in group G. Suppose $u \in i(N)$ is mapped to the element $u + e \in G$ which has an error e. Then establish the method which enables us to obtain the information about where the error e should exist in G provided that we are allowed to use only the data about syndrome $\pi(e)$.*

Before we state our contribution to the above problem, we provide some definitions used in this section.

Let

$$N = G_0 \leq G_1 \leq G_2 \leq \cdots \leq G_t \leq \cdots \leq G_n = G \tag{4.1}$$

be the composition series of group G which starts from N and let

$$G_e := [N, \pi(e)] \tag{4.2}$$

be the normal subgroup generated by N and $\pi(e)$. Then we can reformulate the above problem 4.1 as follows.

Problem 4.2. *We use the above notations. Let $N \xrightarrow{i} G \xrightarrow{\pi} G/N$ be an extension. Assume $i(N)$ to be a kind of encoding in group G. Suppose $u \in i(N)$ is mapped to the element $u + e \in G$ which has an error e. Then find i in $1 \leqq i \leqq n$ such that*

$$G_e \not\subset G_{i-1}, G_e = G_i \text{ and } G_e \subset G_{i+1}, \tag{4.3}$$

using the information arising from the data in G/N.

Definition 4.1. *In the above situation, let i be the number satisfying (4.3). Then we define the distance which measures errors by the number i.*

Now we state our contribution to the above problem.

Theorem 4.1. *[Main Theorem] In the above problem 4.2, suppose we are given the data $\{H^*(N, A)\}$ for some coefficient G-module A. Then we can deduce the number i such that $G_e = G_i$ using the information arising from the data in G/N, or there exists a computation sequence (may be infinite) which provides the information about the number i.*

5 Examples of error correction

5.1 Free abelian groups of finite rank

Let G be a free abelian group of finite rank k, and N be a subgroup of rank l, $(l < k)$. Now consider the extension,

$$0 \longrightarrow N \longrightarrow G \longrightarrow G/N \longrightarrow 0.$$

Suppose $u \in N$, and we encode u and then obtain $i(u) \in G$. Now $i(u)$ is supposed to be mapped to $i(u) + e$, where e denotes an error. Let $G_e := [N, \pi(e)]$. In order to estimate e, we have only to compute the $x := rank(G_e)$, that is, the rank of G_e. We are allowed to use the following data:

$$H^n(N, \mathbb{Z}) = \begin{cases} \text{a free abelian group of rank}\binom{l}{n}, & \text{if } l > n \geq 0, \\ 0, & \text{otherwise,} \end{cases} \tag{5.1}$$

where $\binom{*}{*}$ denotes the binomial coefficient symbol. In order to measure G_e , we must compute the cohomology of $G_1, G_2, \ldots, G_n$ successively using the spectral sequence.

$$E_2^{p,q} = H^p(G_{i+1}/G_i, H^q(G_i, \mathbb{Z})) \Longrightarrow H^{p+q}(G_{i+1}, \mathbb{Z}).$$

Since $G_{i+1}/G_i \cong \mathbb{Z}$, we know $E_2^{p,q} = 0$ for $p \neq 0, 1$. Therefore only two columns exist in the $E_2^{p,q}$, so there are exact sequences

$$0 \longrightarrow E_2^{1,n-1} \longrightarrow H^n(G_{i+1}, \mathbb{Z}) \longrightarrow E_2^{0,n} \longrightarrow 0. \tag{5.2}$$

Since $H^{n-1}(G_i, \mathbb{Z})$ is $(G_{i+1}/G_i \cong \mathbb{Z})$-trivial, we have

$$\begin{aligned} E_2^{1,n-1} &= H^1(G_{i+1}/G_i, H^{n-1}(G_i, \mathbb{Z}))) \\ &\cong \mathrm{Hom}(\mathbb{Z}, H^{n-1}(G_i, \mathbb{Z})) \cong H^{n-1}(G_i, \mathbb{Z}). \end{aligned}$$

By induction, this group is a free abelian group of rank$\binom{l+i}{n-1}$. As for another term $E_2^{0,n}$, we have

$$E_2^{0,n} = H^0(G_{i+1}/G_i, H^n(G_i, \mathbb{Z})) \cong H^n(G_i, \mathbb{Z}),$$

also by $\mathbb{Z}$-triviality of $H^*(G_i, \mathbb{Z})$. By induction, this group is a free abelian group of finite rank $\binom{l+i}{n}$. It follows that the exact sequence (5.2) of abelian groups splits, and then $H^n(G_{i+1}\mathbb{Z}) \cong H^n(G_i, \mathbb{Z}) \oplus H^{n-1}(G_i, \mathbb{Z})$. Hence $H^n(G_{i+1}, \mathbb{Z})$ is a free abelian group of rank$\binom{l+i+1}{n}$.
If we are given the information of the syndrome $\{H^n(G_e, \mathbb{Z})\}$,which can be computed using $E_2^{p,q} = H^p(G_e/N, H^q(N, \mathbb{Z}))$, then we can deduce the number i such that $G_i = G_e$ by comparing $\{H^*(G_i, \mathbb{Z})\}$ with $\{H^n(G_e, \mathbb{Z})\}$.

5.2 Elementary abelian groups of order p^n

Let p be a prime number and let G be an elementary abelian group of order p^n. It is known that $p \times H_2(G, \mathbb{Z}) = 0$. In almost the same way as in the previous example, we can compute $\{H_2(G_i, \mathbb{Z})\}$ inductively, and then we can show that

$$H_2(G_i, \mathbb{Z}) \cong \text{elementary abelian of order } p^{\frac{n(n-1)}{2}},$$

where $n = \log_p$(the order of G_i).
By theorem 3.1, there exists a spectral sequence which computes $H_n(G_{i+1}, \mathbb{Z})$ satisfying

$$E^2_{p,q} = H_p(G_{i+1}/G_i, H_q(G_i, \mathbb{Z})) \Longrightarrow H_{p+q}(G_{i+1}, \mathbb{Z}). \tag{5.3}$$

Then we have the following filtration which shows the above spectral sequence collapses:

$$0 = F^{-1}H_2 \subset F^0 H_2 \subset F^1 H_2 \subset F^2 H_2 = H_2.$$

Since G_{i+1}/G_i is cyclic, $E^2_{2,0} = H_2(G_{i+1}/G_i, \mathbb{Z}) = 0$. Thus the above filtration of $H_2(G_{i+1}, \mathbb{Z})$ has only two steps. By collapsing we have $E^\infty_{0,2} = E^2_{0,2}$ and $E^\infty_{1,1} = E^2_{1,1}$. Therefore there is an exact sequence

$$0 \longrightarrow E^2_{0,2} \longrightarrow H_2(G_{i+1}, \mathbb{Z}) \longrightarrow E^2_{1,1} \longrightarrow 0. \tag{5.4}$$

As $p \times H_2(G_{i+1}, \mathbb{Z}) = 0$, the term $H_2(G_{i+1}, \mathbb{Z})$ is a vector space over $\mathbb{Z}/p\mathbb{Z}$, thus the outside terms of 5.4 are also vector spaces and then the sequence 5.4 splits. Now we have

$$H_1(G_i, \mathbb{Z}) \cong G_i/[G_i, G_i] \cong G_i \cong \coprod_{n-1} \mathbb{Z}/p\mathbb{Z},$$

where $ord(G_i) = p^{(n-1)}$. Therefore we have

$$E^2_{1,1} = H_1(G_{i+1}/G_i, H_1(G_i, \mathbb{Z})) \cong \coprod_{n-1} H_1(G_{i+1}/G_i, \mathbb{Z}/p\mathbb{Z})$$
$$\cong \coprod_{n-1} \mathbb{Z}/p\mathbb{Z}.$$

As for the term $E^2_{0,2}$, we have

$$E^2_{0,2} = H_0(G_{i+1}/G_i, H_2(G_i, \mathbb{Z})) \cong H_2(G_i, \mathbb{Z}).$$

By induction, $E^2_{0,2}$ is elementary abelian of

$$p^{\frac{(n-1)(n-2)}{2}},$$

and therefore $H_2(G_{i+1}, \mathbb{Z})$ has dimension

$$n - 1 + \log_p(\text{the order of } H_2(G_i, \mathbb{Z})) = p^{\frac{n(n-1)}{2}}.$$

If we are given the information of the syndrome $\{H_2(G_e, \mathbb{Z})\}$, which can be computed using $E^2_{p,q} = H_p(G_e/N, H_q(N, \mathbb{Z}))$, then we can deduce the number i such that $G_i = G_e$ by comparing $\{H_2(G_i, \mathbb{Z})\}$ with $\{H_2(G_e, \mathbb{Z})\}$.

6 Conclusion

We have established an abstract model of the theory of error correcting codes. The proposed abstract model of codes requires only quotient structures of groups and the proposed decoding method on this model requires only homological computation. Therefore we have expanded the subject of error correction to the category of general groups. The future theoretical problem that we should investigate is how to establish a kind of geometric generalization of error correction, or how to give error correction geometric meaning. Some of the future practical problems which we should solve are to reduce the complexity of the proposed decoding method and to apply our method to nonlinear codes.

References

[1] Benson, D. J. and Carlson, J. F., "The Cohomology of Extraspecial Groups" , *Bull. London Math. Soc. 24 (1992), 209-235.*

[2] Brown, K. S., "Cohomology of Groups" , *Springer-Verlag, 1982.*

[3] Cartan, H. and Eilenberg, S., "Homological Algebra", *Princeton University Press, 1956.*

[4] Duursma, I. M. and Kötter, R., "Error-locating pairs for cyclic codes", *IEEE Trans. Inform. Theory, vol. 40 (1994), pp. 1108-1121 .*

[5] Feng, G.-L. and Rao, T. R.N., "Decoding algebraic-geometric codes up to the designed minimum distance", *IEEE Trans. Inform. Theory, vol. 39 (1993), pp. 37-45.*

[6] Hammons Jr., A. R., Kumar, P. V., Calderbank, A. R., Sloan, N. J. A., and Solé, P., "The $\mathbb{Z}_4$-linearity of Kerdock, Preparata, Goethals, and related codes", *IEEE Trans. Inform. Theory, vol. 40(1994), pp. 301-319.*

[7] Lyndon, R. C., "The Cohomology Theory of Group Extensions", *Duke Math. Journal. 15 (1948), pp. 271-292.*

[8] MacLane, S., "Homology" *Springer-Verlag, 1963.*

[9] Pellikaan, R., "On decoding by error location and dependent sets of error location and dependent sets of error positions", *Discrete Math., no. 106/107 (1992), pp. 369-381.*

[10] Spanier, E., "Algebraic Topology" *McGraw-Hill, 1966.*

New extremal doubly even [80,40,16] codes with an automorphism of order 13 *

Stoyan N. Kapralov
Technical University, Gabrovo, Bulgaria, kapralov@tugab.bg

Radka Russeva
Konstantin Preslavsky University, Shoumen, Bulgaria
r.russeva@fmi.shu-bg.net

Valentina Radeva
Higher Military School, Shoumen, Bulgaria, V_Radeva@abv.bg

Abstract

In this work we present a construction of extremal doubly-even [80,40,16] codes, possessing an automorphism of order 13 and give 35 new examples of such codes.

1 Introduction.

The extended quadratic residue code QR_{80} of length 80 [3] was the only known doubly-even [80,40,16] code for a long time. In [1] Gulliver, Harada and Kim found three new extremal doubly-even codes $B_{80,1}$, $B_{80,2}$ and $B_{80,3}$ as a bordered double circulant [80,40] ones. Other extremal codes were obtained via automorphisms. All the odd primes p dividing the order of the group of a doubly-even [80,40,16] code are 79, 19, 13, 7, 5 and 3. QR_{80} is the unique up to equivalence extremal doubly-even code of length 80, possessing an automorphism of order $p = 79$. There exist exactly 11 doubly-even [80,40,16] codes with an automorphism of order $p = 19$ [5]. We have been intrigued to investigate the next value of $p = 13$. There are six previously known codes with an automorphism of order 13: QR_{80}, $B_{80,1}$, $B_{80,2}$, $B_{80,3}$ and two codes announced in [6] with an automorphism of order 39. Let denote these codes by $R_{80,1}$, $R_{80,2}$. In this work, using the field with 2^{12} elements, we construct 35 new doubly-even [80,40,16] codes, possessing an automorphism of order 13. Thus we improve the lower bound for the number of known doubly-even [80,40,16] codes from 17 to 52.

*Research supported by The Technical University of Gabrovo, Project III.7/25.03.2002, and by The Konstantin Preslavsky University, Project 15/14.03.2002.

2 Results.

We make use of the method for constructing self-dual codes via an automorphism of odd prime order developed by Huffman and Yorgov in [2, 7, 8]. Suppose $\mathcal{C}$ is a doubly-even [80,40,16] code with an automorphism σ of order 13. It can be shown that σ has 6 cycles and 2 fixed points in its decomposition. Up to equivalent code we may assume $\sigma=(1,2\ldots 13)(14,15\ldots 26)\cdots(66,67\ldots 78)$. Let $E_\sigma(\mathcal{C})$ be the set of those vectors in $\mathcal{C}$, which have even weight in each cycle of σ. Denote $F_\sigma(\mathcal{C})=\{v\in\mathcal{C}|\ \ v\sigma=v\}$. The code $\mathcal{C}$ can be decomposed as a direct sum of its subcodes $F_\sigma(\mathcal{C})$ and $E_\sigma(\mathcal{C})$. Each vector v from $F_\sigma(\mathcal{C})$ is constant on any cycle of σ. Define the map $\pi: F_\sigma(\mathcal{C})\rightarrow F_2^8$, where for $v\in F_\sigma(\mathcal{C})$ $\pi(v)$ is the binary vector of length 8 obtained by choosing a coordinate from each cycle of v. Then $\pi(F_\sigma(\mathcal{C}))$ is a binary self-dual [8,4] code. Since $\mathcal{C}$ is doubly-even and the multiplicative order of σ is $13\equiv 1(mod4)$, the code $\pi(F_\sigma(\mathcal{C}))$ is also doubly-even one [8]. According to [4] it is equivalent to the Hamming code H_8. As the group of H_8 is triply transitive we can fix the generator matrix of $F_\sigma(\mathcal{C})$ in the form

$$gen(F_\sigma(\mathcal{C}))=\begin{pmatrix} J & & & & & J & 1 & 1\\ & J & & & J & & 1 & 1\\ & & J & & J & J & 0 & 1\\ & & & J & J & J & 1 & 0 \end{pmatrix},$$

where J is the all-one vector of length 13 and the blanks are zero's.

Let P be the binary cyclic code of length 13, generated by $x+1$. Since the parity check polynomial $\frac{x^{13}-1}{x+1}$ is irreducible, P is a field with 2^{12} elements. We let $E_\sigma(\mathcal{C})^*$ be the code $E_\sigma(\mathcal{C})$ with the last two coordinates deleted. The restriction of every vector $v\in E_\sigma(C)^*$ on the cycles of σ can be viewed as an element $v_0+v_1x+\ldots+v_{12}x^{12}$ from P. In this way we define the map $\varphi:\ E_\sigma(\mathcal{C})^*\rightarrow P^6$. It follows from [[8], Theorem 2] that the image $\varphi(E_\sigma(\mathcal{C})^*)$ is a self-dual [6,3] code over P with respect to the following inner product $(u,v)=u_1v_1^{2^6}+u_2v_2^{2^6}+\ldots+u_6v_6^{2^6}=0$.

We consider the elements $\beta=x+x^3+x^4+x^5+x^8+x^9+x^{10}+x^{12}$ and $\gamma=x+x^3+x^{10}+x^{11}$ from P of multiplicative order 63 and 65, respectively. Therefore $\beta\gamma$ is a primitive element in the field and $P\setminus\{0\}=\{\beta^i\gamma^j$, for $0\le i\le 62$ and $0\le j\le 64\}$. By a computer search we establish that any 3-weight vector in $\varphi(E_\sigma(\mathcal{C})^*)$ generates a vector of length less then 16 in $\mathcal{C}$. So $\varphi(E_\sigma(\mathcal{C})^*)$ is an [6,3,4] M.D.S. code over the field P and any three coordinates are information positions. By row reducing, up to equivalent code, we obtain the generator matrix of $\varphi(E_\sigma(\mathcal{C})^*)$ in the form:

$$G=\begin{pmatrix} \delta^{i_1} & & & \beta^{k_1} & \beta^{k_2}\delta^{i_4} & \beta^{k_3}\delta^{i_5}\\ & \delta^{i_2} & & \beta^{k_4} & \beta^{k_5}\gamma^{s_1} & \beta^{k_6}\gamma^{s_2}\\ & & \delta^{i_3} & \beta^{k_7} & \beta^{k_8}\gamma^{s_3} & \beta^{k_9}\gamma^{s_4} \end{pmatrix},$$

where the blanks are zero's, $\delta=\gamma^{13}=1+x^4+x^{10}+x^{12}$, $0\le i_j\le 4$, for $j=1,2,3$, $0\le k_t\le 62$ for $t=1,\ldots,9$ and $0\le s_l\le 64$ for $l=1,2,3,4$. Because of the self-duality of the code $\varphi(E_\sigma(\mathcal{C})^*)$ we obtain the following restrictions:

$$\begin{gathered}\beta^{k_1}+\beta^{k_2}+\beta^{k_3}=0,\ \ \beta^{k_4}+\beta^{k_5}+\beta^{k_6}=0,\ \ \beta^{k_7}+\beta^{k_8}+\beta^{k_9}=0,\\ \beta^{k_1+k_4}+\beta^{k_2+k_5}\gamma^{13i_4-s_1}+\beta^{k_3+k_6}\gamma^{13i_5-s_2}=0,\\ \beta^{k_1+k_7}+\beta^{k_2+k_8}\gamma^{13i_4-s_3}+\beta^{k_3+k_9}\gamma^{13i_5-s_4}=0 \text{ and}\\ \beta^{k_4+k_7}+\beta^{k_5+k_8}\gamma^{s_1-s_3}+\beta^{k_6+k_9}\gamma^{s_2-s_4}=0\end{gathered}$$

To get a generator matrix for $E_\sigma(\mathcal{C})^*$ we replace any element g_{ij} from G by a 12×13 binary circulant type matrix with first row corresponding to $\varphi^{-1}(g_{ij})$ for $i=1,2,3$ and $j=1,2,\ldots,6$. So the generator matrix of the code $\mathcal{C}$ has the form

$$gen(\mathcal{C})=\begin{pmatrix} gen(E_\sigma(\mathcal{C})^*) & 0\\ & gen(F_\sigma(\mathcal{C})) \end{pmatrix}$$

For different values of the parameters in G we construct 35 new inequivalent doubly-even [80,40,16] codes. Denote them by $\mathcal{C}_1,\mathcal{C}_2,\ldots,\mathcal{C}_{35}$. The corresponding values of the parameters are given in Table 1.

The weight enumerator of a doubly-even [80,40,16] code is uniquely determined in [3], with 97565 minimal weight vectors. Let D_{16} be the number of ordered pairs codewords of minimal weight in distance 16 in a code. In Table 1 we give the invariants D_{16} for the codes constructed. They are different for all of the codes. In Table 2 we present the invariants D_{16} for the previously known doubly-even [80,40,16] codes with an automorphism of order 13. They are different to the invariants in Table 1. Thus we prove that the constructed 35 codes are new. We improve the lower bound for the number of known doubly-even [80,40,16] codes from 17 to 52. The results are summarized in the next theorem.

Theorem 2.1 *1)There are at least 52 nonequivalent extremal doubly-even [80,40,16] binary codes. 2)There are at least 41 nonequivalent extremal doubly-even [80,40,16] binary codes with an automorphism of order 13.*

References

[1] T. A. Gulliver, M.Harada and J.-L.Kim, Construction of New Extremal Self-Dual Codes, Discrete Math. (to appear)

[2] W.C.Huffman, Automorphisms of codes with application to extremal doubly-even codes of length 48, IEEE Trans. Inform. Theory, vol 28, 1982, pp. 511-521.

[3] C.L.Mallows and N.J.A.Sloane, An upper bound for self-dual codes, Inform.Contr., vol.22, 1973, pp. 188-200.

[4] V.Pless, A classification of self-orthogonal codes over GF(2), Discrete Math.,vol.3, 1972, pp. 209-246.

[5] R.A.Dontcheva and M.Harada, Extremal Doubly-Even [80,40,16] Codes with an Automorphism of Order 19., preprint.

[6] R.A.Dontcheva, private communication, Apr. 12, 2002.

[7] V.Y.Yorgov, Binary self-dual codes with automorphisms of odd order (in Russian), Probl.Pered.Inform. vol. 19, 1983, pp. 11-24. English translationin in Probl.Pered. Inform. Transm. 19, 1983, pp. 260-270.

[8] V.Y.Yorgov, A method for constructing inequivalent self-dual codes with applications to length 56, IEEE Trans.Inform. Theory, vol. 33, 1987, pp. 77-82.

Table 1. Parameters in $G = gen(\varphi(E_\sigma(\mathcal{C}_i)^*))$ and the invariant D_{16} for the doubly-even [80,40,16] codes $\mathcal{C}_i$

Code	i_1, i_2, i_3	k_1, k_2, k_3, i_4, i_5	k_4, k_5, k_6, s_1, s_2	k_7, k_8, k_9, s_3, s_4	D_{16}
$\mathcal{C}_1$	2,4,2	0,0,0,1,4	14,41,3,24,62	14,41,3,2,42	20044440
$\mathcal{C}_2$	3,4,1	0,23,23,1,4	4,55,58,40,29	4,7,54,62,49	19979310
$\mathcal{C}_3$	4,4,4	1,2,26,0,1	4,27,56,35,28	16,35,45,60,29	19923540
$\mathcal{C}_4$	1,2,2	1,26,2,0,2	21,38,62,39,52	29,41,34,6,59	19967610
$\mathcal{C}_5$	0,2,0	1,26,2,2,1	5,39,54,32,14	11,10,33,54,47	19920030
$\mathcal{C}_6$	3,0,1	1,26,2,4,1	7,56,9,8,54	12,45,41,54,4	20035080
$\mathcal{C}_7$	1,0,1	1,26,2,4,1	46,57,48,56,23	61,29,3,60,9	19975410
$\mathcal{C}_8$	2,0,4	2,1,26,0,1	42,22,27,35,17	58,39,3,6,52	19952790
$\mathcal{C}_9$	4,0,2	2,1,26,0,1	43,9,55,52,47	46,38,15,59,24	19987110
$\mathcal{C}_{10}$	3,4,0	2,1,26,0,1	62,24,61,37,44	34,33,43,16,30	19952400
$\mathcal{C}_{11}$	4,3,0	2,1,26,3,0	49,37,43,8,56	56,40,61,46,32	20014800
$\mathcal{C}_{12}$	3,0,3	2,1,26,4,1	32,10,29,29,25	8,30,57,49,35	19955130
$\mathcal{C}_{13}$	1,2,2	2,1,26,4,4	13,44,17,35,51	18,55,5,29,44	19990230
$\mathcal{C}_{14}$	0,4,4	2,26,1,1,0	3,57,43,47,59	48,29,50,61,26	19945770
$\mathcal{C}_{15}$	4,1,1	2,26,1,1,0	39,24,17,31,63	30,54,31,27,28	20031960
$\mathcal{C}_{16}$	1,4,0	2,26,1,2,3	35,36,28,15,6	27,4,56,12,13	19969170
$\mathcal{C}_{17}$	1,1,0	2,26,1,4,4	0,25,25,28,6	2,34,62,7,44	19975800
$\mathcal{C}_{18}$	4,3,1	26,1,2,1,3	5,37,48,1,50	17,40,3,16,7	19967220
$\mathcal{C}_{19}$	1,0,3	26,1,2,4,1	55,34,35,28,15	15,47,27,25,8	20027670
$\mathcal{C}_{20}$	3,0,4	1,5,11,3,1	49,43,37,20,32	45,36,27,35,35	20010120
$\mathcal{C}_{21}$	2,4,1	1,58,20,0,1	62,23,32,25,20	25,54,50,23,48	20011290
$\mathcal{C}_{22}$	0,3,4	3,15,62,3,0	43,31,16,16,21	57,8,30,33,27	19983210
$\mathcal{C}_{23}$	3,3,2	3,19,54,2,4	28,58,25,29,43	52,5,50,57,4	19970340
$\mathcal{C}_{24}$	3,0,3	5,38,42,1,4	46,16,5,47,19	16,32,38,27,50	20007390
$\mathcal{C}_{25}$	2,0,2	7,61,39,1,2	13,43,20,53,2	48,26,35,37,11	20002710
$\mathcal{C}_{26}$	4,2,2	24,3,9,1,2	14,39,11,59,19	2,58,42,38,23	20003490
$\mathcal{C}_{27}$	3,4,3	29,61,3,2,4	19,25,59,48,28	47,33,50,56,26	20033520
$\mathcal{C}_{28}$	4,2,3	45,1,49,4,2	51,36,20,1,4	30,52,47,37,32	19966440
$\mathcal{C}_{29}$	1,3,4	50,3,59,0,2	46,61,1,29,4	26,29,57,23,56	20000760
$\mathcal{C}_{30}$	2,4,1	52,43,11,4,3	3,14,41,54,9	28,25,58,6,62	19963710
$\mathcal{C}_{31}$	2,2,2	54,9,27,3,3	58,39,3,8,23	23,40,26,42,62	19974630
$\mathcal{C}_{32}$	1,1,4	54,15,57,4,1	53,27,31,13,15	20,47,49,18,63	20003100
$\mathcal{C}_{33}$	1,1,0	58,1,20,4,2	12,28,9,19,59	21,56,31,14,32	19952010
$\mathcal{C}_{34}$	3,3,3	58,29,7,2,0	15,18,17,14,35	36,39,62,58,41	19990230
$\mathcal{C}_{35}$	3,3,0	59,50,3,4,4	9,24,3,52,52	34,34,0,52,52	19962540

Table 2.

Code	D_{16}	Code	D_{16}	Code	D_{16}
QR_{80}	21104850	$B_{80,1}$	20062380	$B_{80,2}$	20117370
$B_{80,3}$	20034300	$R_{80,1}$	20138430	$R_{80,2}$	20062380

Permutation decoding of the binary codes of triangular graphs

J. D. Key
Department of Mathematical Sciences
Clemson University
Clemson SC 29634, U.S.A.
keyj@clemson.edu

J. Moori and B. G. Rodrigues
School of Mathematics, Statistics
and Information Technology
University of Natal-Pietermaritzburg
Pietermaritzburg 3209, South Africa
moori@nu.ac.za and rodrigue@nu.ac.za

Abstract

We examine the binary codes obtained from the adjacency matrix of the triangular graph $T(n)$ for $n \geq 5$ and show that permutation decoding can be used for these codes by finding explicit PD-sets.

1 Introduction

For any n, the triangular graph $T(n)$ is defined to be the line graph of the complete graph K_n, i.e. the vertices are the 2-subsets of $\Omega = \{1, 2, \ldots, n\}$ and vertices $\{a, b\}$ and $\{c, d\}$ are adjacent if they have one letter from Ω in common. The valency is $2(n-1)$ and the graph is strongly regular. The binary codes formed from the span of the adjacency matrix of these graphs have been examined in [9, 5, 3, 4, 1, 2]. The dimension and weight enumerator of these codes are easily determined. Here we examine the codes and their duals further, and show how the case $n = 6$ distinguishes itself. We show that S_n is the full automorphism group of the code for $n \geq 5$ except in the case $n = 6$. We also look at the question of minimum-weight generators for the code, and for its dual, and obtain explicit permutation-decoding sets for the code.

The code is also that of the 1-$(\frac{n(n-1)}{2}, 2(n-2), 2(n-2))$ design $\mathcal{D}$ obtained by taking the rows of the adjacency matrix as the incidence vectors of the blocks; the automorphism group of this design will contain the automorphism group of the graph, the latter of which is easily seen to be S_n. Similarly, the automorphism group of the code will contain S_n. However for $n = 6$ the group of the design and code is larger than the group of the graph (S_6), and we will use the words of weight-3 in the dual code to explain this.

2 Terminology

An incidence structure $\mathcal{D} = (\mathcal{P}, \mathcal{B}, \mathcal{I})$, with point set $\mathcal{P}$, block set $\mathcal{B}$ and incidence $\mathcal{I}$ is a t-(v,k,λ) design, if $|\mathcal{P}| = v$, every block $B \in \mathcal{B}$ is incident with precisely k points, and every t distinct points are together incident with precisely λ blocks. The **code C_F of the design** $\mathcal{D}$ over the finite field F is the space spanned by the incidence vectors of the blocks over F. If the point set of $\mathcal{D}$ is denoted by $\mathcal{P}$ and the block set by $\mathcal{B}$, and if $\mathcal{Q}$ is any subset of $\mathcal{P}$, then we denote the incidence vector of $\mathcal{Q}$ by $v^{\mathcal{Q}}$. Thus $C_F = \langle v^B \mid B \in \mathcal{B} \rangle$.

If a code C over a field of order q is of length n, dimension k, and minimum weight d, then we write $[n,k,d]_q$ for C. A **generator matrix** matrix for the code is a $k \times n$ matrix made up of a basis for C. The **dual** code $C^\perp$ is the orthogonal under the standard inner product $(,)$. A **check** (or **parity-check**) matrix for C is a generator matrix H for $C^\perp$; the **syndrome** of a vector $y \in F^n$ is Hy^T. Two codes are **isomorphic** if they can be obtained from one another by permuting the coordinate positions. Any code is isomorphic to a code with generator matrix in so-called **standard form**, i.e. the form $[I_k \,|\, A]$; a check matrix then is given by $[-A^T \,|\, I_{n-k}]$. The first k coordinates are the **information symbols** and the last $n-k$ coordinates are the **check symbols**. An **automorphism** of a code C is an isomorphism from C to C.

Permutation decoding was first developed by MacWilliams [7] and involves finding a particular set of automorphisms of the code, called a PD-set. The method is described fully in [8, Chapter 15] and [6, Section 8]. A **PD-set** for a t-error-correcting code C is a set $\mathcal{S}$ of automorphisms of C which is such that every possible error vector of weight $s \le t$ can be moved by some member of $\mathcal{S}$ to another vector where the s non-zero entries have been moved out of the information positions. Thus every t-set of coordinate positions is moved by at least one member of $\mathcal{S}$ to a t-set consisting only of check-position coordinates. Such a set, if it exists, will fully use the error-correction potential of the code: see [6, Theorem 8.1].

The algorithm for permutation decoding is as follows: given a t-error-correcting $[n,k,d]_q$ code C with generator matrix G in standard form and check matrix H. Thus $G = [I_k|A]$ and $H = [A^T|I_{n-k}]$, for some A, and any vector v of length k is encoded as vG. Suppose x is sent and y is received and at most t errors occur. Let $\mathcal{S} = \{g_1, \ldots, g_s\}$ be the PD-set. Compute the syndromes $H(yg_i)^T$ for $i = 1, \ldots, s$ until an i is found such that the weight of this vector is t or less. Now look at the information symbols in this vector, and obtain the codeword c that has these information symbols; decode y as cg_i^{-1}.

3 The binary codes

Let n be any integer and let $T(n)$ denote the triangular graph with vertex set $\mathcal{P}$ the $\binom{n}{2}$ 2-subsets (or duads) of a set Ω of size n. The 1-design $\mathcal{D} = (\mathcal{P}, \mathcal{B})$ will have point set $\mathcal{P}$ and for each point $\{a,b\} \in \mathcal{P}$, $a \ne b$, $a, b \in \Omega$, a block $\overline{\{a,b\}}$ defined in the following way:

$$\overline{\{a,b\}} = \{\{a,x\},\{b,y\} \mid x \ne a,\, b;\ y \ne a,\, b\}.$$

Thus $\mathcal{B} = \{\,\overline{\{a,b\}} \mid a, b \in \Omega,\ a \ne b\}$. The incidence vector of the block $\overline{\{a,b\}}$ is then

$$v^{\overline{\{a,b\}}} = \sum_{x \ne a} v^{\{a,x\}} + \sum_{y \ne b} v^{\{b,y\}}.$$

If a, b, c are distinct points in Ω, write

$$v^{\overline{\{a,b,c\}}} = v^{\{a,b\}} + v^{\{b,c\}} + v^{\{a,c\}}.$$

Proposition 1 *Let C be the binary code obtained by the row span of an adjacency matrix for the triangular graph $T(n)$, where $n \ge 5$.*

If $n = 2m$ then C is a self-orthogonal $[\binom{2m}{2}, 2m-2, 4(m-1)]_2$ code with weight distribution the zero vector and, for m even,

$$< 4(m-1), \binom{2m}{2} >, < 8(m-2), \binom{2m}{4} >, \ldots, < m^2, \frac{1}{2}\binom{2m}{m} >$$

and, for m odd,

$$< 4(m-1), \binom{2m}{2} >, < 8(m-2), \binom{2m}{4} >, \ldots, < m^2-1, \binom{2m}{m-1} >.$$

If n is odd, then C is a $[\binom{n}{2}, n-1, n-1]_2$ code with weight distribution the zero vector and

$$< n-1, n >, \ldots, < 2i(n-2i), \binom{n}{2i} >, \ldots,$$

where $1 \le i \le (n-1)/2$.

The minimum weight of $C^\perp$ is 3 and any word of the form $v^{\overline{\{a,b,c\}}}$ is in $C^\perp$. If $n \ne 6$, these are all the words of weight 3 in $C^\perp$, and the number of words of weight 3 is thus $\binom{n}{3}$. If $n = 6$, further words of weight 3 have the form $v^{\{a,b\}} + v^{\{c,d\}} + v^{\{e,f\}}$ where $\Omega = \{a,b,c,d,e,f\}$; in this case there are 35 words of weight 3.

The automorphism group of C is S_n unless $n = 6$, in which case it is $PGL_4(2) \cong A_8$.

(Here $< i, j >$ denotes j vectors of weight i.)

Concerning the question of bases of minimum-weight vectors for C and $C^\perp$, it is clear that if n is even then C has a basis of minimum-weight vectors, since the incidence vectors of the blocks are the minimum-weight vectors and span C by definition.

Proposition 2 *For $n \ge 5$, C has a basis of minimum-weight vectors. The code $C^\perp$ has a basis of minimum-weight vectors for n odd, but not for n even.*

4 Permutation decoding

To obtain specific PD-sets for the codes we order the points as follows:

$$P_1 = \{1,n\}, P_2 = \{2,n\}, \ldots, P_{n-1} = \{n-1,n\}, \tag{1}$$

first, followed by the set

$$P_n = \{1,2\}, P_{n+1} = \{1,3\}, \ldots, P_{2n-2} = \{2,3\}, \ldots, P_{\binom{n}{2}} = \{n-2,n-1\}. \tag{2}$$

The generator matrix for $C^\perp$, using the words of weight 3 (with $\jmath$ if n is even), is then a check matrix for C in standard form. The generator matrix for C will then also be in standard form, with the first $n-1$ coordinates the information symbols for n odd, and the first $n-2$ for n even. We find PD-sets for the codes in S_n:

Proposition 3 *For $n \geq 5$ PD-sets can be found for the code C. Using the ordering of the point set $\mathcal{P}$ given in Equations (1) and (2), the following sets of permutations in S_n in the natural action on the points $\mathcal{P}$ are PD-sets for C.*

1. *For $n \geq 5$ odd, a PD-set of n elements is*

$$\mathcal{S} = \{1_G\} \cup \{(i,n) \mid 1 \leq i \leq n-1\}.$$

2. *For $n \geq 6$ and even, a PD-set of $n^2 - 2n + 2$ elements is*

$$\mathcal{S} = \{1_G\} \cup \{(i,n) \mid 1 \leq i \leq n-1\} \cup \{[(i,n-1)(j,n)]^{\pm 1} \mid 1 \leq i,j \leq n-2\}.$$

References

[1] E. F. Assmus, Jr. and J. D. Key. *Designs and their Codes.* Cambridge: Cambridge University Press, 1992. Cambridge Tracts in Mathematics, Vol. 103 (Second printing with corrections, 1993).

[2] E. F. Assmus, Jr. and J. D. Key. Designs and codes: an update. *Des. Codes Cryptogr.*, 9:7–27, 1996.

[3] A. E. Brouwer and C. J. van Eijl. On the *p*-rank of the adjacency matrices of strongly regular graphs. *J. Algebraic Combin.*, 1:329–346, 1992.

[4] A. E. Brouwer and J.H. van Lint. Strongly regular graphs and partial geometries. In D.M. Jackson and S.A. Vanstone, editors, *Enumeration and Design*, pages 85–122. Toronto: Academic Press, 1984. Proc. Silver Jubilee Conf. on Combinatorics, Waterloo, 1982.

[5] Willem H. Haemers, René Peeters, and Jeroen M. van Rijckevorsel. Binary codes of strongly regular graphs. *Des. Codes Cryptogr.*, 17:187–209, 1999.

[6] W. Cary Huffman. Codes and groups. In V. S. Pless and W. C. Huffman, editors, *Handbook of Coding Theory*, pages 1345–1440. Amsterdam: Elsevier, 1998. Volume 2, Part 2, Chapter 17.

[7] F. J. MacWilliams. Permutation decoding of systematic codes. *Bell System Tech. J.*, 43:485–505, 1964.

[8] F. J. MacWilliams and N. J. A. Sloane. *The Theory of Error-Correcting Codes.* Amsterdam: North-Holland, 1983.

[9] Vladimir D. Tonchev. *Combinatorial Configurations Designs, Codes, Graphs.* Pitman Monographs and Surveys in Pure and Applied Mathematics, No. 40. New York: Longman, 1988. Translated from the Bulgarian by Robert A. Melter.

On Dominant Error Sequences for Some ISI Channels with White Gaussian Noise

Victor D.Kolesnik, University for Air Space Instrumentation,
St.Petersburg, Russia

Abstract

We consider the binary data transmission through the Gaussian channels with the intersymbol interference (ISI). Given the target polynomial $T(D)$, we assume that the linear equalization is used for getting the partial response (PR) model of this channel. The algorithm Viterbi provides a symbol detection for the PR channel. We show that for lowpass PR channels (e.g., wire telephone links, magnetic channels with the perpendicular recordings and so on) the dominant error sequences for large enough signal/noise ratio are mainly bursts of length 2 or greater.

1. Introduction

We consider binary channels with the intersymbol interference and assume that the channel output $r(t)$ is represented by the following linear model:

$$r(t) = \sum_k (a_k - a_{k-1}) g(t - kT_0) + w(t),$$

where $...,a_{k-1}, a_k, ...$ are input binary symbols, $a_k \in \{-1,+1\}$, spaced by interval T_0, $g(t)$ is the step response of the channel and $w(t) \sim N(0, \sigma^2)$ is the white Gaussian noise. The function $h(t) = g(t) - g(t - T_0)$ is called a dibit response. Then $r(t) = \sum_k a_k h(t - kT_0) + w(t)$. The channel is called a lowpass if the spectral density of $h(t)$ is concentrated in low frequencies. Two examples of step responses for lowpass ISI channels are given below

$$g(t) = \begin{cases} max(0, 1 - exp(-\frac{t}{T_0 D_0})), & \text{for unloaded telephone wire link, (a)} \\ \text{erf}(\frac{2t\sqrt{\log 2}}{T_0 D_0}), & \text{for perpendicular magnetic recording, (b)} \end{cases} \quad (1)$$

where D_0 is the channel parameter responsible for the interference power and $\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x \exp(-y^2)dy$ is the error function.

Even in absence of noise the channel filter creates a signal distortion because of interference especially for small T_0 and large D_0. The equalization is usually used for decreasing interference length and simplifying the maximum likelihood detection. Let $a(D) = \sum_k a_k D^k$, $h(D) = \sum_k h_k D^k$, $h_k = h(kT_0)$, and $r(D)$, $w(D)$ be the correspondent $D-$transforms of sampled signals $r(t)$ and $w(t)$. Then, $r(D) = h(D)a(D) + w(D)$. In order to get the partial response (PR) channel the signal $r(D)$ at the channel output is processed by the linear filter (equalizer) $E(D)$:

$$y(D) = E(D)r(D) = T(D)a(D) + n(D),$$

where $T(D) = E(D)h(D) = T_0 + T_1 D + ... + T_L D^L$ is so called target polynomial and $n(D) = E(D)w(D)$ is a filtered Gaussian noise with the covariation $R_n(D) = \sigma^2 E(D)E(D^{-1})$. Given target $T(D)$, the equalizer can be found by a standard procedure (see, for example, [3]).

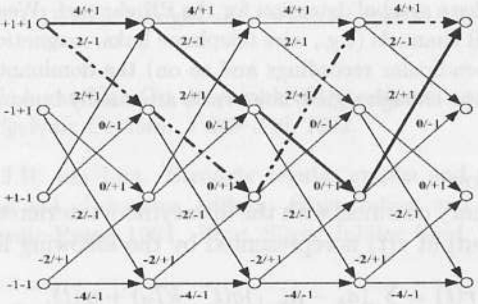


Fig.1 Trellis for PR channel with the target $T(D) = 1 + 2D + D^2$

The noiseless output $T(D)a(D)$ of the partial response channel can be represented by the trellis diagram with 2^L states. Any path in the trellis corresponds to some information sequence $a(D)$. Given the received sequence $y(D)$, the detector searches the most probable path in the trellis. Two paths corresponding to information words $\mathbf{a}_i, \mathbf{a}_j$ differ in so called input error sequence $\mathbf{e}_a = \frac{\mathbf{a}_i - \mathbf{a}_j}{2}$.The most popular for PR channels are the Viterbi detector and $BCJR$ detector [4] that additionally supplies decisions by its confidence values.

It is known that the target polynomial $T(D) = 1 + 2D + D^2$ is a good enough for channels (1a) and (1b). Four sections of the trellis corresponding to this target is shown in Fig.1. Two paths at the lowest Euclidean distance are also shown in this picture, these paths differ in the input error sequence $\mathbf{e}_a = \frac{a_i - a_j}{2} = ...00+-00...$. The output error sequence $\mathbf{e}_y$ is the convolution of $\mathbf{e}_a$ and $\mathbf{T} = (1\ 2\ 1)$: $\mathbf{e}_y = \mathbf{e}_a * \mathbf{T} = (...00++--00...)$, where $(+)$ denotes $+1$ and $(-)$ denotes -1. The information word $\mathbf{a}_0 = (+-++)$ is transmitted by the sequence $\mathbf{y} = \mathbf{a}_0 * \mathbf{T} = (...4\ 2\ 0\ 2\ 4...)$. An input error $\mathbf{e}_a$ appears if the detector carries out the decision in a favour of $\mathbf{a}_1 = (++++)$, $\mathbf{a}_2 = (-+++)$ or $\mathbf{a}_3 = (--++)$: $\mathbf{e}_{a1} = (0+00)$, $\mathbf{e}_{a2} = (-+00)$, or $\mathbf{e}_{a3} = (-000)$. The set of squared Euclidean distances for the codeword $\mathbf{a}_0$ is $S(\mathbf{a}_0) = \{d_{a1}^2 = 6, d_{a2}^2 = 4, d_{a3}^2 = 6\}$, where $d_{ai}^2 = ||\mathbf{e}_{ai} * \mathbf{T}||^2$. For other codewords we have a different set of distances, e.g., $S(\mathbf{a}_1) = \{6, 6, 20\}$. In general, the error sequences are of infinite length. We write $\mathbf{e}_a$ as a finite length sequence assuming that it is zero everywhere except positions where they are equal to $\mathbf{e}_a$.

The target polynomial may be chosen in many ways. Usually it is chosen to minimize the error probability at the output of the detector. The problem for finding the best target was considered in the literature (see, e.g., [1],[2]). It was found that spectral properties of the dibit response and the target should be similar, however, not obviously coincide. If the channel is lowpass than the optimal target should be the lowpass filter.

The input error sequence $\mathbf{e}_a = (\varepsilon_1, \varepsilon_2, ..., \varepsilon_N) \in \{-1,0,+1\}^N, \varepsilon_1, \varepsilon_N \neq 0$, is called a dominant if it has the much greater probability of appearance among all input error sequences for all information words of a given length. The notion of dominant error sequences is informal and usually is employed only at a weak noise. It is easy to prove that for relatively weak Gaussian noise the probability of $\mathbf{e}_a$ is as follows:

$$\Pr(\mathbf{e}_a) \simeq c\exp(-\frac{d_a^2}{2\sigma^2}),\ d_a^2 = ||\mathbf{e}_a * \mathbf{T}||^2, \qquad (2)$$

where σ^2 is the variance of noise at the equalizer output. Therefore, the dominant error sequences can be found by the analysis of squared distances d_a^2 for the given target $T(D)$.

Our aim is to formalize the notion and find a criteria for searching dominant error sequences in a way convenient for error prediction. Moreover, we would show that for lowpass channels these dominant error sequences are mainly solid bursts of length greater or equal 2.

2. Dominant error sequences

Definition. Let $\mathbf{e}_a$ be a concatenation of two subsequences: $\mathbf{e}_a = (\mathbf{e}_h, \mathbf{e}_b)$. We call $\mathbf{e}_h$ the head and $\mathbf{e}_b$ the body of $\mathbf{e}_a$. Let $\mathbf{E}_h, \mathbf{E}_b$ be two sets of subsequences $\mathbf{e}_h, \mathbf{e}_b$ and assume that $\mathbf{E}_b$ does not contain a zero subsequence. We call the set $\mathbf{E}_D = \{\mathbf{e}_h, \mathbf{e}_b\}$ the $\{\mathbf{E}_h, \mathbf{E}_b\}$–dominant set of sequences if the conditional probability $\Pr(\mathbf{e}_b \in \mathbf{E}_b|\mathbf{e}_h)$, for $\mathbf{e}_h \in \mathbf{E}_h$, is greater 1-$\alpha$ and $\Pr(\mathbf{e}_b \in \mathbf{E}_b|\mathbf{e}'_h)$ for any $\mathbf{e}'_h \notin \mathbf{E}_h$, is less than α for some small $\alpha > 0$. In other words, $\mathbf{e}_a = (\mathbf{e}_h, \mathbf{e}_b)$ is a dominant $\{\mathbf{E}_h, \mathbf{E}_b\}$–error sequence if nonzero body lies in $\mathbf{E}_b$ with very high probability only when the head belongs to $\mathbf{E}_h$.

Unfortunately, this definition depends on α and for some α there might not exist $\mathbf{E}_h, \mathbf{E}_b$. However, it becomes valuable and convenient in applications for channels with a weak and specific noise. The following examples illustrate this definition.

Example 1. Let $N = 3$, $\mathbf{E}_h = \{(0\ 1),(0\ \text{-}1)\}$ and $\mathbf{E}_b$=$\{(1),(\text{-}1)\}$. Assume that $\Pr(\mathbf{e}_b = \pm 1|\mathbf{e}_h = (0\ e)) > 1 - \alpha$ for any $e = \pm 1$ and small enough $\alpha > 0$, while $\Pr(\mathbf{e}_b = \pm 1|\mathbf{e}_h = (\text{-}e\ e)) = \Pr(\mathbf{e}_b = \pm 1|\mathbf{e}_h = (e\ \text{-}e)) \leq \alpha$ and $\Pr(\mathbf{e}_b = \pm 1|\mathbf{e}_h = (0\ 0)) \leq \alpha$. This means that $\{\mathbf{E}_h, \mathbf{E}_b\}$–dominant sequences are solid bursts of length 2. Hence, for this channel the dominant are error sequences of the form (0 1 -1 0) or (0 -1 1 0).

Example 2. Let $N > 3, \mathbf{E}_h = \{(0\ 1),(0\ \text{-}1)\}$ and $\mathbf{E}_b$ be the set of nonzero sequences of length $N - 2$. Assume that $\Pr(\mathbf{e}_b \neq 0|\mathbf{e}_h = (0\ e)) > 1 - \alpha$ for any $e = \pm 1$ and small enough α. For this channel the $\{\mathbf{e}_h, \mathbf{E}_b\}$–dominant sequences are bursts of length $2, ..., N - 1$. When some error sequence starts with the head $\mathbf{e}_h \in \mathbf{E}_h$, the body belongs to $\mathbf{E}_b$, i.e., they are nonzero sequences of length $N - 2$ with the high enough probability.

The main property of channels with $\{\mathbf{E}_h, \mathbf{E}_b\}$–dominant error sequences from Examples 1,2 is the exact prediction of the error in the body-subsequence if we have information about errors in the head-subsequence. If the head of some error sequence equals (0 1) or (0 -1), then one can predict with probability greater than 1-α that the next subsequence (the body) is in error.

The main result is in the following theorem. It relates to the lowpass ISI channels and shows that the dominant input error sequences are bursts of length 2 or greater.

Theorem 1. Let L be the degree of $T(D)$ and $E_{2L+1} = \{\mathbf{e} = (e_0, e_1, ..., e_{2L+1})\}$ be the set of sequences over the ternary alphabet {-1,0,1} with $e_0 = e_{L+2} = ... = e_{2l+1} = 0$. If the following inequality is valid

$$\min_{\mathbf{e} \in E_{2L+1}: \varepsilon_1 = 1, \varepsilon_2 = -1} ||\mathbf{e} * \mathbf{T}||^2 < \min_{\mathbf{e} \in E_{2L+1}: \varepsilon_1 = 1, \varepsilon_2 = 0} ||\mathbf{e} * \mathbf{T}||^2, \tag{3}$$

then the dominant error sequences for $T(D)$ and for a weak noise are $\{\mathbf{E}_h, \mathbf{E}_b\}$-dominant error sequences, where $\mathbf{E}_h = \{(0\ 1),(0\ \text{-}1)\}$ and $\mathbf{E}_b$ is the set of nonzero sequences of length L. This means that the dominant error sequences are of the form $\mathbf{e} = (..., 0, e_i, e_{i+1}, ..., e_{i+L-1}, 0, ...)$, where e_i and $e_{i+1}$are both nonzero. In other words, if (3) is valid, then the dominant error sequences are bursts of length 2 or greater.

For illustration we consider PR channels with $T_1(D) = 1 + 2x + x^2$ and $T_2(D) = 1{+}2x{+}1x^2{+}0.5x^3$. For the first polynomial $\min_{\mathbf{e} \in E_{2L+1}: \varepsilon_1 = 1, \varepsilon_2 = -1} ||\mathbf{e} * \mathbf{T}_1||^2 = 4$ and $\min_{\mathbf{e} \in E_{2L+1}: \varepsilon_1 = 1, \varepsilon_2 = 0} ||\mathbf{e} * \mathbf{T}_1||^2 \geq 6$. For the second polynomial we have $\min_{\mathbf{e} \in E_{2L+1}: \varepsilon_1 = 1, \varepsilon_2 = -1} ||\mathbf{e} * \mathbf{T}_2||^2 = 3.5$ and $\min_{\mathbf{e} \in E_{2L+1}: \varepsilon_1 = 1, \varepsilon_2 = 0} ||\mathbf{e} * \mathbf{T}_2||^2 \geq 6.25$. Hence, dominant error sequences are bursts of length at least two and we may expect that for $T_2(D)$ the appearance of bursts as dominant errors will be observed for less SNR than for $T_1(D)$. The other example we have for the target $T_3(D) = 1 + D - D^2 - D^3 : \min_{\mathbf{e} \in E_{2L+1}: \varepsilon_1 = 1, \varepsilon_2 = -1} ||\mathbf{e} * \mathbf{T}_3||^2 = \min_{\mathbf{e} \in E_{2L+1}: \varepsilon_1 = 1, \varepsilon_2 = 0} ||\mathbf{e} * \mathbf{T}_3||^2 = 4$. We cannot affirm that for this target the dominant error sequences are bursts. Note that this channel is not a lowpass channel.

The inequality (3) may be reformulated in a spectral language, where it is a direct consequence of the lowpass property of the target filter. It is evident that $||\mathbf{e} * \mathbf{T}||^2$ is the energy of the output of the filter with the pulse response $T(D)$ fed with the sequence $\mathbf{e}$. It is easy to conclude that the sequence $\mathbf{e} \in E_{2L+1}$ meeting the conditions $\varepsilon_1 = 1, \varepsilon_2 = -1$ has more high frequency spectral components than the sequence $\mathbf{e} \in E_{2L+1}$ meeting the conditions $\varepsilon_1 = 1, \varepsilon_2 = 0$. The inequality (3) reflects that $T(D)$ corresponds to the filter with less transfer in low frequencies, i.e., to the lowpass filter. Therefore, the following consequence is true.

Consequence. The dominant error sequences for the lowpass ISI channel with the weak Gaussian noise are bursts of length 2 or greater.

The more detailed analysis shows that the bursts are mainly solid (comprising of nonzero symbols only) bursts.

3. Simulation results

We simulate the transmission through telephone wire link (see (1a)) at SNR=19 and 20 dB with $D_0 = 2$ for 2 target polynomials $T_1(D) = 1{+}2x{+}x^2$ and $T_2(D) = 1 + 2x + 1x^2 + 0.5x^3$. The 31-tap equalizer is found for each target by the Least Mean Square approach [3] with the oversampling rate 1:10. The best value of a sampling phase is searched to achieve the best SNR at the output of the detector. The detector for the PR channel is developed on the base of the $BCJR$ [4] algorithm.

The binary stream of errors at the output of the detector is considered as the stream of 8-bit byte errors. For each two adjacent bytes B_{curr} and B_{next} we calculate statistical probability $\Pr(B_{next} \neq 0|B_{curr})$ which is the conditional probability of error in the next byte for a given current error byte pattern. Probabilities $\Pr(B_{next} \neq 0|B_{curr})$ are presented in the following table, where stars denote unknown binary symbols:

B_{curr}	$T_1(D)$ 19 dB	$T_1(D)$ 20 dB	$T_2(D)$ 19 dB	$T_2(D)$ 20 dB
$(********)$	0.0362	0.0157	0.0193	0.0071
$(*******1)$	0.5991	0.5789	0.5192	0.5104
$(******10)$	0.0652	0.0318	0.0373	0.0135
$(******01)$	0.9601	0.9815	0.9720	0.9906
$(******11)$	0.4730	0.4846	0.4889	0.4947

For these channels we have $\{\mathbf{E}_h, \mathbf{E}_b\}$-dominant error sequences, where $\mathbf{E}_h = \{(0\ 1),(0\ \text{-}1)\}$ and $\mathbf{E}_b$ is the set of nonzero 8-bit bytes. These error events corrupt at least two adjacent bytes. This can be effectively employed for error prediction in the multi-step decoding of RS codes in ISI channels (see [5]).

References

[1] J.Fitzpatric, J.K.Wolf and L.Barbosa, "New equalizer targets for samples magnetic recording system," Proc. of the 25th Asilomar Conference on Signal Systems and Computers, pp.30-34, Nov.1991.

[2] Hideki Sawaguchi, Yasutaka Nishida, Hisashi Takano and Hajime Aoi, "Performance Analysis of Modified PRML channels for Perpendicular Recording Systems," The Digest of the INTERMAG2000, pp.24aA-06.

[3] B.Farhang-Boroujeny, "Adaptive Filters, Theory and Applications," John Willey & Sons, 1998.

[4] L.R.Bahl, J.Cocke, F.Jelinek and J.Raviv, "Optimal Decoding of Linear Codes for Minimizing Symbol Error rate," IEEE *Trans. on Information Theory*, vol.20, pp.284-287, Mar,1974.

[5] V.D.Kolesnik, "Multi-Step Decoding in ISI Channels," This issue.

On Error-Detecting Properties of RS-Codes

Victor D.Kolesnik

University of Air Space Instrumentation,
St.Petersburg, Russia

Abstract

We consider Reed-Solomon (N,K,d)–codes, $d = N - K + 1$, over $GF(q)$, $q = 2^m$, with the algebraic error correction of all t–error patterns, $t \leq t_0 \leq \frac{d-1}{2}$. It is well known that RS codes have the high error detection capability, i.e., almost always when the number of errors exceeds t_0 the algebraic decoder (AD_{t_0}) detects these error patterns as uncorrectable. We make this statement quantitative. For every $t > t_0$ we found a fraction $f_{t_0}(t)$ of t–error patterns that cannot be detected by the AD_{t_0}. We show that for long enough RS–codes having large minimal distance the fraction $f_{t_0}(t)$ is very small. This can be effectively used for achieving error correction beyond the minimum distance bound. We also consider the fraction $g_{t_0}(t)$ of uncorrected t-error patterns, $t > t_0$, and show that this value may be estimated by the same technique as used for finding $f_{t_0}(t)$.

1. Introduction

Let d be the minimal distance of the $q-ary$ linear (N,K,d)–code $\mathbf{C}$. Assume that the decoder employs Hamming balls of radius $t_0 \leq \left[\frac{d-1}{2}\right]$ centered at codewords as decision regions. Let $\mathbf{c} \in \mathbf{C}$, $\mathbf{r} = \mathbf{c} + \mathbf{e}$ be sent and received words. The decoder corrects any error sequence $\mathbf{e}$ of Hamming weight $wt(\mathbf{e}) = t$ if $t \leq t_0$. For $wt(\mathbf{e}) > t_0$ the error sequence $\mathbf{e}$ can be detected and the received word $\mathbf{r}$ can be marked as uncorrected (or erroneous) if $\mathbf{e}$ does not lie in any decision region. For the $q - ary$ linear N–space let us denote by $S_t(\mathbf{x})$ the radius t sphere centered at $\mathbf{x}$ and by $B_{t_0}(\mathbf{c})$ the radius t_0 ball centered at $\mathbf{c}$. Then $\mathbf{e}$, $wt(\mathbf{e}) = t > t_0$, can be detected if it does not lie in any intersection $S_t(\mathbf{0}) \cap B_{t_0}(\mathbf{c})$ for $\mathbf{c} \in \mathbf{C}$. We call the value

$$f_{t_0}(t) = \frac{|\cup_{\mathbf{c} \in \mathbf{C}} \{S_t(\mathbf{0}) \cap B_{t_0}(\mathbf{c})\}|}{|S_t(\mathbf{0})|} \tag{1}$$

a fraction of undetected t–errors in decoding with t_0-error correction. It is evident that $f_{t_0}(t) = 0$ for $t \le d - t_0 - 1$. The case $t_0 = 0$ was intensively studied (see, e.g., [1],[2] and references therein). It is well known, that for many codes, particularly for Reed-Solomon (RS) codes, $f_{t_0}(t)$ stays very small for $t_0 > 0$ even for t much greater then t_0. Although this statement is well known and popular there is no a quantitative study of that (see [4], Ch.14, for a sight to this problem).

Note, that the classical algebraic decoding algorithm for BCH codes (based on the Berlecamp-Massey method , see e.g., [4]) supports the decoding in Hamming balls of radius $t_0 \le \frac{d-1}{2}$.We denote such algorithm as AD_{t_0}.

In this report we consider AD_{t_0} and calculate $f_{t_0}(t)$ for $q-ary$ linear codes using its weight distribution $A(w)$. We show that for long enough $RS-$codes the value $f_{t_0}(t)$ is very small. This observation can be successively employed for improving decoding schemes ([5]) for a product of $RS-$codes.

2. Fraction of undetected error patterns

Let $B_{t_0}(\mathbf{c}|w)$ be the same as $B_{t_0}(\mathbf{c})$ under condition that $wt(\mathbf{c}) = w$.The cardinality $I(w, t_0, t)$ of the intersection $S_t(\mathbf{0}) \cap B_{t_0}(\mathbf{c}|w)$ depends on t_0, t and the weight w of $\mathbf{c}$, but does not depend on $\mathbf{c}$ itself:

$$\begin{aligned} I(w, t_0, t) &= |S_t(\mathbf{0}) \cap B_{t_0}(\mathbf{c}|w)| = \\ &= \sum_{u,v,r} \binom{w}{v} \binom{w-v}{r} \binom{N-w}{u} (q-2)^r (q-1)^u, \qquad (2) \\ u+v+r &= t,\, w-v+u \le t_0,\, w-2v+t-t_0 \le r \le t-v. \end{aligned}$$

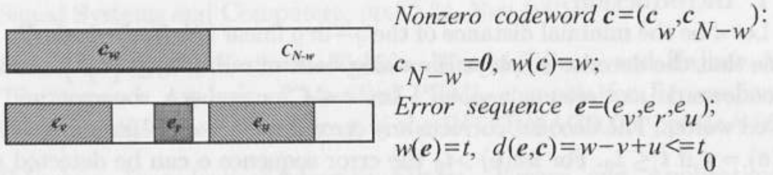

c_w
C_{N-w}
e_v
e_r
e_u
Nonzero codeword $\boldsymbol{c}=(\boldsymbol{c}_w,\boldsymbol{c}_{N-w})$:
$\boldsymbol{c}_{N-w}=\boldsymbol{0}$, $w(\boldsymbol{c})=w$;
Error sequence $\boldsymbol{e}=(e_v,e_r,e_u)$;
$w(\boldsymbol{e})=t$, $d(\boldsymbol{e},\boldsymbol{c})=w-v+u<=t_0$

Fig.1 On the calculation of intersection cardinality

The calculation leading to the expression (2) is illustrated by the diagram (see Fig.1), where two words are shown: the weight w codeword $\mathbf{c} = (\mathbf{c}_w, \mathbf{c}_{N-w})$, $\mathbf{c}_w \ne \mathbf{0}$, and the error sequence with three nonzero parts $\mathbf{e} = (\mathbf{e}_v, \mathbf{e}_r, \mathbf{e}_u)$. Similarly shading parts of words coincide and the unshaded parts are zero subvectors. Here we denote by $\mathbf{x}_m$ a $q-ary$ vector having a support of length m.

We have $v \ge w + u - t_0 = w + t - v - r - t_0 \ge t - t_0$, $v = t - r - u \le t$, hence $r \ge w - 2v + t - t_0$, $r = t - v - u \le t - v$ and

$$\begin{aligned} I(w, t_0, t) &= |S_t(\mathbf{0}) \cap B_{t_0}(\mathbf{c}|w)| = \qquad (3) \\ &= \sum_{v=t-t_0}^{t} \sum_{r=w-2v+t-t_0}^{t-v} \binom{w}{v} \binom{w-v}{r} \binom{N-w}{t-v-r} (q-2)^r (q-1)^{t-r-v}. \end{aligned}$$

The numerator in (1) can be written in a following way:

$$|\cup_{\mathbf{c}\in\mathbf{C}} \{S_t(\mathbf{0}) \cap B_{t_0}(\mathbf{c})\}| = |\cup_w \{\cup_{\mathbf{c}\in\mathbf{C}, wt(\mathbf{c})=w} \{S_t(\mathbf{0}) \cap B_{t_0}(\mathbf{c}|w)\}\}|.$$

Since balls $B_{t_0}(\mathbf{c}|w)$ do not intersect, we have

$$|\cup_{\mathbf{c}\in\mathbf{C}} \{S_t(\mathbf{0}) \cap B_{t_0}(\mathbf{c})\}| = \sum_{w=\max\{d,t-t_0\}}^{t+t_0} A(w) I(w, t_0, t), \; t > t_0, \qquad (4)$$

where $A(w)$ is the number of codewords in $\mathbf{C}$ of Hamming weight w and we take into account that the intersection $S_t(\mathbf{0}) \cap B_{t_0}(\mathbf{c}|w)$ is empty for $|w - t| > t_0$.

The fraction $f_{t_0}(t)$ of undetected t–error patterns, $t > t_0$, may be found from (1),(4) and from the expression for the size of radius t sphere in Hamming space: $|S_t(\mathbf{0})| = \binom{N}{t}(q-1)^t$.

3. Fraction of undetected error patterns for RS-codes

The calculation of $f_{t_0}(t)$ uses the weight distribution of a code. Fortunately, the weight distribution for $RS-$codes is known (see, e.g., [3]):

$$A(w) = (q-1)\binom{N}{w} \sum_{j=0}^{w-d} (-1)^j \binom{w-1}{j} q^{w-d-j}, \; w \ge d. \qquad (5)$$

Then, the fraction of undetected error patterns for $RS-$codes is as follows:

$$f_{t_0}(t) = \frac{\sum_{w=\max\{d,t-t_0\}}^{t+t_0} A(w) I(w, t_0, t)}{|S_t(0)|}. \qquad (6)$$

For $t_0 = \frac{d-1}{2}$ the function $f_{t_0}(t)$ can be directly calculated. The results are presented in Fig.2 for some $RS-$codes.

We can simplify calculations to get an approximation for $f_{t_0}(t)$ for RS-codes. First, we note that

$$A(w) = (q-1)\binom{N}{w} q^{w-d} \sum_{j=0}^{w-d} \left(\frac{-1}{q}\right)^j \binom{w-1}{j} < (q-1)\binom{N}{w} q^{w-d}. \tag{7}$$

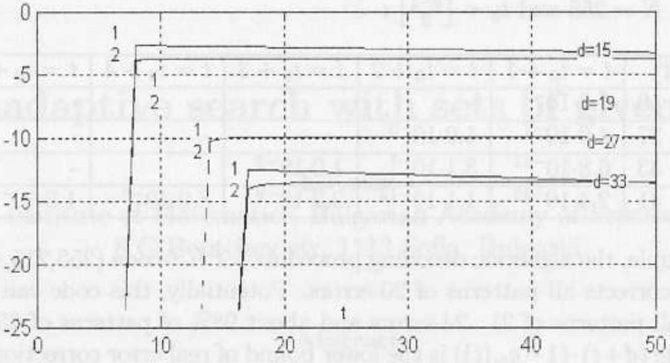
The upper bound in (7) is very tight for large enough N and w. For some natural suggestions the doubled single term in the sum (3) corresponding to $v = w - t_0$ and $r = t + t_0 - w$ gives a good enough bound for $I(w, t_0, t)$:

$$I(w, t_0, t) \le 2\binom{w}{t_0}\binom{t_0}{t + t_0 - w}(q-1)^{t+t_0-w}.$$

Hence, the following approximation is valid:

$$\begin{aligned} f_{t_0}(t) &= \frac{1}{|S_t(\mathbf{0})|} \sum_{w=\max\{d,t-t_0\}}^{t+t_0} A(w) I(w, t_0, t) \simeq \\ &\simeq \frac{2t_0(q-1)\binom{t+t_0}{t_0}\binom{N}{t+t_0} q^{t+t_0-d}}{\binom{N}{t}(q-1)^t} = \frac{2t_0(q-1)}{q^{d-t_0}} \cdot \frac{\binom{N-t}{t_0}}{\left(1 - \frac{1}{q}\right)^t}. \end{aligned} \tag{8}$$

If error vectors of equal weights are of the same probabilities, then $f_{t_0}(t)$ coincides with the probability $P_{er}(t)$ of undetected $t-$error patterns if decision regions are the radius t_0-balls in a $q-ary$ Hamming space. This probability together with the simplified bound (8) is shown in Fig. 2 as a function of t for $t_0 = \frac{d-1}{2}$, $N = 255$, $q = 256$ and minimal distances $d = \{15, 21, 27, 33\}$. For example, let $N = 255$, $d = 27$, $t_0 = 13$ and $t = 18$, then the fraction $f_{13}(18) = P_{er}(18)$ of undetected errors of weight 18, as well as for all other $t, 13 < t < 50$, is very close to 10^{-10}. The approximating bound is in the range $(2\times10^{-10}, 10^{-9})$.



-log10(Per(t)) for (N,K,d) PS-code with N=255, K=N-d+1.
1 - simplified upper bound, 2 - exact value

Fig.2 Probability of undetected errors

4. Fraction of uncorrected error patterns for RS-codes

The algebraic AD_{t_0} decoder corrects all error patterns with $wt(\mathbf{e}) \le t_0$. However, there exist many patterns of weight $t > t_0$ that could be corrected, e.g., by using the exhaustive search or list decoding. So, the error sequence $\mathbf{e}$, $wt(\mathbf{e}) = t$, not belonging to any ball $B_t(\mathbf{c}')$, $\mathbf{c}' \in C$, can be corrected since $t = d(\mathbf{r}, \mathbf{c}) < d(\mathbf{r}, \mathbf{c}')$. The fraction $g_{t_0}(t)$ of error sequences that cannot be corrected because they belong to the intersections $I(w, t, t)$ is given by (??), where $t_0 = t$:

$$g(t) \le \frac{1}{|S_t(\mathbf{0})|} \sum_{w=d}^{2t} A(w) I(w, t, t), \tag{9}$$

where

$$\begin{aligned} I(w, t, t) &= |S_t(\mathbf{0}) \cap B_t(\mathbf{c}|w)| = \\ &= \sum_{v=0}^{t} \sum_{r=w-2v}^{t-v} \binom{w}{v}\binom{w-v}{r}\binom{N-w}{t-v-r}(q-2)^r(q-1)^{t-r-v} \end{aligned} \tag{10}$$

We have the inequality in (9) because Hamming balls of radius greater than $\frac{d-1}{2}$ may have nonzero intersection. By this reason the bound works well only for relatively small values t.

The following table shows the upper bound (9) for some RS-codes with

$q = 256$, $N = 255$ and $t_0 = \left[\frac{d-1}{2}\right]$:

$g_{t_0}(t)$	$t = t_0 + 1$	$t = t_0 + 2$	$t = t_0 + 3$	$t = t_0 + 4$	$t = t_0 + 5$
$d = 19$	$1.3{\cdot}10^{-2}$	-	-	-	-
$d = 27$	$4.0{\cdot}10^{-7}$	$1.9{\cdot}10^{-3}$	-	-	-
$d = 33$	$6.8{\cdot}10^{-11}$	$3.1{\cdot}10^{-7}$	$1.0{\cdot}10^{-3}$	-	-
$d = 41$	$2.4{\cdot}10^{-16}$	$1.1{\cdot}10^{-12}$	$3.2{\cdot}10^{-9}$	$8.0{\cdot}10^{-6}$	$1.9{\cdot}10^{-2}$

For example, the algebraic decoding procedure for RS-code (255,215,41) over $GF(2^8)$ corrects all patterns of 20-errors. Potentially, this code can correct almost all patterns of 21...24-errors and about 98% of patterns of 25-errors. The value $(d+t)\cdot(1-g_{t_0}(t))$ is the lower bound of real error correction excess that can be achieved by decoding beyond the minimum distance bound.

References

[1] J.K.Wolf, A.M.Michelson and A.H.Leveque, "On the Probability of Undetected Error for Linear Block Codes", IEEE *Trans. on Commun.* COM-30, pp.317-324,1982.

[2] T.Kloeve, "A Lower Bound on the Probability of Undetected Error", Sixth Joint Swedish-Russian International Workshop on Information Theory, Aug.22-27, 1993, Moelle, Sweden, pp. 362-366.

[3] F.J.MacWilliams and N.J.A. Sloan, "The Theory of Error-Correcting Codes", North-Holland, 1977.

[4] R.E.Blahut, Theory and Practice of Error Control Codes, Addison-Wesley Publ.Company, 1984.

[5] V.D.Kolesnik, "Multi-Step Decoding in ISI Channels," This issue.

[6] V.D.Kolesnik, "On Dominant Error Sequences for Some ISI Channels with White Gaussian Noise", This issue.

Nonadaptive search with sets of given sum

E.Kolev
Institute of Matematics, Bulgarian Academy of Sciences
8 G.Bontchev str, 1113 Sofia, Bulgaria

Abstract

We consider nonadaptive search for unknown element x from the set $A = \{1, 2, 3, \ldots, 2^n\}$, $n \geq 3$. For fixed integer S the "questions" are of the form: does x belong to set B such that $B \subset A$ and the sum of the elements of B equals S?

1 Introduction

Consider the set $A = \{1, 2, 3, \ldots, 2^n\}$, $n \geq 3$ and let $x \in A$ be unknown element. For given natural number S we are allowed to ask whether x belongs to a set B such that $B \subset A$ and $\sum_{y \in B} y = S$. All questions are stated in advance, i.e. this is nonadaptive search. Call the set of subsets $B_1, B_2, \ldots B_m$ *good set of weight* S if for all $i = 1, 2, \ldots, m$ we have $\sum_{y \in B_i} = S$ and the unknown element x can be found using all of the elements of this set as questions. Note also, that since the sum of all elements of A is $2^{n-1}(2^n+1)$ then if $B_1, B_2, \ldots, B_m$ is good set of weight S then $\overline{B}_1, \overline{B}_2, \ldots, \overline{B}_m$ where $\overline{B}_i = A \setminus B_i$ is good set of weight $2^{n-1}(2^n+1) - S$.

There are two problems of interest:

Problem 1. Find all S for which a good set of weight S exists.

Problem 2. Find all S for which a good set of weight S and cardinality n exists.

2 Problem 1

It is clear that if for any $a, b \in A$ there exists question B such that $a \in B$ and $b \notin B$ or $a \notin B$ and $b \in B$ then the set of all possible questions is good. Thus, it is easy to prove the following proposition.

Proposition 1. Good set of weight S exists iff

$$S \in [2^n - 1, 2^{n-1}(2^n + 1) - (2^n - 1)].$$

3 Problem 2

For each set B consider characteristic vector $(a_1, a_2 \ldots a_{2^n})$ where $a_i = 1$ if $i \in B$ and $a_i = 0$ otherwise. Also, for the set $B_1, B_2, \ldots, B_n$ consider characteristic matrix G having as rows the characteristic vectors of $B_1, B_2, \ldots, B_n$. It is clear that if the set $B_1, B_2, \ldots, B_n$ is good set of cardinality n then the columns of G are all binary vectors of length n. Define weight of matrix G as

$$wt(G) = \sum_{i=1}^{2^n} wt(a_i).i$$

where a_i are vector columns of G.

Proposition 2. If a good set of weight S and cardinality n exists then

$$S \in \left[2^{2n-2} + 2^{n-2} - \frac{\binom{2n-1}{n-1}}{2}, 2^{2n-2} + 2^{n-2} + \frac{\binom{2n-1}{n-1}}{2} \right].$$

Proof: If a good set of weight S exists then the characteristic matrix G is of weight nS. Thus, to find minimal S we have to find minimum of $wt(G)$. It is clear that this minimum is achieved when the first column of G is all-one vector, next n columns are the vectors of weight $n-1$, and so on, the last column is all-zero vector. To find the weight of such matrix we have to compute

$$1 + \frac{(1+\binom{n}{n-1})(2+\binom{n}{n-1})}{2} + \frac{(1+\binom{n}{n-1}+\binom{n}{n-2})(2+\binom{n}{n-1}+\binom{n}{n-2})}{2} + \cdots$$

$$\cdots + \frac{(1+\binom{n}{n-1}+\binom{n}{n-2}+\cdots+\binom{n}{1})(2+\binom{n}{n-1}+\binom{n}{n-2}+\cdots+\binom{n}{1})}{2}$$

It is not difficult to find that the above sum equals $n\left(2^{2n-2} + 2^{n-2} - \frac{\binom{2n-1}{n-1}}{2}\right)$

and we get the assertion of the proposition. ◇

Denote by G_i the matrix with columns all binary vectors of weight i. Let $\overline{G_i}$ be the matrix such that $G + \overline{G_i} = J$ where J is the matrix all entries of which are ones. If $n = 2k+1$ then it is easy to see that

$$G = (G_{i_1} G_{i_2} \ldots G_{i_{n+1}}) \tag{1}$$

where $G_{i_1} G_{i_2} \ldots G_{i_{n+1}}$ is a permutation of the matrices G_0, $\overline{G}_0$, G_1, $\overline{G}_1, \ldots G_k$, $\overline{G}_k$ is characteristic matrix of good set. Note that if $wt(G) = nS$ then $wt(\overline{G}) = (2^{n-1}(2^n-1) - n)S$. In the case $n = 3$ for all integers in the interval [13, 23] of Proposition 2 there exists good set of the form (1). For example, if we denote by G_S the characteristic matrix of good set of weight S then

$$G_{13} = (\overline{G}_0 \overline{G}_1 G_1 G_0) = \begin{pmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

$$G_{14} = (\overline{G}_0 \overline{G}_1 G_0 G_1) = \begin{pmatrix} 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$G_{15} = (\overline{G}_1 \overline{G}_0 G_0 G_1) = \begin{pmatrix} 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$G_{16} = (\overline{G}_1 G_0 \overline{G}_0 G_1) = \begin{pmatrix} 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

$$G_{17} = (G_0 \overline{G}_0 \overline{G}_1 G_1) = \begin{pmatrix} 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$G_{18} = (\overline{G}_0 G_1 G_0 \overline{G}_1) = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \end{pmatrix}$$

$$G_{19} = \overline{G}_{17}, \quad G_{20} = \overline{G}_{16}, \quad G_{21} = \overline{G}_{15}, \quad G_{22} = \overline{G}_{14}, \quad G_{23} = \overline{G}_{13}.$$

Remark 1. It is easy to show that $\frac{\binom{2n-1}{n-1}}{2}$ is an integer iff n is not power of 2.

Remark 2. In the case $n = 5$ the interval from Proposition 2 is [201, 327]. For all but 14 values of $S \in [201, 327]$ a chtracteristic matrix of good set of weight S of the form (1) exists.

We conjecture that for all integers S in the interval from Proposition 2 there exists a good set of weight S.

Multistage decoding for error-free Elias construction.

Sergey Kovalev

St. Petersburg University on Aerospace Instrumentation,
B.Morskaja 67, St. Petersburg, Russia,
e-mail : s-kovalev@nwgsm.ru

Abstract. We consider a multistage decoding algorithm for the Elias iterative codes. In the proposed algorithm the decoding bit error probability decreases faster with increasing code length than in the original Elias decoder. Thus the code rate of the Elias codes can be increased significantly while keeping the same output bit error probability. Simulation results are presented.

1. Introduction .

In 1954, Elias [1] was first to propose a class of error correcting codes and their simple decoding algorithm, such that for given ε and R_b the decoding bit error probability was guaranteed to be less than ε while the code rate R >R_b. The construction of the Elias codes is quite simple. Let us consider an iteration of the extended Hamming codes of length 2^m, 2^{m+1}, 2^{m+2},..., 2^{m+s-1} (m>1). This s-dimensional iterative code possesses the code length N, the code rate R and the minimum distance D:

$$N= 2^{s(m+(s-1)/2)}\ ;\ R=\left(1-\frac{m+1}{2^m}\right)\left(1-\frac{m+2}{2^{m+1}}\right)...\left(1-\frac{m+s}{2^{m+s-1}}\right);\ D= 4^s\ ;$$

Elias considered the binary symmetric channel with transition probability p. Each dimension of the iterative code was decoded with correction of one error (or less). Decoding starts from the shortest length. The main Elias' result is based on the following proposition : if the average bit error probability in the code word of the first dimension is less than ½, then the decoding bit error probability tends to zero with increase of the dimension. The "threshold" value of p for each m was calculated precisely by Berlekamp[2]. The efficiency of the Elias construction can be increased by choosing stronger codes in the first several iterations [3].

In this paper we investigate an improvement of the Elias construction using a more complicated decoding algorithm which is closed to a simplified turbo decoding algorithm for iterative block codes.

2. The decoding algorithm.

Consider a set of binary linear block codes $g_i\,(n_i,\, k_i,\, d_i)$, $i=1,2,...,s$, of length n_i, dimension k_i and minimum Hamming distance d_i . Let us denote the m-dimensional iterative code by $G(N,K,D)=g_1\otimes g_2\otimes...\otimes g_s$. The main code parameters, such as the iterative length N , the code rate R, the information dimension K and the minimum distance D are equal to the product of the corresponding parameters of the iterated codes.

Decoding of iterative codes is based on decoding of conventional block codes in each dimension using information from decoders in other dimensions. The basic operation of the multistage decoding scheme is recalculation of the hard decisions and symbol reliabilities by a decoder of the code g_i in the dimension i using information from the previous dimension or the previous stage.

We assume that the s-dimensional iterative code is used for transmission over a memoryless binary symmetric channel (BSC) with transition probability p. Let $X \in$ G, $X_i\in\{-1,+1\}$ be transmitted codeword and Y, $Y_i\in\{-1,+1\}$ be the corresponding output of BSC. For the multistage decoding we introduce the following notations :

Y^j denotes s-dimensional matrix of soft decisions for j-th stage; $Y^0=Y$;
Z^j denotes s- dimensional matrix of hard decisions for Y^j :
$Z^j_i= -1$ if $Y^j_i<=0$; $Z^j_i= +1$ if $Y^j_i>0$; $Z^0= Y^0$.

Our algorithm consists of the following steps:

1. Generate the initial matrices Y^0 and Z^0.
2. $j=1$.
3. $i=1$.
4. Extract all vectors y^{j-1} from Y^{j-1} in the dimension I, and decode independently each y^j by a decoder for g_i . The decoder recalculates y^j using y^{j-1} according to a decision rule. The new vectors y^j update Y^{j-1} in each dimension immediately and form Y^j after decoding in the last dimension.
5. $i=i+1$. If $i\leq s$ then go to 4.
6. The result of the decoding stage is an updated matrix Y^j.
7. If ($Z^j \in G$) or ($j \geq$ *prescribed upper bound*) then go to 9.
8. $j=j+1$. Go to 3.
9. The end of decoding. The final result is Z^j.

Different approaches to the "soft in-soft out" decoding of the conventional code yield different decoding algorithms for the iterated codes. Let us consider a simple decoding decision rule for the block codes which uses symbol reliabilities and do not use complex float-point operations such as *exp()*, *log()*, multiplication and division.

Let $\boldsymbol{e}$ be a binary error vector such that $e_i\in\{0,+1\}$, $(z_0(-1)^{e_0},z_1(-1)^{e_1},...)\in g$;
$W(y,e)=\sum_{i|e_i=1} y_i$ be the generalized weight of the vector $\boldsymbol{e}$. Let us consider the calculation of y^j_k using $y^{j-1}$and z^{j-1}. For each k we find two error vectors $\boldsymbol{e}^0$ and $\boldsymbol{e}^1$ such that $e^0_k=0$, $e^1_k=1$, $\boldsymbol{e}^0$ has minimum generalized weight among all error vectors with $e_k=0$, $\boldsymbol{e}^1$ has minimum generalized weight among all error vectors with $e_k=1$. For the Hamming codes of moderate lengths these vectors can be found using simple known algorithms [5].
We propose the following decision rule [4]:

$$y^j_k= y^0_k-\ z^{j-1}_k\,(W(y^{j-1},e^1)-\ W(y^{j-1},e^0))/2; \qquad (1)$$

3.Simulation results.

The described above algorithm was checked by computer simulations for several Elias codes obtained from the extended Hamming codes of lengths 4,..,64. The simulation results for multistage decoding are shown in the following Table. More than 10^8 channel symbols were processed for each simulation. The number of stages was restricted to 10. Results for the Elias decoder were adopted from [2].

P_{output}	m=3, P_{input} =0.08	m=4, P_{input} = 0.04	m=5, P_{input} =0.019
Elias decoding , s=2	0.019	0.0098	0.0049
Multistage decoding, s=2	0.002	0.00038	0.000071
Elias decoding, s=3	0.0098	0.0049	0.0024
Multistage decoding, s=3	0*	-	-

*- 10^5 code words were simulated. It is more convenient to estimate the code rate gain. For example, multistage decoding of (32*32)-code is equivalent to the Elias decoding of (16*32)-code with P_{input} = 0.04, P_{output} = 0.01. At the same time, $R_{(16*32)}$ =0.56, $R_{(32*32)}$=0.66 . Other example :
G1=(8*16*32) ,G2=(16*16). The Elias decoding of G1 code for P_{input} = 0.08, P_{output} = 0.01 has the same performance as multistage decoding of G2 code. Note, N1=4096, N2=256, R1=0.33, R2=0.49.

4.Conclusion remarks.

As follows from the simulation results, the proposed algorithm is essentially better than the Elias decoder. Moreover, the decision rule (1) is not much worse than the classic symbol-by-symbol MAP-decoding which usually supposes an analysis of each codeword of the code (or the dual code) and requires complicated floating point operations. The computer simulations implemented for AWGN channel and iterated extended Hamming codes (16*16*16) show almost the same results for rule (1) and for MAP-decoding. For $E_b/N_0 = 1.5dB$ the bit error probability $P_b = 2*10^{-5}$ was obtained for the both decoding rules [4,6]. Other comparison was done for the single parity check (SPC) iterative codes. Simulation results for (8*8*8*8*8) iterative SPC code and MAP decoding rule obtained $P_b = 10^{-5}$ for $E_b/N_0 = 2dB$ [7], whereas the rule (1) achieves only $P_b = 10^{-4}$. Thus, the proposed algorithm may be useful for communication systems with restricted complexity of the decoding scheme.

References.

[1] P. Elias. Error-free coding. IRE Trans. Inform. Theory, vol IT-4, pp29-37,Sept.1954.

[2] E.R.Berlekamp. Algebraic coding theory. New York: McGraw-Hill, 1968.

[3] A.B.Cooper, W.C.Gore. Iterated codes with improved performance. IEEE Trans. Inform. Theory, vol. 24, pp. 116-118, 1978.

[4] S.Kovalev An improved decoding procedure for iterative codes, Proc.7-th Joint Swedish-Russian Int. Workshop on Information Theory, pp.158-160, St.Petersburg, Russia, 1995.

[5]J. Snyders, "Reduced lists of error patterns for maximum likelihood soft decoding", IEEE Trans. Inform. Theory, vol. IT-37, pp. 1194-1200, 1991.

[6] S.Fedorenko and V.Kolesnik, Multi-Step Decoding of the Iteration of Hamming Codes, Proc.7-th Joint Swedish-Russian Int. Workshop on Information Theory, pp.80-83, St.Petersburg, Russia, 1995.

[7] D.Rankin, T.A.Gulliver. Single Parity Product Codes. IEEE Trans. Comm., vol. 49, no. 8, pp. 1354-1362, 2000.

The Kerdock Codes and Separating Systems

A.Krasnopeev. McMaster University, Gamilton, Canada.
Yu.L.Sagalovich.
Institute for Information Transmission Problems,
Russian Academy of Sciences.
19, Bol. Karetny str., 101447, Moscow GSP-4 , Russia. E-mail: sagal@iitp.ru

Abstract

It is shown that one-shortened Kerdock code is (2, 2)– and (2, 1)– separating system. The parameters θ, θ^* of the separation obtained.

I. $(2, 2, \theta)$-separating systems

Let us consider the binary code C of length n. The code C is $(2, 2, \theta)$–separating system $(-s.s)$ [1, 2, 3], if the arbitrary ordered quadruple of distinct vectors

$$\begin{array}{l} s_1 = (a_1^{(1)}, a_2^{(1)}, \ldots, a_i^{(1)}, \ldots, a_n^{(1)}) \\ s_2 = (a_1^{(2)}, a_2^{(2)}, \ldots, a_i^{(2)}, \ldots, a_n^{(2)}) \\ s_3 = (a_1^{(3)}, a_2^{(3)}, \ldots, a_i^{(3)}, \ldots, a_n^{(3)}) \\ s_4 = (a_1^{(4)}, a_2^{(4)}, \ldots, a_i^{(4)}, \ldots, a_n^{(4)}) \end{array} \tag{1}$$

contains not less than θ so-called regular columns of the form

$$(aa\bar{a}\bar{a})^T.$$

Let n_x, n_y, n_u be the numbers of columns respectively

$$(a\bar{a}a\bar{a})^T \quad \text{or} \quad (a\bar{a}\bar{a}a)^T,$$

$$(a\bar{a}aa)^T \quad \text{or} \quad (a\bar{a}aa)^T,$$

$$(aa\bar{a}a)^T \quad \text{or} \quad (aaa\bar{a})^T.$$

$$a = 0,\ 1; \quad \bar{a} = 1,\ 0.$$

Denote

$$s = ((s_1 + s_2 + s_3 + s_4) \bmod 2, \tag{2}$$

$w(s)$ as the weight of vector s and $d(s_i, s_j) = d_{ij}$ as the distance between vectors s_i, s_j respectively.

It is easy to see, that

$$w(s) = n_y + n_u, \tag{3}$$

$$d_{12} = n_x + n_y, \quad d_{34} = n_x + n_u, \tag{4}$$

$$4\theta = d_{13} + d_{14} + d_{23} + d_{24} - 2d_{12} - 2d_{34} + 2n_x. \tag{5}$$

We obtain from (4) and (5)

$$4\theta = d_{13} + d_{14} + d_{23} + d_{24} - d_{12} - d_{34} - (n_y + n_u). \tag{6}$$

If d and D are respectively the minimal and maximal distances of code C, then

$$4d - 2D - (n_y + n_u) > 0 \tag{7}$$

is the sufficient condition to a code to form a $(2,2,\theta)$-s.s. for some θ.

Let us consider the one-shortened Kerdock ($\mathbf{K}(2^m-1, 2^{2m-1})$) code [4] of length $n = 2^m - 1$ with 2^{2m-1} codewords, m is even.

It is easy to prove that the list of distances of the $\mathbf{K}(2^m-1, 2^{2m-1})$ includes only following values

$$0, \quad 2^{m-1} - 2^{(m-2)/2}, \quad 2^{m-1}, \quad 2^{m-1} + 2^{(m-2)/2}. \tag{8}$$

The following lemmas are obvious:

Lemma 1. *The* $\mathbf{K}(2^m-1, 2^{2m-1})$ *is the subcode of the one-shortened second-order Reed-Muller code, because the Kerdock code is the subcode of the second-order Reed-Muller code.*

Lemma 2. *Maximal distance of the one-shortened second-order Reed-Muller code is equal* $3 \cdot 2^{m-2}$.

Let the vectors in quadruple (1) belong to $\mathbf{K}(2^m-1, 2^{2m-1})$

The Kerdock code is not linear code. Therefore vector s in (2) might not belong to $\mathbf{K}(2^m-1, 2^{2m-1})$, and $w(s)$ in (3) is not necessary contained in the list (8). But vectors (1) belong to one-shortened second-order Reed-Muller code (see Lemma 1). Hence by Lemma 2 $n_y + n_u \leq 3 \cdot 2^{m-2}$. Thus, we have taking into account the equations (6) and (7) and Lemma 2: The $\mathbf{K}(2^m-1, 2^{2m-1})$ is $(2,2,\theta)$–s.s. when

$$4\theta \geq 4(2^{m-1} - 2^{(m-2)/2}) - 2(2^{m-1} + 2^{(m-2)/2}) - 3 \cdot 2^{m-2} = 2^{m-2} - 3 \cdot 2^{m/2} > 0. \tag{9}$$

This inequality is fulfilled by $m \geq 8$. Eq. (9) gives for $m = 8$ and 10 respectively $\theta \geq$ 4 and 40, whereas computing analysis gives respectively $\theta \geq 12$ and 48.

II. $(2,1,\theta^*)$-separating systems

The code C is $(2,1,\theta)$–separating system ($-s.s$), if the arbitrary ordered triple of distinct vectors

$$\begin{aligned} s_1 &= (a_1^{(1)}, a_2^{(1)}, \ldots, a_i^{(1)}, \ldots, a_n^{(1)}) \\ s_2 &= (a_1^{(2)}, a_2^{(2)}, \ldots, a_i^{(2)}, \ldots, a_n^{(2)}) \\ s_3 &= (a_1^{(3)}, a_2^{(3)}, \ldots, a_i^{(3)}, \ldots, a_n^{(3)}) \end{aligned}$$

contains not less than θ^* regular columns of the form

$$(a\bar{a}\bar{a})^T.$$

Let us denote n_y', n_u' the number of columns respectively

$$(a\bar{a}a)^T \quad \text{or} \quad (\bar{a}a\bar{a})^T,$$

$$(aa\bar{a})^T \quad \text{or} \quad (\bar{a}\bar{a}a)^T.$$

$$a = 0,\ 1; \quad \bar{a} = 1,\ 0.$$

It is easy to see, that

$$d_{12} = \theta^* + n_y', \quad d_{13} = \theta^* + n_u', \quad d_{23} = n_y' + n_u'.$$

Hence

$$2\theta^* = d_{12} + d_{13} - d_{23}.$$

Finally, the $\mathbf{K}(2^m-1, 2^{2m-1})$ code is $(2,1,\theta^*)$–s.s. when

$$2\theta^* \geq 2(2^{m-1} - 2^{(m-2)/2}) - 2^{m-1} - 2^{(m-2)/2} = 2^{m-1} - 3 \cdot 2^{(m-2)/2} > 0. \tag{10}$$

This inequality is fulfilled by $m \geq 4$. The values of Eq. (10) coincide with the results of computing analysis for $m = 4, 6, 8, 10$.

III. Conclusion

Now we can formulate the problem of existence of some linear subcode $\mathbf{L}$ of second-order Reed-Muller code ($\mathbf{RM}(m,2)$), such that Kerdock code and more over the $\mathbf{K}(2^m-1, 2^{2m-1}) \subset \mathbf{L} \subset \mathbf{RM}(m,2)$. With this assumption is we have the opportunity for improving of parameter θ of the (2, 2)-separation.

References

1. Yu. L. Sagalovich, "A method of increasing the reliability of finite automata," *Probl. Peredachi Inf.* **1**, No. 2, 27–35 (1965).

2. Yu. L. Sagalovich, *State Encoding and Reliability of Automata* [in Russian], Svyaz', Moscow (1975).

3. Yu. L. Sagalovich,"Separating Systems" *Probl. Peredachi Inf.* **30**, No. 2, 14–35 (1994).

4. F.J. MacWilliams, N.J.A. Sloane, *The Theory of Error-Correcting Codes, Part II, North Holland* (1977).

On decomposition of $(n, 4^{n-1}, 2)_4$ MDS codes and double-codes [1]

D. S. Krotov [2]

The union of two disjoint $(n, 4^{n-1}, 2)$ MDS codes in $\{0,1,2,3\}^n$ is a complete double-code. If adjacency graph of the complete double-code is not connected then the double-code can be decomposed onto complete double-codes of smaller dimensions and the $(n, 4^{n-1}, 2)$ MDS codes are also decomposed.

Let $A \overset{df}{=} \{0,1,2,3\}$ and n be a natural number. For $\bar{x} = (x_1, x_2, ..., x_n)$ we use the following notations: $\bar{x}^{(k)}\#y \overset{df}{=} (x_1, ..., x_{k-1}, y, x_{k+1}, ..., x_n)$ and $\bar{x}^{(k_1,k_2,...,k_t)}\#(y_1, y_2, ..., y_t) \overset{df}{=} (...((\bar{x}^{(k_1)}\#y_1)^{(k_2)}\#y_2)...)^{(k_t)}\#y_t$.

Edge (*i-edge*) of A^n is a set of four elements of A^n which differ in only one (ith) coordinate. Let $edge_i(\bar{x})$ denote the i-edge containing $\bar{x}$ from A^n. If $S \subset A^n$ then $edge_i(S) \overset{df}{=} \bigcup_{\bar{x}\in S} edge_i(\bar{x})$ and $\Psi_{i,j;\bar{x}} S \overset{df}{=} \{(b,c) \in A^2 : \bar{x}^{(i,j)}\#(b,c) \in S\}$.

A set $S \subset A^n$ is called a $(n, 4^{n-1}, 2)_4$ *MDS code* if each edge of A^n contains exactly one element from S. A set $S \subset A^n$ is called a *double-code* if each edge of A^n contains zero or two elements from S. A double-code $S \subset A^n$ is called *complete* if each edge of A^n contains exactly two elements from S. If a double-code is a subset of some complete double-code we call it *complemented*. If a double-code is complemented and nonempty and can not be partitioned onto more than one nonempty double-codes we will call it *simple*.

Note. The union of two disjoint $(n, 4^{n-1}, 2)_4$ MDS codes is always a complete double-code. The reverse statement is not true for $n \geq 3$.

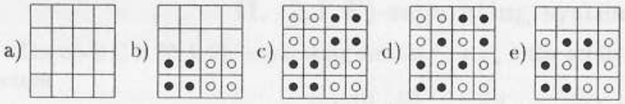


Figure 1: The black circles identify the elements of double-codes in A^2. The results of operation $\backslash_1$ are identified by the white circles.

Example. Figure 1 shows all double-codes in A^2 up to permutations of rows and columns. The double-codes a)-d) are complemented and e) is not. The double-codes c) and d) are complete. The double-codes b) and d) are simple.

Proposition 1. *(1) The supplement of a complete double-code in A^n is a complete double-code.*
(2) A double-code $S \subset A^n$ is complete if and only if $|S| = |A^n|/2 = 2^{2n-1}$.
(3) Let $\bar{S} = \{S_1, ..., S_k\}$ be a partition of complemented double-code S onto simple ones, S_0 be a simple double-code and $S_0 \subseteq S$. Then $S_0 \in \bar{S}$.

Proof. (1) follows from the definition of complete double-code.

(2) is obvious if one considers the partition of A^n onto i-edges where i is fixed.

(3) For arbitrary double-code $S' \subset A^n$ we consider the graph $G(S')$ with vertex set S', two vertex being adjacent if and only if they differ in exactly one coordinate. The degree of $G(S')$ is n. To a simple double-code corresponds a connected graph. To the partition of S onto simpe double-codes corresponds the partition of $G(S)$ onto connected components. The subgraph $G(S_0)$ of $G(S)$ is connected and has degree n. Consequently, it contains all edges of $G(S)$ beginning in S_0. It means that $G(S_0)$ is a connected component of $G(S)$. △

For $S \subset A^n$ and $i \in \{1, ..., n\}$ we denote $\backslash_i S \overset{df}{=} edge_i(S)\backslash S$.

Proposition 2. *Let $S, S' \in A^n$ be double-codes and $i, i' \in \{1, ..., n\}$. Then*
(1) $S \cap \backslash_i S = \emptyset$;
(2) $\backslash_i \backslash_i S = S$;
(3) $|S| = |\backslash_i S|$;
(4) $S \subseteq S'$ if and only if $\backslash_i S \subseteq \backslash_i S'$;
(5) S is complete if and only if $\backslash_i S$ is complete double-code;
S is complete iff $\backslash_i S = A^n \backslash S$; (6) S is complemented iff $\backslash_i S$ is complemented double-code;
(7) S is simple if and only if $\backslash_i S$ is simple double-code;
(8) if S is simple then either $\backslash_i S = \backslash_{i'} S$ or $\backslash_i S \cap \backslash_{i'} S = \emptyset$;
(9) if S is complemented then $\backslash_i \backslash_{i'} S = \backslash_{i'} \backslash_i S$;
(10) S is complete if and only if $|S| > 0$ and $\backslash_j \backslash_{j'} S = S$ for each $j, j' \in \{1, ..., n\}$.

Proof. (1) is clear.

The set $edge_i(S) = edge_i(\backslash_i S) = S \cup \backslash_i S$ can be partitioned onto i-edges. Each edge of the partition has two elements from S and the other two from $\backslash_i S$. It follows (2) and (3).

(4). Let $S \subseteq S'$. Then $edge_i(S) \subseteq edge_i(S')$. Each edge $edge_i(\bar{x})$, $\bar{x} \in edge_i(S)$, contains two elements from S and the other two from $\backslash_i S$. They are also elements from S' and $\backslash_i S'$ respectively. So, each element from $\backslash_i S$ is in $\backslash_i S'$. The reverse statement is proved by the same way.

(5) follows from (1), (3), and Proposition 1(2).

(6). First we will show that $\backslash_i S$ is a double-code. Let $edge_j(\bar{x})$ be arbitrary edge, where $j \in \{0, ..., n\}$ and $\bar{x} = (x_1, ..., x_n) \in A^n$. If $j = i$ then $|edge_j(\bar{x}) \cap S| = |edge_j(\bar{x}) \cap \backslash_i S| \in \{0, 2\}$. Let $j \neq i$. It is clear that $\Psi_{i,j;\bar{x}} S$ is a double-code in A^2 and $\Psi_{i,j;\bar{x}} \backslash_i S = \backslash_1 \Psi_{i,j;\bar{x}} S$. Furthermore, the fact that S is complemented implies that $\Psi_{i,j;\bar{x}} S$ is complemented too. It is easy to check (Fig. 1(a-d)) that $\backslash_1 \Psi_{i,j;\bar{x}} S$ is a double-code. Consequently, $|edge_j(\bar{x}) \cap \backslash_i S| = |edge_2(x_i, x_j) \cap \backslash_1 \Psi_{i,j;\bar{x}} S| \in \{0, 2\}$, and $\backslash_i S$ is a double-code by the definition. The fact that $\backslash_i S$ is complemented follows from (5) and (4).

(7) By (6) we may assume that S and $\backslash_i S$ are complemented double-codes. Let S be non-simple. And let $S = S_1 \cup S_2$, where S_1 and S_2 are disjoint nonempty double-codes. Since S_1 and S_2 are complemented, then $\backslash_i S_1$ and $\backslash_i S_2$ are also complemented double-codes. It follows from (4) and (2) that $\backslash_i S_1 \cap \backslash_i S_2 = \backslash_i (S_1 \cap S_3)$. Therefore $\backslash_i S_1$ and $\backslash_i S_2$ are disjoint and the double-code $\backslash_i S = \backslash_i S_1 \cup \backslash_i S_2$ is not simple, which proves the statement.

(8) Let $S \subseteq S''$, where S'' is a complete double-code. It follows from (4) that $\backslash_i S \subseteq \backslash_i S'' = A^n \backslash S''$. On the other hand $\backslash_{i'} S \subseteq \backslash_{i'} S'' = A^n \backslash S''$. By Proposition 1(3) the simple double-codes $\backslash_i S$ and $\backslash_{i'} S$ belong to the partition of the complete double-code $A^n \backslash S''$ onto simple double-codes. Therefore they are either coincident or disjoint.

(9) One need check only that $\backslash_i \backslash_{i'} S_{i,i';\bar{x}} = \backslash_{i'} \backslash_i S_{i,i';\bar{x}}$ for each $\bar{x} \in A^n$, where $S_{i,i';\bar{x}} \overset{df}{=} \{x^{(i,i')}\#(b,c) : b, c \in A\}$. Equivalently, $\backslash_1 \backslash_2 \Psi_{i,i';\bar{x}} S = \backslash_2 \backslash_1 \Psi_{i,i';\bar{x}} S$ for all $\bar{x} \in A^n$. The last can be checked directly taking into account that $\Psi_{i,i';\bar{x}} S$ is a complemented double-code (Fig. 1a-d).

(10) If S is a complete double-code then $|\Psi_{i,j;\bar{x}} S| = 8$ (Fig. 1(c,d)) for all $i, j \in \{1, ..., n\}$, $i \neq j$, and $\bar{x} \in A^n$. Therefore, $\backslash_i \backslash_j S = S$.

The reverse statement. For the contradiction let S be non-complete. Then there exist $\bar{x}, \bar{z} \in A^n$ such that $edge_1(\bar{x}) \cap S \neq \emptyset$ and $edge_1(\bar{z}) \cap S = \emptyset$. Consider the sequence $\bar{x} = \bar{x}^0, \bar{x}^1, ..., \bar{x}^n = \bar{z}$ such that $\bar{x}^{j-1}$ and $\bar{x}^j_s$ coincide in all coordinates exept jth. There exists $j \in \{2, ..., n\}$ such that $edge_1(\bar{x}^{j-1}) \cap S \neq \emptyset$ and $edge_1(\bar{x}^j) \cap S = \emptyset$. Then $|\Psi_{1,j;\bar{x}} S| \notin \{0, 8\}$ (Fig. 1(a,c,d)), that contradicts to $\backslash_1 \backslash_j S = S$. △

Let S be a complete double-code in A^n, and $S_0 \subseteq S$ be a simple double-code. We will say that i and i' from $\{1, \ldots, n\}$ are *equivalent*, or $i \sim i'$, if $\backslash_i \backslash_{i'} S_0 = S_0$. It follows from Proposition 2(2,7,9) that $\sim$ is an equivalence relation. The sets $S_{\bar{y}}$, $\bar{y} \in \{0,1\}^k$ are defined by the equalities $S_{\bar{0}} \overset{df}{=} S_0$ and $S_{\bar{y}\oplus e_j} \overset{df}{=} \backslash_{i_j,1} S_{\bar{y}}$. This definition is correct by Proposition 1(2,9).

[1] This research was supported by the RFBR under the grant 00-01-00822

[2] Sobolev Institute of Mathematics, pr. Koptyuga 4, Novosibirsk 630090, Russia, e-mail: dkrotov@mail.ru

Let $K_1 = \{i_{1,1}, i_{1,2}, ..., i_{1,n_1}\}, K_2 = \{i_{1,1}, i_{1,2}, ..., i_{1,n_1}\}, \ldots, K_k = \{i_{k,1}, i_{k,2}, ..., i_{k,n_k}\}$ be the equivalence classes of $\sim$. From Proposition 2(10) we have

Proposition 3. *The double-code S is simple if and only if $k = 1$.*

The next proposition is also a corollary of Proposition 2.

Proposition 4. *(1) For each $\bar{y} \in \{0,1\}^k$ the set $S_{\bar{y}}$ is a simple double-code. If $\bar{y}$ is even then $S_{\bar{y}} \subseteq S$. If $\bar{y}$ is odd then $S_{\bar{y}} \subseteq A^n \backslash S$.*
(2) For each $\bar{y}$ from $\{0,1\}^k$ the equality $S_{\bar{y}} = \backslash_{i'} \backslash_{i''} S_{\bar{y}}$ is true if and only if $i', i'' \in K_j$ for some $j \in \{1, ..., k\}$.
(3) $S = \bigcup_{\bar{y} \in \overline{\{0,1\}^k}} S_{\bar{y}}$ and $A^n \backslash S = \bigcup_{\bar{y} \in \{0,1\}^k \backslash \overline{\{0,1\}^k}} S_{\bar{y}}$, where $\overline{\{0,1\}^k}$ is the set of the even elements of $\{0,1\}^k$.

Let $\sigma = \chi_S : A^n \to \{0,1\}$ be the characteristic function of S and $\sigma_j(y_1, ..., y_{n_j}) \overset{df}{=}$ $\overset{df}{=} \sigma(\bar{0}^{(i_1, ..., i_{n_j})} \# (y_1, ..., y_{n_j}))$ $(j = 1, ..., k)$ be its subfunctions.

Proposition 5. *For every i', i'', which belong to different equivalence classes, for every $\bar{x} \in A^n$ and for every $a', a'' \in A$ it holds*

$$\sigma(\bar{x}) \oplus \sigma(\bar{x}^{(i')} \# a') \oplus \sigma(\bar{x}^{(i'')} \# a'') \oplus \sigma(\bar{x}^{(i', i'')} \# (a', a'')) = 0.$$

Proof. Let $S' \subseteq S$ be a simple double-code such that $S' \cup edge_{i'}(\bar{x}) \neq \emptyset$ (Proposition 2 implies its existance). Let us consider the sets $S^2 = \Psi_{i', i''; \bar{x}} S$ and $S'^2 = \Psi_{i', i''; \bar{x}} S'$. Since $\backslash_{i'} \backslash_{i''} S' \neq S'$, by Proposition 2(8) it is true that $\backslash_{i'} \backslash_{i''} S' \cap S' = \emptyset$ and, consequently, $\backslash_1 \backslash_2 S'^2 \cap S'^2 = \emptyset$. Therefore S^2 corresponds to case c) of Figure 1 up to rows and columns permutations. The statement follows from the obvious identity

$$\chi_{S^2}(b', b'') \oplus \chi_{S^2}(c', b'') \oplus \chi_{S^2}(b', c'') \oplus \chi_{S^2}(c', c'') = 0, \quad \forall b', b'', c', c'' \in A. \quad \triangle$$

Proposition 6. *Let $P = \{p_1, ..., p_m\} \subseteq \{1, ..., n\}$, $Q = \{q_1, ..., q_r\} \subseteq \{1, ..., n\}$, and each K_j is disjoint with at least one of P and Q. Let $\bar{x} = \{x_1, ..., x_n\} \in A^n$. Then*

$$\sigma(\bar{0}) \oplus \sigma(\bar{0}^{(p_1, ..., p_m)} \# (x_{p_1}, ..., x_{p_m})) \oplus \sigma(\bar{0}^{(q_1, ..., q_r)} \# (x_{q_1}, ..., x_{q_r}))$$
$$\oplus \sigma(\bar{0}^{(p_1, ..., p_m, q_1, ..., q_r)} \# (x_{p_1}, ..., x_{p_m}, x_{q_1}, ..., x_{q_r})) = 0.$$

Proof. By Proposition 5 we have

$$\bigoplus_{s=1}^{m} \bigoplus_{t=1}^{r} \Big(\sigma(\bar{0}^{(p_1, ..., p_{s-1}, q_1, ..., q_{t-1})} \# (x_{p_1}, ..., x_{p_{s-1}}, x_{q_1}, ..., x_{q_{t-1}}))$$
$$\oplus \; \sigma(\bar{0}^{(p_1, ..., p_{s-1}, q_1, ..., q_t)} \# (x_{p_1}, ..., x_{p_{s-1}}, x_{q_1}, ..., x_{q_t}))$$
$$\oplus \; \sigma(\bar{0}^{(p_1, ..., p_s, q_1, ..., q_{t-1})} \# (x_{p_1}, ..., x_{p_s}, x_{q_1}, ..., x_{q_{t-1}}))$$
$$\oplus \; \sigma(\bar{0}^{(p_1, ..., p_s, q_1, ..., q_t)} \# (x_{p_1}, ..., x_{p_s}, x_{q_1}, ..., x_{q_t})) \Big) = 0.$$

Collecting similar terms proves the statement. $\triangle$

Proposition 7. *For each $\bar{x}$ from A^n it holds*

$$\sigma(x) = \bigoplus_{j=1}^{k} \sigma_j(x_{i_{j,1}}, x_{i_{j,2}}, ..., x_{i_{j,l_j}}) \oplus ((k-1) \bmod 2) \sigma(\bar{0}).$$

Proof. Let $\tilde{x}_j = (x_{i_{j,1}}, x_{i_{j,2}}, ..., x_{i_{j,n_j}})$. Without lost of generality we can assume that $(i_{1,1}, ..., i_{1,l_1}, i_{2,1}, i_{2,2}, ..., i_{1,n_2}, \ldots, i_{k,n_k}) = (1, \ldots, n)$. So, $\bar{x} = (\tilde{x}_1, \tilde{x}_2, ..., \tilde{x}_k)$. By Proposition 6 we have

$$\bigoplus_{j=2}^{k} \sigma(\bar{0}) \oplus \sigma(\tilde{x}_1, ..., \tilde{x}_{j-1}, 0, ..., 0) \oplus \sigma(0, ..., 0, \tilde{x}_j, 0, ..., 0) \oplus \sigma(\tilde{x}_1, ..., \tilde{x}_{j-1}, \tilde{x}_j, 0, ..., 0) = 0$$

and the statement is proved by collecting similar terms. $\triangle$

Proposition 8. *For each $j \in \{1, ..., k\}$ the function σ_j is the characteristic function of a simple complete double-code.*

Proof. σ_j is a subfunction of σ. Therefore it is the characteristic function of some complete double-code S_j. If S_j is not simple then Propositions 3-7 lead to contradiction with the fact that K_j is an equivalence class. $\triangle$

The following theorem is a corollary of Propositions 7 and 8.

Theorem 1. (decomposition of complete double-codes) *a) The characteristic function χ_S of a complete double-code S is representable in the form*

$$\chi_S(\bar{x}) = \bigoplus_{j=1}^{k} \chi_{S_j}(\tilde{x}_j) \oplus \sigma_0,$$

where $\tilde{x}_j = (x_{i_{j,1}}, ..., x_{i_{j,n_j}})$ are disjoint collections of variables from $\bar{x}$, S_j are simple complete double-codes in A^{n_j} and $s_0 \in \{0,1\}$.
b) S is a union of 2^{k-1} simple double-codes.

The next theorem gives the representation of $(n, 4^{n-1}, 2)_4$ MDS codes, which is based on the decomposition of complete double-codes presented in Theorem 1.

Theorem 2. (decomposition of $(n, 4^{n-1}, 2)_4$ MDS codes) *Let S be a complete double-code in A^n, C be a $(n, 4^{n-1}, 2)_4$ MDS code, and $C \subset S$ then*

$$C = \{(x_1, \ldots, x_n) \,|\, (x_{i_{j,1}}, x_{i_{j,2}}, ..., x_{i_{j,n_j}}, y_j) \in C_j, \; j = 1, ..., k; \; (y_1, ..., y_k) \in C_0\},$$

where C_j is a $(n_j + 1, 4^{n_j}, 2)_4$ MDS code for $j = 1, ..., k$, C_0 is a $(k, 4^{k-1}, 2)_4$ MDS code, and $k, n_j, i_{j,s}$ are specified by Theorem 1.

Several Properties of Public-Key Cryptosystems based on quadratic orders [*]

Kshevetsky A.S. [†]

Abstract. A possibility of construction of a signature based on the public-key cryptosystem with quadratic decryption time [2] is analysed.

1 Introduction

J. Buchmann and H. C. Williams suggested to use quadratic orders for cryptographic purposes. They applied a Diffie-Hellman like key exchange cryptosystem and a RSA like public-key cryptosystem as described in [5] and [4]. Both schemes have cubic time of encryption and decryption. Here and below calculation time means time, measured in bit operations and depending on the bit length of input parameters. Then D. Hühlein, M. Jacobson, Jr. and S. Paulus suggested an ElGamal like public-key cryptosystem in [3] with encryption and decryption of cubic time and pointed out a way for building signature, based on this cryptosystem. And finally, S. Paulus and T. Takagi proposed a new public-key cryptosystem in [2] with cubic encryption time and quadratic decryption time as an alternative to RSA and ElGamal cryptosystems.

We showed, that the size of a cipher text is more than six times greater than the size of a message text in the cryptosystem [2].

There is an open question about signature based on the cryptosystem with quadratic decryption time. It is proved in the paper, that there is no way to create signature schemes based on that cryptosystem.

The section 2 of this paper reminds common issues of quadratic field theory. The section 3 contains description and analysis of proposed cryptosystems based on quadratic orders. Analysis of signatures is in the section 4.

2 Imaginary Quadratic Fields

Imaginary quadratic field is the extension of the field of rational numbers $\mathbb{Q}$ with an element $\sqrt{\delta}$, δ being in $\mathbb{Z}^-$ and $-\delta$ being not a full square. Elements of quadratic field are represented as $r + s\sqrt{\delta}$, where $r, s \in \mathbb{Q}$. Norm of an element α of quadratic field is defined as $N(\alpha) = r^2 - \delta s^2$.

Let Δ be discriminant of quadratic field. Discriminant satisfies the equality: $\Delta = 0, 1 \; mod \; 4$. Discriminant is said fundamental discriminant Δ_1, if $\Delta_1 = 1 \; mod \; 4$ or $\Delta_1/4 = 2, 3 \; mod \; 4$. A non fundamental discriminant Δ_f can be decomposed in a fundamental discriminant Δ_1 and square of a number f, called conductor: $\Delta_f = \Delta_1 f^2$.

Ring of integral numbers in imaginary quadratic field is defined by $\mathcal{O}_\Delta = \mathbb{Z} + \frac{\Delta + \sqrt{\Delta}}{2}\mathbb{Z}$, it is called quadratic order. Integral ideal in quadratic order has the form $m(a\mathbb{Z} + \frac{b+\sqrt{\Delta}}{2}\mathbb{Z})$, where $m, a \in \mathbb{Z}^+$, $b \in \mathbb{Z}$, $-a < b < a$ and $\Delta = b^2 \; mod \; 4a$. Ideal is called primitive, if $m = 1$. (m, a, b) is called the standard representation of ideal, (a, b) - the one of primitive ideal. The norm of ideal equals to $m^2 a$. Primitive ideal is called reduced if $N(a) \leq N(\frac{b+\sqrt{\Delta}}{2})$, where $N(\alpha)$ denotes norm of an element α of quadratic field. The norm of primitive reduced ideal is less than $\sqrt{|\Delta_f|/3}$. Primitive ideal with the norm less than $\sqrt{|\Delta_f|/4}$ is always reduced.

An ideal $\mathfrak{a}$ is called coprime to a number f, if $gcd(N(\mathfrak{a}), f) = 1$. Coprime to f ideals form an abelian group $\mathcal{J}_{\Delta_f}(f)$ in the quadratic order with the discriminant Δ_f. Principal coprime to f ideals form a subgroup $\mathcal{P}_{\Delta_f}(f)$ in $\mathcal{J}_{\Delta_f}(f)$. A quotient group $Cl(\Delta_f) = \mathcal{J}_{\Delta_f}(f)/\mathcal{P}_{\Delta_f}(f)$ is classes group. Its order is denoted by $h(\Delta_f)$. A class group in the quadratic order with fundamental discriminant Δ_1 is $Cl(\Delta_1) = \mathcal{J}_{\Delta_1}(f)/\mathcal{P}_{\Delta_1}(f)$. There is only one reduced ideal in every class. Given an ideal from class, one could evaluate the reduced ideal in the class. This operation is called reduction Red_Δ. Hence, every class of $Cl(\Delta_1)$ can be uniquely identified with reduced ideal. Classes arithmetics can be performed as reduced ideals arithmetics. For example, multiplication of two classes is done by multiplication of their reduced ideals and consequent reduction.

Let consider the relation between ideals in $\mathcal{O}_{\Delta_1}$ and $\mathcal{O}_{\Delta_f}$. It is possible to define isomorphisms for ideals between groups $\mathcal{J}_{\Delta_f}(f)$ and $\mathcal{J}_{\Delta_1}(f)$. For an ideal $\mathfrak{a} \in \mathcal{J}_{\Delta_f}(f)$ there is an isomorphism $\phi(\mathfrak{a}) = \mathfrak{a}\mathcal{O}_{\Delta_1} = \mathfrak{u} \in \mathcal{J}_{\Delta_1}(f)$. And for an ideal $\mathfrak{u} \in \mathcal{J}_{\Delta_1}(f)$ there is an inverse to ϕ isomorphism $\phi^{-1}(\mathfrak{u}) = \mathfrak{u} \cap \mathcal{O}_{\Delta_f} = \mathfrak{a} \in \mathcal{J}_{\Delta_f}(f)$. Equality for superposition $\phi \cdot \phi^{-1} = \phi^{-1} \cdot \phi = 1$ holds.

Map ϕ can be thought as homomorphism of class group $Cl(\Delta_f)$ into $Cl(\Delta_1)$. This is the mapping many-to-one. The kernel of this homomorphism is a group of ideals mapping to the subgroup of principal ideals $\mathcal{P}_{\Delta_1}(f)$ in $\mathcal{J}_{\Delta_1}(f)$ by homomorphism ϕ.

When it is mentioned about ideals, it implies *isomorphisms* ϕ and ϕ^{-1}, and when it is mentioned about classes of ideals, it implies *homomorphism* ϕ.

Let $f = q$ be a prime and $\sqrt{|\Delta_1|/3} < q$. Then all reduced ideals in $\mathcal{O}_{\Delta_1}$ are coprime to q and hence belong to $\mathcal{J}_{\Delta_1}(q)$. In this case $h(\Delta_q) = h(\Delta_1)(q - \epsilon(\Delta_1, q))$, where $\epsilon(\Delta_1, q)$ denotes the Kronecker symbol. That is, $(q - \epsilon(\Delta_1, q))$ classes of $Cl(\Delta_q)$ are mapped to one class of $Cl(\Delta_1)$.

Operations of reduction $Red_\Delta(\mathfrak{a})$, isomorphisms $\phi(\mathfrak{a})$ and $\phi^{-1}(\mathfrak{a})$ have quadratic bit complexity $O(log^2(N(\mathfrak{a})))$, multiplication of ideals $\mathfrak{a} \times \mathfrak{b}$ - $O(log^2(max(N(\mathfrak{a}), N(\mathfrak{b}))))$, multiplication of a number $\alpha \in \mathcal{O}$ and ideal $\mathfrak{a}$ - $O(log^2(max(N(\alpha), N(\mathfrak{b}))))$. Because the norm of any reduced ideal is less than $\sqrt{|\Delta|/4}$, bit complexity of operations with them is $O(log^2(\Delta))$, .

Isomorphisms ϕ, ϕ^{-1} can be performed, if both numbers Δ_1 and f are known only. Operation of reduction Red_Δ requires knowledge of a discriminant Δ.

Algorithms for these operations can be found in [2], [6].

3 Cryptosystems, based on quadratic orders

J. Buchmann and H. C. Williams suggested a Diffie-Hellman like key exchange cryptosystem and a RSA like public-key cryptosystem in [5] and [4]. Cryptosystems are built in multiplicative group of classes $Cl(\Delta_f)$. Operations with classes are equivalent to operations with reduced ideals, as in every class only one reduced ideal exists and this is the ideal with the minimal norm. Cryptosystems have cubic time $O(log^3\Delta_f)$ of encryption and decryption due to exponentiation, used in Diffie-Hellman and RSA schemes.

D. Hühlein, M. Jacobson, Jr. and S. Paulus improved technique of cryptosystems applying over quadratic orders in [3]. They involved two orders with discriminants Δ_1, $\Delta_q = \Delta_1 q^2$ respectively and isomorphisms ϕ, ϕ^{-1}, which map ideals between orders $\mathcal{O}_{\Delta_1}$ and $\mathcal{O}_{\Delta_q}$. Suggested an ElGamal like public-key cryptosystem is built in a group of classes $Cl(\Delta_q)$. Encryption is done in $Cl(\Delta_q)$, it maps a message ideal $\mathfrak{m}$ to a cipher ideal $\mathfrak{c}$. Decryption is done in $Cl(\Delta_1)$:

- the reduced cipher ideal $\mathfrak{c}$ from $Cl(\Delta_q)$ is mapped to a reduced ideal $\mathfrak{u}$ in $Cl(\Delta_1)$ by the isomorphism ϕ;

- decryption algorithm (exponentiation, multiplication and reduction of ideals) is applied in $Cl(\Delta_1)$ to $\mathfrak{u}$ to produce reduced ideal $\mathfrak{u}_1$;

- the ideal $\mathfrak{u}_1$ is mapped by ϕ^{-1}, providing the message ideal $\mathfrak{m}$.

Mapping of ideals into $Cl(\Delta_1)$ allow to increase decryption performance. Decryption has cubic bit complexity $O(log^3\Delta_1)$ and encryption has cubic complexity of $O(log^3\Delta_q) \simeq O(27 \; log^3\Delta_1)$. Complexity is cubic due to exponentiation, used in ElGamal scheme. Note, that cryptosystems by J. Buchmann and H. C. Williams could be modified to work by similar manner.

Recently, S. Paulus and T. Takagi proposed a new public-key cryptosystem [2] with quadratic decryption time as an alternative to RSA and ElGamal cryptosystems. Their cryptosystem is a further development of [3]. It uses the fact that $(q - \epsilon(\Delta_1, q))$ classes of $Cl(\Delta_q)$ are mapped to one class of $Cl(\Delta_1)$ by ϕ. Encryption is done in $Cl(\Delta_q)$ and has cubic bit complexity $O(log^3\Delta_q)$ due to exponentiation. Decryption technique is the same as in above work [3] except that decryption algorithm in $Cl(\Delta_1)$ does not include exponentiation and has quadratic bit complexity $O(log^2\Delta_1)$. Consider the cryptosystem in details according to [1].

- **Key generation.** Choose random big primes $p = 3 \; mod \; 4$ and $q > \sqrt{p/3}$. Let $\Delta_1 = -p$, $\Delta_q = \Delta_1 q^2$. And let k and l be the bit lengths of $\sqrt{|\Delta_1|/4}$ and $(q - \epsilon(\Delta_1, q))$ respectively. Secret parameters are factorization of Δ_q: (Δ_1, q). Public parameters are $(\Delta_q, k, l, \mathfrak{p})$, where $\mathfrak{p}$ is a random ideal from the kernel of the isomorphism ϕ. The ideal $\mathfrak{p}$ is evaluated by choosing a random number $\alpha \in \mathcal{O}_{\Delta_1}$ and computing $\mathfrak{p} = \phi^{-1}(\alpha\mathcal{O}_{\Delta_1})$.

[*] Some results of this paper were announced on XLIV Scientific Conference of MIPT in [1].

[†] Moscow Institute of Physics and Technologies, e-mail: alex@dgap.mipt.ru

- **Encryption.** First, a text message m is transformed to a reduced ideal $\mathfrak{m}$ in $\mathfrak{I}_{\Delta_q}(q)$, an ideal $\mathfrak{u} = \phi(\mathfrak{m})$ being reduced in $\mathfrak{I}_{\Delta_1}(q)$. This can be done as follows.

  Let the message m be a number of $k-3$ bit length. Then the number $x = 1.M$ is formed by bit concatenation of 1 and the message m. A prime number a is generated: $a = 3 \ mod \ 4, x \leq a < 2x$. The difference $d = a - x$ is stored. The bit length of a is at most $k-1$, that is, $a < \sqrt{|\Delta_1|/4}$. Try to solve the equation $b^2 = \Delta_q \ mod \ 4a$ for b by the formulae $b = \pm\Delta_q^{(a+1)/4} \ mod \ a$. If the founded b is not the square root (the probability is $1/2$), repeat generation of the prime a again. Construct a message ideal $\mathfrak{m} = (a, b)$. The ideal $\mathfrak{m}$ is reduced, $N(\mathfrak{m}) = a < \sqrt{|\Delta_q|/4}$, and belongs to $\mathfrak{I}_{\Delta_q}(q)$, $N(\mathfrak{m}) = a < \sqrt{|\Delta_1|/3} < q$. Note, that the norm $N(\phi(\mathfrak{m})) = N(\mathfrak{m}) = a < \sqrt{|\Delta_1|/4}$ and, hence, $\phi(\mathfrak{m})$ is the reduced ideal in $\mathfrak{I}_{\Delta_1}(q)$.

  Generate a random number r with the bit length of $l-1$. Inequality $r < (q - \epsilon(\Delta_1, q))$ holds.

  Encryption is done by multiplication of the message ideal $\mathfrak{m}$ and the ideal $\mathfrak{p}$ from the kernel of the isomorphism ϕ, exponentiated in the power r. Reduction operation is performed after every multiplication to reduce tho norm of resulting ideal and, therefore, to increase performance. That is, $\mathfrak{c} = Red_{\Delta_q}(\mathfrak{m}\mathfrak{p}^r) \in \mathfrak{I}_{\Delta_q}(q)$.

  So, encrypted cipher text is represented by the reduced ideal $\mathfrak{c} = (a_1, b_1)$, and the difference d, required to restore initial m from the prime number a.

- **Decryption.** Decryption is done as follows:

  – apply the isomorphism ϕ to the cipher ideal $\mathfrak{c}$: $\phi(\mathfrak{c}) = \mathfrak{u}_1 \in \mathfrak{I}_{\Delta_1}(q)$;

  – make reduction $Red_{\Delta_1}(\mathfrak{u}_1) = \mathfrak{u}$; ideals $\mathfrak{u}_1$ and $\mathfrak{u} = \phi(\mathfrak{m})$ belong to the same class of $Cl(\Delta_1)$;

  – make $\phi^{-1}(\mathfrak{u}) = \mathfrak{m}$.

  – restore $m = a - d$ from the ideal $\mathfrak{m} = (a, b)$

The advantage of this cryptosystem is quadratic decryption time.
The drawbacks are:

- The ciphertext $\mathfrak{c}$ takes the size 6 time greater than the size of original message m. The bit length of original message is $L_m \simeq k = \log_2 \sqrt{|\Delta_1|/4} \simeq \frac{1}{2} log_2|\Delta_1|$. The bit length of the ciphertext $\mathfrak{c} = (a_1, b_1)$ is $L_C = log_2 a_1 + log_2 b_1 + log_2 d \simeq 2 log_2 a_1 \simeq 2 log_2 \sqrt{|\Delta_q|} \simeq 2 \cdot 3 log_2 \sqrt{|\Delta_1|} = 3 log_2 |\Delta_1| = 6k$.

- The ideal $\mathfrak{p}$ from the kernel of the homomorphism ϕ is published. Currently, there are no evidences that it could compromise the scheme. Until safety of publishing of $\mathfrak{p}$ is not proved, it is a potential flaw.

- The person encrypting a message can not verify that given ideal $\mathfrak{p}$ actually belongs to the kernel of the homomorphism ϕ.

- The technique of transformation of the message m into ideal requires calculation of a square root b modulo prime: $\Delta_q = b^2 \ mod \ 4a$. Because of $\Delta_q = 1 \ mod \ 4$, direct calculation is not applicable and algorithm of solving is probabilistic and has average cubic bit complexity. It could be found in [8] for example.

- As we convert a message to a prime number, we should provide additional open information, namely, the difference d, in order to make the inverse conversion. This additional information is sent as open text.

4 Analysis of signatures

In the works [5] and [4] there were proposed public-key cryptosystems and signatures in multiplicative groups, namely in the group of classes $Cl(\Delta_f)$, based on well-known Diffie-Hellman, RSA and ElGamal schemes. The public-key cryptosystem [2] uses properties of homomorphism between two groups of classes $Cl(\Delta_1)$ and $Cl(\Delta_f)$ that allow to decrypt a ciphertext with quadratic time.

The isomorphism (homomorphism) ϕ and the inverse isomorphism ϕ^{-1} can be performed if decomposition of $\Delta_q = \Delta_1 q^2$ is known only. It can be said that the knowledge of ϕ and ϕ^{-1} is the secret key in the cryptosystem [2]. In this cryptosystem two classes of the group $Cl(\Delta_q)$, represented by the reduced message ideal $\mathfrak{m}$ and the reduced cipher ideal $\mathfrak{c}$, are mapped by the homomorphism ϕ to a one class of the group $Cl(\Delta_1)$. The message ideal $\mathfrak{m}$ is mapped by the isomorphism ϕ to a reduced ideal in $\mathfrak{I}_{\Delta_1}(q)$ and can be recovered by the inverse isomorphism ϕ^{-1}.

Our task is to consider signatures based on the cryptosystem [2]. We will suggest that decomposition and mappings ϕ and ϕ^{-1} are the secret key of signer.

Let describe a common scheme of signing in this framework.

- A message m is transformed to a reduced ideal $\mathfrak{m} \in \mathfrak{I}_{\Delta_q}(q)$, representing a class in the group $Cl(\Delta_q)$;

- The message ideal $\mathfrak{m}$ is mapped by ϕ to an ideal $\mathfrak{u} \in \mathfrak{I}_{\Delta_1}(q)$, the ideal $\mathfrak{u}$ representing a class in $Cl(\Delta_1)$;

- Some evaluations are made with $\mathfrak{u}$, preferably without exponentiation. We remind here that we deal with classes; classes are represented by reduced ideals. So, every step of evaluations should be followed by reduction in order to decrease the norm of ideals and therefore to decrease the time of calculations. The resulted reduced ideal $\mathfrak{u}_1$ will represent the same class, as the one represented by $\mathfrak{u}$ (in the case of simple reduction of $\mathfrak{u}$ as in [2] it is done). Or the ideal $\mathfrak{u}_1$ will represent an another class of $Cl(\Delta_1)$.

- The resulted reduced ideal $\mathfrak{u}_1$ is mapped by ϕ^{-1} and then is reducing to a signature ideal $\mathfrak{s} \in \mathfrak{I}_{\Delta_q}(q)$, representing a class of the group $Cl(\Delta_q)$.

The task of a verifier is to make sure that the ideal $\mathfrak{s}$ actually corresponds to the message m. More precisely, the verifier should check that two classes of $Cl(\Delta_q)$, one, represented by the reduced message ideal $\mathfrak{m}$ and another represented by the reduced signature ideal $\mathfrak{s}$, are in a relation defined by the signing scheme. This relation can be one of following:

- The message ideal $\mathfrak{m}$ and the signature ideal $\mathfrak{s}$ represent two classes from a subgroup of $Cl(\Delta_q)$, which are mapped by the homomorphism ϕ to a one class of $Cl(\Delta_1)$;

- The message ideal $\mathfrak{m}$ and the signature ideal $\mathfrak{s}$ represent two classes from a subgroup of $Cl(\Delta_q)$, which are mapped by the homomorphism ϕ to different classes of $Cl(\Delta_1)$.

There are three statements:

- Verifier could not check that given ideal $\mathfrak{p}$ is actually from the kernel of the homomorphism ϕ. So, an ideal $\mathfrak{p}$ could not be a session specific parameter, it could be a constant public key of signer.

- Verifier could check that two ideals belong to a one subgroup of $Cl(\Delta_q)$ mapped by ϕ to a one class of $Cl(\Delta_1)$ by the only manner: he should multiply one of ideals by the ideal $\mathfrak{p}$ from the kernel of the homomorphism ϕ, exponentiated in a some power r. Calculation of the number r is the taking logarithm made by signer indeed.

- Verifier could check that two ideals belong to different subgroups of $Cl(\Delta_q)$ mapped by ϕ to different classes of $Cl(\Delta_1)$ by the only manner: he should be convinced that the taking logarithm task from $\mathfrak{p}^r$ has no solution.

These three statements prohibit any signature scheme to be constructed.

Let consider the case of composite f. We note here that the decomposition task of the discriminant Δ_f should be computationally infeasible. Hence, the discriminant Δ_f must have big prime conductors. Suppose, the conductor is $f = qp$. Brief analysis of discriminants Δ_1, Δ_q, Δ_q, Δ_{qp} leads to the same problem of detection of relation between two classes of $Cl(\Delta)$.

There is no way to construct a signature scheme based on the public-key cryptosystem [2].

References

[1] *A. S. Kshevetsky*, Implementation of the public-key cryptosystem in imaginary quadratic fields with quadratic decryption time, XLIV Scientific Conference of Moscow Institute of Physics and Technologies (2001).

[2] *S. Paulus, T. Takagi*, A new public-key cryptosystem over a quadratic order with quadratic decryption time // Journal of Cryptology (2000) 13.

[3] *D. Hühlein, M. J. Jacobson, Jr., S. Paulus*, A cryptosystem based on non-maximal imaginary quadratic orders with fast decryption, Advances in Cryptology - EUROCRYPT'98, LNCS 1403, Spriger-Verlag, Berlin (1998), pp.294-307.

[4] *J. Buchmann and H. C. Williams*, Quadratic Fields And Cryptography // London Mathematical Society Lecture Note Series 154, Cambridge University Press, Cambridge (1990), pp.9-26.

On the nonexistence of some optimal arcs in PG(4, 4)

Ivan N. Landjev, Assya Rousseva

Institute of Mathematics and Informatics

Bulgarian Academy of Sciences

8 Acad. G. Bonchev str. bl.8

1113 Sofia, BULGARIA

The aim of this note is to survey the present state of knowledge in the research of the function $n_4(5,d)$, defined as the minimal length of a linear code of dimension 4 and minimum distance d over $\mathbb{F}_4$. In the same time we find the exact value or refine the current estimate for several d's. We approach this problem from its geometric side and consider the equivalent arcs in the projective geometries over the field with four elements.

Let us note that the exact value of $n_4(k,d)$ is known for all $k \leq 4$ for all d [5, 6, 9]. There has been a considerable amount of research on optimal quaternary codes of dimension 5 [1, 2, 8, 10, 11]. At present, the exact value of $n_4(5,d)$ is undecided in the following cases [12]:

$n_4(5,d) = g_4(5,d)$ or $g_4(5,d)+1$ for the following values of d:

17	18		21				33	34			
65	66	67	68	69	70			81	82	83	84
129	130	131	132	133	134	135	136	137	138		
		259	260			263	264	265	266		
269	270	271	272	273	274	275	276	277	278	279	280
281	282	283	284	285	286			289	290	291	292
293	294	295	296	321	322	323	324	325	326	327	328
329	330	331	332	333	334	335	336	337	338	339	340
341	342	343	344	353	354	355		357	358	359	

Let us note that linear codes with parameters $[47,5,33]_4$ and $[48,5,34]_4$ are equivalent to $(47,14)$- and $(48,14)$-arcs in $PG(4,4)$, respectively, which are easily seen to be caps. By a recent result by Bierbrauer and Edel [4], the maximal size of a cap in $PG(4,4)$ is 41, so arcs with these parameters do not exist.

$n_4(5,d) = g_4(5,d)$, $g_4(5,d)+1$ or $g_4(5,d)+2$ for the following values of d:

		71	72	73	74	75			267	268
		287	288			297	298		301	302
345	346	347	348	349	350	351	352			356
			360	361	362					

Recently, H.N Ward proved that there exist no codes with parameters $[98,5,72]_4$ [13]. So, the value $d=72$ should be moved to the next group.

$n_4(5,d) = g_4(5,d)+1$ or $g_4(5,d)+2$ for the following values of d:

			24	37	38	39	40	41	42	43	44
45	46	47	48	49	50	51	52	57	58	59	60
61	62	63					76	89	90	91	92
93	94	95	96	97	98	99	100		102	103	104
105	106	107	108	109	110	111	112	113	114	115	116
117	118	119	120			123	124	141	142	143	144
153	154	155	156	157	158	159	160			163	164
165	166	167	168	169	170	171	172	173	174	175	176
		299	300			303	304	313	314	315	316
317	318	319	320			363	364				

A *multiset in* $PG(k-1,q) = (\mathcal{P},\mathcal{L},I)$ is a mapping $\mathfrak{k}\colon \mathcal{P} \to \mathbb{N}_0$. The integer $\mathfrak{k}(\mathcal{P}) = \sum_{P\in\mathcal{P}} \mathfrak{k}(P)$ is called the *cardinality* of the multiset $\mathfrak{k}$. For a subset $\mathcal{Q}$ of $\mathcal{P}$, we set $\mathfrak{k}(\mathcal{Q}) = \sum_{P\in\mathcal{Q}} \mathfrak{k}(P)$. The integer $\mathfrak{k}(\mathcal{Q})$ is called the *multiplicity* of the subset $\mathcal{Q}$. A point of multiplicity i is called an i-point; i-lines, i-planes etc. are defined in a similar way. A multiset $\mathfrak{k}$ in $\mathrm{PG}(k-1,q)$ is called an $(n,w,k-1,q)$*-arc*, or an (n,w)-arc for short, if

(a) $\mathfrak{k}(\mathcal{P}) = n$;

(b) for each hyperplane Δ in $\mathrm{PG}(k-1,q)$, $\mathfrak{k}(\Delta) \leq w$, and

(c) there is a hyperplane Δ with $\mathfrak{k}(\Delta) = w$.

Given a $(n, w; k-1, q)$ arc, we define by $\gamma_i(\mathfrak{k})$ the maximal multiplicity of an i-dimensional flat in $\mathrm{PG}(k-1, q)$, i.e. $\gamma_i(\mathfrak{k}) = \max_\delta \mathfrak{k}(\delta)$, $i = 0, \ldots, k-1$, where δ runs over all i-dimensional flats in $PG(k-1, q)$.

The existence of a linear $[n, k, d]_q$ code of full length (no coordinate is identically zero) is equivalent to that of $(n, n-d; k-1, q)$-arcs [3]. Moreover two linear codes with the same parameters are semilinearly isomorphic if and only if the corresponding arcs are projectively equivalent. For an $(n, n-d; k-1, q)$ arc $\mathfrak{k}$, let a_i be the number of hyperplanes Δ in $\mathrm{PG}(k-1, q)$ with $\mathfrak{k}(\Delta) = i$, $i = 0, 1, \ldots$, and by λ_j the number of points P from $\mathcal{P}$ with $\mathfrak{k}(P) = j$. The sequence $(a_0, a_1, \ldots)$ is called *the spectrum* of $\mathfrak{k}$.

The following argument will be used throughout this note. Let $\mathfrak{k}$ be an $(n, n-d; k-1, q)$ arc. Fix an i-dimensional flat δ in $\mathrm{PG}(k-1, q)$, with $\mathfrak{k}(\delta) = t$. Let further π be an j-dimensional flat in $\mathrm{PG}(k-1, q)$ with $i + j = k-2$ and $\delta \cap \pi = \emptyset$. Define the projection $\varphi_{\delta,\pi}$ from δ onto π by

$$\varphi_{\delta,\pi}\colon \begin{cases} \mathcal{P} \setminus \delta & \rightarrow \quad \pi \\ Q & \rightarrow \quad \pi \cap \langle \delta, Q \rangle. \end{cases} \tag{1}$$

Note that $\varphi_{\delta,\pi}$ maps $(i+s)$-flats containing δ into $(s-1)$-flats in π. Given a set of points $\mathcal{F} \subset \pi$, define $\mu(\mathcal{F}) = \sum_{\varphi_{\delta,\pi}(P) \in \mathcal{F}} \mathfrak{k}(P)$. If $\mathcal{F}$ is a k'-dimensional flat in π then $\mu(\mathcal{F}) \leq \gamma_{k'+i+1} - t$.

Let $\mathfrak{k}$ be a $(123-i, 31)$-arc in $PG(3, 4)$, $i = 0, 1, 2, 3$. Then its structure can be described as follows. Denote by l a line in $PG(3, 4)$ and by π_j, $j = 0, \ldots, 4$, the planes through l. Let further δ be a plane, different from the planes π_j and let $\mathfrak{b}$ be a plane $(5+i, 1)$-bloking set in δ which has $\mathfrak{b}(l \cap \delta) = 0$. Set

$$\mathfrak{k}(P) = \begin{cases} 1 & \text{if } P \in (\pi_0 \cup \pi_1) \setminus \delta \\ 1 - \mathfrak{b}(P) & \text{if } P \in (\pi_0 \cup \pi_1) \cap \delta \\ 2 & \text{if } P \in (\pi_2 \cup \pi_3 \cup \pi_4) \setminus \delta \\ 2 - \mathfrak{b}(P) & \text{if } P \in (\pi_2 \cup \pi_3 \cup \pi_4) \cap \delta \end{cases} . \tag{2}$$

Hence all arcs with parameters $(123-i, 31)$ can obtained by choosing a particular blocking set $\mathfrak{b}$ in δ. These blocking sets are well-known (see [7]).

Theorem 1. *(i) There exists a unique* $(123, 31)$*-arc in* $PG(3, 4)$.

(ii) Every $(122, 31)$*-arc is extendable.*

(iii) A $(121, 31)$*-arc is either extendable, or else obtained by the above construction, where* $\mathfrak{b}$ *is a Baer subplane.*

(iv) A $(120, 31)$*-arc is either extendable, or else obtained by the above construction, where* $\mathfrak{b}$ *is the indecomposable* $(8, 1)$*-blocking set.*

Clearly, the possible multiplicities of the planes π_j.

$$(\pi_0, \ldots, \pi_4) = (15 - \varepsilon_0, 15 - \varepsilon_1, 31 - \varepsilon_2, 31 - \varepsilon_3, 31 - \varepsilon_4), \tag{3}$$

where $\sum_j \varepsilon_j = i$.

Theorem 2. *There exists no* $(483, 122)$*-arc in* $PG(4, 4)$.

Proof. Assume such an arc, say $\mathfrak{k}$ does exist. Clearly, it has the structure described above. Fix a 122-solid δ_0 and a 31-plane, π say, through the 0-line l in δ_0. Denote by δ_j all solids through π. Consider a projection φ from the 0-line l onto some plane disjoint from l. Set $l_j = \varphi(\delta_j)$, $j = 0, \ldots, 4$. The types of the lines l_j can be obtained from (3). Now we have the following possibilities:

(A) $(\mu(l_0), \ldots, \mu(l_4)) = (122, 122, 122, 122, 119)$;

(B) $(\mu(l_0), \ldots, \mu(l_4)) = (122, 122, 122, 121, 120)$;

(C) $(\mu(l_0), \ldots, \mu(l_4)) = (122, 122, 121, 121, 121)$;

The set $\{X | X \in \cup_{j=1}^{3}, k(X) \geq 29\}$ is a $(9, 3)$-arc. Now a simple counting argument gives that for any point Y on l_4 we have $\mu(Y) \leq 19$. Hence $\mu(l_4) \leq 31 + 4 \cdot 19 = 109$, a contradiction. □

Corollary 3. *There exist no codes with parameters* $[483, 5, 361]_4$ *and* $[484, 5, 362]_4$. *Moreover,* $n_4(5, 362) = 485$ *or* 486, *and* $n_4(5, 361) = 484$ *or* 485.

Theorem 4. *There exists no* $(478, 121)$*-arc in* $PG(4, 4)$.

The proof of this theorem is similar to that of Theorem 2.

Corollary 5. *There exist no codes with parameters* $[478, 5, 357]_4$, $[479, 5, 358]_4$, $[480, 5, 359]_4$ *and* $[481, 5, 360]_4$. *Consequently,* $n_4(5, 357) = 479$, $n_4(5, 358) = 480$, $n_4(5, 359) = 481$, $n_4(5, 360) = 482$ *or* 483.

References

[1] I. Boukliev, New bounds for the minimum length of quaternary linear codes of dimension five, *Discrete Math.* 169(1997), 185–192.

[2] I. Boukliev, R. Daskalov, S. Kapralov, Optimal quaternary linear codes of dimension five, *IEEE Trans. Inform. Theory* 42(1996), 1228-1235.

[3] S. Dodunekov, J. Simonis, Codes and projective multisets, *Electronic Journal of Combinatorics* 5, 1998, no. #R37.

[4] Y. Edel, J.Bierbrauer, 41 is the largest size of a cap in $PG(4,4)$, *Designs, Codes and Cryptography* 16(1999), 51–60.

[5] P. Greenough, R. Hill, Optimal linear codes over $GF(4)$, *Discrete Math.* 125(1994), 187-199.

[6] R. Hill, I. Landjev, On the nonexistence of some quaternary codes, *Applications of Finite Fields* (ed. D. Gollmann), IMA Conference Series 59, Clarendon Press, Oxford, 1996, 85–98.

[7] J.W.P. Hirschfeld, L. Storme, The packing problem in statistics, coding theory and finite projective spaces, in: Finite geometries, Proc. of the Fourth Isle of Thorns Conference, (eds. A. Blokhuis et al.), 2001, Kluwer, 201–246.

[8] I. Landjev, T. Maruta, On the minimum length of quaternary linear codes of dimension five, *Discrete Math.* 202(1999), 145–161.

[9] I. Landjev, T. Maruta, R. Hill, On the nonexistence of quaternary $[51,4,37]$ codes, *Finite Fields Appl.* 2(1996), 96–110.

[10] T. Maruta, On the minimum length of q-ary linear codes codes of dimension five, *Geom. Dedicata* 65(1997), 299–304.

[11] T. Maruta, The nonexistence of $[116,5,85]_4$ codes and $[187,5,139]_4$ codes, Proc. of the Second Int. Workshop on Optimal Codes, Sozopol, 1998, 168–174.

[12] T. Maruta, personal communication.

[13] H.N. Ward, A sequence of unique quaternary Griesmer codes, manuscript.

Soft decision information set decoding algorithms.

S. Lazareva

St.Petersburg State University of Aerospace Instrumentation,
Bolshaia Morskaia str.,67,St.Petersburg,190000,Russia.
E-mail:lazsv@mail.ru

Introduction.

By E_2^n we denote the set of all n-words over the 2-ary alphabet.Speaking of linear codes, we assume that E_2^n is a linear space over the field F_2^n. Let G be a $k \times n$ generator matrix of binary linear k-dimensional *(n,k)* code C and d_0 correspond the Gilbert-Varshamov distance

$$d_0 = min\left\{d : \sum_{i=0}^{d} \binom{n}{i} > 2^{n-k}\right\}.$$

We use a short notation [n] for set $\{1,2,...,n\}$. Let $W \subseteq [n]$ and let A be a matrix with n columns. By $A(W)$ we denote the submatrix A formed by the columns of A labeled with indices from W. Thus, $y(W)$ is the projection of the vector y on its coordinates in W. Let $y = \{y_1, y_2, ..., y_n\}$ be the receiving vector.

An information set is a k-set $W \subset [n]$ such that the corresponding $k \times k$ submatrix $G(W)$ is nonsingular. The remaining n-k coordinates are called a check set.

Let $\Im$ be the collection of the information set of code C. If the error vector e has zeros on $W \in \Im$, we shall be able to find the transmitted codeword c from the received word $y = e+c$. One of the methods to choose the collection $\Im$ is the method of random choice. For hard decision decoding the obvious suggestion is to take random uniformly distributed k-subsets of [n]. We call the following algorithm the covering set decoding [1]. Let

$$L_n(k) = (nlogn)\binom{n}{d_0}/\binom{n-k}{d_0}$$

Covering set decoding:

1. Set *c=0.*
2. Choose randomly a k-subset W. Form a list of codewords $L(W) = \{c \in C | c(W) = y(W)\}$.
3. If there is a $c' \in L(W)$ such that $dist(c', y) < dist(c', y)$, assign $c \leftarrow c'$.
4. Repeat the last two steps $L_n(k)$ times. Output c.

For soft decision decoding the problem of choosing information sets is a different one. Any received output y from the Euclidean space R^n still gives the most probable "hard decision" vector $a = (a_1, a_2, ..., a_n) \in E_2^n$. However, each symbol a_j has its own reliability $v_j = ln\,[p(a_j|y_j)/p(b_j|y_j)]$, that is the log of the likelihood ratio of the more probable symbol a_j to the less probable symbol $b_j = 1 - a_j$. Obviously, different subsets of k positions are not equally reliable. Therefore we should more often include more reliable symbols in information sets and less reliable ones in check set. Let's consider two methods for finding an error-free information subset in a number of random-search trials:

1. Algorithm of Dumer [2] with a concrete choice of parameters.
2. Information set with the help of the Markov process.

Decoding algorithms.

Let the Forney criterium [3] be the choice criterium for the error-free information set. The received reliabilities are normalized so that $0 \leq v_j \leq 1$. The received symbols y_j are arranged according to the decreasing normalized reliability $v_1 \geq v_2 \geq ... \geq v_n$.

Let's define a rule for the choice of an information set for the algorithm of Dumer with a concrete choice of parameters (1). To each symbol of the hard decision a_i we shall assign the probability p_i, corresponding to the following relations:

1. $\sum_{i=0}^{n-1} p_i = k$.
2. $\sum_{i=0}^{n-1} H_2(\gamma_i) = k$, where $H_2(x) = -xlog_2(x) - (1-x)log_2(1-x)$.
3. $\gamma_i = \frac{1}{1+2^{\lambda v_i}}$, λ -is determined from formula 2.
4. $0 < p_i \leq 1 - 2\gamma_i$.

The probabilities corresponding to the relations (1) - (4) ensure a minimum covering E_2^n by ellipsoids [2]. The condition (4) for the choice p_i is specified as follows [4]:

5. $p_i = \eta_i(1-2\gamma_i)^m$, where η_i normalizing factor ensuring the realization of relation 1, m is a positive number.

If m is increasing, an explicit division of the probability p_i into three groups takes place:

A: the choice probabilities are approximately identical and relatively large for all symbols which are included in the given set.

B: the choice probability is close to zero.

C: those which are not included in either in group A or B.

Thus, the choice m determines the size of the groups A,B and C. If $m \to \infty$ the size of set A approaches a certain number h with a choice probability of the symbols $p_i = \frac{k}{h}$, the size of set B approaches *n-h* with a choice probability of the symbols $p_i = 0$. Thus, the problem of information set decoding *(n,k)* code is reduced to information set decoding *(h,k)* code with random uniformly distributed k-subset of [h]. We define the decoding algorithm 1 as follows:

Let T be the maximum number of the selected information sets.

Decoding algorithm 1.

1. Set *c=0.*
2. Choose m, with the help of simulation, so that it ensures minimum probability of the error sequence. *l: = 0.*
3. Form an information set I as follows: we include a symbol a_i in an information set with probability p_i .
4. Compute a codeword $c' = (x_1, x_2, ...x_n) = y(I)G^{-1}(I)G$.
5. If $\sum_{i=1}^{n} v_i(-1)^{1-x_i} > n - d$, than c' is the output. Otherwise $l := l + 1$.
6. If $l > T$, is the output.

Now the method of the dependent choice of information sets with the help of of the Markov process (2) should be considered. It ensures less error probability in case the ratioes signal / noise are large in comparison with algorithm 1(figure 1). The error probability of a symbol y_j in case the value of the reliability v_j is defined by the following ratio: $r_j = \frac{e^{-v_j}}{1+e^{-v_j}}$. The choice probability of the accepted reliabilities can be changed. The rule of the recalculation of the choice probabilities of a symbol into an information set can be described by the Markov process with the following matrix of transitional probabilities A of a size *(n, n)*:

1. $A_{ii} = 1 - r_i, 1 \leq i \leq k, A_{ij} = 0, i \neq j, 1 \leq i, j \geq k.$
2. $A_{ii} = 1 - \sum_{i=1}^{k} A_{ij}, k+1 \leq i \leq n.A_{ij} = 0, i \neq j, k+1 \leq i, j \leq n.$
3. $A_{ij} = \alpha_i(1 - r_i), \alpha_i = \frac{r_i}{\sum_{j=k+1}^{n}(1-r_j)} 1 \leq i \leq k, k+1 \leq j \leq n.$
4. $A_{ij} = A_{ji}, k+1 \leq i \leq n, 1 \leq j \leq k.$

Decoding algorithm 2.

1. Choose the first k of the most reliable symbols which present an information set I, *l: = 0.* Set $c = 0$.
2. Compute a codeword $c' = (x_1, x_2, ...x_n) = y(I)G^{-1}(I)G$.
   If $\sum_{i=1}^{n} v_i(-1)^{1-x_i} > n - d$, than c' is the output. Otherwise $l := l + 1$.
3. Exponentiating $A = A^l$, reduce the matrix to the following : $A_{ii} = \sum_{i=1}^{k} A_{ij}, 1 \leq i, j \leq k.A_{ij} = 0, i \neq j, 1 \leq i, j \leq k.A_{ii} = \sum_{j=k}^{n} A_{ij}, k+1 \leq i, j \leq n.A_{ij} = 0, i \neq j, k+1 \leq i, j \leq n$. The information set is organized as follows: the probability A_{ii} $i \in [1, k]$ is the probability of the symbol entering an information set, the probability A_{ij} is the i symbol from the information set is substituted by the j symbol of the check set. Obtaining an information set I, go to step 1.
4. If $l > T$, c is the output.

Matrix A is a bistochastic one. Therefore, the matrix A becomes stationary at certain degree $l = u$, which means that any element from the matrix $A_{ij} = 1/n$, i.e. if we are not able to decode for u steps, the problem of decoding becomes a case in which all the symbols are equiprobable. According to matrix A, we can estimate an amount of information sets necessary for ensuring the required error probability.

Literature.

1. E.A.Krouk, "Decoding complexity bound for linear block codes, 'Problems of Information Transmission 25 (3) (1989), 251-254./ in Russian

2. Dumer I. Soft-decision decoding via sphero-ellipsoidal coverings.// In Proceedings of the Sixth International Workshop on Algebraic and Combinatorial Coding Theory, pages 98-102, Pskov, Russia, September 1998.

3. D. Forney. Concatenated codes. //Research Monograph 37 The M.I.T. Press, Cambridge, Massachusetts,1966.

4. M.E.E. Dulte Jose de Barros."Soft decision information set decoding of long block codes".// Ph.D. dissertation, Darmstadt University of Technology , June 2000./in German.

5. A.Barg."Complexity issues in coding theory,"in Handbook of Coding Theory, vol.1,V.Pless and W.C.Huffman, Eds.,Amsterdam, The Netherlands: Elsevier Science, 1998, pp. 649–754

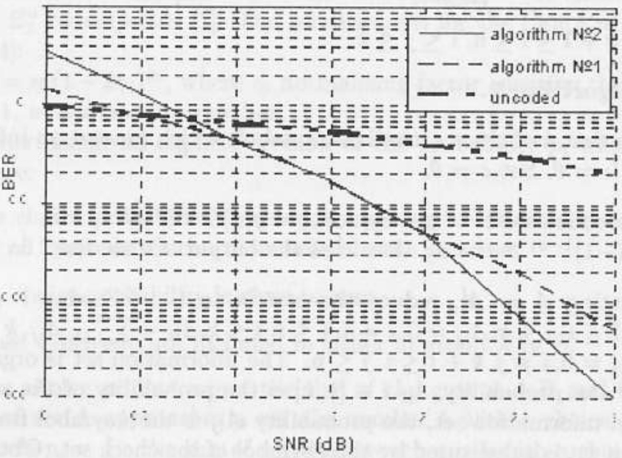


Figure 1: Output bit error rate for the (63,30) BCH code with decoding list of size T=100 for algorithms 1 and 2.

Some tables for (w, r) superimposed codes

Vladimir Lebedev[1]

Abstract

We give some constructions for (w, r) superimposed codes of small size. We also prove new asymptotic upper bound on rate of (w, r) superimposed codes.

1 Introduction

Definition 1. An $N \times T$ (0,1)-matrix C is called a (w, r) *superimposed code of size* $N \times T$, if for any pair of subsets $I, J \subset [T]$ such that $|I| = w$, $|J| = r$ and $I \cap J = 0$, there exists a coordinate $x \in [N]$ such that $c_{xi} = 1$ for all $i \in I$ and $c_{xj} = 0$ for all $j \in J$.

The main problem in the study of superimposed codes is to find the maximal cardinality $T(N, w, r)$ of a (w, r) superimposed code for a given length N, or the minimal length $N(T, w, r)$ of a (w, r) superimposed code for a given cardinality T.

Obviously $N(T, w, r) = N(T, r, w)$; thus we may only consider the case $w \leq r$. There are simple examples of (w, r) superimposed codes. If we take a matrix whose rows are all possible binary vectors of weight w, then this matrix satisfies the (w, r) superimposed property for any r. We call this matrix the trivial (w, r) superimpose code of size $N \times T$, where $N = \binom{T}{w}$. It is easy to prove that $N(w + r + 1, w, r) = \binom{w+r+1}{w}$ for all $w \leq r$ (see [4], [6]).

We now give some constructions for (w, r) superimposed codes.

2 Some constructions for (w, r) superimposed codes

The notion of a super-simple 2-design was introduced in [5]. A $2-(v, k, \lambda)$ design is called super-simple if the intersection of any two blocks has at most two elements. We define a super-simple t-design to be a $t-(v, k, \lambda)$ design in which the intersection of any two blocks has at most t elements.

Theorem 1. [6] *A super-simple* $t-(v, k, \lambda)$ *design is a* $(t, \lambda - 1)$ *superimposed code of size* $N \times v$, *where* $N = \frac{\lambda\binom{v}{t}}{\binom{k}{t}}$.

Corollary 1. There is a $(2, 2)$ superimposed code of size 18×9.

[1] V.S. Lebedev is with the Institute for Problems of Information Transmission of the Russian Academy of Sciences, Bol'shoi Karetnyi per. 19, Moscow 101447, Russia (lebed@iitp.ru).

Our second construction use a notation and constructions of 3-covering array from [10].

Definition 2. t-covering array with alphabet size q, length n and size k consists of k vectors of length n with entries from $\{0, 1, \dots, q-1\}$ with the property that the projection onto any t coordinates contains all q^t possibilities.

Let $g_t(n)$ denote the minimal size of a binary t-covering array of length n.

Theorem 2. *For* $(2,2)$ *superimposed codes*

$$N(2T, 2, 2) \leq N(T, 2, 2) + g_3(T)$$

Proof. Let A be a $(2,2)$ superimposed code of size $N \times T$ and B be a binary 3-covering array of length T. Then it is easy to check that

$$\begin{array}{cc} \hline A & A \\ B & \bar{B} \\ \hline \end{array}$$

is a $(2,2)$ superimposed code (where the bar indicates the complementary array).

The same constructions were considered in [10] for a binary 3-covering array and in [7] for $(2,2)$ superimposed code. But in [7] the matrix B is a complete separate $(1,2)$ code and there are two additional rows with all zeroes and all ones. So our theorem 2 usual gives better values of $(2,2)$ superimposed code's size.

These two constructions give good $(2,2)$ superimposed codes only for small size of codes. But using well known idea of concatenating we can construct good codes from these codes.

Let B be an external q-ary code of size $N_q \times T_q$ and let C be an internal code of size $N_1 \times T_1$ with $T_1 = q$. Then there is a concatenated code $B \diamond C$ of size $N \times T$, where $N = N_q N_1$ and $T = T_q$ i.e. each q-ary element $\theta \in [q]$ in the codebook matrix of the external code B is replaced by the θ-th codeword of the internal code C

The following result (see [3]) which is a natural generalization of Sagalovich's result ([8], $w = r = 2$) and D'yachkov, Macula, Rykov's result ([2], $w = 1$) shows how to use concatenated construction in constructing (w,r) superimposed codes.

Lemma 1 (concatenated construction) [3]. *Let B be a q-ary (w,r) separate code of size $N_q \times T_q$ and C be a binary (w,r) superimposed code of size $N_1 \times T_1$ with $T_1 = q$. Then the concatenated code $B \diamond C$ is a binary (w,r) superimposed code of size $N \times T$, where $N = N_q N_1$ and $T = T_q$.*

Corollary 2 [3]. Let $w, r \geq 1$ and $\lambda \geq 1$ be integers and $q \geq wr\lambda$ be a prime power. Then

$$N(q^{\lambda+1}, w, r) \leq N(q, w, r)\,[wr\lambda + 1]\,.$$

If q is not prime power then we can use

Proposition 2 [6]. *Suppose there are z mutually orthogonal Latin squares of order M and $z + 1 \geq wr$. Then*

$$N(M^2, w, r) \leq N(M, w, r)\,[wr + 1]\,.$$

Let us summarize all known results about the size of $(2,2)$ superimposed codes and $(2,3)$ superimposed codes in the following tables. These tables show that Theorem 2 gives many good examples of $(2,2)$ superimposed codes.

Table 1. Optimal length of some $(2,2)$ superimposed codes.

$T =$	4	5	6	7	8	9
$N(T,2,2) =$	6	10	14	14	14	18

Table 2. Existence of some $(2,2)$ superimposed codes.

$T =$	10	12	16	18	20	22	24	28	32	36	40	44	48	56	60	64
$N(T,2,2) \leq$	20	22	26	30	32	34	38	42	43	48	50	53	61	65	67	68

$T =$	72	80	88	112	128	144	512
$N(T,2,2) \leq$	74	76	80	96	100	109	126

Table 3. Existence of some $(2,3)$ superimposed codes.

$T =$	6	7	10	15	16	21	24	49	225	2197	4096
$N(T,2,3) \leq$	15	21	30	42	48	56	76	147	294	546	624

3 Asymptotic upper bound on rate of (w,r) superimposed codes

The idea of using a recurrent method to obtain an asymptotic upper bound of a rate of $(2,2)$ superimposed code was considered in [9]. The following method is a natural generalization of that idea.

Consider a binary code C of size $N \times T$. Let $x, y > 0$ be integers. Consider $x + y$ fixed codewords and denote by $C_{x,y}$ the submatrix of C of size $N \times (x+y)$. Define a super-distance for the $x + y$ codewords be a number of rows from the matrix $C_{x,y}$ with weight x. Denote by $d_{x,y}$ the minimal super-distance for some $x + y$ codewords from code C. It is easy to see that $d_{1,1}$ is a minimal Hamming distance of the code C.

Theorem 3. *The following asymptotic bound for a rate of binary code with superdistance $d_{x,y}$ is true*

$$R(N, d_{x,y}) \leq 1 - \frac{(x+y)^{x+y} d_{x,y}}{x^x y^y C^x_{x+y} N}.$$

This asymptotic bound gives well known Plotkin bound for case $x = y = 1$. The following Lemma is a natural generalization of the Lemma Kabatianski (see [6]).

Lemma 2. *If there is a (w, r) superimposed code of size $N \times T$ then there is a $(w - x, r - y)$ superimposed code of size $[d_{x,y}/C^x_{x+y}] \times (T - x - y)$.*

For a (w, r) superimposed code C of size $N \times T$, we define its rate $R(C)$ by $R(C) = \frac{\log_2 T}{N}$. Denote by $R(w, r)$ the asymptotic rate for (w, r) superimposed codes.

Theorem 4. *For (w, r) superimposed codes*

$$R(w, r) \leq \min_{0<x<w} \min_{0<y<r} \frac{R(w-x, r-y)}{R(w-x, r-y) + (x+y)^{x+y}/(x^x y^y)}.$$

Proof. It follows from Theorem 3 and Lemma 2 that

$$R(w, r) \leq 1 - \frac{(x+y)^{x+y} N(T-x-y, w-x, r-y)}{x^x y^y N(T, w, r)}.$$

So

$$R(w, r)(1 + \frac{(x+y)^{x+y}}{x^x y^y R(w-x, r-y)}) \leq 1.$$

Corollary 3. For case $w = r$ we have

$$R(w, w) \leq \frac{3}{4^{(w-2)}(3C(2,2)+1) - 1},$$

where $C(2, 2) = 6,2125692$.

Theorem 4 gives better upper bound then the upper bound from [4] for all values w and r. We give some numerical values of the upper bound only for case $w = r$.

Table 4. Asymptotic upper bound on rate of some superimposed codes

(w,r)	(3,3)	(4,4)	(5,5)	(6,6)	(7,7)	(8,8)
$R(w,r) \leq$	0.0386843	0.0095784	0.0023889	0.0005969	0.0001492	0.0000373

Remark. If for case $x = y = 1$ we will use more stronger upper bound from coding theory then the method give us better numerical values then numerical values from table 4. So for case $x = y = 1$ we can use the follows proposition.

Proposition 2. For $2 \leq w \leq r$ the rate

$$R(w, r) \leq \bar{R}(w, r)$$

where $\bar{R}(w, r)$ is defined recurrently:

if we know $\bar{R}(w-1, r-1)$ then $\bar{R}(w, r)$ is the unique solution of equation

$$\bar{R}(w, r) = H_2(1/2 - \sqrt{\frac{2\bar{R}(w,r)}{\bar{R}(w-1, r-1)}(1 - \frac{2\bar{R}(w,r)}{\bar{R}(w-1, r-1)})}),$$

such that $\bar{R}(w, r) < \bar{R}(w-1, r-1)/4$ where $H_2(x) = -x \log_2 x - (1-x)\log_2(1-x)$.

References

[1] A.G D'yachkov, V.V. Rykov, "Bounds on the Length of Disjunct Codes", *Problemy Peredachi Informatsii*, 17, 1982, n. 3, 7-13, (in Russian).

[2] A. D'yachkov, A. Macula, V. Rykov, “New Constructions of Superimposed Code”, *IEEE Trans. Inform. Theory*, vol 46, no. 1, pp. 284-290, 2000.

[3] A. D'yachkov, A. Macula, D. Torney, P. Vilenkin, S. Yekhanin, “New results in the theory of superimposed codes”, *Seventh International Workshop on Algebraic and Combinatorial Coding Theory*, June 18-24. 2000, Bansko (Bulgaria) pp. 126-136.

[4] K. Engel, "Interval Packing and Covering in the Boolean Lattice", *Combinatorics Prob. and Computing*, 5, 1996, 373-384.

[5] H.-D.O.F Gronau, R.S.Mullin, “On super-simple $2-(v, 4, \lambda)$ designs”, J.Combin. Math. Combin. Comput. 11 (1992), 113-121.

[6] H.K. Kim and V. Lebedev, "On Optimal Cover-free Families, Superimposed Codes and Key Distribution Patters", *submitted to J. Comb. Theory, ser. A.*

[7] G. Mago, "Monotone Function in Sequential Circuits", *IEEE Trans. Comput.*, vol 22, no. 10, pp. 928-933, 1973.

[8] Yu.L. Sagalovich “Separating systems”, *Problems of Information Transmission*, vol. 30, no. 2, pp. 105-123, 1994.

[9] Yu.L. Sagalovich, “Completely Separating systems”, *Problemy Peredachi Informatsii*, vol 18 (2), pp. 74-82, 1982. (in Russian).

[10] N.J.A. Sloane, “Covering arrays and intersecting codes”, *Journal of Combinatorial Designs*, vol. 1, pp. 51-63, 1993.

Iteratively Decodable Sliding Codes on Graphs[1]

Michael Lentmaier, Dmitri Truhachev, and Kamil Sh. Zigangirov
Dept. of Information Technology, Lund University
Box 118, SE-22100 Lund, Sweden
e-mail: {michael,dimitri,kamil}@it.lth.se

1 Introduction

When Gallager in the early 60's introduced low-density parity-check (LDPC) block codes [1], together with a probabilistic iterative algorithm to decode them, he demonstrated that good performances could be achieved with rather simple decoding methods. The convolutional code counterpart of Gallager's codes, low-density parity-check convolutional codes, were introduced in [2] and further developed and investigated in [3][4]. These codes form a large family of convolutional codes that includes, as a special case, the well-known turbo codes. The code sequences of the LDPC convolutional codes are produced by an encoder that is sliding over the information sequence and continuously generating the coded bits.

In this paper we consider a class of sliding codes, where instead of trivial single parity-check codes any arbitrary systematic binary codes of rate $R_c > 1/2$ can serve as component codes. For block codes, such a generalization of the original LDPC codes was introduced by Tanner [5]. In contrast to LDPC convolutional codes, which are defined through their parity-check matrix, the presented class of codes includes non-linear codes, that are neither convolutional codes nor allow a definition based on parity-check matrices.

As an illustrative example, we consider in the following in more detail the particular case where the component codes are Hamming codes. The resulting sliding codes are a convolutional code version of the generalized low-density parity-check (GLDPC) block codes, investigated in [6]. As a measure for the asymptotic iterative decoding performance, when the memory of the sliding codes tends to infinity, we calculate bounds on the iterative limits of the considered codes, based on the ideas given in [4]. We also present results from computer simulations for sliding codes with finite memory and compare them to corresponding GLDPC block codes.

2 Code Construction and Encoding Procedure

Both the construction of sliding codes and the encoding procedure can be illustrated by means of an infinite two-dimensional array. For (7,4) Hamming component codes, a sketch of such an array is given in Figure 1. Each column and each row contains seven symbols: one information and six parity-check symbols. The information symbols u_t are placed on the diagonal. In each column three parity-check symbols lie above the diagonal and three below. In the same way, in each row three parity-check symbols are to the left of diagonal and three symbols to the right.

At each time t all symbols above and to the left of the information symbol u_t have already been encoded before. The parity-check symbols v_t^{h1}, v_t^{h2}, and v_t^{h3} are chosen such, that the seven symbols in the tth row form a codeword of the component Hamming code. Analogously, the seven symbols in the tth column form a codeword of the component code. During the encoding procedure the encoder is continuously sliding along the diagonal. At each time

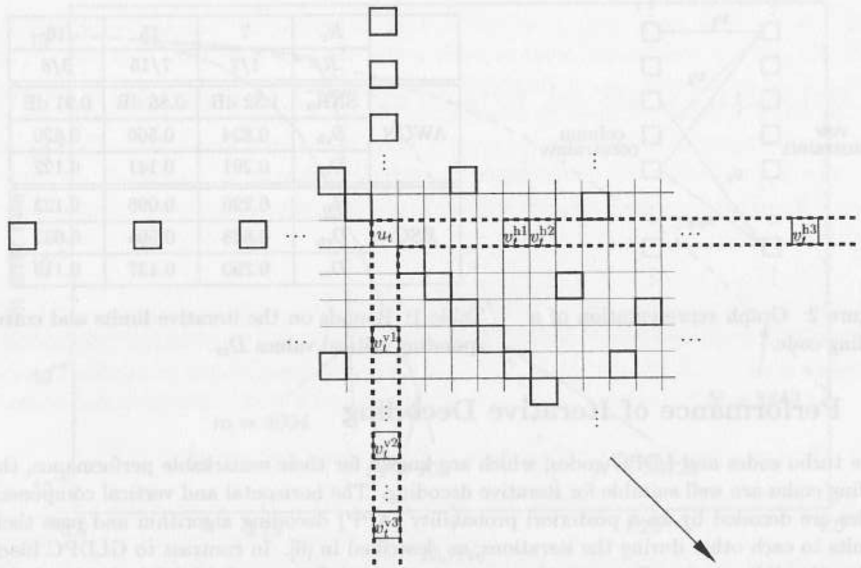


Figure 1: Illustration of the encoding scheme.

instance seven code symbols are generated. The code rate is $R = 2R_c - 1$, where $R_c = 4/7$ is the rate of the component code in our example.

This encoding procedure can be generalized to the case of arbitrary, rate $R_c > 1/2$, length N_c component codes. Then each row (column) consists of N_c symbols. The $K = (2R_c - 1)N_c$ middle symbols are information symbols. To the left (above) and to the right of (below) them are $(1 - R_c)N_c$ parity-check symbols. The array can be divided into three parallel diagonal type strips. The first strip has $(1 - R_c)N_c$ parity-check symbols in each column and row, the second (middle) strip $(2R_c - 1)N_c$ information symbols, and the third, last strip has again $(1 - R_c)N_c$ parity-check symbols in each column and row.

A particular code is now defined by the positions of the symbols in the rows and columns of the array. This can be represented by a corresponding infinite matrix having a one at each position containing a symbol, and zeros elsewhere. Such matrices can be constructed from elementary permutation matrices in a similar way as the parity-check matrices of LDPC convolutional codes [2][3]. We define the memory m of a sliding code to be the largest distance between t and the time index of a symbol used by the encoder at time t, in analogy to the syndrome former memory m_s of a LDPC convolutional code.

The sliding code resulting from the proposed encoding procedure can be described by an infinite Tanner graph and decoded iteratively by the component codes, as other codes on graphs. Another graph representation is given in Figure 2. All vertices correspond to constraints of the encoding scheme illustrated in Figure 1. Left vertices correspond to row constraints and right vertices to column constraints, resulting from a horizontal and vertical component codes, respectively. The edges correspond to code symbols. Each set of edges leaving a particular vertex forms a codeword of the component code. The code definition above assures, that the graph will have at most one edge between each pair of vertices. From this follows directly, that the corresponding Tanner graph has at least girth eight.

[1] This work was supported in part by the Swedish Research Council for Engineering Sciences (Grant 01-3123).

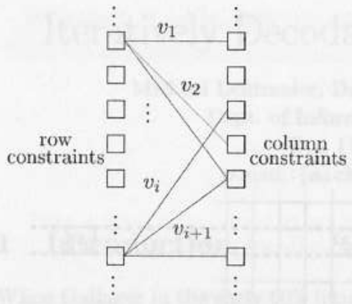


Figure 2: Graph representation of a sliding code.

	N_c	7	15	16
	R	1/7	7/15	3/8
AWGN	$\overline{\mathrm{SNR}_0}$	1.32 dB	0.86 dB	0.91 dB
	D_{ch}	0.824	0.566	0.630
	D_{cr}	0.291	0.141	0.122
BSC	$\underline{p_0}$	0.220	0.098	0.123
	D_{ch}	0.828	0.595	0.657
	D_{cr}	0.290	0.137	0.119

Table 1: Bounds on the iterative limits and corresponding critical values D_{cr}.

3 Performance of Iterative Decoding

Like turbo codes and LDPC codes, which are known for their remarkable performance, the sliding codes are well suitable for iterative decoding. The horizontal and vertical component codes are decoded by an a posteriori probability (APP) decoding algorithm and pass their results to each other during the iterations, as described in [6]. In contrast to GLDPC block codes, the sliding codes allow a continuous transmission (after some initial delay), when being decoded using a pipeline realization, like suggested for LDPC convolutional codes in [2]. In this case, the decoding iterations can be performed by independent processors in parallel.

Following the approach in [4], we calculated bounds on the iterative limits for sliding codes with Hamming components of length $N_c = 7$, 15 and 16, considering a two-phase algorithm. We proved, that if the Bhattacharyya parameter after the first decoding phase becomes smaller than some critical value D_{cr}, then the bit error probability P_b goes to zero at least double exponentially with the number of iterations I. For components of length N_c, the critical value D_{cr} is given by the smallest positive root of the equation

$$D_{cr} = D_{ch} \frac{A'(D_{cr})}{N_c} \ , \qquad (1)$$

where $A(D)$ is the weight enumerator function of the component code and $D_{ch} = \exp(-RE_b/N_0)$ (or $D_{ch} = \sqrt{4p(1-p)}$) is the Bhattacharyya parameter of the statistics received from the AWGN channel (or BSC with crossover probability p). Particularly, for $N_c = 7$,

$$\frac{A'(D_{cr})}{N_c} = 3D^2 + 4D^3 + D^6 \ . \qquad (2)$$

Together with a numerical analysis of the first decoding phase, based on Monte Carlo methods, we obtain upper bounds $\overline{\mathrm{SNR}_0}$ (lower bounds $\underline{p_0}$) on the iterative limits. The results are presented in Table 1. In the derivation of the bounds the decoding of an individual symbol is analyzed using a tree-like graph, which has the same structure for both sliding codes and GLDPC block codes. The bounds on the iterative limit are therefore valid for both cases.

Simulation results for iteratively pipeline decoded sliding codes with (15,11) Hamming component codes are presented in Figure 3. The simulated codes were picked randomly and decoded with 50 iterations. For comparison also results for GLDPC block codes are presented, taken from [6]. Even for a relatively small memory $m = 105$ the sliding codes achieve a bit error rate of 10^{-5} already at signal-to-noise ratios E_b/N_0 around 1 dB.

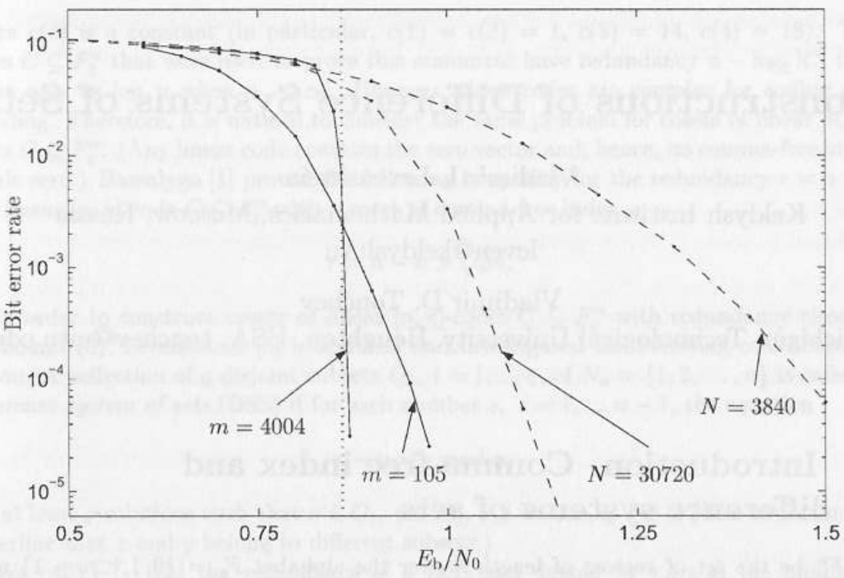


Figure 3: Performance of rate $R = 7/15$, memory m sliding codes (solid) and length N GLDPC block codes (dashed), based on (15,11) Hamming components. The vertical line shows the bound on the iterative limit.

References

[1] R. G. Gallager. *Low-Density Parity-Check Codes.* MIT Press, Cambridge, Mass., 1963.

[2] A. Jimenez and K. Sh. Zigangirov, "Periodic Time-Varying Convolutional Codes with Low-Density Parity-Check Matrices", *IEEE Trans. on Inform. Theory,* vol. IT-45, no. 5, pp. 2181–2190, Sept. 1999.

[3] K. Engdahl and K. Sh. Zigangirov, "On the Theory of Low-Density Convolutional Codes I", *Problems of Information Transmission,* vol. 35, no. 4, pp. 295–310, Oct-Nov-Dec 1999.

[4] M. Lentmaier, D. Truhachev, and K. Sh. Zigangirov. "On the Theory of Low Density Convolutional Codes II", *Problems of Information Transmission,* vol. 37, no. 4, pp. 288–306, October–December 2001.

[5] R. M. Tanner, "A recursive approach to low complexity codes," *IEEE Trans. on Inform. Theory,* vol. IT-27, no. 5, pp. 533–547, September 1981.

[6] M. Lentmaier, and K. Sh. Zigangirov, "On generalized low-density parity-check codes based on Hamming component codes", *IEEE Communications Letters,* Vol. 3, No. 8, pp. 248–250, August 1999.

Constructions of Difference Systems of Sets*

Vladimir I. Levenshtein
Keldysh Institute for Applied Mathematics, Moscow, Russia
leven@keldysh.ru

Vladimir D. Tonchev
Michigan Technological University, Houghton, USA, tonchev@mtu.edu

1 Introduction. Comma-free index and difference systems of sets

Let F_q^n be the set of vectors of length n over the alphabet $F_q = \{0, 1, ..., q-1\}$ and $d(x, y)$ be the Hamming metric. For any $x = x_1 \cdots x_n \in F_q^n$, $y = y_1 \cdots y_n \in F_q^n$, and $i = 0, 1, ..., n-1$, we denote $x_{i+1} \cdots x_n y_1 \cdots y_i$ by $T_i(x, y)$ and call $T_i(x, y)$ a *joint* of x and y. In particular, $T_i(x, x)$ is a *cyclic shift* of x. The *comma-free index* $\rho(C)$ of a code $C \subseteq F_q^n$ is defined as $\min d(z, T_i(x, y))$ where the minimum is taken over all $x, y, z \in F_q^n$ and $i = 1, ..., n-1$. We denote by $M_q(n, \rho)$ the maximum cardinality of a code $C \subseteq F_q^n$ with $\rho(C) = \rho$. The comma-free index $\rho(C)$ allows one to distinguish a current code word from a joint of two code words (and hence to provide synchronization of code words) if at most $\lfloor \rho(C)/2 \rfloor$ errors have occurred in a code word.

The problem of synchronization of words of a code $C \subseteq F_q^n$ was first introduced in the paper by Golomb, Gordon, and Welch [3] where *comma-free codes* are defined as codes with $\rho(C) = 1$ and it was proved that

$$M_q(n, 1) \le \frac{1}{n} \sum \mu(d) q^{d/n} \tag{1}$$

where the sum is taken over all divisors d of n and $\mu(d)$ is the Möbius function. Eastman [2] proved that this bound is tight for every $q \ge 2$ and every odd n. The best known upper and lower bounds on $M_2(n, \rho)$ for $\rho > 1$ are obtained by Levenshtein in [4]. In particular, he proved that for any fixed $\rho \ge 1$,

$$M_2(n, \rho) \gtrsim \frac{1}{c(\rho) e} \frac{2^n}{n} \quad \text{as} \quad n \to \infty \tag{2}$$

where $c(\rho)$ is a constant (in particular, $c(1) = c(2) = 1$, $c(3) = 14$, $c(4) = 18$). The codes $C \subseteq F_2^n$ that were used to prove this statement have redundancy $n - \log_2 |C|$ that grows only as $\log_2 n$ when $n \to \infty$. However, these codes are complex for coding and decoding. Therefore, it is natural to consider the same problem for cosets of linear $[n, k]$-codes $C \subseteq F_q^n$. (Any linear code contains the zero vector and, hence, its comma-free index equals zero.) Bassalygo [1] proved the following inequality for the redundancy $r = n - k$ of a linear $[n, k]$ code $C \subseteq F_q^n$ with a coset of comma-free index ρ:

$$r = n - k > \sqrt{\rho n}. \tag{3}$$

In order to construct cosets of linear $[n, k]$-codes $C \subseteq F_q^n$ with redundancy close to the bound (3), Levenshtein [5] introduced and investigated the following combinatorial notion. A collection of q disjoint subsets Q_i, $i = 1, ..., q$, of $N_n = \{1, 2, \ldots, n\}$ is called a *difference system of sets* (DSS) if for each number s, $s = 1, ..., n-1$, the equation

$$x - y = s \bmod n \tag{4}$$

has at least ρ solutions such that $x \in Q_i$, $y \in Q_j$, $i, j = 1, ..., q$, $i \ne j$. (It is important to underline that x and y belong to different subsets.)

We call $\sum_{i=1}^q |Q_i|$ the *redundancy* of a DSS and denote by $r_q(n, \rho)$ the minimum redundancy of a DSS with parameters n, q, and ρ. A DSS is *optimal* if its redundancy is equal to $r_q(n, \rho)$. Given a DSS with redundancy r, we can define a linear code $C \subseteq F_q^n$ with redundancy r whose information positions have the numbers from $N_n \setminus \cup_{i=1}^q Q_i$, and the remaining positions are zeros. Then the sets Q_i, $i = 1, ..., q$, determine a coset of the linear code C and ensure that the comma-free index of this coset is at least ρ.

A DSS is called *perfect* if for every number s, $1 \le s \le n-1$, the equation (4) has exactly ρ solutions. A DSS is called *regular* if all subsets Q_i are of the same size. We use the notation DSS-(n, m, q, ρ) for a regular DSS with q subsets of size m on the set N_n and parameter ρ; its redundancy equals $r = qm$. Any cyclic (v, k, λ) difference set (cf., e.g. [6]) is a perfect regular DSS-$(v, 1, k, \lambda)$. Thus DSS are a generalization of cyclic difference sets. Levenshtein [5] proved that for any DSS with parameters n, q, and ρ,

$$r_q(n, \rho) \ge \sqrt{\frac{q\rho(n-1)}{q-1}} \tag{5}$$

with equality if and only if the DSS is perfect and regular. It follows that any perfect regular DSS is optimal. In [5], optimal DSS were found for $q = 2$ and $\rho = 1$ or $\rho = 2$, and for all $n \ge 2$, and it was proved that

$$r_2(n, 1) = \rceil \sqrt{2(n-1)} \lceil, \quad r_2(n, 2) = \rceil 2\sqrt{n-1} \lceil. \tag{6}$$

Similar results are not known for $q \ge 3$, although some examples of regular perfect DSS-$(n, m, q, 1)$ have been found by computer. For instance, the sets $Q_0 = \{1, 10\}$,

*Research supported by The Russian Foundation for Basic Research, grant 02-01-00687, and by the NRC COBASE research grant INT-00023241.

$Q_1 = \{2, 19\}$, $Q_2 = \{4, 15\}$ form a regular perfect DSS-(25, 2, 3, 1). It seems natural to expect that for any fixed $q \geq 3$

$$r_q(n, 1) = O(\sqrt{n}) \quad \text{as } n \to \infty. \qquad (7)$$

Note that from (5) (or (3)) it follows that $r_q(n, 1) \gtrsim \sqrt{n}$ as $q \to \infty$. This bound is asymptotically tight for the cyclic difference sets $(q^2 - q + 1, q, 1)$ where $q - 1$ is a prime power. However, in this case q grows as $\sqrt{n}$. For applications in coding theory, it is significant to prove (7) when q is fixed or grows slowly in comparison with n.

2 DSS from cyclic difference sets

Theorem 2.1 *The existence of a cyclic (v, q, ρ) difference set implies that for every $h = 2, 3, \ldots$ there exists a regular DSS-(n, m, q, ρ) with $n = v^h$ and $m = (q^h - 1)/(q - 1)$.*

The proof of this theorem is constructive. The DSS described in Theorem 2.1 is regular but not perfect. In particular, one can use as (v, q, ρ) the cyclic difference sets $(v = p^2 + p + 1, q = p + 1, \rho = 1)$ related to the projective plane of order p, where p is an arbitrary prime power. Then, for a suitable choice of h, the construction above implies that $r_q(n, 1) = O(\sqrt{n})$ for $q = O\left(\frac{\ln n}{\ln(\ln n)}\right)$ when $n \to \infty$.

The next construction gives perfect and regular DSS obtained as partitions of cyclic difference sets.

Lemma 2.2 *Let $D \subseteq \{1, 2, \ldots, n\}$, $|D| = k$, be a cyclic (n, k, λ) difference set. Assume that D can be partitioned into q disjoint subsets $Q_1, \ldots, Q_q$ that are the base blocks of a cyclic pairwise balanced 2-$(n, \{\tau_1, \ldots, \tau_q\}, \lambda_1)$ design, where $\tau_i = |Q_i|$, $i = 1, 2, \ldots, q$. Then the sets $Q_1, \ldots, Q_q$ form a perfect DSS with parameters $(n, \tau_1, \ldots, \tau_q, \rho = \lambda - \lambda_1)$.*

The following theorem gives infinitely many perfect DSS obtained by partitioning the trivial cyclic $(n, n-1, n-2)$ difference set.

Theorem 2.3 *Let $n = mq + 1$ be a prime, and let α be a primitive element of $GF(n)$. The collection of sets $Q_1 = \{\alpha^q, \alpha^{2q}, \ldots, \alpha^{mq} = 1\}$, $Q_2 = \alpha Q_1$, $\ldots$, $Q_q = \alpha^{q-1} Q_1$ is a regular perfect DSS-$(n, m, q, \rho = n - m - 1)$.*

The DSS in Theorem 2.3 is formally optimal but it has redundancy $r = n - 1$. The following example suggests that it is sometimes possible to obtain a DSS with smaller redundancy as a sub-collection of the DSS described in Theorem 2.3.

Example 2.4 Let $n = 19$, $q = 6$, $m = 3$. The DSS from Theorem 2.3 has $\rho = 15$, and the six sets Q_i of size 3 are $\{1, 7, 11\}$, $\{2, 14, 3\}$, $\{4, 9, 6\}$, $\{5, 16, 17\}$, $\{8, 18, 12\}$, $\{10, 13, 15\}$. The two sets $\{1, 7, 11\}$, $\{2, 14, 3\}$ form a regular perfect DSS-(19,3,2,1) with the minimum redundancy $r = 6$ (see (5)).

The following theorem is based on a construction that uses Singer difference sets and finite geometry and yields regular perfect DSS with substantially smaller redundancy $r_q(n, \rho)$.

Theorem 2.5 *For every prime power p and every integer $s > 1$ there exists a perfect regular DSS with parameters $n = \frac{p^{2s+1} - 1}{p - 1}$, $m = p + 1$, $q = \frac{p^{2s} - 1}{p^2 - 1}$, $\rho = \frac{p^{2s-1} - p}{p - 1}$.*

The proof uses a partition of a hyperplane in the projective space $PG(2s, p)$ over $GF(p)$ into disjoint lines, each belonging to a different Singer cycle.

Example 2.6 Let $p = 2$, $s = 2$. We consider $1, \alpha, \alpha^2, \ldots, \alpha^{30}$ as points of $PG(4, 2)$, where α is a primitive element of $GF(2^5)$ defined by the polynomial $x^5 + x^3 + 1$. The following set of 15 points $H = \{1, \alpha, \alpha^2, \alpha^3, \alpha^5, \alpha^7, \alpha^{11}, \alpha^{14}, \alpha^{15}, \alpha^{16}, \alpha^{22}, \alpha^{23}, \alpha^{26}, \alpha^{28}, \alpha^{29}\}$ is a hyperplane in $PG(4, 2)$, and hence a $(31, 15, 7)$ difference set in the multiplicative group of $GF(2^5)$. The following partition of H,

$$H = \{1, \alpha^2, \alpha^{28}\} \cup \{\alpha, \alpha^5, \alpha^{26}\} \cup \{\alpha^3, \alpha^{11}, \alpha^{22}\} \cup \{\alpha^7, \alpha^{14}, \alpha^{23}\} \cup \{\alpha^{15}, \alpha^{16}, \alpha^{29}\}$$

has the property that each of the five 3-subsets is a projective line, and these five lines are the base blocks of a cyclic 2-$(31, 3, 1)$ design under the multiplicative group of $GF(2^5)$. Thus, these five 3-subsets define a regular perfect DSS with parameters $n = 31, m = 3, q = 5, \rho = 6$.

The construction of Theorem 2.5 can be generalized by using partitions of hyperplanes in $PG(n, p)$ into subspaces of dimension larger than one.

References

[1] L.A. Bassalygo, “On the separation of comma-free codes”, *Probl. Peredachi Inform.*, vol. 2, no. 4, pp. 78–79, 1966 (in Russian).

[2] W.L. Eastman, “On the construction of comma-free codes”, *IEEE Trans. on Inform. Theory*, vol. IT-11, pp. 263–266, 1965.

[3] S.W. Golomb, B. Gordon, L.R. Welch, “Comma-free codes”, *Canad. J. Math.*, vol. 10, no. 2, pp. 202–209, 1958.

[4] V.I. Levenshtein, “Bounds for codes providing error correction and synchronization”, *Problemy Peredachi Informatsii*, vol. 5, no. 2, pp. 3-13, 1969 (in Russian).

[5] V.I. Levenshtein, “One method of constructing quasilinear codes providing synchronization in the presence of errors”, *Problemy Peredachi Informatsii*, vol. 7, no. 3, pp. 30–40, 1971 (in Russian).

[6] V. D. Tonchev, “Combinatorial Configurations”, Wiley, New York 1988.

Identifying vertices in the binary n-cube

Iiro Honkala[*]
Department of Mathematics
University of Turku
20014 Turku, Finland
e-mail: honkala@utu.fi

Antoine Lobstein
CNRS and ENST
46 rue Barrault
75013 Paris, France
e-mail: lobstein@infres.enst.fr

Abstract – A code $C \subseteq \mathbb{F}_2^n$ is called r-identifying if the sets $B_r(x) \cap C$, where $B_r(x)$ is the ball of radius r centred at x, are all nonempty and distinct. Let $M_r(n)$ be the minimum cardinality of an r-identifying code in $\mathbb{F}_2^n$. We prove that if $\rho \in [0,1)$ is a constant, then $\lim_{n\to\infty} n^{-1}\log_2 M_{\lfloor \rho n\rfloor}(n) = 1 - H(\rho)$.

1 Introduction

Denote by $\mathbb{F}_2$ the binary alphabet $\{0,1\}$, by d the Hamming distance, and let

$$B_r(x) = \{y \in \mathbb{F}_2^n : d(y,x) \le r\},\ \ S_r(x) = \{y \in \mathbb{F}_2^n : d(y,x) = r\},$$

$$V(n,r) = |B_r(x)| = \sum_{i=0}^{r}\binom{n}{i}.$$

The problem of constructing *identifying codes* in $\mathbb{F}_2^n$ and other graphs was introduced in [14], and has been studied in a number of papers since. We say that a code C is r-identifying if the sets

$$I(x) := B_r(x) \cap C$$

are nonempty for all $x \in \mathbb{F}_2^n$ and no two of them are the same. If $r = n$, there are clearly no r-identifying codes. The minimum cardinality of an r-identifying code in $\mathbb{F}_2^n$ is denoted by $M_r(n)$.

We call C an *r-covering*, if the union of the sets $B_r(c)$, $c \in C$, is the whole space $\mathbb{F}_2^n$. In other words, for all $x \in \mathbb{F}_2^n$, the sets $B_r(x) \cap C$ are nonempty.

In particular, for the same code length n, the set of r-identifying codes is included in the set of r-covering codes.

In the Hamming space, the study of $M_r(n)$ and some related problems have been considered, among others, in [14], [2], [9]–[11], [15]–[17].

It is known that for any fixed r, there are constants α and β (depending on r) such that for all large n,

$$\frac{\alpha 2^n}{V(n,r)} \le M_r(n) \le \frac{\beta 2^n}{V(n,r)},$$

but there is no value of r for which the known values α and β would coincide. For $r = 1$, there is a sequence (n_i) such that $n_i \to \infty$ and

$$\lim_{i\to\infty} \frac{M_1(n_i)}{2^{n_i}/V(n_i,1)} = 1.$$

This follows from [14, Corollary 1] using the covering codes described in [6, Theorem 4.5.8], which in turn are obtained using codes from [8].

In this paper we consider the problem asymptotically, when ρ is a constant, $0 \le \rho < 1$, and $r = \lfloor \rho n \rfloor$. Based on a non-constructive argument, we prove that

$$\lim_{n\to\infty} n^{-1}\log_2 M_{\lfloor \rho n\rfloor}(n) = 1 - H(\rho),$$

where $H(x) = -x\log_2 x - (1-x)\log_2(1-x)$ is the binary entropy function.

2 A Construction of Asymptotically Optimal Identifying Codes

The following theorem is not new: cf. Theorems 12.1.2, 12.2.1 and 20.3.4 and the corresponding Notes in Sections 12.9 and 20.4 in [6] as well as [3]–[5].

Theorem 2.1 *Assume that $L \subseteq \{0,1,2,\dots,n\}$ and $L \ne \emptyset$. There exists a code $C \subseteq \mathbb{F}_2^n$, with cardinality K satisfying*

$$K \le \lceil n2^n \ln 2 / \sum_{i\in L}\binom{n}{i} \rceil, \ \ \textit{such that} \ \bigcup_{c\in C}\bigcup_{i\in L} S_i(c) = \mathbb{F}_2^n.$$

Proof. It follows [7]; see [12]. □

Our construction is based on using codes with the property that every vector in the Hamming space has distance exactly r or $r+1$ to at least one codeword.

Theorem 2.2 *If $0 < r \le n-2$ and $C_0 \subseteq \mathbb{F}_2^n$ has the property that*

$$\bigcup_{c\in C_0}(S_r(c) \cup S_{r+1}(c)) = \mathbb{F}_2^n, \ \textit{then the code}\ \ C = \bigcup_{c\in C_0} S_1(c) \ \ \textit{is r-identifying.}$$

[*]Research supported by the Academy of Finland under grant 44002.

Classification of linear codes by preclassification

Stefano Marcugini, Alfredo Milani, Fernanda Pambianco
Dipartimento di Matematica e Informatica,
Università degli Studi di Perugia,
Via Vanvitelli 1, 06123 Perugia Italy
e-mail: {gino, milani, fernanda}@dipmat.unipg.it

Abstract. We consider the problem of computing the equivalence classes of a set of linear codes. This problem arises when new codes are obtained extending codes of lower dimension. We propose a technique that, exploiting an invariant simple to compute, allows to reduce the computational complexity of the classification process. Using this technique the $[13, 5, 8]_7$, the $[14, 5, 9]_8$ and the $[15, 4, 11]_9$ codes have been classified. The same technique can be applied to the problem of the classification of other structures.

1 Introduction

This paper deals with the problem of classifying sets of linear codes. This problem arise, for example, using computer-based extension processes that construct new codes of dimension d_1 starting from codes of dimension d_2, $d_2 < d_1$. For examples of papers using such technique see [2], [3], [5] and [7]. In particular in [5] and [7], we constructed new near maximum-distance separable (NMDS) codes adding new rows to the generating matrix of NMDS codes of lower dimension. For a description of the properties of the NMDS codes see [3] and references therein.

When extending a code in this way, several equivalent copies of the same code are obtained. A classification step allows to compute the set of nonequivalent codes, but, when the number of examples to classify is high, some strategy has to be adopted to reduce the computational complexity of this phase.

The most direct and simple algorithm that can be used for the classification of a set S of codes keeps a list L of nonequivalent codes. Initially L is empty. All the codes C of S are considered: if there exists a code in L equivalent to C, than C is neglected, otherwise C is included in L. At the end L contains the set of representatives of the equivalence classes of S. The computational complexity of this simple algorithm is $O(|S| \times |L|)$, therefore it can be used only when $|S|$ and $|L|$ are relatively small. In [2], the program described in [1] has been used. It deals with the problem of computing equivalence between codes exploiting invariants and signatures. In [4] a set of invariants has been introduced that allows to determine if binary codes of dimension three are equivalent or not.

To reduce the computational complexity of the classification step, we propose a technique of preclassification based on the use of an invariant. The condition on the invariant is that it must be easier to compute than the equivalence between two codes. In our case we used the minimum weight of the code.

Using the invariant in opportune way, the set S of the codes to classify is partitioned in subsets S_i such that $C_1 \in S_i$ and $C_2 \in S_j$ are not equivalent if $i \neq j$. Then it is sufficient to classify separately the codes in each S_i and the set of the the representatives of the equivalence classes in S is the union of the sets of the representatives of the equivalence classes in S_i. If each S_i contains only one equivalence class, the computational complexity of the classification step is $O(|S|)$. In our practical applications we finished with almost all the subsets S_i containing one or just a few equivalence classes. There is an adjunctive cost, the computation of the invariant for the codes of S and for several truncated codes, but it remains negligible respect to the cost of the classification phase.

Our technique is of general interest. In fact not only different invariants can be applied, but also other computational classification problems can be faced, as long as there is a way to construct substructures preserving the invariant property.

The preclassification technique is described in Section 2. Section 3 contains some experimental results concerning the classification of the $[13, 5, 8]_7$, of the $[14, 5, 9]_8$ and of the $[15, 4, 11]_9$ codes.

2 Preclassification using an invariant

Our aim was the classification of a set of codes S. Two $[n, k, d]_q$ codes, C_1 with generating matrix G_1 and C_2 with generating matrix G_2, have been considered equivalent in monomial sense, i.e. if there exist an invertible $(k, k)-$matrix A, an $(n, n)-$permutation matrix P and a field automorphism φ such that $G_1 = \varphi(AG_2P)$.

To reduce the number of the expensive computations of the equivalence between two codes, we used a numeric invariant easy to compute (the minimum weight of the code in our case) to divide the set of codes S in subsets S_i such that all the codes in S_i have the same value i of the invariant. Then to classify the codes in S it is sufficient to classify separately the codes in each S_i. The best situation is when each S_i contains only one equivalence class because the computational complexity of the classification step becomes $O(|S|)$. But if the invariant is simple this will not be the case. Some S_i will contain a number of equivalence class of the same order of magnitude of S. Then we used the same invariant to further divide each S_i. To do that we exploited the fact that if C_1 and C_2 are equivalent $[n, k]$ codes and $\overline{C}_1$ is an $[n-1, k]$ code obtained from C_1 by truncating, then there exists an $[n-1, k]$ code $\overline{C}_2$ obtained from C_2 by truncating such that $\overline{C}_1$ and $\overline{C}_2$ are equivalent. This fact follows immediately from the definition of equivalence.

Then for each code C in each S_i we computed a first level index defined as the sum of the minimum weights of the n $[n-1, k]$ subcodes obtained truncating C by deleting a column of the generating matrix in all the possible ways. If two codes C_1 and C_2 have different first level index, then they are not equivalent. In this way each S_i can be divided in subsets S_{i_j} such that $C_1 \in S_{i_j}$ and $C_2 \in S_{i_k}$ are not equivalent if $j \neq k$.

The process can be iterated computing the second level index defined as the sum of the minimum weights of the $n*(n-1)/2$ $[n-2, k]$ subcodes obtained by truncating from an $[n, k]$ code C deleting two columns of the generating matrix in all the possible ways and so on. Exploiting the indices of different levels the initial set of codes S is partitioned in subsets containing an ever-decreasing number of equivalence classes.

The computational cost of the computing of the index of order i of an $[n, k]$ code is $O\left(\binom{n}{i}\right)$. In the practical application we verified that it is sufficient to consider relatively small values of i to obtain sets of codes containing one or just a few numbers of equivalence classes. We note that two codes can have the same index of level i, but different indices of level j, $j < i$. Therefore when doing the preclassification it is useful considering all the indices belonging to the interval $[1, i]$ and not only the index of maximum value i.

3 Experimental results

This section describes the application of our preclassification technique for the classification of the $[13,5,8]_7$, of the $[14,5,9]_8$ and of the $[15,4,11]_9$ codes.

All our computations have been done using MAGMA, a computer algebra package developed at the University of Sydney. In [5], extending the 923 nonequivalent $[11,3,8]_7$ codes, we obtained 80326 examples of $[13,5,8]_7$ codes such that any other $[13,5,8]_7$ code is equivalent to one of our examples. In analogous way we obtained 4331 examples of $[14,5,9]_8$ codes and 69471 examples of $[15,4,11]_9$ codes extending respectively the 4181 $[12,3,9]_8$ codes and the 105193 $[14,3,11]_9$ codes found in [6].

Table 1 contains, for each set S of codes, the number of examples to classify, the number of classes obtained, the number of levels used in the preclassification step, the running time T_P, in hours, of the preclassification step, the running time T_C, in hours, of the classification step, and the ratio between the two running times. The duration of the preclassification does not exceed the duration of the classification step.

Code	$\|S\|$	Classes	Levels	T_P	T_C	Ratio
$[13,5,8]_7$	80326	988	6	111	600	18.5%
$[14,5,9]_8$	4331	58	4	3.5	48	7.3%
$[15,4,11]_9$	69471	6585	5	140	168	83.3%

Table 1: Running time of the classification of the codes

Table 2 contains, for each set S of codes, the number of sets obtained in each level of the preclassification step. In the first and in the second case we stopped the preclassification when the number of sets obtained at the current level is almost equal to the number of codes of the previous level. This means that we can expect, as it is confirmed by Table 3, that almost all the sets contain just one class, the desired condition.

		Level					
		0	1	2	3	4	5
Code	$[13,5,8]_7$	13	156	343	565	664	690
	$[14,5,9]_8$	13	39	49	55		
	$[15,4,11]_9$	16	196	681	1464	2570	

Table 2: Number of sets obtained at level k

Table 3 contains, for each set S of codes, the number of sets obtained in the last step of the preclassification that contain k classes. In the first and in the second case almost all the sets contains one class. This means that the computational cost of the classification step is $O(|S|)$. In the third case almost all the sets contains a small number of classes, therefore the computational cost of the classification step is near to $O(|S|)$.

		Classes					
		1	2	3	$4-10$	$11-20$	$21-89$
Code	$[13,5,8]_7$	571	67	17	30	5	
	$[14,5,9]_8$	52	3				
	$[15,4,11]_9$	1690	365	160	262	56	37

Table 3: Number of sets of maximum level containing k classes

Acknowledgments. This research is supported by Italian MIUR and GNSAGA. The authors would like to thank Prof. D. Betten for his hints and suggestions.

References

[1] I. Bouyukliev, Q-extension – strategy in algorithms, Proc. VII ACCT (2000), 84-88.

[2] I. Bouyukliev and J. Simonis, Some new results on optimal codes over F_5, preprint.

[3] S.M. Dodunekov and I. Landjev, Near-MDS codes over some small fields, Discrete Math. 213 (2000), 55-65.

[4] J. Maks and J. Simonis, Polynomial invariants for binary linear codes, Proc. International Workshop on Coding and Cryptography (2001).

[5] S. Marcugini, A. Milani and F. Pambianco, Existence and classification of NMDS codes over $GF(5)$ and $GF(7)$, Proc. VII ACCT (2000), 232–239.

[6] S. Marcugini, A. Milani, F. Pambianco, Classification of the [n,3,n-3]q NMDS codes over GF(7), GF(8) and GF(9), Ars Combinatoria, 61, (2001).

[7] S. Marcugini, A. Milani and F. Pambianco, NMDS Codes of Maximal Lenght over F_q, $8 \leq q \leq 11$, IEEE Trans. Inform. Theory, vol. 48 n. 4 (2002).

Extendability of ternary linear codes

Tatsuya Maruta*

Department of Information Systems, Aichi Prefectural University
Nagakute, Aichi 480-1198, Japan
E-mail: maruta@ist.aichi-pu.ac.jp

Abstract

There are at least four diversities of ternary linear codes of dimension k, minimum distance d with $3 \le k \le 6$, $\gcd(3,d) = 1$, which are always extendable. Every $[n,k,d]_3$ code with diversity $(\theta_{k-2}, 0), (\theta_{k-3}, 2\cdot 3^{k-2})$ or $(\theta_{k-2} + 3^{k-2}, 3^{k-2})$, $d \equiv 1 \pmod 3$, $k \ge 3$ is doubly extendable, where $\theta_j = (3^{j+1} - 1)/2$.

1. Introduction

Let $\mathcal{C}$ be an $[n,k,d]_q$ code, that is a linear code over $\mathrm{GF}(q)$ of length n with dimension k whose minimum Hamming distance is d, where $\mathrm{GF}(q)$ stands for the finite field of order q. The weight distribution of $\mathcal{C}$ is the list of numbers A_i which is the number of codewords with weight i. The weight distribution with $(A_0, A_d, ...) = (1, \alpha, ...)$ is also expressed as $0^1 d^\alpha \cdots$. We only consider *non-degenerate* codes having no coordinate which is identically zero.

The code obtained by deleting the same coordinate from each codeword of $\mathcal{C}$ is called a *punctured code* of $\mathcal{C}$. If there exists an $[n+1, k, d+1]_q$ code $\mathcal{C}'$ which gives $\mathcal{C}$ as a punctured code, $\mathcal{C}$ is called *extendable* (to $\mathcal{C}'$) and $\mathcal{C}'$ is an *extension* of $\mathcal{C}$. $\mathcal{C}$ is *doubly extendable* if its extension is also extendable. Throughout this paper we deal with $[n,k,d]_3$ codes with $\gcd(3,d) = 1$, $k \ge 3$ (see [6] for the case $k \le 2$).

Let $\mathcal{C}$ be an $[n,k,d]_3$ code with $k \ge 3$, $\gcd(3,d) = 1$. Define

$$\Phi_0 = \frac{1}{2} \sum_{3|i, i\neq 0} A_i, \quad \Phi_1 = \frac{1}{2} \sum_{i \not\equiv 0, d \ (\mathrm{mod}\ 3)} A_i.$$

We call the pair (Φ_0, Φ_1) the *diversity* of $\mathcal{C}$. It is known that $\mathcal{C}$ is extendable if $\Phi_1 = 0$ ([3],[4]) or if $\Phi_0 + \Phi_1 < \theta_{k-2} + 3^{k-2}$ ([7],[9]), where $\theta_j = (3^{j+1} - 1)/2$. Our aim is to find all the possibilities of the diversity for which $\mathcal{C}$ is extendable for $k = 3, 4, 5, 6$. Note that $\theta_{-1} = 0, \theta_0 = 1, \theta_1 = 4, \theta_2 = 13, \theta_3 = 40, \theta_4 = 121, \theta_5 = 364$. Our main results are summarized in the following theorems.

Theorem 1.1. *Let $\mathcal{C}$ be an $[n,k,d]_3$ code with diversity (Φ_0, Φ_1), $gcd(3,d) = 1$, $3 \le k \le 6$. Then*
(1) $(\Phi_0, \Phi_1) \in \{(4,0), (1,6), (4,3), (4,6), (7,3)\}$ *when* $k = 3$.
(2) $(\Phi_0, \Phi_1) \in \{(13,0), (4,18), (13,9), (10,15), (16,12), (13,18), (22,9)\}$ *when* $k = 4$.
(3) $(\Phi_0, \Phi_1) \in \{(40,0), (13,54), (40,27), (31,45), (40,36), (40,45), (49,36), (40,54), (67,27)\}$ *when* $k = 5$.
(4) $(\Phi_0, \Phi_1) \in \{(121,0), (40,162), (121,81), (94,135), (121,108), (112,126), (130,117), (121,135), (148,108), (121,162), (202,81)\}$ *when* $k = 6$.
(5) *$\mathcal{C}$ is extendable if* $(\Phi_0, \Phi_1) \in \{(\theta_{k-2}, 0), (\theta_{k-3}, 2\cdot 3^{k-2}), (\theta_{k-2}, 2\cdot 3^{k-2}), (\theta_{k-2} + 3^{k-2}, 3^{k-2})\}$.

Theorem 1.2. *Let $\mathcal{C}$ be an $[n,k,d]_3$ code with diversity (Φ_0, Φ_1), $gcd(3,d) = 1$. Then $\mathcal{C}$ is not extendable if* (1) $\sum_{d<i\equiv d (\mathrm{mod}\ 3)} A_i = 0$ *when* $(\Phi_0, \Phi_1) = (4,3)$ *for* $k = 3$,
(2) $\sum_{d<i\equiv d (\mathrm{mod}\ 3)} A_i < 6$ *when* $(\Phi_0, \Phi_1) \in \{(13,9), (10,15), (16,12)\}$ *for* $k = 4$,
(3) $\displaystyle\sum_{d<i\equiv d (\mathrm{mod}\ 3)} A_i < 18$ *if* $(\Phi_0, \Phi_1) \in \{(40,27), (31,45), (40,45), (49,36)\}$ *for* $k = 5$,
(4) $\displaystyle\sum_{d<i\equiv d (\mathrm{mod}\ 3)} A_i < 24$ *if* $(\Phi_0, \Phi_1) = (40,36)$ *for* $k = 5$,
(5) $\displaystyle\sum_{d<i\equiv d (\mathrm{mod}\ 3)} A_i < 54$ *when* $(\Phi_0, \Phi_1) \in \{(121,81), (94,135), (121,135), (148,108)\}$ *for* $k = 6$,
(6) $\displaystyle\sum_{d<i\equiv d (\mathrm{mod}\ 3)} A_i < 72$ *when* $(\Phi_0, \Phi_1) \in \{(121,108), (112,126), (130,117)\}$ *for* $k = 6$.

Theorem 1.3. *Let $\mathcal{C}$ be an $[n,k,d]_3$ code with diversity (Φ_0, Φ_1), $d \equiv 1$ (mod 3), $k \ge 3$. Then $\mathcal{C}$ is doubly extendable if* $(\Phi_0, \Phi_1) \in \{(\theta_{k-2}, 0), (\theta_{k-3}, 2\cdot 3^{k-2}), (\theta_{k-2} + 3^{k-2}, 3^{k-2})\}$.

For example, a $[200, 6, 130]_3$ code found by Gulliver ([2]) is doubly extendable, for its weight distribution is $0^1 130^{144} 131^{224} 132^{64} 133^{112} 134^{32} 139^{32} 140^{64} 142^{32} 144^8 148^4 150^8 152^4$ (diversity $(40, 162)$). Hence a $[202, 6, 132]_3$ code exists.

Theorem 1.3 can be proved quite similarly to the proof of Theorem 1.2 in [8], so we omit the proof in this paper.

2. Diversities of ternary linear codes of dimension k, $3 \le k \le 6$

We denote by $\mathrm{PG}(r,q)$ the projective geometry of dimension r over $\mathrm{GF}(q)$. A *j-flat* is a projective subspace of dimension j in $\mathrm{PG}(r,q)$. 0-flats, 1-flats, 2-flats, 3-flats and $(r-1)$-flats are called *points*, *lines*, *planes* and *hyperplanes* respectively as usual. We denote by $\mathcal{F}_j$ the set of j-flats of $\mathrm{PG}(r,q)$.

Let $\mathcal{C}$ be an $[n,k,d]_q$ code with a generator matrix G. Then the columns of G can be considered as a multiset of n points in $\Sigma = \mathrm{PG}(k-1,q)$ denoted also by $\mathcal{C}$. An *i-point* is a point of Σ which has multiplicity i in $\mathcal{C}$. Denote by γ_0 the maximum multiplicity of a point from Σ in $\mathcal{C}$ and let C_i be the set of i-points in Σ, $0 \le i \le \gamma_0$. For any subset S of Σ we define *the multiplicity of S with respect to $\mathcal{C}$*, denoted by $m_{\mathcal{C}}(S)$, as

$$m_{\mathcal{C}}(S) = \sum_{i=1}^{\gamma_0} i \cdot |S \cap C_i|,$$

Then we obtain the partition $\Sigma = C_0 \cup C_1 \cup \cdots \cup C_{\gamma_0}$ such that

$$n = m_{\mathcal{C}}(\Sigma), \quad n - d = \max\{m_{\mathcal{C}}(\pi) \mid \pi \in \mathcal{F}_{k-2}\}.$$

Conversely such a partition of Σ as above gives an $[n,k,d]_q$ code in the natural manner if there exists no hyperplane including the complement of C_0 in Σ. Note that the number of hyperplanes π with $m_{\mathcal{C}}(\pi) = i$ is equal to $A_{n-i}/(q-1)$. Since $(n+1) - (d+1) = n - d$, we get the following.

Lemma 2.1. *$\mathcal{C}$ is extendable iff there exists a point $P \in \Sigma$ such that $m_{\mathcal{C}}(\pi) < n - d$ for all hyperplanes π through P.*

[*] This research has been partially supported by Grant-in-Aid for Scientific Research of the Ministry of Education under Contract Number 304-4508-12640137

Let Σ^* be the dual space of Σ (considering $\mathcal{F}_{k-2}$ as the set of points of Σ^*). Then Lemma 2.1 is equivalent to the following:

Lemma 2.2. *$\mathcal{C}$ is extendable iff there exists a hyperplane Π of Σ^* such that*

$$\Pi \subset \{\pi \in \mathcal{F}_{k-2} \mid m_{\mathcal{C}}(\pi) < n-d\}.$$

From now on, we assume that $\mathcal{C}$ is an $[n,k,d]_3$ code with diversity (Φ_0,Φ_1), $\gcd(3,d)=1$, $3 \le k \le 6$. Then $n-d \not\equiv n \pmod 3$. Define F_0 and F_1 as follows.

$$F_0 = \{\pi \in \mathcal{F}_{k-2} \mid m_{\mathcal{C}}(\pi) \equiv n \pmod 3\},\quad F_1 = \{\pi \in \mathcal{F}_{k-2} \mid m_{\mathcal{C}}(\pi) \not\equiv n,\ n-d \pmod 3\}.$$

Then $F = F_0 \cup F_1$ forms a blocking set with respect to lines in the dual space Σ^* of $\Sigma = \mathrm{PG}(k-1,3)$ ([7]), that is, every line of Σ^* meets F in at least one point of Σ^*. The following is straightforward from Lemma 2.2.

Lemma 2.3. *$\mathcal{C}$ is extendable if F includes a hyperplane of Σ^*.*

$\mathcal{C}$ is extendable if $F \cup \{\pi \in \mathcal{F}_{k-2} \mid m_{\mathcal{C}}(\pi) < n-d,\ m_{\mathcal{C}}(\pi) \equiv n-d \pmod 3\}$ includes a hyperplane of Σ^* even if F does not, by Lemma 2.2 (see Example 1 (3)). Note that $|F_0| = \Phi_0$, $|F_1| = \Phi_1$.

Lemma 2.4([7]). *Let L be a line of Σ^* with $F \cap L = \{\pi_1,\dots,\pi_u\}$, $m_{\mathcal{C}}(\pi_i) = s_i$, $1 \le i \le u$. Then $\sum_{i=1}^{u} s_i \equiv n + (u-1)(n-d) \pmod 3$.*

We denote by $\mathcal{F}_j^*$ the set of j-flats of Σ^*, so $\mathcal{F}_j^* = \mathcal{F}_{k-2-j}$, $0 \le j \le k-2$. A subset T of Σ^* is called an *(i,j)-set* if $|T \cap F_0| = i$, $|T \cap F_1| = j$. A line l which forms an (i,j)-set is called an *(i,j)-line*. An *(i,j)-plane* is defined similarly.

Assume $2 \le t \le k-1$ and let $\delta_t \in \mathcal{F}_t^*$. Denote by $c_{i,j}^{(t)}$ the numbers of $(t-1)$-flats which form (i,j)-sets in δ_t and let ${\varphi_s}^{(t)} = |\delta_t \cap F_s|$, $s = 0,1$. $({\varphi_0}^{(t)},{\varphi_1}^{(t)})$ is called the *diversity of δ_t* and the list of $c_{i,j}^{(t)}$'s is called its *spectrum*. An easy counting argument yields the following:

$$\sum_{(i,j)\in\Lambda_{t-1}} c_{i,j}^{(t)} = \theta_t,\quad \sum_{(i,j)\in\Lambda_{t-1}} i c_{i,j}^{(t)} = \theta_{t-1}{\varphi_0}^{(t)},\quad \sum_{(i,j)\in\Lambda_{t-1}} j c_{i,j}^{(t)} = \theta_{t-1}{\varphi_1}^{(t)},$$

$$\sum_{(i,j)\in\Lambda_{t-1}} \binom{i}{2} c_{i,j}^{(t)} = \theta_{t-2}\binom{{\varphi_0}^{(t)}}{2},\quad \sum_{(i,j)\in\Lambda_{t-1}} \binom{j}{2} c_{i,j}^{(t)} = \theta_{t-2}\binom{{\varphi_1}^{(t)}}{2},$$

$$\sum_{(i,j)\in\Lambda_{t-1}} \binom{i+j}{2} c_{i,j}^{(t)} = \theta_{t-2}\binom{{\varphi_0}^{(t)}+{\varphi_1}^{(t)}}{2},$$

where Λ_{t-1} is the set of all possibilities of $({\varphi_0}^{(t-1)},{\varphi_1}^{(t-1)})$. Λ_1 is obtained from Lemma 2.4 as

$$\Lambda_1 = \{(1,0),(0,2),(2,1),(1,3),(4,0)\}.$$

We refer the above simultaneous six equations as $(*)$.

Assume $t=2$. When $c_{0,2}^{(2)} = 0$, the above equations have the four solutions (a-1), (a-5), (a-6), (a-7) in Table 1. When $c_{4,0}^{(2)} = 0$, we get the three solutions (a-2), (a-3) and (a-4) in Table 1.

Table 1

	${\varphi_0}^{(2)}$	${\varphi_1}^{(2)}$	$c_{1,0}^{(2)}$	$c_{0,2}^{(2)}$	$c_{2,1}^{(2)}$	$c_{1,3}^{(2)}$	$c_{4,0}^{(2)}$
(a-1)	4	0	12	0	0	0	1
(a-2)	1	6	2	9	0	2	0
(a-3)	4	3	4	3	6	0	0
(a-4)	4	6	0	3	6	4	0
(a-5)	7	3	1	0	9	1	2
(a-6)	4	9	0	0	0	12	1
(a-7)	13	0	0	0	0	0	13

Theorem 2.5. *Let $\mathcal{C}$ be an $[n,3,d]_3$ code with diversity (Φ_0,Φ_1), $gcd(3,d)=1$. Then*
(1) $(\Phi_0,\Phi_1) \in \{(4,0),(1,6),(4,3),(4,6),(7,3)\}$.
(2) $\mathcal{C}$ is extendable if $(\Phi_0,\Phi_1) \in \{(4,0),(1,6),(4,6),(7,3)\}$.
(3) $\mathcal{C}$ is extendable iff $\sum_{d<i\equiv d \pmod 3} A_i > 0$ when $(\Phi_0,\Phi_1) = (4,3)$.

Proof. When $k=3$, $(\Phi_0,\Phi_1) = ({\varphi_0}^{(2)},{\varphi_1}^{(2)})$ and $\Phi_0+\Phi_1 < \theta_2$, for $\{\pi \in \mathcal{F}_2 \mid m_{\mathcal{C}}(\pi) \equiv n-d \pmod 3\} \ne \emptyset$. Hence we get (1) from Table 1. Since $c_{1,3}^{(2)} + c_{4,0}^{(2)} > 0$ for all cases except when $({\varphi_0}^{(2)},{\varphi_1}^{(2)}) = (4,3)$, we get (2) from Table 1 by Lemma 2.3.
(3) Assume that $(\Phi_0,\Phi_1) = (4,3)$. Since $c_{2,1}^{(2)} = 6$ and since any two (2,1)-lines meet in a point of F, every point of $\delta_2 \setminus F$ is on a (2,1)-line. Hence it follows from Lemma 2.2 that $\mathcal{C}$ is extendable iff $\{\pi \in \mathcal{F}_{k-2} \mid m_{\mathcal{C}}(\pi) < n-d,\ m_{\mathcal{C}}(\pi) \equiv n-d \pmod 3\} \ne \emptyset$, i.e. $\sum_{d<i\equiv d \pmod 3} A_i > 0$. □

Solving $(*)$ recursively, Theorems 1.1 and 1.2 can be proved.

References

[1] M. van Eupen and P. Lisonek, Classification of some optimal ternary linear codes of small length, *Des. Codes Cryptogr.* **10** (1997), 63–84.

[2] T.A. Gulliver, personal communications (2001).

[3] R. Hill, An extension theorem for linear codes, *Des. Codes Cryptogr.* **17** (1999), 151–157.

[4] R. Hill and P. Lizak, Extensions of linear codes, Proc. IEEE Int. Syposium on Inform. Theory (Whistler, Canada, 1995), pp. 345.

[5] J.W.P. Hirschfeld, Projective geometries over finite fields 2nd ed., Clarendon Press, Oxford, 1998.

[6] T. Maruta, On the extendability of linear codes, *Finite Fields and Their Appl.* **7** (2001), 350–354.

[7] T. Maruta, Extendability of linear codes over GF(q) with minimum distance d, gcd(d,q) = 1, submitted.

[8] T. Maruta, A new extension theorem for linear codes, submitted.

[9] J. Simonis, Adding a parity check bit, *IEEE Trans. Inform. Theory* 46 (2000) 1544–1545.

The codes of type Preparata over $GF(4)$. *

Nechaev A.A., Neljubin A.S.
Center of New Informational Technologies
of Lomonosov Moscow State University
119899, Vorobjevy Gory, Moscow, Russia
e-mail: nechaev@cnit.msu.ru

Abstract

For $q = 4$ we construct a code P over the field $F = GF(q)$ whose parameters generalize the parameters of classical binary Preparata code. We explicate the idea of [6]: the code P is a Reed-Solomon representation of a linear over the Galois ring $R = GR(q^2, 4)$ code $\mathcal{P}$ dual to a linear code $\mathcal{K}$ with parameters near (but not equal) to the parameters of generalized linear Kerdock code over R [4, 6, 7].

1 Introduction

Let $R = GR(q^2, 4)$ be a Galois ring with identity e of characteristic 4 and cardinality q^2, $q = 2^l$, $l \geq 1$. The set $\Gamma(R) = \{r \in R\colon\ r^q = r\}$ is called the *p-adic (or Teichmueller) coordinate set* of the ring R. Any element $r \in R$ is a unique sum $r = r_0 + 2r_1$, where $r_t = \gamma_t(r) \in \Gamma(R)$, $t = 0, 1$. If we define $\oplus$ on $\Gamma(R)$ by the rule $u \oplus v = \gamma_0(u + v)$ then $(\Gamma(R), \oplus, \cdot)$ is a field $GF(q)$. In the following we fix the short notation: $F = \Gamma(R)$.

Let $F = \{\omega_0 = 0, \omega_1 = e, \ldots, \omega_{q-1}\}$ and $\gamma_*\colon R \to F^q$ be the map acting on an element $r = r_0 + 2r_1 \in R$ by the rule

$$\gamma_*(r) = (r_1, r_1 \oplus \omega_1 r_0, \ldots, r_1 \oplus \omega_{q-1} r_0). \tag{1}$$

Then $\gamma_*(R)$ is $[q, 2, q-1]_q$ Reed–Solomon code over $F = GF(q)$ and therefore the map $\gamma_*(R)$ is called *RS-map* [7]. Note that if $q = 2$, i.e. if $R = \mathbb{Z}_4$, then γ_* is the so called Gray map from [5].

With any h-code $\mathcal{K} \subseteq R^h$ over the ring R we can associate its *RS-representation* $K = \gamma_*^h(\mathcal{K}) \subseteq F^{qh}$. It is a code of the length qh over F, consisting of all words

$$\gamma_*^h(\vec{u}) = (\gamma_*(u(0)), \ldots, \gamma_*(u(h-1))), \quad \vec{u} \in \mathcal{K}. \tag{2}$$

So K is a concatenation of the code $\mathcal{K}$ over R and a linear over F code $\gamma_*(R)$. Note that if $\mathcal{K}$ is a subgroup of the group $(R^h, +)$ then K is distance invariant. In the last case the Hamming distance $d(K)$ of the code K is equal to the minimum of Hamming weights of nonzero words in K.

If $\mathcal{K}$ is a linear code over R, i.e. $\mathcal{K} \leq {}_R R^h$ (so it is a submodule of the R-module ${}_R R^h$), we call K an *(R, γ_*)-linear code* (and sometimes briefly an *R-linear code*). An R-linear code K is distance invariant but may be nonlinear.

Let now $\mathcal{K}^o$ be a code dual to a linear code $\mathcal{K} \leq {}_R R^h$ relative to a standard scalar product. Then again $\mathcal{K}^o \leq {}_R R^h$ and we shall call the R-linear code $K_\perp = \gamma_*^h(\mathcal{K}^o) \subseteq F^{qh}$ *R-dual* to the (R-linear) code K.

In [2, 3] the $\mathbb{Z}_4$-linearity of classical binary Kerdock $(2^{m+1}, 2^{2(m+1)}, 2^m - 2^\lambda)$-code, where m is odd and $\lambda = [m/2]$ [1], was discovered. Further in the paper [5] it was discovered that the classical binary Preparata code with parameters $(2^{m+1}, 2^{2(2^m-1-m)}, 6)$ is $\mathbb{Z}_4$-dual to the binary Kerdock code. Simultaneously in [4] a generalized Kerdock code $K_q(m)$ over any Galois field $F = GF(q), q = 2^l, l > 1$ with parameters $(n, n^2, ((q-1)/q)(n - \sqrt{n})), n = q^{m+!}$ was constructed. This code has a form $K_q(m) = \gamma_*^h(\mathcal{K}_R(m))$, where $\mathcal{K}_R(m) \leq {}_R R^h$ is a special linear code of the length $h = q^m$, called *base linear code* (see below).

However the attempts to build a generalized Preparata code by analogy with [5] as a code R-dual to $K_q(m)$ were unsuccessful: for $q > 2$ the code $K_q(m)_\perp = \gamma_*^h(\mathcal{K}_R(m)^o)$ has the distance $3(q-1)$ (see [6] and [8] for $R = \mathbb{Z}_{q^2}, q$ — prime, odd). So the formula of the distance of such "generalization" of the Preparata code is not a generalization of that of the distance of the original binary Preparata code: for $q = 2$ we have 3 instead of $6 = 3q$. Nevertheless this very construction was called in [8] the generalization of the Preparata code. We propose some alternative approach to the definition of this notion based on the idea of [6].

Let us consider a parameter $\Delta = \frac{q^m-1}{q-1}$. Note that for $q = 2$ the parameters of the binary Preparata code can be expressed as $(2^{m+1}, 2^{2(2^m-1-m)}, 6) = (q(\Delta+1), q^{2(\Delta-m)}, 3q)$. We show that for $q = 4, (m, q-1) = 1$ there exists a $(q(\Delta+1), q^{2(\Delta-m)}, 3q)$-code over the field F and it is R-dual to a code with parameters near to the parameters of generalized Kerdock code.

2 General constructions.

Here we suppose that $q = 2^l, l \geq 1$. Let $S = GR(q^{2m}, 4)$ be a Galois extension of the degree m of the Galois ring $R = GR(q^2, 4)$ with Teichmueller coordinate set $\Gamma(S) = \{\beta \in S\colon\ \beta^{q^m} = \beta\}$. Any element $\beta \in S$ is a unique sum $\beta = \beta_0 + 2\beta_1$, where $\beta_t = \gamma_t(\beta) \in \Gamma(S)$, $t = 0, 1$. If we define $\oplus$ on $\Gamma(S)$ by the rule $u \oplus v = \gamma_0(u + v)$ then $(\Gamma(S), \oplus, \cdot)$ is a field $GF(q^m)$ and the field $F = \Gamma(R) = \{\beta \in S\colon\ \beta^q = \beta\}$ is a subfield of $Q = \Gamma(S)$.

Let us take an element $\xi \in Q^*$ of order $ord\xi = d$ such that $Q = F(\xi)$ and consider the code $\mathcal{K}_R[\xi]$ of the length $h = d + 1$ over R consisting of all words

$$\vec{v} = (v(0) \ldots v(h-1))$$

such that for some $\alpha \in S$, $c \in R$

$$v(i) = Tr_R^S(\alpha\xi^i) + c, \quad i = \overline{0, h-2}, \quad v(h-1) = c, \tag{1}$$

where $Tr_R^S(x)$ is the *trace-function* from S onto R ($Tr_R^S(x) = \sum_\sigma \sigma(x)$, σ spans the group of automorphisms of S over R).

If $d = q^m - 1$ then $\mathcal{K}_R[\xi] = \mathcal{K}_R(m)$ is the *base linear code* mentioned above.

We consider the case $d = \Delta$. In this situation we shall call $\mathcal{K}_R[\xi]$ the *reduced base code* and denote it by $\mathcal{K}_R^{red}(m)$. Correspondingly we shall call a code $K_q^{red}(m) = \gamma_*^h(\mathcal{K}_R^{red}(m))$ the *reduced (generalized) Kerdock code.* Our main object is the *generalized Pereparata code*

*The work was partially supported by RFFR grants 99-01-00941, 99-01-00382.

$P_q(m)$ which is defined as R-dual to $K_q^{red}(m)$ code: $P_q(m) = \gamma_*^h(\mathcal{P}), \mathcal{P} = \mathcal{K}_R^{red}(m)^o$. It is not difficult to see that $\mathcal{P}$ is a linear code of the length $h = \Delta + 1$ over the ring R with check matrix

$$H = \begin{pmatrix} e & e & e & \dots & e \\ 0 & e & \xi & \dots & \xi^{\Delta-1} \end{pmatrix}$$

3 Main results.

Proposition 1 *The length n and the cardinality C of the reduced Kerdock code $K_q^{red}(m)$ are*

$$n = q(\Delta+1) = \frac{q}{q-1}(q^m + q - 2), \quad C = q^{2(m+1)} = ((q-1)n - q^2 + 2q)^2.$$

If $q = 4, (m, q-1) = 1$, then the distance d of this code satisfies the inequalities

$$4^m - 4^{[\frac{m}{2}]} \geq d \geq 4^m - \frac{17}{3} \cdot 4^{\frac{m}{2}} + 2$$

In comparison with the parameters $(n, n^2, \frac{q-1}{q}(n - \sqrt{n})$ of the generalized Kerdock code over F the cardinality C of our code is larger: $C \simeq (q-1)^2 n^2$, but the distance is smaller. The last inequalities allow to state that for $q = 4$ there is an equality

$$d = \frac{q-1}{q}(n - c(m)\sqrt{n}), \quad \text{where } 6.54 \geq c(m) \geq 0.577 \cdot 2^{m-2\lambda}, \quad \lambda = [m/2].$$

. Apparently the last estimations are rather rough. The following results of calculations for $q = 4$ allow to conjecture that for $m > 4$ really $3 \geq c(m) \geq 2$.

m	n	$4^m - 4^\lambda$	d	$4^m - \frac{17}{3}2^m + 2$	$c(m) = \frac{n - \frac{q-1}{q}d}{\sqrt{n}}$
2	24	12	12	-4	1.77
4	344	240	238	167	1.44
5	1368	1008	962	845	2.31
7	21848	16320	16146	15661	2.17
8	87384	65280	65048	64087	2.21

Our main results about the Preparata codes are the following.

Theorem 2 *If $(m, q-1) = 1$ and $q = 2^l$ then the generalized Prepararta code $P_q(m)$ is a $(q(\Delta+1), q^{2(\Delta-m)}, d \geq 3q)$-code over the field $F = GF(q)$. If m is even then $3q \leq d(P_q(m)) \leq 4(q-1)$.*

More precisely the value of $d(P_q(m))$ was calculated in some particular cases.

Proposition 3 *Under the conditions of Theorem 2 if $m_1|m$ then $d(P_q(m)) \leq d(P_q(m_1))$. If $q = 4$ and m is a multiple of 2, 5 or 7 then $d(P_q(m)) = 3q$.*

Note that if $q = 4, m = 2$ we have still that $P_q(m)$ and $K_q^{red}(m)$ are both $(24, 4^6, 12)$-codes. These results allow us to formulate the following

Conjecture. *The equality $d(P_q(m)) = 3q$ holds for any $q = 2^l$.*

If this conjecture is true then we can say that $\mathbb{Z}_4$-duality of binary Kerdock and Preparata codes is in some sense a casual result. In fact the code R-dual to the generalized Preperata code over $GF(2^l)$ is the reduced Kerdock code $K_q^{red}(m)$ which is equal to the generalized Kerdock code $K_q(m)$ only if $q = 2$.

References

[1] Kerdock A. M. A class of low-rate non-linear codes. *Inform. Control*, **20** (1972), 182–187.

[2] Nechaev A. A. Trace function in Galois ring and noise stable codes (in Russian), *V All-Union Symp. on theory of rings, algebras and modules*, Novosibirsk, p. 97, 1982.

[3] Nechaev A. A. Kerdock code in a cyclic form (in Russian). *Diskr. Math. (USSR)*, 1 (1989), No 4, 123–139. English translation: *Discrete Math. and Appl.*, 1 (1991), No 4, 365–384 (VSP).

[4] Kuzmin A. S., Nechaev A. A. Linearly presented codes and Kerdock code over an arbitrary Galois field of the characteristic 2. *Russian Math. Surveys*, **49** (1994), № 5.

[5] Hammons A. R., Kumar P. V., Calderbank A. R., Sloane N. J. A., Solé P. The $\mathbb{Z}_4$-linearity of Kerdock, Preparata, Goethals and related codes, *IEEE Trans. Inf. Theory*, vol. 40, No 2, pp. 301–319, 1994.

[6] Nechaev A. A., Kuzmin A. S. Linearly presentable codes, *Proceedings of the 1996 IEEE Int. Symp. Inf. Theory and Appl.*, Victoria B. C., Canada, 1996, pp. 31–34.

[7] Nechaev A. A., Kuzmin A. S. Trace-function on a Galois ring in coding theory. *Lecture Notes in Computer Science*, **1255**. Springer, 1997, 277–290.

[8] Bram van Asch, Henk C. A. van Tilborg. Some observations about linear $\mathbb{Z}_{p^2}$ codes. *Third Euro Workshop on Optimal Codes and Related Topics*. 10-17 June, 2001, Sunny Beach, Bulgaria, 5–12.

Low Complexity Tail-biting Trellises for Some Extremal Self-Dual Codes

G.Olocco, A.Otmani *

LACO, Université de Limoges, 87060 Limoges

LRI, Université Paris-Sud, 91405 Orsay

e-mail: `olocco@lri.fr`, `otmani@unilim.fr`

Abstract

We obtain low complexity tail-biting trellises for some extremal self-dual codes for various lengths and fields such as the [12,6,6] ternary Golay code and a [24,12,8] Hermitian self-dual code over GF(4). These codes are obtained from a particular family of cyclic Tanner graphs called necklace factor graphs.

Keywords : self-dual codes, tail-biting trellises, necklace factor graph.

1 Introduction

The representation of linear block codes by trellises is a very powerful description which allows an efficient soft decision decoding. We consider a family of codes introduced in [2] based on the use of short length codes and interleavers. From this family it is possible to extract a sub-family of codes adapted to iterative decoding. Indeed every code of this sub-family is associated with a necklace factor from which a tail-biting trellis can be deduced. Among this family, it is interesting to find codes with the best minimal distance as in [3]. Herein, we obtained some extremal self-dual codes over $GF(2)$ and $\mathbb{Z}_4$.

In this paper, we extend the construction [2] to several fields and we formalize the constraints on the necklace graph given in [3] to get codes with the best minimum distances. By this way, we have low complexity tail-biting trellises for several codes like the [12,6,6] ternary Golay code and a [24,12,8] Hermitian self-dual code over GF(4).

2 Necklace Factor Graph

For an introduction to factor graphs we refer the reader to [4]. We recall that a factor graph of a code C over $GF(q)$ consists of *check nodes* representing local constraints of C, and *variable nodes* which take values in an alphabet. We distinguish between two types of variable nodes: *symbol nodes* which are associated with the symbol of the codewords of C and *state nodes* which are used for computing the codewords of C but which are not transmitted. A variable node is *adjacent* to a check node if the corresponding variable is involved in the corresponding local constraint.

*This work was supported in part by France Télécom R&D.

We consider $[N, K, D]$ linear codes C over $GF(q)$ obtained from a $[n, k, d]$ linear base code B over $GF(q)$ (see [3]). We focus on codes having a *cyclic* factor graph $N_t(C)$ with t *necklaces* like the one given in Figure 1.

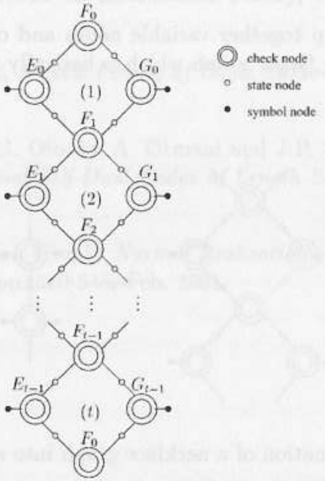


Figure 1: A necklace factor graph of order t.

Each check node represents the base code B. Symbol nodes take values in $GF(q)^k$. State nodes take values in $GF(q)^{\frac{k}{2}}$ if k is even and in $GF(q)^{\lfloor\frac{k}{2}\rfloor}$ or $GF(q)^{\lfloor\frac{k}{2}\rfloor+1}$ if not. All the variable nodes adjacent to a check node form a codeword of B and all the symbol nodes of the $N_t(C)$ form a codeword of C.

Proposition 1 *If all the local constraints of a necklace factor graph representing a code C are self-dual codes over $GF(q)$ then C is also a linear self-dual code.*

Proof It is an application of Theorem 7.3 given in [4] since any necklace graph is a normal factor graph.

□

3 Constraints on the Necklace Graph

Among all the codes that have necklace graphs described in the Section 2, we are particularly interested in those that have the best minimum distances. We meet this requirement firstly by defining the properties that should satisfy a necklace graph, and then by searching exhaustively interleavers that check them.

Property of diffusion: for any check node which has degree four, if one of its adjacent state node has non-zero (Hamming) weight, then at least three of them has non-zero weights.

Property of expansion: for any check node which has degree three, if exactly one state node has non-zero weight, then the symbol node has always a weight greater than a certain constant $b \geq d - k/2$.

4 Low Complexity Tail-Biting Trellises

lemma 1 *Any necklace factor graph $N_t(C)$ of a code C can be put into the form of a t-section tail-biting trellis $T_t(C)$.*

Proof It is sufficient to group together variable nodes and check nodes of the same level (see Figure 2) to obtain a new factor graph which is basically a tail-biting trellis.

□

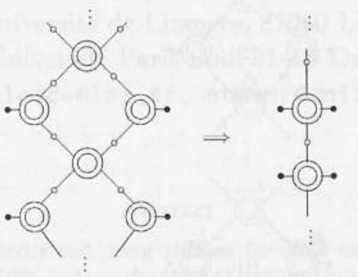


Figure 2: Transformation of a necklace graph into a tail-biting trellis.

There exist several types of complexity for a tail-biting trellis ([1]). We are only interested in the maximum state complexity.

Definition 1 *Let T be a t-section tail-biting trellis with state spaces $S_0, \ldots, S_{t-1}$. The maximum state complexity S_{max} of T is defined as*

$$S_{max} = max\{|S_0|, \ldots, |S_{t-1}|\}.$$

lemma 2 *Let C be a $[N, K, D]$ linear code over $GF(q)$ with necklace graph $N_t(C)$ obtained from a $[n, k, d]$ linear code B. Each states space of the tail-biting trellis $T_t(C)$ deduced from $N_t(C)$ is of size q^k and therefore S_{max} is equal to q^k states.*

The following table gathers the parameters of the obtained codes along with the complexities of their associated tail-biting trellises.

q	B	C	t	$S_{max}(T_t(C))$
3	[4,2,3]	[8,4,3]	2	3^2
3	[4,2,3]	[12,6,6]	3	3^2
3	[4,2,3]	[16,8,6]	4	3^2
3	[4,2,3]	[20,10,6]	4	3^2
3	[12,6,6]	[24,12,9]	2	3^6
4 Euclidean	[4,2,3]	[8,4,3]	2	4^2
4 Euclidean	[4,2,3]	[12,6,6]	3	4^2
4 Euclidean	[4,2,3]	[16,8,6]	4	4^2
4 Hermitian	[6,3,4]	[12,6,4]	2	4^3
4 Hermitian	[8,4,4]	[24,12,8]	3	4^4
5	[6,3,4]	[18,9,6]	3	5^3
5	[8,4,4]	[24,12,8]	3	5^4

References

[1] A.R. Carlderbank, G.D. Forney, A. Vardy, *Minimal Tail-Biting Trellis: The Golay Code and More*, IEEE Transactions on Information Theory, Vol. 45, No. 5, July 1999, pp. 1435-1455.

[2] J.C. Carlach, C. Vervoux, *A New Family of Block Turbo-Codes*, AAECC-13, 1999, pp. 15-16.

[3] E. Cadic, J.C. Carlach, G. Olocco, A. Otmani and J.P. Tillich, *Low Complexity Tail-Biting Trellises of Extremal Self-Dual Codes of Length 24, 32 and 40 over GF(2) and* $\mathbb{Z}_4$, AAECC14, 2001.

[4] G.D. Forney, Jr., *Codes on Graph: Normal Realizations*, IEEE Transactions on Information Theory, Vol. 47, pp. 520-548, Feb. 2001.

Decoding Arbitrary Codes in Rank Metric

Alexei Ourivski
Moscow Institute of Physics and Technology
Institutskii per. 9, 141700 Dolgoprudny, Russia
ourivski@mail.ru

Thomas Johansson
Lund University
Box 118, 221 00 Lund, Sweden
thomas@it.lth.se

Abstract

We present a new technique of decoding codes in rank metric. The decoding problem is modeled as solving a system of quadratic equations in the field $GF(q)$. Several strategies for solving the system by guessing certain unknowns are considered. The result is two new algorithms for decoding a general (n, k) linear rank distance code over $GF(q^N)$ correcting errors of rank r in $O\left((Nr)^3 q^{(r-1)(k+1)}\right)$ and $O\left((k+r)^3 r^3 q^{(r-1)(N-r)}\right)$ operations in $GF(q)$, respectively.

Introduction

Codes in rank metric, introduced by Gabidulin in [1], well suited for correcting burst errors and lattice-pattern errors in parallel channels. Another application of codes in rank metric is public key cryptography, namely they are used in the GPT PKC [2] and its modifications, and in Chen's authentication scheme [3].

In this paper, we first recall the definition of rank codes and formulate the decoding problem for them. Then we show how to model this problem as solving a system of quadratic equations in the base field $GF(q)$. After that we consider several methods for solving the system and estimate their complexities.

1 Rank distance codes

Let $GF(q)$ be the finite field with q elements, and let $GF(q^N)$ be the extension field with q^N elements, q is a power of a prime. Let $\mathbf{x} = (x_1, \dots, x_n)$ be a vector over $GF(q^N)$. The *rank* of $\mathbf{x}$ over $GF(q)$, denoted $r(\mathbf{x}|q)$, is defined to be the maximal number of x_i's that are linearly independent over $GF(q)$. The *rank distance* $d_r(\mathbf{x}, \mathbf{y})$ between two vectors $\mathbf{x}$ and $\mathbf{y}$ in $GF(q^N)^n$ is $d_r(\mathbf{x}, \mathbf{y}) = r(\mathbf{x} - \mathbf{y}|q)$.

Let $\mathcal{C}$ be an (n, k) linear code over $GF(q^N)$. In the sequel, we only consider linear codes. For the code $\mathcal{C}$ the (minimal) *rank distance* $d(\mathcal{C})$ is defined to be $d(\mathcal{C}) = \min\{r(\mathbf{c}|q) \mid \mathbf{c} \in \mathcal{C}, \mathbf{c} \neq \mathbf{0}\}$. If $\mathcal{C}$ has minimal rank distance d, then it can correct all errors $\mathbf{e}$ with $r(\mathbf{e}|q) \leq t = \lfloor (d-1)/2 \rfloor$. We call t the rank error-correcting capability of the code.

The problem of decoding an (n, k) code $\mathcal{C}$ over $GF(q^N)$ with rank distance d may be formulated as follows. Given a length n vector $\mathbf{c}$, find a k-vector $\mathbf{m}$ such that the difference $\mathbf{e} = \mathbf{c} - \mathbf{mG}$ has the smallest possible rank, where $\mathbf{G}$ is a $k \times n$ generator matrix of $\mathcal{C}$. If $r(\mathbf{e}|q) \leq t$, then this is bounded decoding, and exactly one solution exists.

Let $r(\mathbf{e}|q) = r \leq t$. The problem of bounded decoding for $\mathcal{C}$ can be formulated as the problem of finding a codeword of the smallest rank in the code $\mathcal{C}_e$ with generator matrix

$$\mathbf{G}_e = \begin{bmatrix} \mathbf{G} \\ \mathbf{c} \end{bmatrix} = \begin{bmatrix} \mathbf{E}_k & 0 \\ \mathbf{m} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{G} \\ \mathbf{e} \end{bmatrix}, \tag{1}$$

where $\mathbf{E}_k - k \times k$ identity matrix. The code $\mathcal{C}_e$ has rank distance r, and all codewords of rank r are multiples of $\mathbf{e}$, hence they are of the form $\varepsilon\mathbf{e}$, where $\varepsilon \in GF(q^N)\backslash\{0\}$. The value of ε may easily be computed from the two vectors $\varepsilon\mathbf{e}\mathbf{H}^T$ and $\mathbf{c}\mathbf{H}^T = \mathbf{e}\mathbf{H}^T$, where $\mathbf{H}$ is a parity-check matrix of $\mathcal{C}$.

2 Modeling the decoding problem as a set of quadratic equations

Our approach to solving the decoding problem is to find a system of equations on the components of the error vector $\mathbf{e}$, and then to solve that system.

Reduce the generator matrix given in (1) to a systematic form $\mathbf{G}_{syst} = [\mathbf{E}_{k+1}\ \mathbf{R}]$.

Partition $\mathbf{e}$ as $\mathbf{e} = (\mathbf{e}_1\ \mathbf{e}_2)$, where $\mathbf{e}_1$ is the first $k+1$ components of $\mathbf{e}$, and $\mathbf{e}_2$ is the last $n-k-1$ ones. Evidently, $r(\mathbf{e}_1|q) \leq r(\mathbf{e}|q) = r$. Assume that $\mathbf{e}_1 \neq \mathbf{0}$. It can be shown that

$$\mathbf{e}_2 = \mathbf{e}_1\mathbf{R}. \tag{2}$$

Thus, we need to find a vector $\mathbf{e}_1$ with $r(\mathbf{e}_1|q) \leq r$ such that the vector $(\mathbf{e}_1, \mathbf{e}_1\mathbf{R})$ has rank exactly r. The vector $\mathbf{e}$ of rank r can be represented as

$$\mathbf{e} = (x_0, x_1, \dots, x_{r-1}) \begin{bmatrix} \alpha_{0,1} & \cdots & \alpha_{0,k+1} & \alpha_{0,k+2} & \cdots & \alpha_{0,n} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \alpha_{r-1,1} & \cdots & \alpha_{r-1,k+1} & \alpha_{r-1,k+2} & \cdots & \alpha_{r-1,n} \end{bmatrix} = \mathbf{xA},$$

where $x_0, x_1, \dots, x_{r-1} \in GF(q^N)\backslash\{0\}$ are linearly independent over $GF(q)$, and $\mathbf{A}$ is some $r \times n$ matrix over $GF(q)$ of full rank r. Since we are looking for any multiple of $\mathbf{e}$, we can put $x_0 = 1$. The parts $\mathbf{e}_1$ and $\mathbf{e}_2$ are now written as $\mathbf{e}_1 = \mathbf{xA}_1$, $\mathbf{e}_2 = \mathbf{xA}_2$, where $\mathbf{A}_1$ and $\mathbf{A}_2$ are the first $k+1$ and the last $n-k-1$ columns of $\mathbf{A}$, respectively.

For every j, $j = k+2, \dots, n$, from (2) we obtain the equality

$$\mathbf{xA}_1(j)\widetilde{\mathbf{r}}_j = 0, \tag{3}$$

where $\mathbf{A}_1(j) = \begin{bmatrix} \mathbf{A}_1 & \begin{matrix} \alpha_{0,j} \\ \alpha_{1,j} \\ \vdots \\ \alpha_{r-1,j} \end{matrix} \end{bmatrix}$, $\widetilde{\mathbf{r}}_j = \begin{bmatrix} \mathbf{r}_j \\ -1 \end{bmatrix}$, and $\mathbf{r}_j$ is the $(j-k-1)$-th column of $\mathbf{R}$. Taking different indices j in (3) we get a system of $n-k-1$ equations in the unknowns $x_1, \dots, x_{r-1}$, $\mathbf{A}_1$, $\mathbf{A}_2$. We are going to rewrite the system over the base field $GF(q)$.

Let $\mathbf{\Omega}$ be a basis of $GF(q^N)$ over $GF(q)$. For every $b \in GF(q^N)$ put into correspondence elements $\beta_1, \beta_2, \ldots, \beta_N \in GF(q)$, called coordinates of b in the basis $\mathbf{\Omega}$, and write this correspondence as $b \leftrightarrow (\beta_1, \ldots, \beta_N)$. For two elements $b \leftrightarrow (\beta_1, \ldots, \beta_N)$ and $c \leftrightarrow (\varepsilon_1, \ldots, \varepsilon_N)$ the coordinates of the product $cb \leftrightarrow (\delta_1, \ldots, \delta_N)$ are

$$\delta_\ell = (\beta_1, \ldots, \beta_N)\mathbf{\Delta}_\ell(\varepsilon_1, \ldots, \varepsilon_N)^T, \quad \ell = 1, \ldots, N, \tag{4}$$

where $N \times N$ matrices $\mathbf{\Delta}_\ell$ with elements in $GF(q)$ are all non-singular, pairwise different, and depend only on the choice of $\mathbf{\Omega}$, hence known.

For the column $\mathbf{r}_j$ we can associate a $(k+1) \times N$ matrix $\mathbf{\Upsilon}_j^T$ m-th row of which is the representation of m-th component of $\mathbf{r}_j$. Let $-1 \leftrightarrow \boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_N)$.

Let c_{ij} be the i-th component of the vector $\mathbf{A}_1(j)\widetilde{\mathbf{r}}_j$. Denote $x_i \leftrightarrow (\gamma_{1i}, \gamma_{2i}, \ldots, \gamma_{Ni})$. Using (4), we find the representation of $x_i c_{ij} \leftrightarrow (\sigma_1(ij), \sigma_2(ij), \ldots, \sigma_N(ij))$

$$\sigma_\ell(ij) = (\gamma_{1i}, \gamma_{2i}, \ldots, \gamma_{Ni})\mathbf{\Delta}_\ell\left[\mathbf{\Upsilon}_j\, \boldsymbol{\lambda}^T\right](\alpha_{i1}, \alpha_{i2}, \ldots, \alpha_{i,k+1}, \alpha_{ij})^T, \quad i = 1, \ldots, r-1. \tag{5}$$

Since $x_0 = 1$, the formula for $x_0 c_{0j} = c_{0j}$ is simpler, $\sigma_\ell(0, j) = \boldsymbol{\mu}_\ell(j)(\alpha_{01}, \ldots, \alpha_{0,k+1}, \alpha_{0j})^T$, where $\boldsymbol{\mu}_\ell(j)$ is the ℓ-th row of $\left[\mathbf{\Upsilon}_j\, \boldsymbol{\lambda}^T\right]$.

In conclusion, equation (3) for each j, $j = k+2, \ldots, n$, gives the following system of N quadratic equations in the base field $GF(q)$:

$$\sum_{i=1}^{r-1}(\gamma_{1i}, \gamma_{2i}, \ldots, \gamma_{Ni})\mathbf{\Delta}_\ell\left[\mathbf{\Upsilon}_j\, \boldsymbol{\lambda}^T\right]\begin{pmatrix}\alpha_{i1}\\ \vdots\\ \alpha_{i,k+1}\\ \alpha_{ij}\end{pmatrix} + \boldsymbol{\mu}_\ell(j)\begin{pmatrix}\alpha_{01}\\ \vdots\\ \alpha_{0,k+1}\\ \alpha_{0j}\end{pmatrix} = 0, \qquad \ell = 1, \ldots, N. \tag{6}$$

For any solution to system (6) compute $\mathbf{e}_2 = \mathbf{e}_1\mathbf{R}$, and check the rank of $\mathbf{e} = (\mathbf{e}_1\ \mathbf{e}_2)$. If $r(\mathbf{e}|q) = r$, then decoding is successful. Otherwise search for another solution.

If the value $r(\mathbf{e}|q) = r$ is not known beforehand, start decoding assuming that $r = 1$. If no solution was found, assume $r = 2$, and so forth. Complexity of solving system (6) grows exponentially in r, so only the final step determines the overall complexity of decoding.

3 Solving the system of equations

In this section, we suggest different strategies for solving system (6). It is not necessary to use all the equations in the system, since it suffices to find $\mathbf{e}_1$ to recover $\mathbf{e}$. Assume that we use a subset of m elements from the set $k+2, \ldots, n$ for the index j, say $\{j_1, \ldots, j_m\}$.

Strategy 1. Guess values of $\alpha_{i1}, \alpha_{i2}, \ldots, \alpha_{i,k+1}$, and α_{ij} for every $i = 1, \ldots, r-1$, and $j \in \{j_1, j_2, \ldots, j_m\}$. Now we have a system of Nm linear equations in $v_1 = (r-1)N+k+m+1$ unknowns. The system is solvable if there are more equations than unknowns, so we require $m \geq \lceil((r-1)N+k+1)/(N-1)\rceil$.

Solving the system takes about $O(v_1^3)$ operations. In fact, there is no need to guess all $(r-1)(k+m+1)$ unknown α_{ij}. What we need is to guess a matrix $\mathbf{A}_m$ consisting of all $k+1$ columns of $\mathbf{A}_1$ and some m columns of $\mathbf{A}_2$ except the first component of each column. Since $\mathbf{e} = \mathbf{x}\mathbf{B}\mathbf{B}^{-1}\mathbf{A}$ for any non-singular matrix $\mathbf{B}$ over $GF(q)$, we can consider both $\mathbf{A}$ and $\mathbf{A}_m$ to be in row-reduced echelon forms. There exist $M_1 \leq q^{(r-1)(k+m+1-r)+2}$ candidates for $\mathbf{A}_m$ that we have to check. Thus, for any fixed $\mathbf{A}_m$ we solve a linearized system derived from (6). The overall complexity of this strategy of decoding is $W_1 = O(v_1^3 M_1)$ operations in $GF(q)$. Asymptotically, when values r, N and k go large, we have $m \sim r$, $k \sim N$, and

$$W_1 \approx O\left(\left(rN\right)^3 q^{(r-1)(k+1)}\right). \tag{7}$$

Strategy 2. In this strategy we also linearize the quadratic terms in (6) by guessing values of $\gamma_{\ell i}$, $\ell = 1, \ldots, N$, $i = 1, \ldots, r-1$, i. e. the basis for $\mathbf{e}$. So we get a system of Nm linear equations in $v_2 = (k+1+m)r$ unknowns $\alpha_{i1}, \ldots, \alpha_{i,k+1}, \alpha_{ij}$ for every $i = 0, \ldots, r-1$, $j \in \{j_1, j_2, \ldots, j_m\}$. Obviously, m must satisfy the inequality $m \geq \lceil(k+1)r/(N-r)\rceil$.

As in Strategy 1 we do not have to consider all possible bases for $\mathbf{e}$. It is sufficient to look through only non-equivalent bases leading to different $\mathbf{e}$. It is readily shown that there exist $M_2 \leq q^{(r-1)(N-r)+2}$ different bases with $x_0 = 1$.

Altogether, the total complexity of this strategy is $W_2 = O(v_2^3 M_2)$ operations in $GF(q)$. For large k, r, and N we have $m \sim r+1$, $k \sim N$, and

$$W_2 \approx O\left((k+r)^3 r^3 q^{(r-1)(N-r)}\right). \tag{8}$$

Example. Let $\mathcal{C}_1 = (12, 6)$ and $\mathcal{C}_2 = (24, 12)$ be optimal rank codes over $GF(2^{12})$ correcting errors of rank 3 or less. For code $\mathcal{C}_1$ we obtain $W_1 \approx 2^{29}$ and $W_2 \approx 2^{33}$. Code $\mathcal{C}_2$ is decodable in $W_1 \approx 2^{44}$ and $W_2 \approx 2^{36}$ binary operations.

References

[1] E. M. Gabidulin. Theory of Codes with Maximum Rank Distance. *Probl. Inform. Transm., 21(1)*, July 1985, pp. 1–12.

[2] E. M. Gabidulin, A. V. Paramonov, O. V. Tretjakov. Ideals over a Non-Commutative Ring and Their Application in Cryptology, in: D. W. Davies, ed., *Advances in Cryptology – EUROCRYPT'91, LNCS 547*, 1991, pp. 482–489.

[3] K. Chen. A New Identification Algorithm, in: E. P. Dawson, J. Golic, eds., *Proc. Int. Confr. on Cryptography: Policy and Algorithms, LNCS 1029*, 1996, pp. 244–249.

The Switching construction and kernels of q-ary 1-perfect codes *

K.T.Phelps [†] J.Rifà M.Villanueva [‡]

May 30, 2002.

Abstract

The kernel of a q-ary code C can be defined as $K_C = \{x \in \mathbb{F}_q^n \ : \ \lambda x + C = C \quad \forall \lambda \in \mathbb{F}_q\}$. We establish the kernel dimension of different q-ary 1-perfect codes of length n, using the Switching construction.

1. Introduction. Let $\mathbb{F}_q^n$ be a vector space of dimension n over a Galois Field $\mathbb{F}_q = GF(q)$. The *Hamming distance* between vectors $u, v \in \mathbb{F}_q^n$, denoted $d(u,v)$, is the number of coordinates in which u and v differ. The *support* of $x = (x_1, x_2, \dots, x_n) \in \mathbb{F}_q^n$ is $supp(x) = \{i \ : \ x_i \neq 0\}$. A *q-ary code*, C, of length n is simply a subset of $\mathbb{F}_q^n$. The elements of C are called *codewords* and C is called *linear* if it is a linear space over $\mathbb{F}_q$. The *minimum distance* of a code is the smallest distance between a pair of codewords.

A q-ary code C of length n is *perfect* if for some integer $r \geq 0$ every $x \in \mathbb{F}_q^n$ is within distance r from exactly one codeword of C. In [8] it is shown that the only parameters for nontrivial perfect codes are the two Golay codes and the q-ary 1-perfect codes where q is a prime power. The q-ary 1-perfect codes have length $n = \dfrac{q^m-1}{q-1}$, $m \geq 2$, and $r = 1$. They have q^{n-m} codewords and minimum distance 3. The linear 1-perfect codes are unique up to equivalence, they are the well-known *Hamming codes* and exist for all $m \geq 2$. Nonlinear q-ary 1-perfect codes also exist for $q = 2, m \geq 4$ and for q a prime power, $q \geq 3, m \geq 2$ (except for $q = 4$ and 8 if $m = 2$) [14], [13], [9].

Two structural properties of nonlinear codes are the rank and kernel.

The *rank* of a q-ary code C, $r(C)$, is simply the dimension of the subspace spanned by C. Etzion and Vardy [6] established the existence of binary 1-perfect codes of length $n = 2^m - 1$, $m \geq 4$, and rank $r(C) = n - m + s$ for each $s \in \{0, 1, \dots, m\}$. In [12] was established the generalization to the q-ary case, that is, the existence of q-ary 1-perfect codes of length $n = \dfrac{q^m-1}{q-1}$, $m \geq 4$ and rank $r(C) = n - m + s$ for each $s \in \{0, 1, \dots, m\}$. This problem still remain open for $m = 2$ and $m = 3$ if $q \geq 3$.

The *kernel* of a binary code C is defined as $K_C = \{x \in \mathbb{F}_2^n \ : \ x + C = C\}$. If the zero word is in C, then K_C is a linear subspace of C. In general, C can be written as the union of cosets of K_C and K_C is the largest such linear code for which this is true [3]. We will denote the dimension of the kernel of C by $k(C)$. Phelps and LeVan [10] established that for each such $m \geq 4$, there exists a nonlinear binary 1-perfect code of length $n = 2^m - 1$, with a kernel of dimension $k(C) = k$ for each $k \in \{1, 2, \dots, n - m - 2\}$.

The most intuitive approach to constructing nonlinear 1-perfect codes consists of starting with the Hamming code H_m, and *switching* out one specially selected set of codewords $S \subset H_m$ for another set of words S' such that the resulting code $C = (H_m \backslash S) \cup S'$ would still be a 1-perfect code. This idea has been developed from different approaches to construct binary 1-perfect codes, see [1], [2], [6] and [11]. In [7], it was used one generalization of this technique to construct q-ary 1-perfect codes. In [12], was generalized the approach developed in [10] to construct q-ary 1-perfect codes with different ranks. In this article, we will use this construction to construct q-ary 1-perfect codes with kernels of different sizes.

2. Kernel of q-ary codes. First of all, we generalize the definition of the kernel for a q-ary code C.

Definition 1 *The kernel of a q-ary code C is*

$$K_C = \{x \in \mathbb{F}_q^n \ : \ \lambda x + C = C \ \ \forall \lambda \in \mathbb{F}_q\}.$$

It is easy to see that if the zero word is in C, then K_C is a linear subspace of C. As in the binary case, we will denote the dimension of the kernel of C by $k(C)$.

Proposition 1 *Let K_C be the kernel of a q-ary code C. The code C is a union of cosets of K_C, and K_C is the maximal linear subspace of $\mathbb{F}_q^n$ with this property.*

3. Switching construction. Let $\mathbb{F}_q = \{0, \alpha^0, \alpha, \dots, \alpha^{q-2}\}$, where α is a primitive element. Let e_i denote the vector of length n having all components equal to zero, except the i^{th} component, which contains a one. Let H_m be a q-ary Hamming code of length $n = \dfrac{q^m-1}{q-1}$. Let T_i will denote the subspace spanned by the triples through the point i.

Proposition 2 *[12] Given a q-ary Hamming code H_m of length $n = \dfrac{q^m-1}{q-1}$, $m \geq 3$, $q \geq 3$ and $x_i \in H_m \backslash T_i$. Then,*

$$C' = (H_m \backslash (T_i + x_i)) \cup (T_i + x_i + \alpha^j e_i) \tag{1}$$

is a nonlinear q-ary 1-perfect code, $\forall i \in \{1, \dots, n\}$ and $\forall j \in \{0, 1, \dots, q-2\}$.

[*] Research partially supported by CICYT Grant TIC2000-0739-c04-01.

[†] Discrete & Statistical Sciences, Auburn University, Auburn, Al 36849-5307

[‡] Dep. d'Informàtica, Universitat Autònoma de Barcelona, 08193-Bellaterra, Spain

Next, we will see the kernel of the code C' given by (1) is $K_{C'} = T_i$. Actually, we will prove two more general results.

Proposition 3 *Let H_m be a q-ary Hamming code of length $n = \frac{q^m-1}{q-1}$, $m \geq 3$, $q \geq 3$ and let K be a subspace of H_m such that $T_i \subseteq K \subseteq H_m$ and $dimK \leq n-m-1$. Then*

$$C' = (H_m \backslash (K+y)) \cup (K+y+\alpha^j e_i)$$

is a nonlinear q-ary 1-perfect code with kernel K, $\forall i \in \{1,\ldots,n\}$, $\forall j \in \{0,1,\ldots,q-2\}$ and $\forall y \in H_m \backslash K$.

Let H_m be a q-ary Hamming code of length $n = \frac{q^m-1}{q-1}$. The parity check matrix of H_m consist of n pairwise linearly independent columns vectors of length m over $\mathbb{F}_q$. From H_m we can construct a projective space $PG(m-1,q)$ of dimension $m-1$ over $\mathbb{F}_q$, where the points are the columns of the parity check matrix of H_m and three points are in a line if the corresponding columns are linearly dependent [4]. Then, the elements of the support of a word of weight 3 are points that are in a line in the projective space. We will say that $\{1,2,\ldots,k\}$ is a set of *independent points* if the corresponding columns of the parity check matrix are a set of independent vectors, that is if in the projective space no set of three points are colinear.

By Proposition 2, once we have made one switch we have another q-ary 1-perfect code. Actually, it is proved [12] that for all $m \geq 4$, there exist $x_1, x_2, \ldots, x_m$ such that it is possible to make a series of switches. In this case, if $\{1,2,\ldots,m\}$ is a set of independent points of H_m, we can switch $T_1 + x_1$ with $T_1 + x_1 + \alpha^{j_1} e_1$, $T_2 + x_2$ with $T_2 + x_2 + \alpha^{j_2} e_2$, $\ldots$, $T_m + x_m$ with $T_m + x_m + \alpha^{j_m} e_m$, $\forall j_i \in \{0,\ldots,q-2\}$ $\forall i \in \{1,\ldots,m\}$, since $T_i + x_i$ and $T_k + x_k$ are always disjoint for all $k \neq i$.

Proposition 4 *Given a q-ary Hamming code H_m of length $n = \frac{q^m-1}{q-1}$, $m \geq 4$, with $\{1,2,\ldots,m\}$ as a set of its independent points. Then, the nonlinear q-ary 1-perfect code*

$$C' = \left(H_m \backslash \bigcup_{i=1}^{s} (T_i + x_i)\right) \cup \bigcup_{i=1}^{s} (T_i + x_i + \alpha^{j_i} e_i) \tag{2}$$

has kernel $K_{C'} = \cap_{i=1}^{s} T_i$, $\forall s \in \{1,2,\ldots,m\}$ and $\forall j_i \in \{0,1,\ldots,q-2\}$.

From Propositions 3 and 4, we can see that we can obtain q-ary 1-perfect codes C of length $n = \frac{q^m-1}{q-1}$, $m \geq 3$ and $q \geq 3$, with a kernel of dimension $k(C) = k$ for each $k \in \{dimT_i, \ldots, n-m-1\}$ and if $m \geq 4$, $k \in \{dim(\cap_{i=1}^{s} T_i) \; \forall s \in \{2,\ldots,m\}\}$.

In order to know the exact dimensions of the kernels obtained using the switching construction, we need to know the dimension of the subspaces T_i and the dimension of the intersection of some of these subspaces.

Proposition 5 *[12] Given a q-ary Hamming code H_m of length n, the dimension of T_i is $q^{m-1}-1$, $\forall i \in \{1,\ldots,n\}$.*

Proposition 6 *Given a q-ary Hamming code H_m with $\{1,2,\ldots,m\}$ as a set of its independent points, the dimension of $\cap_{i=1}^{s} T_i$ is $(q-1)^{s-1} q^{m-s}$, $\forall s \in \{2,\ldots,m\}$.*

References

[1] S. V. Avgustinovich, F. I. Solov'eva, *On projections of perfect binary codes*, Proc. Seventh Joint Swedish-Russian Int. Workshop on Inform. Theory, St. Petersburg, Russia (1995), 25-26.

[2] S. V. Avgustinovich, F. I. Solov'eva, *On non-systematic perfect binary codes*, Problems of Information Transmission, **32** (1996), no 3, 258-261.

[3] H. Bauer, B. Ganter, F. Hergert, *Algebraic techniques for nonlinear codes*, Combinatorica, **3**(1983), 21-33.

[4] I. F. Blake and R. C. Mullin, The Mathematical Theory of Coding, Academic Press, New York, 1975

[5] G. Cohen, I. Honkala, S. Litsyn and A. Lobstein, *Covering Codes*, North-Holland, Amsterdam, 1997.

[6] T. Etzion, A. Vardy, *Perfect binary codes: constructions, properties, and enumeration*, IEEE Trans. on Information Theory, **40**(1994), 754-763.

[7] T. Etzion, *Nonequivalent q-ary perfect codes*, SIAM J. Discrete Math. **9** (1996), no 3, 413-423.

[8] F.I. MacWilliams and N.J. Sloane, *The theory of Error-Correcting codes*, North-Holland, New York, 1977.

[9] B. Lindström *On group and nongroup perfect codes in q symbols*, Math. Scand., **25**(1969), 149-158.

[10] K. T. Phelps, M. LeVan, *Kernels of nonlinear Hamming codes*, Designs, Codes and Cryptography, **6**(1995), 247-257.

[11] K. T. Phelps, M. LeVan, *Switching equivalence classes of perfect codes*, Designs, Codes and Cryptography, **16**(1999), 179-184.

[12] K. T. Phelps, M. Villanueva, *Ranks of q-ary 1-perfect codes*, to appear in Designs, Codes and Cryptography, 2001.

[13] J. Schonheim, *On linear and nonlinear single-error-correcting q-nary perfect codes*, Info. and Control, **12** (1968) 23-26.

[14] J. L. Vasil'ev, *On nongroup close-packed codes*, Problemy Kibernetiki, 8 (1963), 337-339.

TWO APPROACHES TO BLOCK CIPHER ANALYSIS

Rostovtsev A. G. and Makhovenko E. B.
St. Petersburg State Polytechnic University
rostovtsev@ssl.stu.neva.ru

1. Block cipher analysis

Block ciphers are the main instrument for providing information confidentiality. Security of block ciphers is based on the difficulty of key computation when plaintexts and corresponding ciphertexts are known. The complexity of this problem is confirmed by the difficulty of determination computable metric, showing the distance between the solution and the tested key, because small key variations induce random ciphertext variations.

For providing security the number of encryption rounds is usually accepted large enough. But nobody has proved yet that cipher strength wouldn't decrease under increasing of number of rounds; this is cryptographic "folklore".

The universal cryptanalysis methods can be classified as statistical and algebraic. The statistical methods (differential [2], linear [5], key schedule [4], slide [3] use metric "on the average" and hence require large number of (plaintext/ciphertext) pairs. To counteract these methods one can periodically change a key, so obtaining of required number of texts becomes impossible.

Algebraic methods (Andelman – Reeds cryptanalysis [1], lattice cryptanalysis [8]) often use Polia's "generalization/reduction" method [6], so key changing doesn't strengthen a cipher in relation to these methods. Algebraic cryptanalysis needs to determine goal Boolean function, which is true only when tested key equals the solution, and to embed the underlying algebraic structure from $\mathbf{F}_2$ into an ordered set. Cryptanalysis technique implies searching of maximum of goal function [9] or rejecting of key set during single test [8]. Cryptanalysis problem is reduced to the problem of Boolean formula minimization. Note that method, suggested in [1], where extended goal function maximum is computed, provides exponential growth of complexity as a function of number of rounds; this lack can be eliminated [9].

2. Cryptanalysis methods

Let $G_n = \mathbf{F}_2[x_1, ..., x_n]/(x_1^2 \oplus x_1, ..., x_n^2 \oplus x_n)$, where $\oplus$ is addition in $\mathbf{F}_2$, be a ring of Zhegalkin polynomials. Each polynomial in G_n divides zero, but this ring possesses the property of unique factorization (irreducible polynomials have degree n), 1 is a unique invertible element in G_n. Boolean function can be uniquely represented as polynomial in G_n. The map $G_n \to \mathbf{F}_2$, computing polynomial value by giving binary meanings to its variables, is ring homomorphism.

Binary operations in G_n can be extended into ordered sets **Q** (rationals) and $\mathbf{Z}_2$ (2-adic integers):

$$a \oplus b \to |a-b|, \qquad ab \pmod 2 \to ab. \tag{1}$$

First operation is not associative, hence ordered set is not a ring, but has "characteristic" 2. Really computation in **Q** needs to use approximate numbers (small digits are eliminated). Analogous approximation in $\mathbf{Z}_2$ leads to quotient-ring $\mathbf{Z}/2^m\mathbf{Z}$ for some integer m under 2-adic valuation (the smallest valuation is $-m$, the largest one is 0).

Assume that one (plaintext/ciphertext) pair uniquely determines an n-bit key (in average 1.36 n bit texts are needed [7]) and cipher has no equivalent keys. Encryption and decryption process can be written as a composition of polynomial sets in G_n, depending on key bits, if plaintext and ciphertext bits are known. Let u_i be the i-th bit of intermediate text, produced from plaintext encryption on the first half of rounds, and v_i be the i-th bit of intermediate text, produced from ciphertext decryption on the second half of rounds. Let goal function be $H = \prod_{i=1}^{O(n)} (u_i \oplus v_i \oplus 1)$. It is obvious that $H = 1$ if and only if the tested key equals the solution.

Suggested cryptanalysis methods are based on finding of (local) maximum of goal functions $H_\mathbf{Q}$, $H_{\mathbf{Z}_2}$, embedded into **Q** and $\mathbf{Z}_2$ correspondingly. If $H = 1$, then $H_\mathbf{Q} = 1$ and $\mathrm{val}(H_{\mathbf{Z}_2}) = 0$, where for $H_{\mathbf{Z}_2} = 2^a b$ (b is odd) $\mathrm{val}(H_{\mathbf{Z}_2}) = -a$.

Searching of $H_\mathbf{Q}$ maximum is auxiliary iterative procedure, which allows to determine key bit estimations for some initial key approximation. For $H_\mathbf{Q}$ the initial approximation $\mathbf{k}^*$ has almost all key digits equal to 0.5. Algorithm is as follows:

1. Let $H^* = H(\mathbf{k}^*)$ be extended goal function and $\{H_i\} = \varnothing$.
2. For $i = 1$ to n compute extended goal functions H_i^0, H_i^1 for i-th key digit equal to 0 and 1 correspondingly. If $H_i^0 < H^* < H_i^1$, then $H_i = H_i^1$; if $H_i^1 < H^* < H_i^0$, then $H_i = H_i^0$.
   a. If $\{H_i\} \neq \varnothing$, then set $H_{\max} = \max H_i$ and find position j and meaning K_j of corresponding key digit. Change j-th digit of $\mathbf{k}^*$ to K and go to step 1.
   b. If $\{H_i\} = \varnothing$, then return $\{K_j\}$.

If set of initial approximations $\{\mathbf{k}^*\}$ is sufficiently large, then frequency ratio for estimated key bits can be found.

Chosen plaintext cryptanalysis method contains the following main stages.

Precomputation stage 1, where for arbitrary keys **k***, chosen plaintext **x** and corresponding ciphertext y matrix (p_{ij}) is computed for all key bits. Here p_{ij} $(i, j \in \mathbf{F}_2)$ is frequency ratio of obtaining key bit estimation i if real key bit is j. Matrix (p_{ij}) can be considered as the simple function of key with respect to encryption algorithm.

Stage 2 for unknown key, chosen plaintext and known ciphertext produces vector $\mathbf{P} = (P_0, P_1)$ for all key bit, where P_i is frequency ratio for estimation of key bit to be i.

Stage 3 includes solving of equation $\mathbf{P} = (p_{ij})\boldsymbol{\pi}$ for each key bit to find vector $\boldsymbol{\pi} = (\pi_0, \pi_1)$, where π_i is probability for key bit to be i.

Stage 4 consists of key set ordering in accordance with their probabilities and testing ordered keys.

This method does not allow to speed-up key computation if $\boldsymbol{\pi} = (0.5, 0.5)$, but sometimes allows to reject a subset of keys. This is possible when matrix (p_{ij}) as function of key allows to get information about $\boldsymbol{\pi}$ under assumption that unknown key belongs to the key subset.

The method of 2-adic extension is similar to the described one. It differs in few positions: initial approximation is such that almost all key digits are equal to 2; binary key estimation digits are to be chosen so that goal function has (local) maximum of $\mathrm{val}(H_{\mathbf{Z}_2})$; 2-adic valuations of corresponding functions instead of H, H^* are used.

These methods are based on the following assumption: if estimation is close to known key, it is close to unknown key too. This assumption looks fair because estimation computation doesn't require key knowledge. Some ciphers have positive bias defined as difference between probability of estimated bit be equal to the true key bit and 0.5. It allows to increment the complexity of key computation in comparison with enumeration. Described methods can be implemented for hash-function inversion too.

These methods are based on rational and 2-adic quasi-linearization: influence of key digits in small-degree monomials sometimes is greater than influence of key-digits in large-degree ones, because the signs of high-degree monomials are almost random.

3. Experiment outcomes

The offered methods were tested experimentally for 64-bit and 16-round substitution-permutation cipher. Each round consists of text and key XOR-addition, permutation $x_i \to x_{23i(\mathrm{mod}64)}$, extreme 4-bit substitution (0, 13, 11, 8, 3, 6, 4, 1, 15, 2, 5, 14, 10, 12, 9, 7) and 25 bit rotation (in some experiments rotation was round-dependent: $25r$ for round r). The substitution has differentials with probability less or equal to 0.25 and no possible differentials with weight 2. Absolute value of linear sums bias is less or equal to 0.25; absolute value of linear sums bias of weight 2 is less or equal to 0.125. This cipher seems to be strong in relation to differential and linear cryptanalysis. There are no known attacks, allowing to speed-up its cryptanalysis in comparison with enumeration under few known plaintexts.

Non-associativity of (1) causes errors in key bit estimations. To reduce this lack the output substitution bit polynomials were presented in specific form with reduced number of additions modulo 2.

Experiment, including stage 1 for rational and 2-adic extensions, shows that the consequence of key bit estimations is unbalanced: it contains more zeroes than units and looks like random one. The number of obtained estimations of key bits must be at least $O(\varepsilon^{-2})$ for bias ε. Frequency ratio of the event that found key bit estimation is valid, exceeds 0.5 in average. Vector of key bit frequency ratio was not estimated. Experiment shows that frequency ratio dependence on the number of encryption rounds is not monotone. The experimental bias is about 6% and is almost the same both for constant and round-dependent rotations. Rational and 2-adic extensions have approximately equal average biases.

This allows concluding that the key can be computed faster than by enumeration for single chosen plaintext.

References

1. D. Andelman and J. Reeds, On the cryptanalysis of rotor machines and substitution-permutation networks, *IEEE Transactions on Information Theory*, v. IT-28, 1982, pp. 578–584.
2. E. Biham and A. Shamir, Differential cryptanalysis of DES-like cryptosystems, *Advances in Cryptology — CRYPTO '90*, LNCS, v. 537, Springer-Verlag, 1991, pp. 2–21.
3. A. Biryukov and D. Wagner, Slide attacks, *Fast software encryption — FSE'99*, LNCS, v. 1636, 1999, pp. 245–259.
4. J. Kelsey, B. Schneier and D. Wagner, Key-schedule cryptanalysis of IDEA, G-DES, GOST, SAFER, and Triple-DES, *Advances in Cryptology — CRYPTO '96*, LNCS, v. 1109, Springer-Verlag, 1996, pp. 237–251.
5. M. Matsui, Linear cryptanalysis method for DES cipher, *Advances in Cryptology — EUROCRYPT '93*, LNCS, v. 765, 1994, pp. 386–397.
6. G. Polia, *Mathematical discovery*, Nauka, Moscow, 1976 (in Russian).
7. A. G. Rostovtsev and E. B. Makhovenko, *Introduction to public-key cryptology*, Mir & Semja, St. Petersburg, 2001. (in Russian).
8. A. G. Rostovtsev, Lattice cryptanalysis, *Information technology security*, 1997, v. 2, 53–55 (in Russian).
9. A. G. Rostovtsev, Method of iterated hash-function inversion, *Proceedings of "Methods and tools of information security providing"* conference, St. Petersburg, 2001, pp. 114–117 (in Russian).

A Construction of g-th MDS Codes from Matroids

Keisuke SHIROMOTO

Department of Electronics and Informatics
Ryukoku University
Seta, Otsu 520-2194, JAPAN;
keisuke@rins.ryukoku.ac.jp

Abstract

In this paper, we give a relationship between the generalized Hamming weights for linear codes over finite fields and the rank functions of matroids. We also consider a construction of g-th MDS codes from m-paving matroids.

1 Introduction

The closed connection between matroid theory and coding theory has been discussed by many researchers. For instance, Greene ([2]) gave a proof of the MacWilliams identity ([4]) for the Hamming weight enumerator of a linear code by using the Tutte polynomial of the corresponding matroid. Barg ([1]) studied the relation between the support weight enumerator of a linear code and the Tutte polynomial of the matroid. In addition, he showed the MacWilliams equation of the support weight enumerator in a simple form. In [7], Rajpal studied paving matroids and the corresponding linear codes.

The generalized Hamming weights of a linear code were introduced by Wei ([10]). The weights are natural extensions of the concept of minimum Hamming weights of linear codes. Many applications of the generalized Hamming weights are well-known. They are useful in cryptography (cf. [10]), in trellis coding (cf. [3]), etc. The generalized Hamming weights have been determined for binary Hamming codes, MDS codes, Golay codes, Reed-Muller codes and their duals ([10]).

The g-th maximum distance separable (MDS) code was defined by Wei ([10]) as a linear code which meets the generalized Singleton bound on the g-th generalized Hamming weight. In [9], Tsfasman and Vlădut gave a construction of the codes from algebraic-geometric codes.

In this paper, we consider the generalized Hamming weights for the m-paving codes. We also look for a construction of the codes from matroid theory. Then we give some examples of the codes.

2 GHW and m-Paving Matroids

We begin by introducing matroids, as in [6]. A *matroid* is an ordered pair $M = (E, \mathcal{I})$ consisting of a finite set E and a collection $\mathcal{I}$ of subsets of E satisfying the following three conditions:

(I1) $\emptyset \in \mathcal{I}$.

(I2) If $I \in \mathcal{I}$ and $I' \subseteq I$, then $I' \in \mathcal{I}$.

(I3) If I_1 and I_2 are in $\mathcal{I}$ and $|I_1| < |I_2|$, then there is an element e of $I_2 - I_1$ such that $I_1 \cup e \in \mathcal{I}$.

The members of $\mathcal{I}$ are the *independent sets* of M, and a subset of E that is not in $\mathcal{I}$ is called *dependent.* A minimal dependent set in M is called a *circuit* of M, and a maximal independent set in M is called a *base* of M. For a subset X of E, we define the *rank* of X as follows:

$$r(X) := \max\{|Y| \ : \ Y \subseteq X, \ Y \in \mathcal{I}\}.$$

Throughout this paper, let $\mathbb{F}_q$ be a finite field of q elements. For an $m \times n$ matrix A over $\mathbb{F}_q$, if E is the set of column labels of A and $\mathcal{I}$ is the set of subsets X of E for which the multiset of columns labelled by X is linearly independent in the vector space $\mathbb{F}_q^m$, then $M[A] := (E, \mathcal{I})$ is a matroid and is called *vector matroid* of A (cf. [6]).

For a vector $\boldsymbol{x} = (x_1, \ldots, x_n) \in \mathbb{F}_q^n$ and a subset $D \subseteq \mathbb{F}_q^n$, we define the *supports* of $\boldsymbol{x}$ and D respectively as follows:

$$\begin{aligned} \mathrm{supp}(\boldsymbol{x}) &:= \{i \mid x_i \neq 0\}, \\ \mathrm{Supp}(D) &:= \bigcup_{\boldsymbol{x} \in D} \mathrm{supp}(\boldsymbol{x}). \end{aligned}$$

Let C be an $[n, k]$ code over $\mathbb{F}_q$. For each g, $1 \leq g \leq k$, the g-th *generalized Hamming weight* (GHW) $d_g(C)$ is defined by Wei ([10]) as follows:

$$d_g(C) := \min\{|\mathrm{Supp}(D)| \ : \ D \text{ is an } [n, g] \text{ subcode of } C\}.$$

For any g, the following bound is well-known as the *generalized Singleton bound* ([10]):

$$d_g(C) \leq n - k + g.$$

Now, we introduce the connection between the generalized Hamming weights of a linear code and matroid theory. It is usual, for studying the relationship between linear codes and matroids, to deal with the matroid of a generator matrix of a linear code ([1], [7], etc.). In this paper, however, we shall study the rank $n - k$ matroid $M[H]$ of a parity-check matrix H of an $[n, k]$ code C to focus on the generalized Hamming weights of C. Since it finds that $M[H]$ is determined by C (not the chosen parity-check matrix H), we shall represent $M[H] = M_C$. However, a linear code C has more information than the matroid M_C. Indeed, a matroid is the vector matroid of several linear codes. It is also clear that the dual matroid $(M_C)^*$ corresponds to the matroid $M_{C^\perp}$ of the dual code $C^\perp$.

Theorem 2.1 *Let $M_C = M[H]$ be the vector matroid of a parity-check matrix H for an $[n,k]$ code C over $\mathbb{F}_q$. Then $d_g(C) = \delta$ for a g, $1 \leq g \leq k$, if and only if the following two conditions hold:*
(1) for any $(\delta-1)$-subset X of $E(M_C)$, $r(X) \geq \delta - g$;
(2) there exists a δ-subset Y of $E(M_C)$ with $r(Y) = \delta - g$.

Example 2.2 Let M_C be a uniform matroid $U_{n-k,n}$, that is, a matroid on an n-element set E, any $(n-k)$-element subset of E of which is a base. For any $(n-k+g-1)$-element subset X, it follows that $r(X) = n-k$ for every g, $1 \leq g \leq k$. There exists an $(n-k+g)$-element subset Y such that $r(Y) = n-k$ for every g. Therefore we have that $d_g(C) = n-k+g$ for every g. Consequently it follows that C is an MDS code.

An m-paving matroid was introduced by Rajpal ([8]) and the matroid is a generalization of a paving matroid, that is, a rank r matroid whose circuits have cardinality r or $r+1$.

Definition 2.3 A rank r matroid M is *m-paving* for $m \leq r$ if all circuits of M have cardinality exceeding $r - m$.

It is not difficult to show that any uniform matroid $U_{r,n}$ is 0-paving, and any paving matroid is 1-paving. These are the only 0-paving and 1-paving matroids. In [8], Rajpal showed that if G is a generator matrix of a first-order Reed-Muller code $R(1,m)$, then the matroid $M[G]$ is a maximal binary $(m-2)$-paving matroid.

For $m \leq n-k$, we define an *m-paving code* as an $[n,k]$ code C over $\mathbb{F}_q$ such that the matroid M_C is an m-paving. From the above argument, it is clear that the dual code $R(m-2,m)$ of a Reed-Muller code $R(1,m)$ is an $(m-2)$-paving code.

On the generalized Hamming weights of an m-paving code, we shall prove a bound.

Theorem 2.4 *If an $[n,k]$ code C over $\mathbb{F}_q$ is an m-paving code, then*

$$d_g(C) \geq n - k + g - m \tag{1}$$

for any $g, 1 \leq g \leq k$.

We consider a special class of linear codes defined as follows:

Definition 2.5 ([10]) Let C be an $[n,k]$ code over $\mathbb{F}_q$. For g, C is called a *g-th MDS code* if $d_g(C) = n-k+g$.

It is well-known that an MDS code is also a g-MDS code for any g and a g-MDS code is always a g'-th MDS code for any g', $g' \geq g$.

The following proposition is due to Tsfasman and Vlădut (Corollary 4.1 in [9]).

Proposition 2.6 *If C is an $[n,k,d]$ code and $r = n+2-k-d$, then the dual code $C^{\perp}$ is an r-th MDS code.*

Now we give a construction of g-th MDS codes from m-paving matroids. That also indicates a duality for g-th MDS codes. From Theorem 2.1, it is not difficult to prove the following lemma.

Lemma 2.7 *Let C be an $[n,k]$ code. Then C is a g-th MDS code if and only if $r(X) = n-k$ for any $(n-k+g-1)$-element subset $X \subseteq E(M_C)$.*

By using the lemma, we have the following theorem which is a generalizaton of the above proposition.

Theorem 2.8 *Let C be an $[n,k]$ code over $\mathbb{F}_q$. If C is a g-paving code for $1 \leq g \leq \min\{n-k, k-1\}$, then $C^{\perp}$ is a $(g+1)$-th MDS code.*

References

[1] A. Barg, The matroid of supports of a linear code, *Applicable Algebra in Engineering, Communication and Computing*, **8** (1997) pp. 165–172.

[2] C. Greene, Weight enumeration and the geometry of linear codes, *Studies in Applied Mathematics* **55** (1976) pp. 119–128.

[3] T. Kasami, T. Tanaka, T. Fujiwara and S. Lin, On the optimum bit orders with respect to the state complexity of trellis diagrams for binary linear codes, *IEEE Trans. Inform. Theory* **39** (1993) pp. 242–245.

[4] F. J. MacWilliams, A theorem on the distribution of weights in systematic code, *Bell Syst. Tech. J.* **42** (1962) 654.

[5] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*, North-Holland, Amsterdam 1977.

[6] J. G. Oxley, *Matroid Theory*, Oxford University Press, Oxford, 1992.

[7] S. Rajpal, On paving matroids and a generalization of MDS codes, *Discrete Applied Mathematics* **60** (1995) pp. 343–347.

[8] S. Rajpal, On binary k-paving matroids and Reed-Muller codes, *Discrete Mathematics* **190** (1998) pp. 191–200.

[9] M. A. Tsfasman and S. G. Vlădut, Geometric approach to higher weights, *IEEE Trans. Inform. Theory* **41** (1995) pp. 1564–1588.

[10] V. K. Wei, Generalized Hamming weights for linear codes, *IEEE Trans. Inform. Theory* **37** (1991) pp. 1412–1418.

[11] V. Wei, Generalized Hamming weights; Fundamental open problems in coding theory, *Arithmetic, geometry and coding theory* (Luminy, 1993) pp. 269–281.

[12] D. J. A. Welsh, *Matroid Theory*, Academic Press, London, 1976.

On McWilliams-type identities for orbit codes

A. Yu. Serebryakov, V. M. Sidelnikov

It is well known that McWilliams identities have numerous application in the coding theory. Sidelnikov [2] proposed an extension of McWilliams identities to a certain class of orbit group codes. Our aim is to generalize this results to a larger class of group.

Let G be a finite group, $\rho: G \to GL(V)$ — its finite dimensional unitary representation. Denote $g\mathbf{v} = \rho(g)\mathbf{v}$ where $g \in G$, $\mathbf{v} \in V$. Denote a space X as an orbit of a certain vector $\mathbf{a} \in V$: $X = G\mathbf{a}$. Let H be a subgroup of G. Then the orbit code C in the space X is defined as the orbit of $\mathbf{a}$ under the group action of H, i.e. $C = H\mathbf{a}$.

We consider the case when G is the n-wise direct product, i.e. $G = G_0 \times \ldots \times G_0$, $\rho_0 : G_0 \to GL(V_0)$ is a finite dimensional representation of the group G_0 and ρ is a representation of the group G in the space $V = V_0 \oplus \ldots \oplus V_0$, $\rho(g_1, \ldots, g_n)(v_1, \ldots, v_n) = (\rho_0(g_1)v_1, \ldots, \rho_0(g_n)v_n)$ $(g_i \in G_0,\ v_i \in V_0)$. The problem of deriving McWilliams-type identities for the orbit code C is reducible [3] to the problem of finding a monomial basis in the space of polynomials over the set $X_0 = G_0 v_0$ $(v_0 \in V_0)$, namely a basis $f_1, \ldots, f_r \in \mathbb{C}[X_0]$, such that

1). $gf_i = \chi_i(g) f_{j(g,i)}$, $\chi_i(g) \in \mathbb{C}$ for any $g \in G$;

2). the basis is a multiplicative group, i.e.

$$f_i \cdot f_j = f_{k(i,j)}.$$

It is known that this way McWilliams-type identities could be derived for Abelean groups as well as for two-dimensional irreducible representation of the quaternion group Q_8.

We consider left regular representations of groups $G_0 = S_3$, Q_8, D_{2n} (dihedral group). We prove for these representation that the monomial basis exists and its elements can be expressed explicitely as linear combinations of matrix entries of irreducible representations. In all the cases being considered we construct monomial bases which allows us to deive Mcwilliams-type identities for the considered matrix groups.

An Abelean group defined by elements of a monomial basis with multiplication as a group operation, can be considered as dual to the priginal group G_0, its order is just $|G_0|$. For left regular representations of the grops S_3, Q_8, D_{2n} we obtain this way Abelean groups $\mathbb{Z}_6$, $\mathbb{Z}_2 \oplus \mathbb{Z}_2 \oplus \mathbb{Z}_2$, $\mathbb{Z}_2 \oplus \mathbb{Z}_n$, correspondingly, which are dual to the original ones.

Example. For $G_0 = S_3$ we have that its left regular representation is the sum of 4 irreducible representations. The matrix elements of these representations form the basis of space $\mathbb{C}[G_0]$. We can describe these functions as vectors of its values:

$m_{11} = (1, -1/2, -1/2, -1, 1/2, 1/2);$
$m_{12} = (0, -\sqrt{3}/2, \sqrt{3}/2, 0, \sqrt{3}/2, -\sqrt{3}/2);$
$m_{21} = (0, \sqrt{3}/2, -\sqrt{3}/2, 0, \sqrt{3}/2, -\sqrt{3}/2);$
$m_{22} = (1, -1/2, -1/2, 1, -1/2, -1/2);$
$\chi_0 = (1, 1, 1, 1, 1, 1);$
$\chi_1 = (1, 1, 1, -1, -1, -1).$

We define $f_0 = \chi_0$, $f_1 = m_{11} + im_{12}$, $f_2 = im_{21} + m_{22}$, $f_3 = \chi_1$, $f_4 = -im_{21} + m_{22}$, $f_5 = m_{11} - im_{12}$. Then $f_0, f_1, \ldots, f_5$ is monomial basis. In particular, $f_i \cdot f_j = f_{i \oplus j 6}$. With the help of this basis we can derive McWilliams-type identities for the group $G = S_3 \times \ldots \times S_3$.

It is easy to show that in the space of finite dimensional representation of finite nilpotent group over the field $\mathbb{C}$ there exists a basis $f_1, \ldots, f_r$ such that for any $g \in G$ and any $i \in \{1, \ldots, r\}$ there exists $j \in \{1, \ldots, r\}$ for which holds $gf_i = \chi_i(g) f_j$. This is implicit by the fact that any irreducible representation of finite nilpotent group G is induced by some one-dimensional representation of subgroup $H \in G$ [3]. The following problem remains open: whether it is always possible to choose basis in the space $\mathbb{C}[X_0]$ of functions on the orbit X_0 of a given finite nilpotent group G_0 in such a way that its elements form a group with multiplication as a group operation (elements are multiplied as functions). This suppresedly leads to McWilliams-type group for any nilpotent group.

This work was supported by RFBR, project N 02-01-00687 and INTAS, N 00-738.

REFERENCES

1.F.J. Mac Williams, N.J.A. Sloane, The theory of error-correcting codes. Am.-N.Y.-Oxford, 1979.

2. Sidel'nikov V. M. MacWilliams-type identities for linear p-ary codes in non-Hamming spaces. — Seventh International Workshop on Algebraic and Combinatorial Theory, 18-24 June 2000, Bansko, Bulgaria. — pp. 275-278.

3. Kirillov A. A. Elements of representation theory. Moscow, "Nauka", 1972 (in Russian).

Results on minihypers following from results on t-fold $(n-k)$-blocking sets in $PG(n,q)$

L. Storme

Ghent University, Dept. of Pure Maths and Computer Algebra, Krijgslaan 281, 9000 Ghent, Belgium.

(ls@cage.rug.ac.be, http://cage.rug.ac.be/~ls)

(joint work with S. Ferret, P. Sziklai and Zs. Weiner)

Abstract

Minihypers in finite projective spaces are used to obtain results on linear codes meeting the Griesmer bound. Minihypers are particular classes of t-fold $(n-k)$-blocking sets in finite projective spaces. Our goal is to use characterization results on t-fold $(n-k)$-blocking sets to obtain characterization results on minihypers.

1 Introduction

In coding theory, the Griesmer bound states that if there exists an $[n,k,d;q]$ code for given values k,d and q, then $n \geq \sum_{i=0}^{k-1} \left\lceil \frac{d}{q^i} \right\rceil = g_q(k,d)$, where $\lceil x \rceil$ denotes the smallest integer greater than or equal to x.

The question arises whether there exists a linear $[n,k,d;q]$ code with length n equal to the lower bound $g_q(k,d)$. This coding-theoretical problem can be translated into a problem on *minihypers in projective spaces*. Let $PG(t,q)$ be the t-dimensional projective space over the finite field of order q.

Definition 1 *Let F be a set of f points in $PG(t,q)$, where $t \geq 2$ and $f \geq 1$. If $|F \cap H| \geq m$ for every hyperplane H in $PG(t,q)$ and $|F \cap H| = m$ for some hyperplane of $PG(t,q)$, then F is called an $\{f,m;t,q\}$*-minihyper.

Hamada showed that for $d = q^{k-1} - \sum_{i=1}^{h} q^{\lambda_i}$, there is a correspondence between linear $[n,k,d;q]$ codes meeting the Griesmer bound and $\{\sum_{i=1}^{h} v_{\lambda_i+1}, \sum_{i=1}^{h} v_{\lambda_i}; k-1, q\}$-minihypers; where $v_l = (q^l - 1)/(q-1)$, for any integer $l \geq 0$. Let $G = (g_1 \cdots g_n)$ be a generator matrix for a linear $[n,k,d;q]$ code, $d < q^{k-1}$, meeting the Griesmer bound.

Then the set $PG(k-1,q) \setminus \{g_1, \ldots, g_n\}$ is the minihyper linked to the code meeting the Griesmer bound.

Strong results for general values of n,k,d and q were obtained by Hamada, Helleseth and Maekawa, and by Ferret and Storme.

Theorem 2 (a) (Hamada, Helleseth, Maekawa [4, 5]) *A $\{\sum_{i=1}^{h} v_{\lambda_i+1}, \sum_{i=1}^{h} v_{\lambda_i}; k-1, q\}$-minihyper, with $h \leq \sqrt{q}$, is the disjoint union of a λ_0-,...,λ_h-dimensional subspace of $PG(k-1,q)$.*

(b) (Ferret and Storme [1]) *Let F be a $\{\sum_{i=0}^{s} \epsilon_i v_{i+1}, \sum_{i=0}^{s} \epsilon_i v_i; t, q\}$-minihyper, where $\sum_{i=0}^{s} \epsilon_i < 2\sqrt{q}$, $q > q_0$. Then F consists of the disjoint union of either:*

(1) ϵ_s spaces $PG(s,q), \epsilon_{s-1}$ spaces $PG(s-1,q), \ldots, \epsilon_0$ points,

(2) one subgeometry $PG(2l+1, \sqrt{q})$, for some integer l with $1 \leq l \leq s$, ϵ_s spaces $PG(s,q), \ldots, \epsilon_{l+1}$ spaces $PG(l+1,q), \epsilon_s - \sqrt{q} - 1$ spaces $PG(l,q), \epsilon_{l-1}$ spaces $PG(l-1,q), \ldots, \epsilon_0$ points,

(3) one subgeometry $PG(2l, \sqrt{q})$, for some integer l with $1 \leq l \leq s$, ϵ_s $PG(s,q), \ldots, \epsilon_{l+1}$ $PG(l+1,q), \epsilon_l - 1$ $PG(l,q), \epsilon_{l-1} - \sqrt{q}$ $PG(l-1,q), \epsilon_{l-2}$ $PG(l-2,q), \ldots, \epsilon_0$ points.

2 Multiple blocking sets in finite projective spaces

Definition 3 *A t*-fold $(n-k)$-blocking set *in $PG(n,q)$ is a set B of points of $PG(n,q)$ intersecting every k-dimensional subspace in at least t points. A t-fold $(n-k)$-blocking set B of $PG(n,q)$ is called* minimal *when no proper subset of B is still a t-fold $(n-k)$-blocking set.*

A 1-fold $(n-k)$-blocking set of $PG(n,q)$ is also simply called an $(n-k)$-blocking set *of $PG(n,q)$.*

Minihypers in finite projective spaces are particular examples of t-fold $(n-k)$-blocking sets in projective spaces.

Theorem 4 (Hamada [4, Theorem 2.5]) *Let k be any integer, $1 \leq k < n$. If F is a $\{\sum_{i=0}^{k} \epsilon_i v_{i+1}, \sum_{i=1}^{k} \epsilon_i v_i; n, q\}$-minihyper, with $0 \leq \epsilon_i \leq q-1$, $i = 0, \ldots, k$, then every $(n-k)$-dimensional subspace of $PG(n,q)$ intersects F in at least ϵ_k points.*

Recently, the following result on $(n-k)$-blocking sets was obtained.

Theorem 5 (Szőnyi and Weiner [6]) *Let B be a minimal $(n-k)$-blocking set in $PG(n,q)$, $q = p^h$, $p > 2$ prime, $h \geq 1$, of size less than $3(q^{n-k}+1)/2$. Then every subspace that intersects B in at least one point, intersects B in $1 \pmod p$ points.*

This $1 \pmod p$ result gives important information which can be used to obtain characterization results of minimal $(n-k)$-blocking sets.

3 New results on t-fold $(n-k)$-blocking sets

The results of Theorem 5 have been extended to the following theorem.

Theorem 6 (Ferret, Storme, Sziklai and Weiner [2]) *Let B be a minimal t-fold $(n-k)$-blocking set in $PG(n,q)$, $q = p^h$, $p > 2$ prime, $h \geq 1$, of size less than $(t+3/2)(q^{n-k}+1)$. Then every k-dimensional subspace intersects B in $t \pmod p$ points, and any subspace of dimension less than k intersects B in $0, 1, \ldots, t \pmod p$ points.*

The preceding result presently has been used to characterize 2-fold 2-blocking sets B in $PG(4,q)$, q square.

Theorem 7 (Ferret, Storme, Sziklai and Weiner [2]) *A minimal 2-fold 2-blocking set B in $PG(4,q)$, q square, of size at most $2(q^2 + q\sqrt{q} + q + \sqrt{q} + 1)$, is either*

(1) *the disjoint union of two Baer cones with point vertex r_i and with base a Baer subplane in a plane not passing through r_i, $i = 1, 2$,*

(2) *the disjoint union of a subgeometry $PG(4, \sqrt{q})$ and a Baer cone with point vertex r_1 and with base a Baer subplane in a plane not passing through r_1,*

(3) *the disjoint union of two subgeometries $PG(4, \sqrt{q})$.*

The goal of theorems of this type on t-fold $(n-k)$-blocking sets in $PG(n,q)$ is to obtain improvements to the known results (Theorem 2) on minihypers in finite projective spaces.

References

[1] S. Ferret and L. Storme, Minihypers and linear codes meeting the Griesmer bound: Improvements to results of Hamada, Helleseth and Maekawa, *Des. Codes Cryptogr.* **25** (2002), 143-162.

[2] S. Ferret, L. Storme, P. Sziklai and Zs. Weiner, Results on minihypers following from results on t-fold $(n-k)$-blocking sets in $PG(n,q)$, (In preparation).

[3] N. Hamada, A characterization of some $[n,k,d;q]$-codes meeting the Griesmer bound using a minihyper in a finite projective geometry, *Discr. Math.* **116** (1993), 229-268.

[4] N. Hamada and T. Helleseth, A characterization of some q-ary codes ($q > (h-1)^2, h \geq 3$) meeting the Griesmer bound, *Math. Japonica* **38** (1993), 925-940.

[5] N. Hamada and T. Maekawa, A characterization of some q-ary codes ($q > (h-1)^2, h \geq 3$) meeting the Griesmer bound: Part 2, *Math. Japonica* **46** (1997), 241-252.

[6] T. Szőnyi and Zs. Weiner, Small blocking sets in higher dimensions, *J. Combin. Theory, Ser. A* **95** (2001), 88-101.

Low-degree Testing or Distance to Reed-Solomon Codes

Cédric Tavernier

Projet codes, INRIA Rocquencourt
B.P. 105, 78150 Le Chesnay, France
email : Cedric.Tavernier@inria.fr

Abstract

We consider the field $\mathbb{F}_q$. Let $f : \mathbb{F}_q \to \mathbb{F}_q$ for which we only know a fraction of input and output. We suppose that q is large. We would like to give an answer to the following question: does there exist a polynomial of degree d which is very closed to the function f, and we would like to give an approximation of this distance, or equivalently, if we consider the smallest linear code of block length $q-1$ containing both $ev(f)$ and every codeword of the Reed-Solomon code $[q-1, d+1]_q$ we would like to give an approximation of the minimal distance between this last code and the Reed-Solomon code $[q-1, d+1]_q$.

1 Introduction, The Basic Univariate Test

We want to test whether f is a polynomial of total degree d. M. Kiwi [2] describe equivalent tests that achieve this goal. Let P_d denote the set of polynomials from $\mathbb{F}_q$ to $\mathbb{F}_q$ of total degree d, and $C_f(d)$ the smallest linear code of block length $q-1$ containing both $ev(f)$ and every codeword of the Reed-Solomon code $C(d) = [q-1, d+1]_q$, $C_f(d) \stackrel{def}{=} \{\phi ev(f) + \theta g \mid g \in C$ and $\phi, \theta \in \mathbb{F}_q\}$. Here is these equivalent tests.

- Basic Univariate Test [3]: Randomly pick $d+2$ distinct points $x_0, \ldots, x_{d+1}$ in $\mathbb{F}_q$. Then, accept if there exists a polynomial in P_d that agrees with f on $x_0, \ldots, x_{d+1}$, and reject otherwise.

- Basic Univariate Test: let $C(d)$ be the code whose elements are of the form $(p(x) : x \in \mathbb{F}_q)$ where p ranges over P_d. Randomly choose a dual codeword $\lambda \in C(d)^\perp$ of weight $d+2$. Then, accept if $\lambda \in C_f(d)^\perp$, and reject otherwise.

Recall that the minimal distance of a code C is the minimum weight of the codewords in C, and is denoted $\mathrm{wt}(C)$. We denote $\rho\mathrm{wt}(C)$ the relative minimum distance of a code C as the minimal distance of a code C divided by its block length. So if we denote $\Delta(f, P_d)$ the normalized distance, we see that $\Delta(f, P_d) = \rho\mathrm{wt}(C_f(d) \setminus C(d))$.

Theorem 1 *[3] Given a positive integer d, a finite field $\mathbb{F}_q$ of size at least $d+2$ and a function $f : \mathbb{F}_q \to \mathbb{F}_q$, if f satisfies*

$$Pr\left[\exists g \in \mathbb{F}_{2^n}^{(d)}[x] \text{ such that } g(x_i) = f(x_i)\ \forall i \in \{0, \ldots, d+1\}\right] \geq 1 - \delta,$$

where the probability is taken over the uniform distribution over all $d+2$-tuples $\{x_0, \ldots, x_{d+1}\}$ of distinct elements from $\mathbb{F}_q$, then $\Delta(f, P_d) \leq \delta$ thus $\rho wt(C_f(d) \setminus C(d)) \leq \delta$.

The testers above establish that univariate testing can be done in polynomial time (in d), and probes f in only $\mathcal{O}(d)$ places [3], but from the point of view of testing it is not very useful, since it is not very "different" from interpolation.

2 Test based on evenly spaced points over prime field

We now describe a tester which only works for fields of the form $\mathbb{F}_p$ for a prime p [3].

Definition 1 *We say that a set of points $\{x_0, \ldots, x_n\}$ is evenly spaced if $\exists h$ such that $x_i = x_0 + i * h$.*

Lemma 1 *Given a positive integer d and a prime $p \geq d+2$. The points $\{(x_i, y_i) | i \in \{0, \ldots, d+1\}; x_i = x + i * h; x_i, y_i \in \mathbb{F}_p\}$ lie on a degree d polynomial if and only if $\sum_{i=0}^{d+1} \alpha_i y_i = 0$, where $\alpha_i = (-1)^{(i+1)} \binom{d+1}{i}$.*

Theorem 2 *Given a positive integer d, a prime $p \geq d+2$ and a function $f : \mathbb{F}_p \to \mathbb{F}_p$ such that*

$$\underset{x,h \in \mathbb{F}_{2^n}}{P}\left[\sum_{i=0}^{d+1} \alpha_i \cdot f(x_i) = 0\right] \geq 1 - \delta \text{ where } \delta \leq \frac{1}{2(d+2)^2},$$

then $\Delta(f, P_d) \leq 2\delta$, or equivalently $\rho wt(C_f(d) \setminus C(d)) \leq 2\delta$.

In particular, the bound above implies that the tester resulting from this theorem would need to probe f in $\mathcal{O}(d^3)$. We get the following Evenly-Spaced-Test:

```
Repeat $\mathcal{O}(d^2\log(1/\beta))$ times
Pick $x, h \in \mathbb{F}_p \times \mathbb{F}_p$ and verify that $\sum_{i=0}^{d+1} \alpha_i \cdot f(x + i * h) = 0$
Reject if any of the test fails.
```

Theorem 3 *If the output of a program can be expressed by a low-degree polynomial correctly on all its inputs from $\mathbb{F}_p$, then it is passed by Evenly-Spaced-Test. If the output of the program is not $\mathcal{O}(\frac{1}{d^2})$-close to a univariate polynomial, then with probability $1-\beta$, it is rejected by Evenly-Spaced-Test.*

3 Evenly-Spaced-Test for Extension of Prime Fields

We now extend the last results to the field $\mathbb{F}_q = \mathbb{F}_{p^n}$. ω denote a primitive element of $\mathbb{F}_{p^n}$.

Definition 2 *We say that a set of distinct points $\{x_0, \ldots, x_n\}$ is regularly spaced if there exist $x, h, \omega \in \mathbb{F}_{p^n} \times \mathbb{F}^*_{p^n} \times \mathbb{F}^*_{p^n}$, such that $x_0 = x$ et $x_i = x + \omega^{i-1} * h$ pour $i \in \{1, \ldots, d+1\}$.*

Theorem 4 *Let d an integer such that $p^n > d+1$ and a function $f : \mathbb{F}_{p^n} \to \mathbb{F}_{p^n}$. Let $\{x_0, \ldots, x_{d+1}\}$ a regularly-spaced set with $x_0 = x$ and $x_i = x + h \cdot \omega^{i-1}$. Let $y_i = f(x_i)$, $i \in \{0, \ldots, d+1\}$. The set of (x_i, y_i) lie on a degree at most d polynomial if and only if $\sum_{i=0}^{d+1} \alpha_i(\omega, d) \cdot y_i = 0$ where α_i are given by the following recurrence $B_1^0 = B_1^1 = 1$, $B_i^0 = B_{i-1}^0/\omega^{d-i+2}$, $B_i^i = B_{i-1}^{i-1}/(\omega^{i-1} - \omega^d)$ and $B_i^t = B_{i-1}^{t-1}/A_{i-1}^{t-1} - B_{i-1}^t/A_{i-1}^t$ where $A_i^t = \omega^{t-1} - \omega^{t+d-i}$ $t \in \{0, \ldots, i\}$, then $\alpha_t(\omega, d) = B_{d+1}^t$.*

Proof: By linearity we can consider that we test the polynomial X^d. For $j \in \{1, \ldots, d\}$ et $s \in \{1, \ldots, d-j+2\}$, we define the function $f^{(j)}(x_{s-1}, \ldots, x_{s+j-1}) = \frac{f^{(j-1)}(x_{s-1}, \ldots, x_{s+j-2}) - f^{(j-1)}(x_s, \ldots, x_{s+j-1})}{x_{s-1} - x_{s+j-1}}$ with $f^{(0)}(x_i) = f(x_i) = x_i^d$, $i \in \{0, \ldots, d+1\}$. We show this theorem by recurrence: If $j = 1$ we see that $f^{(j)}(x_{s-1}, x_s)$ is the sum of all monomial in x_{s-1}, x_s of degree $d-1$, we suppose that $f^{(j)}(x_{s-1}, \ldots, x_{s+j-1})$ is the sum of all monomial of degree $n-j$. Now at rank $j+1$, for any monomial $x_{s-1}^{i_{s-1}} x_s^{i_s} \ldots x_{s+j-1}^{i_{s+j-1}}$ of $f^{(j)}(x_{s-1}, \ldots, x_{s+j-1})$ we have the monomial $x_{s+j}^{i_{s-1}} x_s^{i_s} \ldots x_{s+j-1}^{i_{s+j-1}}$ of $f^{(j)}(x_{s-1}, \ldots, x_{s+j-1})$ with $i_{s-1} + \ldots + i_{s+j-1} = n-j$, and
$x_{s-1}^{i_{s-1}} x_s^{i_s} \ldots x_{s+j-1}^{i_{s+j-1}} - x_{s+j}^{i_{s-1}} x_s^{i_s} \ldots x_{s+j-1}^{i_{s+j-1}} = (x_{s-1} - x_{s+j}) \cdot M_{s-1,s+j}^{i_{s-1}-1} \cdot x_s^{i_s} \ldots x_{s+j-1}^{i_{s+j-1}}$ where $M_{s-1,s+j}^{i_{s-1}-1}$ is the sum of all monomials of degree $i_{s-1}-1$ in x_{s-1}, x_{s+j}, so we see that $M_{s-1,s+j}^{i_{s-1}-1} \cdot x_s^{i_s} \ldots x_{s+j-1}^{i_{s+j-1}}$ is a sum of monomial of degree $n-j-1$ in $x_s, \ldots, x_{s+j-1}$. We get that $f^{(j+1)}$ is the set of all monomials of degree $n-j-1$. Thus if f is a polynomial of degree at most d then $f^{(d+1)}(x_0, \ldots, x_{d+1}) = 0$ The construction of $f^{(j)}$ immediately gives the proof of the converse and states that there exists $\alpha_i(\omega, d)$ which never depends of h since $x_i - x_j = h \cdot (\omega^{i-1} - \omega^{j-1})$ and such that $\sum_{i=0}^{d+1} \alpha_i(\omega, d) \cdot y_i = 0$.

Theorem 5 *Given a positive integer d, a integer n such that $p^n \geq d+2$ and a function $f : \mathbb{F}_{p^n} \to \mathbb{F}_{p^n}$ such that*

$$\underset{x,h \in \mathbb{F}_{p^n}}{P}\left[\sum_{i=0}^{d+1} \alpha_i(\omega, d) \cdot f(x_i) = 0\right] \geq 1 - \delta \quad ou \quad \delta \leq \frac{1}{2(d+2)^2},$$

then $\Delta(f, P_d) \leq 2\delta$, or equivalently $pwt(C_f(d) \setminus C(d)) \leq 2\delta$.

4 Extending the tester to multivariate polynomials

Theorem 6 *Given a finite field $\mathbb{F}_q$, such that $q > md$ and a function $f : \mathbb{F}_q^n \to \mathbb{F}_q$ such that*

$$\underset{x,h \in \mathbb{F}_q^n}{P}\left[\sum_{i=0}^{d+1} \alpha_i(\omega, d) \cdot f(x_i) = 0\right] \geq 1 - \delta \quad where \quad \delta \leq \frac{1}{2(d+2)^2}.$$

then $\Delta(f, P_d^n) \leq 2\delta$, or equivalently $pwt(C_f(d) \setminus C(d)) \leq 2\delta$. Here $C(d)$ denote the Reed-Muller code $R[d, m]_q$.

In conclusion we can say that the theorems 3 is true for these last cases and the proof is similar to Sudan's proof. Unfortunately These tests above are usefull only if the probability δ is very clothed to 0.

References

[1] T. Jakobsen. Cryptanalysis of block ciphers with probalistic non linear relations of low degree. In H. Krawczyk, editor, *Crypto'98*, number 1462 in LNCS, pages 347–362. Springer, 1998.

[2] M. Kiwi. Testing and weight distribution of dual codes. In *Technical Report TR-97-010*, 1997.

[3] M. Sudan. *Efficient Checking of Polynomial and proofs and the hardness of Approximation Problems.* PhD thesis, University of California, Berkeley, 1992.

[4] M. Sudan. Improved low degree testing and its application. In *Technical Report*, 1997.

Upper bounds on the size of insertion/deletion correcting codes

Ludo Tolhuizen, Philips Research Laboratories, Eindhoven, The Netherlands

May 28, 2002

1 Introduction

Let Q be a set with q elements. The Levenshtein distance $d_L(\mathbf{x}, \mathbf{y})$ of two finite strings $\mathbf{x}$ and $\mathbf{y}$ of elements from Q is the minimum number of deletions plus insertions of symbols that is required to transform $\mathbf{x}$ into $\mathbf{y}$. In this paper, we will give upper bounds on $M_q(n, s)$, the maximum size (or cardinality) of a code $C \subset Q^n$ for which any two distinct codewords have Levenshtein distance exceeding $2s$. We will use the following notation and results that can be found in [1] (where also references are given to the first publications of these results).

The size of a set A will be denoted by $|A|$. The number of runs of equal symbols in the vector $\mathbf{x}$ will be denoted by $r(\mathbf{x})$. So, for example, $r(001200000) = 4$. It is well-known that

$$|\{\mathbf{x} \in Q^n \mid r(\mathbf{x}) = r\}| = \binom{n-1}{r-1} q(q-1)^{r-1}. \tag{1}$$

For each $\mathbf{x} \in Q^n$ and each $t \in \{0, 1, \ldots, n-1\}$, $D_t(\mathbf{x})$ denotes the set of all vectors in Q^{n-t} that can be obtained by deleting t symbols from $\mathbf{x}$. The size of $D_t(\mathbf{x})$ depends on $\mathbf{x}$. As an example, $D_1(\mathbf{0})$ consists of the all-zero vector only, while $D_1(0101\ldots01)$ has as many as n elements. It is known that [2]

$$\text{for each } n \in \mathbb{N} \text{ and each } \mathbf{x} \in Q^n, \quad |D_t(\mathbf{x})| \geq \sum_{i=0}^{t} \binom{r(\mathbf{x}) - t}{i}. \tag{2}$$

One can prove (2) by induction, using that $D_t(\mathbf{x})$ contains all vectors with no deletion from the first run and t deletions from the $r-1$ subsequent runs, and all vectors with a single deletion from the first run, no deletion in the second run, and $t-1$ deletions from the $r-2$ subsequent runs. The bound in (2) is sharp in the sense that for each t and $r \geq t$, there exists an integer n and a vector $\mathbf{x} \in Q^n$ such that $r(\mathbf{x}) = r$ and $|D_t(\mathbf{x})|$ satisfies (2) with equality.

For each $\mathbf{x} \in Q^n$ and $t \geq 0$, $I_t(\mathbf{x})$ denotes the set of all vectors in Q^{n+t} that can be obtained from $\mathbf{x}$ by insertion of t symbols. Surprisingly, $|I_t(\mathbf{x})|$ does not depend on $\mathbf{x}$, and in fact [1]

$$\text{for all } \mathbf{x} \in Q^n \text{ and each } t \geq 0, \quad |I_t(\mathbf{x})| = \sum_{i=0}^{t} \binom{n+t}{i} (q-1)^i. \tag{3}$$

Equation 3 can be proved by induction. In the induction step, it is used that $I_t(x_1, x_2, \ldots, x_n) = \bigcup_{\alpha \neq x_1} \{\alpha\mathbf{y} \mid \mathbf{y} \in I_{t-1}(x_1, \ldots, x_n)\} \cup \{x_1\mathbf{z} \mid \mathbf{z} \in I_t(x_2, \ldots x_n)\}$, which implies that $|I_t(x_1, \ldots, x_n)| = (q-1)|I_{t-1}(x_1, \ldots, x_n)| + |I_t(x_2, \ldots, x_n)|$.

The following proposition will be used in the derivation of the upper bounds.

Proposition 1 *If $d_L(\mathbf{x}, \mathbf{y}) > 2s$, then $I_s(\mathbf{x}) \cap I_s(\mathbf{y}) = D_s(\mathbf{x}) \cap D_s(\mathbf{y}) = \emptyset$*

Indeed, suppose there is a vector $\mathbf{z}$ in $I_s(\mathbf{x}) \cap I_s(\mathbf{y})$. Then $\mathbf{x}$ can be transformed with s insertions to $\mathbf{z}$, and $\mathbf{z}$ can be transformed with s deletions to $\mathbf{y}$, and so $d_L(\mathbf{x}, \mathbf{y}) \leq 2s$. The statement about $D_s(\mathbf{x})$ and $D_s(\mathbf{y})$ is proved similarly.

2 The upper bounds

Theorem 1 *If $1 \leq s \leq n$, then $M_q(n, s) \leq q^{n+s} / \sum_{i=0}^{s} \binom{n+s}{i} (q-1)^i$.*

Proof. Let $C \subset Q^n$ have minimum Levenshtein distance exceeding $2s$. Then we have that

$$q^{n+s} \geq |\cup_{\mathbf{x} \in C} I_s(\mathbf{x})| = \sum_{\mathbf{x} \in C} |I_s(\mathbf{x})| = |C| \sum_{i=0}^{s} \binom{n+s}{i} (q-1)^i.$$

The first equality holds because $I_s(\mathbf{x}) \cap I_s(\mathbf{y}) = \emptyset$ for any two distinct words $\mathbf{x}$ and $\mathbf{y}$ from C. The second equality follows from (3). □

The following upper bound is due to Levenshtein [3]:

Theorem 2 *If $1 \leq s \leq r+1 \leq n$, then*

$$M_q(n, s) \leq \frac{q^{n-s}}{\sum_{i=0}^{s} \binom{r-s+1}{i}} + q \sum_{i=0}^{r-1} \binom{n-1}{i} (q-1)^i.$$

Proof. The first term in the right-hand side is an upper bound to the number of code words with at least $r+1$ runs of equal symbols. It is obtained by combining (2) and the fact that $D_s(\mathbf{x}) \cap D_s(\mathbf{y}) = \emptyset$ for any two distinct words $\mathbf{x}$ and $\mathbf{y}$ in a code with minimum Levenshtein distance exceeding $2s$. The second term equals the number of words in Q^n with at most r runs, which clearly is an upper bound the number of words in C with at most r runs. □

We will modify Theorem 2 by giving another upper bound to the number of code words with at most r runs. To this end, we observe that insertion of a single symbol in a vector $\mathbf{x}$ results in a vector with at most $r(\mathbf{x}) + 2$ runs (equality occurs if and only if between two equal symbols a different symbol is inserted). From this observation, it readily follows (formally by induction on s) that insertion of s symbols in $\mathbf{x}$ results in a vector with at most $r(\mathbf{x}) + 2s$ runs.
Now, let C be a q-ary code with minimum Levenshtein distance exceeding $2s$ with the additional property that $r(\mathbf{c}) \leq r$ for all codewords $\mathbf{c}$. For each $\mathbf{x} \in C$, each element from $I_s(\mathbf{x})$ has at most $2s + r$ runs of equal symbols. As $I_s(\mathbf{x}) \cap I_s(\mathbf{y}) = \emptyset$ for all distinct $\mathbf{x}$,$\mathbf{y}$ in C, it follows that

$$|C| \sum_{i=0}^{s} \binom{n+s}{i} (q-1)^i = \sum_{\mathbf{x} \in C} |I_s(\mathbf{x})| = |\cup_{\mathbf{x} \in C} I_s(\mathbf{x})| \leq |\{\mathbf{z} \in Q^{n+s} \mid r(\mathbf{z}) \leq 2s + r\}| =$$

$$= q \sum_{i=0}^{r+2s-1} \binom{n+s-1}{i} (q-1)^i, \text{ where the final equality follows from (1).}$$

As a consequence, we have the following modification of Theorem 2:

Theorem 3 *If $1 \leq s \leq r+1 \leq n$, then*

$$M_q(n, s) \leq \frac{q^{n-s}}{\sum_{i=0}^{s} \binom{r-s+1}{i}} + \frac{q \sum_{i=0}^{r+2s-1} \binom{n+s-1}{i} (q-1)^i}{\sum_{i=0}^{s} \binom{n+s}{i} (q-1)^i}.$$

Theorems 2 and 3 contain a parameter r over which we minimize. For later use, we remark that in both theorems, the first term is decreasing in r, while the second term is increasing in r.

3 Asymptotic behaviour of the upper bounds

In this section, we consider the asymptotic versions of the upper bounds from Theorems 1,2 and 3. We denote the q-ary entropy function by $h_q(x)$, that is,

$$h_q(x) = -x\log_q(x) - (1-x)\log_q(1-x) + x\log_q(q-1).$$

Moreover, we write $h_q^*(x) = h_q\left(\min(x, 1-\frac{1}{q})\right)$. As is well known, for each $\lambda \in (0,1)$ we have

$$\lim_{n\to\infty}\frac{1}{n}\log_q\left(\sum_{i=0}^{\lambda n}\binom{n}{i}(q-1)^i\right) = h_q^*(\lambda) \tag{4}$$

For each $\sigma \in (0,1)$, we define

$$\mu_q(\sigma) := \limsup\{\frac{1}{n}\log_q(M_q(n, \lfloor\sigma n\rfloor)) \mid n \in \mathbb{N}\}.$$

Let $\sigma \in (0,1)$. The following inequalities readily follow from combination of (4) with Theorem 1, 2 and 3, respectively, where we take $r = \rho n$ in the latter two theorems.

$$\mu_q(\sigma) \le u_q(\sigma) := (1+\sigma)\left(1 - h_q^*(\frac{\sigma}{1+\sigma})\right). \tag{5}$$

For each $\rho \in (\sigma, 1]$, we have

$$\mu_q(\sigma) \le \max\left(a_q(\sigma,\rho), h_q^*(\rho)\right), \text{ where } a_q(\sigma,\rho) = 1 - \sigma - \log_q(2)\cdot(\rho-\sigma)h_2^*(\frac{\sigma}{\rho-\sigma}). \tag{6}$$

For each $\rho \in (\sigma, 1]$, we have

$$\mu_q(\sigma) \le \max\left(a_q(\sigma,\rho), b_q(\sigma,\rho)\right), \text{ where } b_q(\sigma,\rho) = (1+\sigma)\left[h_q^*(\frac{\rho+2\sigma}{1+\sigma}) - h_q^*(\frac{\sigma}{1+\sigma})\right]. \tag{7}$$

The right-hand sides of (6) and (7) can be optimized over ρ. It follows from the remark at the end of Section 2 that for each fixed σ, $a_q(\sigma,\rho)$ in non-increasing in ρ, while $h_q^*(\rho)$ and $b_q(\sigma,\rho)$ both are non-decreasing in ρ. These facts makes determination of the optimal value of ρ relatively easy, as can be seen in the following theorems.

Theorem 4 *If* $1-\sigma \le h_q(\sigma)$, *then* $\inf\{\max(a_q(\sigma,\rho), h_q^*(\rho) \mid \rho \in (\sigma,1)\} = h_q(\sigma)$.
If $1-\sigma > h_q(\sigma)$, *then the minimum of the function* $\max(a_q(\sigma,\rho), h_q^*(\rho))$ *is attained for* $\rho = \rho_q(\sigma)$, *where* $\rho_q(\sigma)$ *is the element in* $(\sigma, 1-\frac{1}{q})$ *that satisfies* $a_q(\sigma,\rho_q(\sigma)) = h_q^*(\rho_q(\sigma))$.

Proof. Combination of the monotonicity of $a_q(\sigma,\rho)$ and $h_q^*(\rho)$, the fact that $a_q(\sigma,\rho) \to 1-\sigma$ if $\rho \downarrow \sigma$, and the obvious inequality $a_q(\sigma, 1-\frac{1}{q}) < 1 = h_q(1-\frac{1}{q})$. □

For describing the value of ρ that minimizes (7), we define $\sigma_q^* = \frac{1}{q}(q-1-(q+1)\sigma)$. As $\frac{\rho+2\sigma}{1+\sigma} \ge \frac{q-1}{q}$ if and only if $\rho \ge \sigma_q^*$, we have that $b_q(\sigma,\rho) = u_q(\sigma)$ if and only if $\rho \ge \sigma_q^*$.

Theorem 5 *If* $\sigma \ge \frac{q-1}{2q+1}$, *or* $a_q(\sigma,\sigma_q^*) \ge u_q(\sigma)$, *then for all* $\rho \in (\sigma,1)$, *we have* $\max(a_q(\sigma,\rho), b_q(\sigma,\rho)) \ge u_q(\sigma)$.
If $a_q(\sigma,\sigma_q^*) < u_q(\sigma)$ *and* $1-\sigma > b_q(\sigma,\sigma)$, *then the minimum of the function* $\max(a_q(\sigma,\rho), b_q(\sigma,\rho))$ *is attained for* $\rho = \gamma_q(\sigma)$, *where* $\gamma_q(\sigma)$ *is the element in* $(\sigma,1)$ *satisfying* $a_q(\sigma,\gamma_q(\sigma)) = b_q(\sigma,\gamma_q(\sigma))$.
If $a_q(\sigma,\sigma_q^*) < u_q(\sigma)$ *and* $1-\sigma \le b_q(\sigma,\sigma)$, *then* $\inf\{\max(a_q(\sigma,\rho), b_q(\sigma,\rho)) \mid \rho \in (\sigma,1)\} = b_q(\sigma,\sigma)$.

Proof. If $\sigma \ge \frac{q-1}{2q+1}$, then $\sigma \ge \sigma_q^*$, and so $b_q(\sigma,\rho) = u_q(\sigma)$ for each $\rho \ge \sigma$.
Next, assume that $\sigma < \frac{q-1}{2q+1}$. If $\rho \ge \sigma_q^*$, then $b_q(\sigma,\rho) = u_q(\sigma)$. If $a_q(\sigma,\sigma_q^*) \ge u_q(\sigma)$, then the monotonicity of a_q implies that $a_q(\sigma,\rho) \ge u_q(\sigma)$ for all $\rho \in (\sigma,\sigma_q^*)$. Hence, in this case, $\max(a_q(\sigma,\rho), b_q(\sigma,\rho)) \ge u_q(\sigma)$ for all $\rho \ge \sigma$.
The final two claims follow from monotonicity, and the facts that $a_q(\sigma,\rho) \to 1-\sigma$ if $\rho \downarrow \sigma$, and $b_q(\sigma,\sigma_q^*) = u_q(\sigma)$. □

3.1 More results for the binary case

In this section, we concentrate on the binary case ($q = 2$). We give some analytical results and plot the graphs of the upper bounds from (5), (6) and (7), the latter two optimized over ρ.

Let σ_2 be the solution to the equation $1-3x = h_2(3x)$. As stated in [3], for each ρ and σ for which $\rho \ge \sigma \ge \sigma_2$, we have

$$\max(a_2(\sigma,\rho)), h_2^*(\rho)) \ge h_2(3\sigma_2). \tag{8}$$

Indeed, if $\rho \ge 3\sigma_2$, then (8) trivially holds. If $\rho < 3\sigma_2$, then $\rho < 3\sigma$ and so $a_2(\sigma,\rho) = 1-\rho \ge 1-3\sigma_2 = h_2(3\sigma_2)$. Numerically, we find that $\sigma_2 = 0.07570$. For each $\sigma \ge \sigma_2$ we have

$$u_2(\sigma) \le u_2(\sigma_2) = 0.68058 < 0.77291 = h_2(3\sigma_2). \tag{9}$$

Combining (8) and (9), we see that for each $\sigma \ge \sigma_2$, (5) yields a sharper upper bound than (6).

Let σ_0 be the solution of the equation $a_2(\sigma,\sigma_2^*) = a_2(\sigma,\frac{1}{2}(1-3\sigma)) = u_2(\sigma)$. Numerically, we find that $\sigma_0 \approx 0.079642$, and that $a_2(\sigma,\frac{1}{2}(1-3\sigma)) \ge u_2(\sigma)$ for all $\sigma \ge \sigma_0$. Combining this with Theorem 5, and the fact that $\sigma_0 \ge \sigma_2$, we conclude the following.

Theorem 6 *For* $q = 2$ *and* $\sigma \ge \sigma_0 \approx 0.079642$, *for no value of* ρ *the bounds from (6) or (7) are sharper than (5).*

Numerical results suggest that in the binary case, (7) yields a sharper upper bound than (6) for all $\sigma < \sigma_0$.

Figure 1 shows the graphs of the upper bounds from (5), (6) and (7) as function of $\sigma \in (0, 0.1)$. (For $\sigma > 0.1$, the bound $u(\sigma)$ from (5) is the sharpest of all). The graph labelled was "MRRW" was obtained as follows. For any two vectors **x** and **y** of equal length, $d_L(\mathbf{x},\mathbf{y})$ is at most twice the Hamming distance between **x** and **y**. Therefore, any upper bound on the size of a q-ary code of length n with Hamming distance exceeding s is an upper bound on $M_q(n,s)$. The graph labelled with MRRW corresponds to the best known asymptotic upper bound on the rate of a binary code with relative Hamming distance σ, due to [4]. It is seen to be less sharp than the three bounds above for small values of σ, but it is sharper for larger values of σ. In particular, the MRRW bound implies that $\mu_2(\sigma) = 0$ for each $\sigma \ge 0.5$. This fact cannot be proved with the other bounds (note that $u_2(0.5) = 0.122556$).

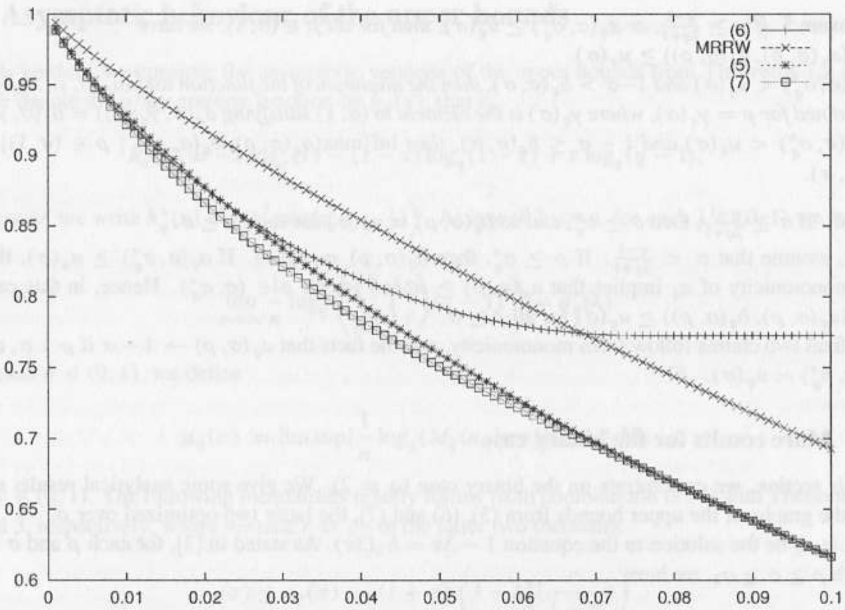


Figure 1: Upper bounds on $\mu_2(\sigma)$

Acknowledgements

I thank Vladimir Levenshtein for sending me a preprint of [3]. The comments of Henk Hollmann and Jack van Lint greatly improved the quality of this paper.

References

[1] Vladimir I. Levenshtein, "Efficient Reconstruction of Sequences from Their Subsequences and Supersequences", *Journal of Combinatorial Theory, Series A*, Vol. 93, pp. 310–322, 2001.

[2] Daniel S. Hirschberg and Mireille Reigner, "Tight bounds on the number of string subsequences", Journal of Discrete Algorithms, Vol. 1, No. 1, 2000, pp. 123–132.

[3] Vlamidir I. Levenshtein, "Bounds for Deletion-Insertion-Correcting Codes", IEEE Int. Symp. on Information Theory 2002, Lausanne, Switzerland.

[4] R.J. McEliece, E.R. Rodemich, H.C. Rumsey, Jr. and L.R.Welch, "New upper bounds on the rate of a code via the Delsarte-MacWilliams inequalities", *IEEE Trans. Inform. Th.*, Vol. 23, 1977, pp. 157–166.

On STS(21) of Wilson type

Svetlana Topalova *, Rossen Zlatarsky

Institute of Mathematics and Informatics, BAS, Bulgaria,
svetlana@moi.math.bas.bg, rossen@moi.math.bas.bg

Abstract

Steiner triple systems of Wilson type and with automorphisms of order 3 with 3 fixed points and 4 fixed blocks are constructed and classified with respect to automorphism group order and resolvability. The results lead to an improvement of Wilson's lower bound on the number of nonisomorphic STS(21).

1 Introduction

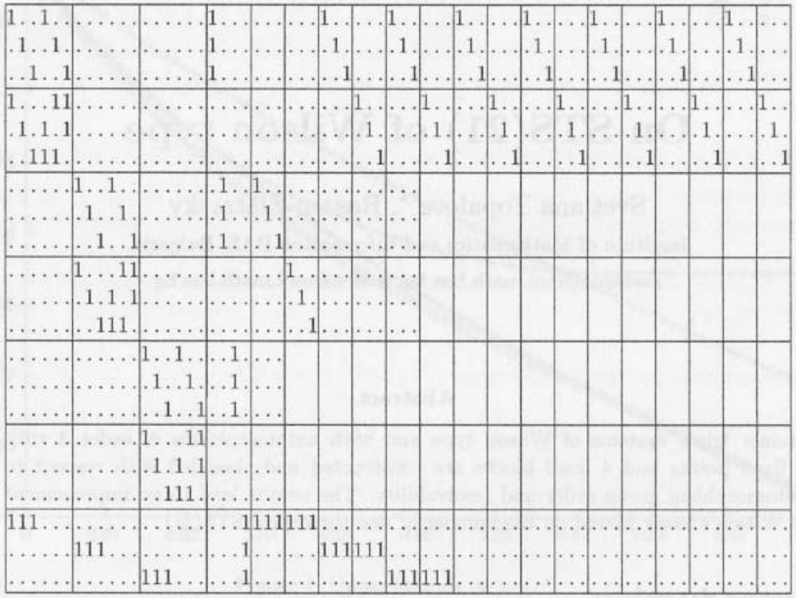
A Steiner triple system of order v (denoted STS(v)) is a 2-(v,3,1) design (see for instance [1]). Its resolutions (if such exist) are called Kirkman triple systems of order v (KTS(v)). Each KTS(v) corresponds to an optimal equidistant $(\frac{v-1}{2},\ v,\ \frac{v-1}{2}-1)$code over Z_7. A 2-(v,k,λ) design is *doubly-resolvable* if it has two distinct resolutions such that each pair of parallel classes, one of the first, and the other of the second resolution, have at most one common block.

Wilson showed in 1974 that the nonisomorphic 2-(21,3,1) designs are at least 2160980 [8]. Classifications of several other smaller classes of STS(21) have been done by several authors, i.e. Mathon, Phelps and Rosa [3], [4], Tonchev [5], [6], Kapralov and Topalova [2], Topalova and Zlatarski [7]. The aim of such classifications was to examine the properties of designs with nontrivial groups, and in particular the resolvable ones.

The classification of Wilson type designs is interesting from two points of view – on the one hand it is still not known if a doubly-resolvable $KTS(21)$ exists or not, on the other hand, knowing the number of STS(21) with certain automorphisms improves Wilson's lower bound on the number of the nonisomorphic STS(21).

*This work was partially supported by the Bulgarian National Science Fund under Contract No I-803/1998.

Figure 1: The start matrix



2 Construction and Results

Wilson type [8] STS(21) have three subsystems STS(7). We suppose an automorphism α of order 3 with three fixed points and four fixed blocks, and acting on the points as

(1,2,3)(4,5,6)...(16,17,18)(19)(20)(21),

and on the blocks as

(1,2,3)...(19,20,21)(22)(23)(24)(25)(26,27,28)...(68,69,70).

To construct the designs we start from the matrix in Fig.1. where for the sake of better readability dots stand instead of zeros. The blank rectangles are what we fill in, in all acceptable ways. The two authors used different approaches to generate the designs. The first one constructed ten possible tactical configurations, and then extended them to designs, while the second one just used the specifics of the Wilson picture we start from. We obtained the same result - 485 nonisomorphic designs.

A classification of the nonisomorphic designs with respect to the automorphism group order is presented in Table 1. A classification of the resolvable designs is presented in Table 2, where *Des* is the number of designs with *Res* nonisomorphic resolutions and order of the automorphism group *Aut*. The row denoted *KTS* contains the whole number of nonisomorphic *KTS*(21) due to these *Des* designs. None of the designs is doubly resolvable.

Table 1: Order of the automorphism group of Wilson type designs with an automorphism of order 3 with 3 fixed points and 4 fixed blocks

Aut	3	6	9	12	18	21	24	42	48	72	126	144	294	882	1008	All
Des	317	126	3	1	7	2	11	6	2	5	1	1	1	1	1	984

Table 2: Kirkman Triple Systems of order 21 with an automorphism of order 3 with 3 fixed points and 4 fixed blocks

Aut	3	3	3	3	3	3	6	6	6	6	18	18	21	24	24	42	42	42	72	126	294	882	1008	All
Res	1	2	3	5	6	9	1	2	8	10	1	9	1	1	2	1	2	3	11	4	2	4	18	
Des	23	7	3	1	1	1	3	4	1	1	1	1	1	1	1	2	1	1	1	1	1	1	1	58
KTS	23	14	9	5	6	9	3	8	8	10	1	9	1	1	2	2	2	3	11	4	2	4	18	137

Wilson finds out [8] that the whole number of designs resulting from his construction (including the isomorphic ones) should be $30^3 7!6!16942080$. A design with no automorphisms has $3!(7!)^3$ isomorphic ones among them, so that there are at least

$$\frac{30^3 7!6!16942080}{3!(7!)^3} = 2160980$$

nonisomorphic STS(21).

Knowing something about the number of Wilson type STS(21) with nontrivial automorphisms, we can use the more precise formula

$$\sum_{aut} \frac{N_{aut} 3!(7!)^3}{aut} = 30^3 7!6!16942080.$$

Here N_{aut} denotes the number of designs with a full automorphism group of order *aut*. The results from the classifications of Wilson type designs (this work and [7]) show that there are at least 1448 nonisomorphic ones with automorphism groups as shown in Table 3. Using this data and the above formula we can calculate that there are at least 2 161 908 STS(21) of Wilson type.

References

[1] Colbourn Ch., Mathon R., Steiner systems, *The CRC Handbook of Combinatorial Designs*, CRC Press, New York, 1996, 66-75.

On the Automorphism Group of Projective Planes of Prime Order

Svetlana Topalova

Institute of Mathematics and Informatics

Bulgarian Academy of Sciences

e-mail: svetlana@moi.math.bas.bg *

Abstract

It is proved that up to isomorphism there is only one projective plane of a prime order p which possesses an automorphism of order p fixing one point.

1 Introduction

There is an old conjecture that projective planes of prime orders are unique. One such plane is known for each prime p, namely the Desarguesean plane which has a very rich group of automorphisms [2]. It was proved in [3] that if a projective plane of prime order has a doubly transitive group of automorphisms, it is Desarguesean. It is obvious that the existence of certain automorphisms determines the Desarguesean nature of the plane. It is proved in this paper that the existence of a smaller subgroup of automorphisms of order p also leads to uniqueness of the plane.

A projective plane of order p corresponds to a 2-$(p^2+p+1, p+1, 1)$ design. For the basic concepts and notations concerning the theory of combinatorial designs refer, for instance, to [1] or [4].

2 A useful presentation of a projective plane

Without loss of generality, we can consider an incidence matrix of any projective plane of the form:

Figure 1.

$$\begin{array}{cccccccccccccc}
1 & 1 & 1 & 1 & \dots & 1 & 1 & 1 & o & o & o & \dots & o & o & o \\
1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & l & o & o & \dots & o & o & o \\
1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & o & l & o & \dots & o & o & o \\
1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & o & o & l & \dots & o & o & o \\
 & & & & \dots & & & & & & & \dots & & & \\
 & & & & \dots & & & & & & & \dots & & & \\
1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & o & o & o & \dots & l & o & o \\
1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & o & o & o & \dots & o & l & o \\
1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & o & o & o & \dots & o & o & l \\
O & L & O & O & \dots & O & O & O & a_{1,1} & a_{1,2} & a_{1,3} & \dots & a_{1,p-2} & a_{1,p-1} & a_{1,p} \\
O & O & L & O & \dots & O & O & O & a_{2,1} & a_{2,2} & a_{2,3} & \dots & a_{2,p-2} & a_{2,p-1} & a_{2,p} \\
O & O & O & L & \dots & O & O & O & a_{3,1} & a_{3,2} & a_{3,3} & \dots & a_{3,p-2} & a_{3,p-1} & a_{3,p} \\
 & & & & \dots & & & & & & & \dots & & & \\
 & & & & \dots & & & & & & & \dots & & & \\
O & O & O & O & \dots & L & O & O & a_{p-2,1} & a_{p-2,2} & a_{p-2,3} & \dots & a_{p-2,p-2} & a_{p-2,p-1} & a_{p-2,p} \\
O & O & O & O & \dots & O & L & O & a_{p-1,1} & a_{p-1,2} & a_{p-1,3} & \dots & a_{p-1,p-2} & a_{p-1,p-1} & a_{p-1,p} \\
O & O & O & O & \dots & O & O & L & a_{p,1} & a_{p,2} & a_{p,3} & \dots & a_{p,p-2} & a_{p,p-1} & a_{p,p}
\end{array}$$

where $a_{i,j}, \quad i,j = 1,2,...p$ are $p \times p$ matrices which have exactly one 1 and $p-1$ zeros in each row and column, l is the all-one vector of dimension p, $L = l^t$, o is the all-zero vector of dimension p, and $O = o^t$.

Let us denote by A the submatrix

$$A = \begin{pmatrix}
a_{1,1} & a_{1,2} & a_{1,3} & \dots & a_{1,p-2} & a_{1,p-1} & a_{1,p} \\
a_{2,1} & a_{2,2} & a_{2,3} & \dots & a_{2,p-2} & a_{2,p-1} & a_{2,p} \\
 & & & \dots & & & \\
 & & & \dots & & & \\
a_{p-2,1} & a_{p-2,2} & a_{p-2,3} & \dots & a_{p-2,p-2} & a_{p-2,p-1} & a_{p-2,p} \\
a_{p-1,1} & a_{p-1,2} & a_{p-1,3} & \dots & a_{p-1,p-2} & a_{p-1,p-1} & a_{p-1,p} \\
a_{p,1} & a_{p,2} & a_{p,3} & \dots & a_{p,p-2} & a_{p,p-1} & a_{p,p}
\end{pmatrix}$$

Without loss of generality we can assume that $a_{1,i} = a_{i,1} = I, i = 1,2,...,p$, where I is the identity matrix of order p.

3 Construction of a projective plane of prime order p with an automorphism of order p with one fixed point

Consider the incidence matrix of a projective plane as in Fig.1. Assume that there is an automorphism α of order p fixing point 0 and block 0 and transforming via a nontrivial

*This work was partially supported by the Bulgarian National Science Fund under Contract No I-803/1998.

permutation φ_n of order p the points within each group $np+1, np+2, ..., np+p$, $n = 0, 1, 2, 3, ..., p$.

Let us denote by b_n the set of blocks $np+1, np+2, ..., np+p$, $n = 1, 2, 3, ..., p$. As points $1, 2, ...p$ are transformed into one another, the sets b_n should also be transformed into one another. On the other hand points $p+1, p+2, ...2p$ are transformed into one another, and therefore the blocks within each set b_n $(n = 0, 1, 2, 3, ..., p-1)$ should be transformed into one another too. Thus we can consider α as the product of two automorphisms α_1 fixing points $0, 1, 2, ..., p$ and α_2 fixing points $0, p+1, p+2, ..., 2p$. Both α_1 and α_2 fix blocks $0, 1, ..., p$.

We can assume that α_1 acts on the points and blocks as

$$(0)(1)(2)...(p)(p+1, p+2, ..., 2p)(2p+1, 2p+2, ..., 3p)...(p^2+1, p^2+2, ..., p^2+p).$$

That is why all the elements of A are circulant matrices. Without loss of generality we can set $a_{1,i} = a_{i,1} = I, a_{2,i} = a_{i,2} = I_i, i = 1, 2, ..., p$, where I_i is the identity matrix of order p shifted $i-1$ times to the right.

We can assume that the automorphism α_2 transforms the first points as (0)(1,2, ..., p) and thus it should shift each row of circulants in such a way that each column of circulants is transformed into the next one. As we already know the first two columns of A, we know the permutations, i.e. α_2 acts on the points as

$$(0)(1, 2, ..., p)(p+1)(p+2)...(2p)(m_1^1, m_2^1, ..., m_p^1)...(m_1^{p-1}, m_2^{p-1}, ..., m_p^{p-1}),$$

where $m_i^j = (j+1)p+1+s_i^j$, $s_1^j = 0$, $s_i^j = s_{i-1}^j + j \ (mod\, p)$, $i = 1, 2, ..., p, j = 1, 2, ..., p-1$. So we can construct all the columns of A in a unique way.

Theorem 3.1 *There is only one projective plane of order p possessing an automorphism of order p with one fixed point.*

Proof. Follows from the construction above. As an illustration the matrix obtained for $p = 11$ is presented below. There the integer i denotes a circulant matrix of order 11 with first row containing 1 in the i-th position, $i = 1, 2, ..., 11$.

$$A = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 \\ 1 & 3 & 5 & 7 & 9 & 11 & 2 & 4 & 6 & 8 & 10 \\ 1 & 4 & 7 & 10 & 2 & 5 & 8 & 11 & 3 & 6 & 9 \\ 1 & 5 & 9 & 2 & 6 & 10 & 3 & 7 & 11 & 4 & 8 \\ 1 & 6 & 11 & 5 & 10 & 4 & 9 & 3 & 8 & 2 & 7 \\ 1 & 7 & 2 & 8 & 3 & 9 & 4 & 10 & 5 & 11 & 6 \\ 1 & 8 & 4 & 11 & 7 & 3 & 10 & 6 & 2 & 9 & 5 \\ 1 & 9 & 6 & 3 & 11 & 8 & 5 & 2 & 10 & 7 & 4 \\ 1 & 10 & 8 & 6 & 4 & 2 & 11 & 9 & 7 & 5 & 3 \\ 1 & 11 & 10 & 9 & 8 & 7 & 6 & 5 & 4 & 3 & 2 \end{pmatrix}$$

The author wants to thank Ivan Landjev for some very useful remarks and advices.

References

[1] Beth Th., Jungnickel D., Lenz H., *Design Theory*, Cambridge University Press, 1993.

[2] Mathon R., Rosa A., 2-(v,k,λ) designs of small order, *The CRC Handbook of Combinatorial Designs*, Boca Raton, FL., CRC Press, 1996, 3-41.

[3] Ostrom T.G., Wagner A., *Math.Zeitschr.*, 1959, 186-199.

[4] Tonchev V.D., *Combinatorial configurations*, Longman Scientific and Technical, New York, 1988.

Computing the minimum distance of linear codes

Alfred Wassermann
Department of Mathematics
University of Bayreuth
Germany

May 31, 2002

Abstract

We present algorithms based on lattice point enumeration for the efficient computation of the weight distribution and the minimum distance of linear binary or ternary codes.

1 Introduction

Throughout this article let C be a linear code of length n and dimension k over the finite field $\mathbb{F}$, where $\mathbb{F}$ is equal to GF(2) or GF(3), with minimum distance d. Let $H = H_C$ be a $(n-k) \times n$ *parity check matrix* of the code C over $\mathbb{F}$, i. e. $C = \{y \in \mathbb{F}^n \mid H \cdot y = 0\}$. The *Hamming weight* of a codeword $x \in C$ is denoted by $w(x)$. It is well known that in general the computation of the minimum distance of a linear code is a difficult task: in [1] it is shown that the problem

> *Weight Distribution*: Given a $(n-k) \times n$ matrix H over $\mathbb{F}$ and an integer $s > 0$, is there a vector $x \in \mathbb{F}^n$ with $w(x) = s$ such that $x \cdot H^\top = 0$?

is NP-complete. In [11] the problem

> *Minimum Distance*: Given a $(n-k) \times n$ matrix H over $\mathbb{F}$ and an integer $s > 0$. Is there a nonzero vector $x \in \mathbb{F}^n$ with $w(x) < s$ such that $x \cdot H^\top = 0$?

is proved to be NP-complete.

There are several algorithms known which address the weight distribution problem or the minimum distance problem. A deterministic algorithm based on an idea of Brouwer is in [2, p. 31]. Further, there are several probabilistic algorithm, see for example Canteaut and Chabaud [4] and Leon [8]. In [11] Vardy mentions the connection of the minimum distance problem to the problem of finding the nearest vector in a lattice.

Here, we compute the weight distribution of linear codes over GF(2) by transforming the problem into a shortest vector problem in a lattice. Then, the shortest vector problem is solved by lattice basis reduction followed by lattice point enumeration.

Lattice point enumeration was already successfully applied by the author in the construction of combinatorial designs [3, 12] and the solving of the market split problem [13].

Of course, any algorithm for the weight distribution problem can also be used to compute the minimum distance of a binary linear code. However, a modification of the lattice point enumeration algorithm computes the minimum distance of a linear code directly. Moreover, this algorithm works for binary and ternary linear codes. A preliminary version of the algorithm was previously published in German [2].

2 Lattice point enumeration

Let $\mathbb{R}^n$ be the n-dimensional $\mathbb{R}$-vector space with the ordinary inner product $\langle .,. \rangle$. A discrete, additive subgroup $L \subset \mathbb{R}^n$ is called a *lattice*. Every lattice L is generated by a set of linearly independent vectors $b_1, \ldots, b_m \in L$, the *basis* of $L = L(b_1, \ldots, b_m) = \{x_1 b_1 + \cdots + x_m b_m \mid x_1, \ldots, x_m \in \mathbb{Z}\}$. The celebrated lattice basis reduction algorithm [7] (called LLL algorithm) computes in polynomial time a new basis of the lattice L which consists of short vectors. Given a reduced basis of a lattice there are algorithms to enumerate all lattice vectors (alos called points) with norm below a given bound, see [5, 6, 9, 10, 12, 13]. In [13] there is a detailed description how to solve linear Diophantine systems with lattice point enumeration.

3 Computing the weight distribution

Let s be an integer with $s > 0$. Let C be defined by a generator matrix $G \subset \mathrm{GF}(2)^{n \times k}$. We will formulate the weight distribution problem, i. e. "Does a codeword $x \in C$ exist such that $w(x) = s$?", as a shortest lattice vector problem.

Any solution of the weight distribution problem is also a solution of the following linear Diophantine system:

$$\left(\begin{array}{ccc|ccc|ccc} & & & 1 & & 0 & 2 & & 0 \\ & G & & & \ddots & & & \ddots & \\ & & & 0 & & 1 & 0 & & 2 \\ \hline 0 & \cdots & 0 & 1 & \cdots & 1 & 0 & \cdots & 0 \end{array}\right) \cdot x = \left(\begin{array}{c} 1 \\ \vdots \\ 1 \\ n-s \end{array}\right) \tag{1}$$

$$\text{and } x_i \in \{0,1\} \text{ for } 1 \le i \le n+k\,, \quad x_i \in \mathbb{Z} \text{ for } n+k < i \le 2n+k\,.$$

The system (1) has $n+1$ rows and $2n+k$ unknowns. It is easy to see that solutions of (1) are codewords of C with weight equal to s: The left part of the matrix (1) are the columns of the generator matrix. The columns of the right part are needed because we have to work over GF(2), i. e. we have to compute all integer linear combinations of G modulo 2. The middle part of the matrix (1) corresponds to slack variables. A column of this part is multiplied by 1 if in the corresponding row the codeword contains a 0. It is multiplied by 0 if in the corresponding row the codeword contains a 1. With the last row of the system (1) it is ensured that exactly $n-s$ slack variables attain the value 1. That is, exactly s entries in the codeword contain the entry 1.

The size of the system (1) can be slightly reduced if we use the generator matrix in standard form. With Gaussian elimination over GF(2) we can always bring the matrix G into the form $G = \left(\frac{I_k}{A}\right)$ where I_k is the matrix of unity in $\mathrm{GF}(2)^{k \times k}$ and A is a $(n-k) \times k$ matrix over GF(2). Then we can avoid some columns in the system (1), because it is sufficient to solve

$$\left(\begin{array}{ccc|ccc|ccc|ccc} & & & 1 & & 0 & & & & & & \\ & I_k & & & \ddots & & & 0 & & & 0 & \\ & & & 0 & & 1 & & & & & & \\ \hline & & & & & & 1 & & 0 & 2 & & 0 \\ & A & & & 0 & & & \ddots & & & \ddots & \\ & & & & & & 0 & & 1 & 0 & & 2 \\ \hline 0 & \cdots & 0 & 1 & \cdots & 1 & 1 & \cdots & 1 & 0 & \cdots & 0 \end{array}\right) \cdot x = \left(\begin{array}{c} 1 \\ 1 \\ \vdots \\ 1 \\ n-s \end{array}\right) . \tag{2}$$

$$\text{and } x_i \in \{0,1\} \text{ for } 1 \le i \le k+n\,, \quad x_i \in \mathbb{Z} \text{ for } k+n < i \le 2n\,.$$

The right part of the matrix in (2) contains the $(n-k) \times (n-k)$ diagonal matrix with all entries equal to 2. Together, the matrix in (2) has $k + n + (n-k) = 2n$ columns.

The systems (1) or (2) can now be solved by lattice point enumeration as described in [13].

If the linear code C is cyclic, than we can improve the efficiency of the algorithm by introducing further equations: if we write the codewords of C as binary strings, then any codeword x in the binary code C of length n with $w(x) = s$ contains a substring of length $\lceil \frac{n}{s} \rceil$ which has the form $100\ldots0$, i. e. the letter one followed by $\lceil \frac{n}{s} \rceil - 1$ zeroes. If the binary code C is known to be cyclic and x is a codeword in C then there exists a permutation σ in the group of automorphisms of C such that the codeword $\sigma(x)$ starts with the above substring. Therefore, $x_1 = 1$ and $x_i = 0$ for $1 \leq i \leq \lceil \frac{n}{s} \rceil - 1$. We can introduce these restrictions in the system (2) by removing the first column of the matrix corresponding to slack variables and by setting in the right hand side vector the coefficients at the positions $2, 3, \ldots \lceil \frac{n}{s} \rceil - 1$ equal to zero.

4 Computing the minimum distance

Let C be a binary or ternary linear code. With a variation of the lattice point enumeration algorithm in [13] we can compute the minimum distance of C.

For this we note that in the case of a binary code we have $-1 \equiv 1 \pmod 2$ and in the case of a ternary case $-1 \equiv 2 \pmod 3$. Thus, codewords of binary or ternary codes can be represented by vectors with entries in $\{0, 1, -1\}$.

Let $\mathbb{F}$ be equal to GF(2) or GF(3), and p be equal to 2 or 3, respectively. We define the lattice spanned by the columns of the $(n+k) \times (k+n)$ matrix

$$L_C = \left(\begin{array}{c|c} G & pI_n \\ \hline I_k & 0 \end{array} \right), \tag{3}$$

where G is a $n \times k$ generator matrix of the code C. Any lattice vector $v \in L_C$ with $v_i \in \{0, 1, -1\}$ for $1 \leq i \leq n$ corresponds to a codeword $v_C \in C$ and $w(v_C)$ equals the number of nonzero entries in the first n coefficients of v. Now, the minimum distance problem can now be solved by finding a nonzero lattice vector with the least number (> 0) of nonzero entries in the first n rows.

For the lattice basis reduction we multiply the upper part of (3) by a large constant N:

$$\left(\begin{array}{c|c} N \cdot G & N \cdot pI_n \\ \hline I_k & 0 \end{array} \right). \tag{4}$$

If the lattice (4) is reduced with the original LLL algorithm [7], we can choose $N > (2)^{(n+k)/2} \cdot p$ and the reduced lattice basis will have the form

$$\left(\begin{array}{c|c} 0 & N \cdot L' \\ \hline * & * \end{array} \right), \tag{5}$$

with a $n \times k$ matrix consisting of zeroes only in the upper left part of (5) and irrelevant entries in the lower k rows. Since we only want to enumerate nonzero codewords we can delete the first k columns in (5). We are not interested in the lower k rows and can remove them, too. Therefore, we have to find the lattice vectors v in the lattice $L' \in \mathbb{Z}^n$ of rank n with $\|v\|_\infty = 1$ which contain the minimal number of nonzero entries. If $w(v_C) = s$ then we also know that $\|v\|_2^2 = s$.

We can compute the minimum distance of C by setting initially the upper bound F of the lattice point enumeration algorithm to an upper bound for the minimum distance of C. This can be the weight of the shortest vector $\not\equiv 0 \pmod p$ in the reduced basis consisting of L' or some theoretical upper bound on the minimum distance of C.

Then the backtracking of the lattice point enumeration algorithm as described in [13] is started. If a lattice vector $v \in L'$ with $\|v\|_\infty = 1$ and $\|v\|_2^2 \leq F$ is found during the enumeration then it is printed and F is set to $F := \|v\|_2 - 1$ and the backtracking is continued. If it is known that the minimum distance of C is a multiple of some integer q, then we even can set $F := \|v\|_2 - q$ in this situation.

Further improvements can be attained by a variation of the lattice point enumeration in [13]. For an integer $0 < t < n$ and a vector $v \in \mathbb{R}^n$ we define $\max_t(v)$ to be the sum of the t largest entries of v in absolute values. Let F be an upper bound on the minimum distance of the code C and $b^{(1)}, b^{(2)}, \ldots, b^{(n)}$ be a basis of the lattice L'. Theorem 2 in [13] can be adapted to the minimum distance problem in the following way, where we take the notation from [13].

Theorem 1 *Let $1 \leq t \leq n$. If for fixed $u_t, u_{t+1}, \ldots, u_n \in \mathbb{Z}$ there exist coefficients $u_1, u_2, \ldots, u_{t-1} \in \mathbb{Z}$ with*

$$\|\sum_{i=1}^{n} u_i b^{(i)}\|_\infty \leq 1 \quad \textit{and} \quad \|\sum_{i=1}^{n} u_i b^{(i)}\|_2 \leq F,$$

then for all $y_t, y_{t+1}, \ldots, y_n \in \mathbb{R}$:

$$\left|\sum_{i=t}^{n} y_i \|w^{(i)}\|_2^2\right| \leq 1 \cdot \max_F(\sum_{i=t}^{n} y_i w^{(i)}). \tag{6}$$

Proof: We have $\langle w^{(l)}, w^{(i)} \rangle = \langle w^{(i)}, w^{(i)} \rangle = c_i$ for $1 \leq l < i \leq n$. If there exist $u_1, u_2, \ldots, u_n \in \mathbb{Z}$ with $\|w^{(1)}\|_\infty = \|\sum_{i=1}^n u_i b^{(i)}\|_\infty \leq 1$ and simultaneously $\|w^{(1)}\|_2 \leq \mathbb{F}$, then for an arbitrary vector $v \in \mathbb{R}^n$ $|\langle w^{(1)}, v \rangle| \leq \max_F(v)$. It follows

$$|\sum_{i=t}^{n} y_i c_i| = |\sum_{i=t}^{n} y_i \langle w^{(i)}, w^{(i)} \rangle| = |\sum_{i=t}^{n} y_i \langle w^{(1)}, w^{(i)} \rangle| = |\langle w^{(1)}, \sum_{i=t}^{n} y_i w^{(i)} \rangle| \leq \max_F(\sum_{i=j}^{n} y_i w^{(i)}).$$

□

Therefore, we can add in the enumeration algorithm of [13] the test (6). Experiments show that with Theorem 1 the minimum distance of quadratic residue codes can easily be determined for values of n at least up to 100. Also, the backtracking part of the enumeration can be done in parallel.

References

[1] E. R. Berlekamp, R. J. McEliece, and H. C. A van Tilborg. On the inherent intractability of certain coding problems. *IEEE Trans. Information Theory*, 24:384–386, 1978.

[2] A. Betten, H. Fripertinger, A. Kerber, A. Wassermann, and K-H. Zimmermann. *Codierungstheorie – Konstruktion und Anwendung linearer Codes.* Springer-Verlag, Heidelberg, 1998.

[3] A. Betten, A. Kerber, R. Laue, and A. Wassermann. Simple 8-designs with small parameters. *Designs, Codes and Cryptography*, 15:5–27, 1998.

[4] A. Canteaut and F. Chabaut. A new algorithm for finding minimum-weight words in a linear code: application to primitive narrow-sense BCH codes of length 511. Rapport de recherche 2685, INRIA, 1995.

[5] U. Fincke and M. Pohst. Improved methods for calculating vectors of short length in a lattice, including a complexity analysis. *Mathematics of Computation*, 44:463–471, 1985.

[6] R. Kannan. Minkowski's convex body theorem and integer programming. *Math. Operations Research*, 12:415–440, 1987.

[7] A. K. Lenstra, H. W. Lenstra Jr., and L. Lovász. Factoring polynomials with rational coefficients. *Math. Ann.*, 261:515–534, 1982.

[8] J. S. Leon. A probabilistic algorithm for computing minimum weights of large error-correcting codes. *IEEE Trans. Information Theory*, 34:1354–1359, 1989.

[9] H. Ritter. Breaking knapsack cryptosystems by max-norm enumeration. In *1st International Conference of the Theory and Appications of Cryptology – Pragocrypt '96, Lecture Notes in Computer Science*, pages 480–492, Prague, 1996. CTU Publishing House, Prague.

[10] H. Ritter. *Aufzählung von kurzen Gittervektoren in allgemeiner Norm.* PhD thesis, Universität Frankfurt, 1997.

[11] A. Vardy. The intractablitiy of computing the minimum distance of a code. *IEEE Trans. Information Theory*, 43:1757–1773, 1997.

[12] A. Wassermann. Finding simple t-designs with enumeration techniques. *J. Combinatorial Designs*, 6:79–90, 1998.

[13] A. Wassermann. Attacking the market split problem with lattice point enumeration. *J. Combinatorial Optimization*, 6:5–16, 2002.

The Separability Function of the Standard cyclic N-ary Gray code

A. J. VAN ZANTEN,
Delft University of Technology, Faculty of Information Technology and Systems
Department of Mathematics,
P.O. Box 5031, 2600 GA Delft, TheNetherlands

I NENGAN SUPARTA[*]
Delft University of Technology, Faculty of Information Technology and Systems
Department of Mathematics,
P.O. Box 5031, 2600 GA Delft, TheNetherlands

Abstract

A sharp lower bound is derived for the cyclic list distance between two codewords, having Hamming distance m, in the standard N-ary Gray code of length n, for $1 \leq m \leq n$ and for even values of N. The bound generalizes the binary separability function.

1 Introduction

A binary Gray code of length n is an ordered sequence (*list*) of all 2^n n-bits strings (*codewords*) such that successive codewords differ in exactly one bit position. The best known example of such an ordered code is the *binary reflected Gray code* (cf. e.g.[4], [6] and also Section 2), sometimes called *standard binary Gray code*. A question of theoretical as well as of practical relevance is the following. If two codewords in a Gray code, or in any ordered code, differ in m positions, how far are they separated from each other in the list of codewords? The larger this list distance of the code, the smaller the number of bit errors will be when transmitting codewords by means of analog signals (cf.[6]). Stated more precisely, when we index the codewords in the list from 0 until $2^n - 1$, and if two codewords $\mathbf{g}_i$ and $\mathbf{g}_j$ have Hamming distance $d_H(\mathbf{g}_i, \mathbf{g}_j) = m$, can we find a bounding function b such that the list distance $d_L(\mathbf{g}_i, \mathbf{g}_j) \geq b(m)$, for $1 \leq m \leq n$? Of course, the most interesting bounding function is a function giving sharp lower bounds for all values of m, i.e. such that for every m-value there exists at least one pair of codewords with list distance $b(m)$. The question of finding this uniquely determined function is called the *separability problem* (cf. [5], [6]). We shall use the term *separability function* for a function b - occasionally denoted as $b(m)$ - yielding sharp lower bounds for $1 \leq m \leq n$. In [5] Yuen solves the separability problem for the binary standard Gray code. The separability function in this case appears to be $\lceil \frac{2^m}{3} \rceil$. The derivation of this expression is accomplished by making use of the *index system* of the standard Gray code, i.e. the relationship between a codeword $\mathbf{g}_i$ and its index i, $0 \leq i \leq 2n - 1$ (cf. e.g.[4]). Along similar lines, Cavior in [1] derives a sharp upper bound for the list distance in this code, being $2^n - \lceil \frac{2^m}{3} \rceil$, $1 \leq m \leq n$. In both papers the list of codewords is interpreted as a linear (non-cyclic) list, which implies that $d_L(\mathbf{g}_i, \mathbf{g}_j)$ is defined as $|i - j|$. Now, it is well known that the standard Gray code is a *cyclic* Gray code, i.e. also the last codeword differs form the first one in precisely one bit position. Therefore it is natural to introduce the *cyclic list distance* defined as

$$D_L(\mathbf{g}_i, \mathbf{g}_j) = min\ \{|i - j|,\ 2^n - |i - j|\}, \tag{1}$$

(cf. also [3]). With respect to this notation the results of Yuen and Cavior can be combined in the following implication

$$d_H(\mathbf{g}_i, \mathbf{g}_j) = m \rightarrow D_L(\mathbf{g}_i, \mathbf{g}_j) \geq \left\lceil \frac{2^m}{3} \right\rceil. \tag{2}$$

We call this implication the *separability property* of the standard binary Gray code.

In the next we shall derive a property generalizing (2), which holds for the standard N-ary Gray code when N is even. Although an index system for this code is known (cf.[2]), it will appear that such a system is not needed to prove the result. Throughout the text, the terms list and Gray code (which is represented by the list) are interchangeable. The columns of this list are numbered from right to left by $1, 2, \ldots, n$.

[*] On leave of absence from Dept. of Mathematics of IKIP Negeri Singaraja, Bali- Indonesia

2 Standard N-ary Gray code

An N-ary Gray code, $N > 0$, of length N is an ordered list of all N^n codewords of length n over the set of integers $S = \{0, 1, \ldots, N - 1\}$, such that each codeword differs from the previous one in exactly one position. If also the last codeword differs from the first word of the list in one position, one speaks of a *cyclic N-ary Gray code*. More in particular, one can require the *minimal change condition*, i.e. that if $\mathbf{g}_i$ and $\mathbf{g}_{i+1}$ differ in in the j^{th} position, one either has $\mathbf{g}_{i+1,j} = \mathbf{g}_{i,j} + 1$ or $\mathbf{g}_{i+1,j} = \mathbf{g}_{i,j} - 1\ mod\ N$, for all values of i with $0 \leq i \leq N^n$, where we identify the codewords with index N^n and with index 0.

A well known minimal change N-ary Gray code of length n, $n \in \mathbb{Z}^+$, is the code $G(n, N)$ which is recursively defined as

$$G(n, N) = \begin{pmatrix} 0 & G(n-1, N) \\ 1 & G(n-1, N)^R \\ 2 & G(n-1, N) \\ \vdots & \vdots \\ N-1 & G(n-1, N)^* \end{pmatrix}, \quad G(1, N) = \begin{pmatrix} 0 \\ 1 \\ 2 \\ \vdots \\ N-1 \end{pmatrix}, \tag{3}$$

where $G(n-1, N)^R$ denotes the list $G(n-1, N)$ in reversed order, and where $*$ stands for R, only when N is even whereas $*$ can be omitted when N is odd. This code is called the *standard N-ary Gray code*. It is obvious that $G(n, N)$ is a cyclic Gray code only if N is even. It will also be obvious that $G(n, N)$ generalizes the standard binary Gray code. From now on we only consider even values of N.

3 Equivalence and contraction of ordered codes

Let $V_{n,N}$ denote the set of all cyclic minimal-change N-ary Gray codes of length n. We define the following transformations which map $V_{n,N}$ onto itself:

(i) if p is a permutation of the integers $1, 2, \ldots, n$, then pG is the code obtained by permuting the columns of $G \in V_{n,N}$ according to p;

(ii) if c is the cyclic permutation $(0, 1, 2, \ldots, N-1)$, then c_iG is the code obtained by permuting the integers in the i^{th} column of G according to c, $1 \leq i \leq n$;

(iii) if b is the permutation $(0,\ N-1)(1,\ N-2)\ \dots\ (\frac{N}{2}-1, \frac{N}{2})$, then b_iG is the code obtained by permuting the integers in the i^{th} column of G according to b, $1 \le i \le n$.

These transformations define a group of order $n!(2N)^n$. Codes which can be transformed into each other by applying one or more of the transformations (i)-(iii) are called *equivalent codes*.

Let G be some code in $V_{n,N}$. Take two k-strings $\mathbf{a} := a_1a_2\dots a_k \in S^k$, and $\mathbf{i} := i_1i_2\dots i_k$, with $1 \le i_1 < i_2 < \dots < i_k \le n$, for some k-value, $1 \le k \le n$. We call $\mathbf{a}$ a *bit pattern* and $\mathbf{i}$ a *position vector*. We now consider the sublist of G consisting of all codewords which have a_j on position i_j for $1 \le j \le k$. Omitting the common bit pattern $\mathbf{a}$ from these words yields an ordered code with word length $n-k$. We shall call this code the *contraction* of G with respect to the pair $(\mathbf{a}, \mathbf{i})$, and we denote this code by $G(n, N;\ \mathbf{a}, \mathbf{i})$. With respect to these notions we have the following properties.

Proposition 3.1. *Equivalent codes satisfy the same separability property.*

Proposition 3.2. *Let $G(n, N)$ be the standard N-ary Gray code and let N be even. Then for any pair $(\mathbf{a}, \mathbf{i})$ the contraction $G(n, N;\ \mathbf{a}, \mathbf{i})$ is a cyclic Gray code equivalent to the standard Gray code $G(n-k, N)$.*

The proofs of these propositions are elementary and are omitted here.

4 Separability function for $G(n, N)$

We are ready now to prove our main result.

Theorem 4.1. *Let $G(n, N)$ be the standard N-ary Gray code of length n, and let N be even. If the Hamming distance between two codewords $\mathbf{g}$ and $\mathbf{h}$ satisfies $d_H(\mathbf{g}, \mathbf{h}) = m$, then the list distance between $\mathbf{g}$ and $\mathbf{h}$ satisfies $D_L(\mathbf{g}, \mathbf{h}) \ge \left\lceil \frac{N^m}{N^2-1} \right\rceil$. Moreover, this lower bound is sharp for all m-values with $1 \le m \le n$.*

Proof. We prove the Theorem in two steps.
A. First we take $m = n$. For $n = 1$ and $n = 2$ the statement is trivial. Assume the statement is true for all values less than $n > 2$. Let $\mathbf{g}$ and $\mathbf{h}$ be two codewords with $d_H(\mathbf{g}, \mathbf{h}) = n$, and let furthermore the list distance $D_L(\mathbf{g}, \mathbf{h})$ be as small as possible. If we write $\mathbf{g} = g_ng_{n-1}\mathbf{v}$ and $\mathbf{h} = h_nh_{n-1}\mathbf{w}$, it follows that $d_H(\mathbf{v}, \mathbf{w}) = n-2$. Hence, considered as codewords of $G(n-2, N)$ the list distance $D_L(\mathbf{v}, \mathbf{w})$ of $\mathbf{v}$ and $\mathbf{w}$ is at least $\left\lceil \frac{N^{n-2}}{N^2-1} \right\rceil$, by induction assumption. Now, in particular, since $D_L(\mathbf{g}, \mathbf{h})$ was taken as small as possible, it follows either that $\mathbf{g} \in g_ng_{n-1}G(n-2, N)$, $\mathbf{h} \in h_nh_{n-1}G(n-2, N)$ or that $\mathbf{g} \in g_ng_{n-1}G(n-2, N)^R$, $\mathbf{h} \in h_nh_{n-1}G(n-2, N)^R$, whereas $D_L(\mathbf{v}, \mathbf{w})$ is equal to $\left\lceil \frac{N^{n-2}}{N^2-1} \right\rceil$. Therefore, we may conclude that

$$D_L(\mathbf{g}, \mathbf{h}) = N^{n-2} + \left\lceil \frac{N^{n-2}}{N^2-1} \right\rceil = \left\lceil \frac{N^n}{N^2-1} \right\rceil.$$

By the principle of mathematical induction the Theorem has been proved now for the case $m = n$.

B. If $m < n$, then $\mathbf{g}$ and $\mathbf{h}$ are equal in $k := n-m$ positions, indicated by some position vector $\mathbf{i} := (i_1, i_2, \dots, i_k)$. The corresponding values of the coordinates will be given by $\mathbf{a} = (a_1, a_2, \dots, a_k)$. Now, we consider the contraction $G(n, N;\ \mathbf{a}, \mathbf{i})$. Let $\mathbf{v}$ and $\mathbf{w}$ be the codewords in this contraction which correspond to $\mathbf{g}$ and $\mathbf{h}$ respectively. So, we have $d_H(\mathbf{v}, \mathbf{w}) = m$. By Proposition 3.1 $G(n, N;\ \mathbf{a}, \mathbf{i})$ is equivalent to $G(m, N)$. By Proposition 3.2 and part A of this proof, we have that $D_L(\mathbf{v}, \mathbf{w}) \ge \left\lceil \frac{N^m}{N^2-1} \right\rceil$ in this contracted code. Hence, we have a fortiori the same inequality

for $D_L(\mathbf{g}, \mathbf{h})$, since in $G(n, N)$ the codewords corresponding to codewords of $G(n, N;\ \mathbf{a}, \mathbf{i})$ will, in general, be interlaced by codewords which have no counterpart in $G(n, N;\ \mathbf{a}, \mathbf{i})$. Finally, by selecting codewords $\mathbf{g}$ and $\mathbf{h}$ which differ in positions $1, 2, \dots, m$, we obtain an example showing that the lower bound is sharp for all relevant m-values. □

Corollary 4.1. *(Yuen, Cavior) The separability function of the standard binary Gray code is equal to $\left\lceil \frac{2^m}{3} \right\rceil$.*

References

[1] S. R. Cavior, "A upper bound associated with errors in Gray code", IEEE Trans. Inform. Theory, vol. IT-21, p.596, 1975.

[2] I. Floris, "Reflected Number System", IEEE Trans. Electron Comput., vol. EC-5, pp.79-82, 1956.

[3] F. P. Preparata and J.Nievergelt, "Difference-preserving codes", IEEE Trans. Inform. Theory, vol. IT-20, pp.643-649, 1974.

[4] E. M. Reingold, J.Nievergelt and N. Deo, "Combinatorial Algorithms: Theory and Practice", Englewood Cliffs, NL: Prentice-Hall, 1977.

[5] C. K. Yuen, "The separability of Gray code", IEEE Trans. Inform. Theory, vol. IT-20, p.668, 1974.

[6] A. J. van Zanten, "Index system and separability of constant weight Gray codes", IEEE Trans. Inform. Theory, vol. 37, No. 4, pp 1229-1233, 1991.

New Constructions of Combinatorial Designs and Related Codes

Eric Zhi Chen

School of Engineering, Kristianstad University, 291 88 Kristianstad, Sweden
Eric.Chen@tec.hkr.se

Abstract— New constructions of Steiner systems, majority logic decodable codes, and constant weight codes are presented. Comparisons with known results as well as examples are also given.

1. Introduction

Error control coding is one of the efficient methods to combat errors in data transmission and data storage systems[1,2]. It is well known that there exists close relationship between coding theory and combinatorial theory[3]. The results in combinatorial theory have been used to construct good error control codes, and the results in coding theory have also been used to give new combinatorial designs. For example, the majority logic decodable difference set cyclic codes, Euclidean and Projective geometry codes are constructed from the block designs on the geometries.

In this paper, some new constructions of Steiner systems are presented, and new majority logic decodable codes and constant weight codes based on Latin squares and block designs are constructed. Comparisons with known results are made and some examples are also given.

2. Combinatorial Designs

Let X be a set of *v* objects (or points, varieties). A *t*-design is a collection of distinct *k*-subsets called blocks of X such that any *t*-subset of X is contained in exactly λ blocks. Such a *t*-design is written as a *t-(v, k, λ)* design. A Steiner system is a *t*-design with λ= 1, and a *t-(v, k, 1)* design is usually called an S(*t, k, v*).

An incomplete block design is an arrangement of *v* distinct objects into *b* blocks such that each block contains exactly *k* different objects, each object occurs in exactly *r* different blocks. Furthermore, if each pair of distinct objects occurs together in exactly λ blocks, the block design is called balanced incomplete block design, or BIBD. Such a block design is written as BIBD(*v, b, r, k, λ*). There are two elementary relations on the five parameters

$$vr = bk \qquad (2.1)$$

$$r(k-1) = \lambda (v-1) \qquad (2.2)$$

A design is said to be resolvable if *b* blocks can be partitioned into *r* sets of *m = b/r* blocks such that each set contains every object exactly once. For a resolvable BIBD,

$$v \geq b - r + 1 \qquad (2.3)$$

If equation (2.3) holds with equality, the design is said to be affine resolvable, and it is written as ARBIBD(*v, b, r, k, λ*).

A BIBD is called symmetric if *v = b* (or, equivalently, *r = k*). It can be written as SBIBD(*v, k, λ*), or (*v, k, λ*)-design. A BIBD(*v, b, r, k, λ*) with λ= 1 is a Steiner system with *t* =2, and is usually called an S(*2, k, v*). A block design is called partially balanced incomplete block design (PBIBD) if pairs of distinct objects may occur together in different numbers of blocks.

A block design can be described by its incidence matrix $A = (a_{ij})$, i =1, 2, …, v, j = 1, 2, …, b. Let $x_1, x_2, \ldots, x_v$ be objects, and $B_1, B_2, \ldots, B_b$ blocks. Then $a_{ij}=1$ if $x_i \in B_j$, and 0 otherwise. So each row of the incidence matrix has *r* 1's, each column has *k* 1's, and each pair of distinct rows has λ common 1's.

3. Majority Logic Decodable Codes from Latin Squares

An *m×m* array (matrix) is a Latin square of order *m*, if each of the numbers *1, 2, …, m* occurs exactly once in each row and column. Two Latin squares A = (a_{ij}) and B = (b_{ij}) of order *m* are orthogonal if for every pair (a, b) ∈ {(1,1), (1,2), …, (m, m)} there exists unique indices i and j such that (a, b) = (a_{ij}, b_{ij}). A set of Latin squares is mutually orthogonal if any two of them are orthogonal[4,5].

Let *N(m)* denote the maximum number of mutually orthogonal Latin squares (MOLS) of order *m*. It is known that *N(m)* $\leq m - 1$, and for prime powers, this upper bound is attained.

Given a set of *t* MOLS of order *m*. Let V be the set of objects, V = {1, 2, …, m×m}. A PBIBD, with parameters $v = m^2$, $b = (t+2)m$, $r = t+2$, $k = m$, *and* λ_1= 0 and λ_2= 1, can be constructed from the set of t MOLS as follows:(i) Write these objects in an *m×m* matrix; (ii) Each row of the matrix forms a block; (iii) Each column of the matrix forms a block; (iv) For each Latin square L_i in the set of t MOLS, i=1, 2, …, t, superimpose it on the matrix of objects, and take the objects from the matrix having the same symbol in L_i as blocks. The following example shows the construction.

Example 3.1. m = 4 and N(4) = 3. There are 3 mutually orthogonal Latin squares of order 4:

$$\begin{array}{ccc} \begin{array}{cccc} 1&2&3&4\\ 2&1&4&3\\ 3&4&1&2\\ 4&3&2&1 \end{array} & \begin{array}{cccc} 1&2&3&4\\ 3&4&1&2\\ 4&3&2&1\\ 2&1&4&3 \end{array} & \begin{array}{cccc} 1&2&3&4\\ 4&3&2&1\\ 2&1&4&3\\ 3&4&1&2 \end{array} \\ L_1 & L_2 & L_3 \end{array}$$

The 4×4 = 16 objects are arranged in the follow matrix:

$$\begin{array}{cccc} 1&2&3&4\\ 5&6&7&8\\ 9&10&11&12\\ 13&14&15&16 \end{array}$$

So the total (3 + 2) × 4 or 20 blocks are constructed: {1,2,3,4}, {5,6,7,8}, {9,10,11,12}, {13,14, 15,16} from rows; {1,5,9,13},{2,6,10,14},{3,7,11,15},{4,8,12,16} from columns; {1,6,11,16}, {2,5,12,15},{3,8,9,14},{4,7,10,13} from L_1; {1,7,12,14}, {2,8,11,13}, {3,5,10,16},{4,6,9,15} from L_2; {1,8,10,15},{2,7,9,16},{3,6,12,13},{4,5, 11,14}from L_3. In fact, this block design is an ARBIBD(16, 20, 5, 4, 1). If *m* is a prime power, there is a set of *m - 1* MOLS of order *m* and an ARBIBD(m^2, $m^2 + m$, $m+1$, m, 1) can always be constructed as shown above. If a set of t < m - 1 MOLS is used to construct the design, then it is a PBIBD with parameters $v = m^2$, $b = (t + 2)m$, $r = t + 2$, $k = m$, $\lambda_1 = 0$ and λ_2=1.

Latin squares can be used to construct various types of codes. Hsiao, Bossen, and Chien[6] presented a construction of majority logic decodable codes based on MOLS. To construct a code of minimum distance d and with m^2 information digits, take a set of *d - 3* MOLS of order *m*. Let A be the incidence matrix for the PBIBD derived from these *d - 3* MOLS as above. Then following parity check matrix defines a majority logic decodable [m^2 + (d -1)m, m^2] code, which is:

$$H = [A^T \; I] \qquad (3.1)$$

For example, for m = 4, N(4) = 3. So majority logic decodable [24, 16, 3], [28, 16, 4], [32, 16, 5], and [36, 16, 6] codes can be constructed with a set of 0, 1, 2, 3 MOLS of order 4, respectively.

From the parity check matrix in (3.1), we know that m parity check equations from the same set do not have any common information digit. Therefore, d – 1 new information digits can be added to form an [m^2 + (d -1)m + d -1, m^2 + d-1] code, where m^2 information digits have a distance of d, and d -1 extended information digits have a distance of m +1. For example, m = 4. We have majority logic decodable [26, 18, 3], [31, 19, 4], [36, 20, 5], and [41, 21, (5,6)] codes, where 2, 3, 4, and 5 information digits have a distance of 5, respectively. When d is small, then the extended information digits have relatively large distance and thus are protected against more errors than m^2 original information digits. They can be used for important information digits.

4. New Constructions of Steiner systems and Constant Weight Codes

A constant weight code is a block code whose codewords all have the same Hamming weight. Let A(n, d, w) be the maximum number of codewords in a binary code of block length *n*, constant weight *w*, and minimum distance of at least *d*. Good constant weight codes can be constructed from combinatorial designs. The followings are two upper bounds on constant weight codes[1].

$$A(n,d,w) \le \left\lfloor \frac{n}{w} A(n-1,d,w-1) \right\rfloor \qquad (4.1)$$

Constant weight codes are closely related to combinatorial designs. For example, it is well known that

$$A(n,2\delta,w) \le \frac{n(n-1)\ldots(n-w+\delta)}{w(w-1)..\delta} \qquad (4.2)$$

and equality holds if and only if a Steiner system S(*w*-δ+*1*, *w*, *n*) exits.

Next, we discuss some new constructions of Steiner systems and constant weight codes derived from MOLS and known block designs.

Let us start with a simple example to show the idea of our constructions. Consider m = 4. With a pair of MOLS of order 4, we can construct a PBIBD as we did in Example 3.1. We know that there is no common object in any pair of blocks from the same set and there is exactly one common object in any pair of blocks from different sets. So we can add one new object to each block in the same set and get:

Rows: {1,2,3,4,**17**},{5,6,7,8, **17**},{9,10,11,12,**17**},{13,14,15,16,**17**};
Columns: {1,5,9,13,**18**},{2,6,10,14,**18**},{3,7,11,15,**18**},{4,8,12,16,**18**};
L_1: {1,6,11,16,**19**},{2,5,12,15,**19**},{3,8,9,14,**19**},{4,7,10,13,**19**};
L_2: {1,7,12,14,**20**},{2,8,11,13,**20**},{3,5,10,16,**20**}{4,6,9,15,**20**};

It is obvious that it turns to be a BIBD(16, 20, 5, 4, 1) if we interchange the blocks and objects.

Theorem 4.1. If there exists a set of *k-2* MOLS of order *m* and an S(*2, k, m*), there exists an S(*2, k, mk*).

Corollary 4.2 Any S(2, q, q^t) can be constructed recursively by MOLS for integer t > 1 and prime power q.

Theorem 4.3 If there exists a set of *k-2* MOLS of order *m* and an S(*2, k, m + 1*), there exists an S(*2, k, mk +1*).

Corollary 4.4 There exists an S(*2, q + 1, q^4 + q^3 + 1*) for any prime power *q*.

Theorem 4.5 If there exists a set of *k-2* MOLS of order *m* and an S(*2, k, m + k*), there exists an S(*2, k, mk + k*).

Corollary 4.6 There exist following Steiner systems:
(1)S(2, 7, 7^{t+1} -6) and S(2,7, 7^{t+1} -42) for integer t > 1.
(2)S(2, 8, 8^{t+1} -7) and S(2, 8, 8^{t+1} -56) for integer t > 1.
(3)S(2, 9, 9^{t+1} -8) and S(2, 9, 9^{t+1} -72) for integer t > 1.

Example 4.1. It is known Hyperbolic system S(2, 2^n , 2^{2n+1} - 2^n) exists for integer n > 1. So there exists an S(2, 8, 120). Choosing m = 112, 119 and 120, and by Theorems 4.1, 4.3. and 4.5, we get Steiner systems S(2, 8, 804), S(2, 8, 953) and S(2, 8, 960).

The Steiner systems constructed above can be used to obtain optimal constant weight codes, if we take blocks as codewords. Further, the basic idea used above can be used to construct good constant weight codes.

Theorem 4.7 If there exists a set of *k-2* MOLS of order *m* and an S(*2, k, m + t*), 1 < t < k, then

$$A(mk+t,2k-2,k) \ge \left[\frac{(mk+t)(mk+t-1)}{k(k-1)}\right] - \left[\frac{k(k-1)-t(t-1)}{k}\right] \qquad (4.3)$$

For examples, it is known that there exists S(2, 6, 31). Let m = 27, t = 4. From Theorem 4.7, we have A(166, 10, 6) ≥ 910. The upper bound is A(166,10, 6) ≤ 913 and equality holds if and only if S(2, 6, 166) exists. Let m = 17, t = 4. From Theorem 4.7 and Steiner system S(2, 5, 21), we have A(89, 8, 5) ≥ 390. The upper bound is A(89, 10, 6) ≤ 391.

Smith et al [8] have presented comparison results on constant weight codes with n ≤ 63 and w ≤ 8. Table 4.1 only lists some examples constructed using the method discussed above. In the table, UB denotes the upper bound, Previous Best gives the best known results prior to the results presented here as "New Result". They are very close to the upper bounds.

Table 4.1 New Constant Weight Codes

N	D	w	New Result	Previous Result [8]	UB
42	10	6	55	49	56
48	10	6	70	56	72
54	10	6	87	65	90
56	12	7	71	57	72
63	12	7	88	60	90

5. Conclusion

Combinatorial designs can be used to construct good codes. Better majority logic decodable codes and good constant weight codes have been constructed based on mutually orthogonal Latin squares and Steiner systems. The same idea has also led to new constructions of several families of Steiner systems. Comparisons with known results are made to show the improvements and many examples are also given to demonstrate the constructions.

References

[1] F.J. MacWilliams, and N.J.A. Sloane, "The theory of error-correcting codes", North-Holland, 1979

[2] S. Lin, and D.J. Costello, Jr., "Error control coding: fundamentals and aplications", Prentice Hall, 1983

[3] H. HALL, Jr.: "Combinatorial theory", 2nd edition, John & Son, 1986

[4]Charles Colbourn and Jeff Dinitz, "Mutually orthogonal latin squares: a brief survey of constructions", J. of Stat. Planning and Inference, vol.95, pp.9-48, 2001

[5] A. E. Brouwer, and G. H. J. van Rees, "More mutually orthogonal Latin squares", Discrete Math., vol. 39, pp.263-281, 1982

[6] M. Y. Hsiao, D. G. Bossen, and R. T. Chien, "Orthogonal Latin square codes", IBM J. Res. and Devel., 14, pp.390-394, 1970

[7]A. E. Brouwer, J. B. Shearer, and N.J.A. Sloane, "A new table of constant weight codes", IEEE Trans. Inform. Theory, vol. 36, pp.1334-1380, 1990

[8] D. Smith, A. Sakhnovich, S. Perkins, D. Knight, and L. Hughes, "Application of coding theory to the design of frequency hopping lists", Tech. Report UG-M-02-01, February, 2002

[9]N.J.A. Sloane, Table of bounds for constant weight codes, http://www.research.att.com/~njas/codes/Andw

Binary extended perfect codes of length 16 by generalized concatenated construction.

D. V. Zinoviev and V. A. Zinoviev

Institute for Problems of Information Transmission

Russian Academy of Sciences

Bol'shoi Karetnyi, 19, GSP-4, Moscow, 101447, Russia

zinov@iitp.ru

Abstract

We enumerate the binary extended nonlinear perfect codes of length 16, obtained by the generalized concatenated (shortly GC) construction. There are 15 different types of such codes, defined by the pairs of MDS codes $A_i : (4, 2, 64)_4$. For every pair, we evaluate the number of the non-equivalent codes of this type. Overall there are 285 non-equivalent such GC codes.

1 Introduction

One of the interesting open problem of the algebraic coding theory is *the classification of nonlinear binary perfect codes with Hamming parameters.* Even for the smallest nontriv length $n = 15$ or $n = 16$ (for the extended codes) this problem is very far from the : solution. Hergert [1] has found all the non-equivalent Vasiliev's codes of length 15: th are 19 such non-equivalent codes (including the linear code). Malugin [2] showed that number of non-equivalent extended Vasiliev's codes of length 16 is equal to 13 (including linear code). Phelps [3] enumerated all perfect codes of length 15, obtained by the doubl construction due to Solov'eva and Phelps: there are 963 of such codes, including the lin code. In [4] all additive perfect binary codes have been classified (3 codes for $n = 15$) and

[5] all Z_4-linear extended binary perfect codes have been found (2 codes for $n = 16$). Malu; [6] enumerated all the different (not non-equivalent) nonlinear perfect codes of length obtained from Hamming code by the simultaneous translate of non-overlapping compone: of different directions: there are 131224432 such codes.

2 Enumeration of binary perfect GC codes

The binary extended perfect GC codes [7-9] of length 16 are based on two quaternary M] codes $A : (4, 2, 64)_4$. Let $E = \{0, 1\}$ and $E_a = \{0, 1, 2, 3\}$. Define

$$\mathbf{b}(0,0) = (1000),\ \mathbf{b}(0,1) = (0111),\ \ \mathbf{b}(1,0) = (0100),\ \mathbf{b}(1,1) = (1011),$$
$$\mathbf{b}(2,0) = (0010),\ \mathbf{b}(2,1) = (1101),\ \ \mathbf{b}(3,0) = (0001),\ \mathbf{b}(3,1) = (1110).$$

For a given $A : (4, 2, 64)_4$ define the *odd half-code* $C(A)$ as:

$$C(A) \;=\; \{(\mathbf{b}(a_1, e_1)\,|\,\mathbf{b}(a_2, e_2)\,|\,\mathbf{b}(a_3, e_3)\,|\,\mathbf{b}(a_4, e_4)) :\ (a_1, a_2, a_3, a_4) \in A,\ (e_1, e_2, e_3, e_4) \in E$$

Let we have two arbitrary MDS codes $A : (4, 2, 64)_4$ and $A' : (4, 2, 64)_4$ over $E_a = \{0, 1, 2,$; The resulting binary extended perfect GC code $C = C : (16, 4, 2048)$ is the union of t odd half-code $C_{odd} = C(A)$ and the *even half-code* $C_{even} = C(A') + \mathbf{s}$ where the vec: $\mathbf{s} = (1000|1000|1000|1000)$ is fixed. We denote $C = (C(A), C(A') + \mathbf{s})$.

Let S_n be the full group of permutations of n elements. We need two groups: G_4 $S_4 \rtimes (S_4)^4$ and $G_{16} = S_{16} \rtimes (S_2)^{16}$. Using the map $(E_a)^4 \to E^{16}$ described above, it is easy see that G_4 is a subgroup of S_{16}, and, therefore the action of G_4 on E^{16} is defined.

Definition 1 *For any subgroup G of G_{16} we say, that codes $C, C' \subseteq E^{16}$ are G-*equivalent *$C = \mathbf{g}\,C'$ for some $\mathbf{g} \in G$.*

The next result (which has been obtained also by V. N. Potapov; unpublished; [10]) s: us that there are 15 types of cascade codes $C : (16, 4, 2^{11})$.

Lemma 1 *There exist 5 non-equivalent MDS codes $A : (4, 2, 64)_4$, denoted $A_1, \ldots, A_5$.*

Define the *canonical half-codes* $C_i = C(A_i)$, $i = 1, \ldots, 5$. For $i \in \{1, 2, 3, 4, 5\}$ define:

$$P_i \;=\; Stab_{G_4}(A_i),\ \ Q_i \;=\; Stab_{G_4}(C_i),\ \ Q_i^s \;=\; Stab_{G_4}(C_i + \mathbf{s}).$$

For any $i, j \in \{1, 2, 3, 4, 5\}$, we have the following double coset decomposition:

$$G_4 \;=\; \bigcup_k Q_i^s \mathbf{d}_{ij}^{(k)} Q_j, \text{ where } k = 1, \ldots, m_1(i, j),$$

$m_1(i,j)$ is the number of $(Q_i^s\text{-}Q_j)$-double cosets of G_4 and $\{\mathbf{d}_{ij}^{(k)}\} = Q_i^s \backslash G_4 / Q_j$ is the fi
set of their representatives.

Let $\mathcal{C}$ (respectively $\mathcal{C}_0$) be the set of all cascade codes $C : (16, 4, 2048)$ (respectively, w
the zero word). If a code $C \in \mathcal{C}$ does not have a zero word then $C + x \in \mathcal{C}_0$ $(x \in C)$ does.

We say that $\mathbf{h} = (\mathbf{h}_1 \,|\, \mathbf{h}_2 \,|\, \mathbf{h}_3 \,|\, \mathbf{h}_4)$, $\mathbf{h}_i) \in E^4$ is *even (odd)* if all weights $\mathrm{wt}(\mathbf{h}_i)$, $i = 1, 2,$
are respectively even (odd). We consider only those shifts $\mathbf{h} \in H_{16}$ under whose acti
$\mathcal{C}$ is closed. It implies that $\mathbf{h}$ is either an odd or even. The subset of even shifts H_{16}^{ev}
a subgroup of H_{16}. Any odd shift can be presented in the form $\mathbf{h} + \mathbf{s}$, where $\mathrm{h} \in H_{16}^{ev}$ a
$\mathbf{s} = (1000|1000|1000|1000)$. By construction, we see that a shift by vector $\mathbf{h} = (\mathbf{h}_1 \,|\, \mathbf{h}_2 \,|\, \mathbf{h}_3 \,|$
where $\mathrm{wt}(\mathbf{h}_i) \in \{0, 4\}$ for all $i = 1, 2, 3, 4$ does not change the code. We denote the subgr
of such vectors by H_{16}^0. Set $H^{ev} = H_{16}^{ev} / H_{16}^0$ (a factor-group of 256 elements). Then
action of group H_{16}^{ev} on $\mathcal{C}$ is the same as that of H^{ev} and the action of H_{16} is the same
$H = H^{ev} \times \{\mathbf{0}, \mathbf{s}\}$, where $\mathbf{0} \in E^{16}$ is the zero vector. Thus, we can consider the gr
H^{ev} instead of H_{16}^{ev}, assuming that H^{ev} is made of elements $\{\mathbf{h} = (\mathbf{h}_1 \,|\, \mathbf{h}_2 \,|\, \mathbf{h}_3 \,|\, \mathbf{h}_4)) \in H_1^e$
where $\mathbf{h}_i \in \{(0000), (1100), (1010), (1001)\}$, $i = 1, 2, 3, 4$. For $i \in \{1, 2, 3, 4, 5\}$ define the
$H_i = \{\mathbf{h} \in H$: there exist $\mathbf{a} = \mathbf{a}_\mathbf{h} \in G_4$ such that $C_i + \mathbf{s} + \mathbf{h} = \mathbf{a}_\mathbf{h}\,(C_i + \mathbf{s})$.

Proposition 1 *For any vector* $\mathbf{h} \in H^{ev}$ *there exists a permutation* $\phi(\mathbf{h}) \in G_4$ *such th*
$C_i + \mathbf{h} = \phi(\mathbf{h})\, C_i$, *for any canonical half-code* $C_i = C(A_i)$ $(i = 1, 2, 3, 4, 5)$.

Proposition 2 *(the action of shifts). For any* $(C_i + \mathbf{s},\ \mathbf{g}\, C_j)$ *we have:* (i) $(C_i + \mathbf{s},\ \mathbf{g}\, C_j) + \mathbf{h}$
$\mathbf{a}_\mathbf{h}\,(C_i + \mathbf{s}, \mathbf{g}'\, C_j)$, *for any* $\mathbf{h} \in H_i$, *where* $\mathbf{g}' = \mathbf{a}_\mathbf{h}{}^{-1} \mathbf{g}\, \phi(\mathbf{h}_1)$ *and* $\mathbf{h}_1 = \mathbf{g}^{-1}(\mathbf{h})$. (ii)$(C$
$\mathbf{s},\ \mathbf{g}\, C_j) + \mathbf{s} = \mathbf{g}\,(C_j + \mathbf{s},\ \mathbf{g}^{-1} \phi(\mathbf{h})\, C_i) + \mathbf{h}$, *where* $\mathbf{h} = \mathbf{s} + \mathbf{g}(\mathbf{s}) \in H^{ev}$, *and* $C_i + \mathbf{h} = \phi(\mathbf{h})($

Our goal is to find all the G_{16}-non-equivalent cascade codes $C : (16, 4, 2048)$. For
subgroup $G \in G_{16}$ and any $C \in \mathcal{C}$ define the G-orbit of C in $\mathcal{C}$: $\mathrm{Orb}_G(C) = \{gC : g \in$
Clearly any G_{16}-orbit of $\mathcal{C}$ has representatives from $\mathcal{C}_0$.

Lemma 2 *Any* $(G_4 \rtimes H)$*-orbit of* $\mathcal{C}$ *has one representative* $C_{ij}^{(k\ell)}$, *where* $i, j \in \{1, \ldots, 5\}$, i
and $\ell = 1, \ldots, m_2(i,j)$, *where* $m_2(i,j)$ *is the number of* $(G_4 \rtimes H)$*-orbits and* $C_{ij}^{(k\ell)}$ *are th*
representatives.

Lemma 3 *(computational results). The number* $m_2(i,j) = m_2(j,i)$ *is equal to:*

$$m_2(1,1) = 10,\ \ m_2(2,1) = 16,\ \ m_2(3,1) = 16,\ \ m_2(4,1) = 8,\ \ m_2(5,1) = 6,$$

$$m_2(1,2) = 16,\ \ m_2(2,2) = 31,\ \ m_2(3,2) = 40,\ \ m_2(4,2) = 18,\ \ m_2(5,2) = 13,$$

$$m_2(1,3) = 16,\ \ m_2(2,3) = 40,\ \ m_2(3,3) = 66,\ \ m_2(4,3) = 16,\ \ m_2(5,3) = 25,$$

$$m_2(1,4) = 8,\ \ m_2(2,4) = 18,\ \ m_2(3,4) = 16,\ \ m_2(4,4) = 11,\ \ m_2(5,4) = 7,$$

$$m_2(1,5) = 6,\ \ m_2(2,5) = 13,\ \ m_2(3,5) = 25,\ \ m_2(4,5) = 7,\ \ m_2(5,5) = 13.$$

Proposition 3 *(exceptional cases). There are 24 exceptional codes, partitioned into 13 di ferent orbits. The codes from the same orbit are equivalent to each other under action of som permutation from* S_{16}/G_4.

Theorem 1 *There are 285 non-equivalent binary extended perfect codes* $C : (16, 4, 2^{11})$, *o tained by the generalized concatenated construction. The number of non-equivalent such cod of the type* (i,j) *is not more than* $m_2(i,j)$, *where* $m_2(i,j)$ *is given in Lemma 3.*

REFERENCES

1. *Hergert F.* The equivalence classes of the Vasiliev codes of length 15// in: Lectu Notes in Mathematics. Combinatorial Theory, Springer - Verlag, Berlin-Heidelberg-New Yor 1982. V. 969. P. 176–186.

2. *Malugin S.A.* On the equivalence classes of perfect binary codes of length 15//, i preparation.

3. *Phelps K.* An enumeration of 1-perfect binary codes of length 15// Australian Journ of Combinatorics. 2000. V. 21. P. 287–298.

4. *Borges J. and Rifà J.* A characterization of 1-perfect additive codes",*IEEE Trans. Information Theory*, vol. 45, pp. 1688-1697, 1999.

5. *Krotov D.S.* Z_4-linear perfect codes// Discretnyi Analiz i Issledovanie Operatzyi. 200 Ser. 1. V. 7. N° 4. P. 78–90 (in Russian).

6. *Malugin S.A.* On enumeration of perfect binary codes of length 15// Discretnyi Anal i Issledovanie Operatzyi. 1999. Ser. 1. V. 6. N° 2. P. 48-73 (in Russian).

7. *Zinoviev V.A.* On generalized concatenated codes// in: Coll. Math. Societatis Jàn Bolyai. V. 16. Topics in Inform. Theory. Keszthely, Hungary. 1975. P. 587–592.

8. *Phelps K.* A general product construction for error correcting codes// SIAM J. Alg braic and Discrete Methods. 1984. V. 5. N° 1. P. 224–228.

9. *Zinoviev V.A., Lobstein A.* On generalized concatenated constructions of perfect bina nonlinear codes// Problems of Inform. Transm. 2000. V. 36. N° 4. P. 59–73.

10. *Solov'eva F.I.* Personal communication// 2001. October.

AUTHOR INDEX

NOTES