Some distance problems in coding theory

SOME DISTANCE PROBLEMS IN CODING THEORY

C.L.M. van PUL

SOME DISTANCE PROBLEMS

IN CODING THEORY

SOME DISTANCE PROBLEMS

IN CODING THEORY

PROEFSCHRIFT

ter verkrijging van de graad van doctor
aan de Technische Universiteit Eindhoven,
op gezag van de rector magnificus, prof.
dr. F.N. Hooge, voor een commissie
aangewezen door het college van dekanen
in het openbaar te verdedigen op

vrijdag 16 januari 1987 te 16.00 uur

door

CORNELIS LEONARDUS MARIA van PUL

geboren te Rotterdam.

Dit proefschrift is goedgekeurd
door de promotoren:

Prof. dr. J.H. van Lint
en
Prof. dr. ir. H.C.A. van Tilborg

ABSTRACT

This thesis is concerned with four topics from coding theory. The first one of these, treated in Chapter 1, is that of coding in an imperfect computer memory with stuck-at-defects and random errors. This coding problem finds its origin in a paper by Kusnetsov and Tsybakov (1974). After a short historical overview in Section 1.1, a description of the problem and some related problems is given in Section 1.2. The Sections 1.3 up to 1.5 deal with lower (i.e., constructions) and upper bounds for the various functions defined in Section 1.2. The function $A(n,d)$, i.e., the largest size of any binary code of length n and minimum distance d, plays an important rôle in these sections.

In Chapter 2 we treat two constructions for constant weight codes. These constructions result in improved lower bounds on the function $A(n,d,w)$, i.e., the largest size of any binary constant weight code of length n, minimum distance d and constant weight w. This function plays an important rôle in determining upper bounds on the function $A(n,d)$ (e.g.: Linear Programming Bound and Johnson bound).

In Chapter 3 we give the complete solution of a problem formulated by Ahlswede, El Gamal and Pang in 1984. They define a constant distance code pair $(\mathcal{A},\mathcal{B})$ as a pair of binary codes of length n such that for some $\delta \in \mathbb{N}$, $0 \leq \delta \leq n$,

$$\forall_{\underline{a} \in \mathcal{A}} \forall_{\underline{b} \in \mathcal{B}} [d(\underline{a},\underline{b}) = \delta] .$$

They prove that for such a code pair $|\mathcal{A}| \cdot |\mathcal{B}| \leq 2^{2\lfloor \frac{n}{2} \rfloor}$. With the help of coding theory Hall and van Lint gave a nice proof of this inequality and moreover characterized all code pairs for which equality holds. Since for these code pairs $\delta = \lfloor \frac{n}{2} \rfloor$ or $\lceil \frac{n}{2} \rceil$, the question remained: "what happens when δ is fixed?". Chapter 3 gives an answer to this question.

In Chapter 4 we discuss a problem which arose in connection with comma-free codes. Let $W_n(q)$ denote the maximal number of codewords in any q-ary comma-free code of length n. Eastman (1965) proved that

$$W_n(q) = \frac{1}{n} \sum_{d|n} \mu(d) q^{n/d} =: B_n(q) \text{ if } n \text{ is odd.}$$

For even wordlength n the situation is much more complicated. In 1984 Golomb and Tang proved that

$$W_{2k}(q) < B_{2k}(q) \text{ if } q > t(k) + k,$$

where $t(k)$ is the maximal cardinality of any $\{0,1,*\}$ tournament code of length k. Chapter 4 deals with the problem of determining lower and upper bounds on $t(k)$, $k \in \mathbb{N}$.

In order to make this thesis self-contained, we start with a short introduction to coding theory in Chapter 0.

CONTENTS

CHAPTER 0

INTRODUCTION

The purpose of this introduction is to make the reader familiar with some of the notions of coding theory; for a course in coding theory we refer the reader to [1] and [2]. We restricted ourselves to the binary case.

Let $\mathbb{F}_2^n$ be the n-dimensional vector space over $\mathbb{F}_2$. A block code C of length n over $\mathbb{F}_2$ is a subset of $\mathbb{F}_2^n$. The elements of C are called codewords. The set of elements of $\mathbb{F}_2$ is called the alphabet of the code C.

A k-dimensional linear subspace of $\mathbb{F}_2^n$ is called a binary linear block code or binary [n,k]-code.

The Hamming-weight wt($\underline{x}$) of a vector $\underline{x} \in \mathbb{F}_2^n$ is the number of non-zero coordinates of $\underline{x}$. The Hamming-distance $d(\underline{x},\underline{y})$ of two vectors $\underline{x}$ and $\underline{y}$ in $\mathbb{F}_2^n$ is defined by $d(\underline{x},\underline{y}) := \mathrm{wt}(\underline{x} \oplus \underline{y})$. In words: $d(\underline{x},\underline{y})$ is the number of coordinate places in which $\underline{x}$ and $\underline{y}$ differ. The minimum distance d of a code C is defined by

$$d := \min \{d(\underline{x},\underline{y}) \mid \underline{x} \in C,\ \underline{y} \in C,\ \underline{x} \neq \underline{y}\}.$$

A block code of length n and minimum distance d is called an (n,d)-code. An (n,d)-code with M codewords, we call an (n,M,d)-code. An (n,M,d)-code of which all codewords have the same Hamming-weight, w say, is called a constant weight code or an (n,M,d,w)-code. A linear [n,k]-code with minimum distance d is called an [n,k,d]-code (the minimum distance in a linear code equals the minimum weight among all non-zero codewords).

In the vector space $\mathbb{F}_2^n$ we define an innerproduct (,) in the usual way .

$$\forall_{\underline{x}\in \mathbb{F}_2^n} \quad \forall_{\underline{y}\in \mathbb{F}_2^n} \quad [(\underline{x},\underline{y}) := x_1y_1 \oplus x_2y_2 \oplus \ldots \oplus x_ny_n] ,$$

where $\oplus$ denotes the usual addition in $\mathbb{F}_2$. If C is an $[n,k]$-code, then the dual code $C^{\perp}$ of C is defined by

$$C^{\perp} := \{\underline{x} \in \mathbb{F}_2^n \mid \forall_{\underline{y}\in C} \, [(\underline{x},\underline{y}) = 0]\}.$$

The code $C^{\perp}$ is an $[n,n-k]$-code.

A <u>generator matrix</u> G of an $[n,k]$-code C is a $k \times n$ matrix, the rows of which form a basis of C. A <u>parity-check matrix</u> H of a linear code C is a generator matrix of the code $C^{\perp}$. Both G and H define the code C. The matrices G and H satisfy $GH^T = 0$ (evaluated in $\mathbb{F}_2$).

Block codes are used for reliable transmission of information over noisy channels. Examples of noisy channels are: telephone wires, telegraph wires, computer memories, etc. A simple model of such a channel is the binary symmetric channel, i.e., a channel over which we can send two different symbols 0 and 1 and for which there is a probability p that a transmitted 0 (resp.1) is interpreted by the receiver as a 1 (resp.0). The following figure illustrates the information-transmission scheme.

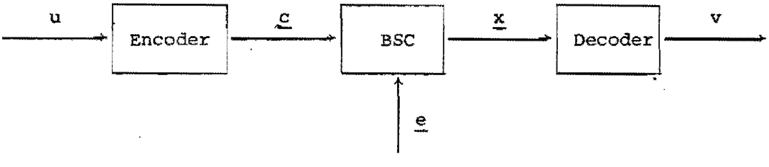


Fig. 1.

We use the following notation:

$u \in \{0,1,\ldots,M-1\} =: U$ the input message set, $\underline{c} \in \mathbb{F}_2^n$ a channel input word, $\underline{x} \in \mathbb{F}_2^n$ a channel output word, $v \in \{0,1,..,M-1\}$ the output message and $\underline{e} \in \mathbb{F}_2^n$ an error vector describing the noise on the binary symmetric channel.

The channel input word $\underline{c}$ and channel output word $\underline{x}$ are related as follows

$$\underline{x} = \underline{c} \oplus \underline{e} \;,$$

where $\oplus$ is the usual addition in $\mathbb{F}_2$ which operates on the vectors componentwise.

In order to protect the information, sent over the BSC channel, one can use the codewords of a binary (n,M,d)-code C as channel input words. A one-to-one mapping Φ, Φ: $U \to C$, is used to map any message $u \in U$ onto a codeword $\Phi(u) = \underline{c} \in C$. The function Φ is called an encoding function for C. A particular decoding function Ψ, $\Psi: \mathbb{F}_2^n \to U$, for C can be defined by

$$v = \Psi(\underline{x}) \;:=\; \Phi^{-1}(\underline{c}') \;,$$

where $\underline{c}'$ is the (not necessarily unique) codeword of C which lies closest to $\underline{x} = \underline{c} + \underline{e}$. If $\mathrm{wt}(\underline{e}) \leq \frac{d-1}{2}$ then one easily sees that $\underline{c}'$ is equal to $\underline{c}$ and hence v is equal to u. We say that C is a $\lfloor\frac{d-1}{2}\rfloor$- error-correcting code.

The decoding principle described above is known as maximum likelihood decoding. It requires the determination of the (not necessarily unique) codeword $\underline{c}'$ of C, which lies closest to the received channel output word $\underline{x}$. This is a laborious task if the cardinality of C is big and C has no structure whatsoever. The linear structure of a code can be utilized to make the decoding somewhat easier.

Let C be a binary linear code with parity check matrix H. For every $\underline{x} \in \mathbb{F}_2^n$ we call $\underline{x}H^T$ the syndrome of $\underline{x}$. From the above we have that the codewords of C are characterized by syndrome $\underline{0}$. The syndrome is an important tool in decoding received vectors $\underline{x}$. Since C is a sub-group of $\mathbb{F}_2^n$ we can partition $\mathbb{F}_2^n$ into cosets of C. Two vectors $\underline{x}$ and $\underline{y}$ are in the same coset iff they have the same syndrome ($\underline{x}H^T = \underline{y}H^T \Leftrightarrow \underline{x} \oplus \underline{y} \in C$). Therefore, if a vector $\underline{x}$ is received, where $\underline{x} = \underline{c} \oplus \underline{e}$, $\underline{c} \in C$, then $\underline{x}$ and $\underline{e}$ have the same syndrome. It follows, that for maximum likelihood decoding of $\underline{x}$ one must choose a vector $\underline{e}'$ of minimal weight in the coset with syndrome $\underline{x}H^T$ and then decode $\underline{x}$ as $\Phi^{-1}(\underline{x} \oplus \underline{e}')$. The vector $\underline{e}'$ is called the <u>coset leader</u>. Again if $\mathrm{wt}(\underline{e}) \leq \frac{d-1}{2}$ then $\underline{e}'$ is equal to $\underline{e}$ and hence we will decode $\underline{x}$ correctly.

Since time is money, we must in general keep the time needed for the transmission of information as short as possible. Let C be a binary (n,M,d)-code. Then the rate R of C, defined by

$$R := n^{-1} \log M ,$$

is a measure for the efficiency of the code C. Since, for a message $u \in U$, with $|U| = M$, we need on the average $\log M$ bits to distinguish u from all other messages in U, the number $n(1-R)$ gives an indication of the loss of time in transmission when the code C is used for error protection. It will be clear that the higher the rate of C the lower the error-correcting capability of C. So knowledge of the following two functions is of the utmost importance.

A(n,d) := maximum number of codewords in any binary code (linear or non-linear) of length n and minimum distance d,

and

B(n,d) := maximum number of codewords in any linear binary code of length n and distance d.

REFERENCES

[1] van LINT, J.H.: *Introduction to Coding Theory*. New York Heidelberg Berlin: Springer, 1982.

[2] MacWILLIAMS, F.J. and SLOANE, N.J.A.: *The Theory of Error-correcting Codes*. Amsterdam-New York-Oxford: North-Holland, 1977.

CHAPTER 1

COMPUTER MEMORIES WITH "STUCK-AT" DEFECTS AND RANDOM ERRORS

1.1 INTRODUCTION

In this chapter we consider the problem of reliable storage of information in an imperfect binary computer memory. We consider a memory that is composed of a very large number of binary memory cells which are partitioned into memory units of n cells. (n the block-length of the error-correcting code to be used). We are concerned with two types of imperfections that affect individual memory cells. The first type is a defective memory cell that is unable to store information; its current value cannot be changed. Such cells are called stuck-at cells. We distinguish between stuck-at-0 and stuck-at-1 cells. When a 1 is written into a stuck-at-0 cell an error results. The second type of imperfection is a noisy cell which is occasionally in error. The distinction between these two types of imperfections is that stuck-at defects are permanent, while errors caused by noise are transient.

By testing a memory unit it is possible to determine the locations and natures of the stuck-at cells. The side information that describes the state of the defects can be incorporated in the decoding or in the encoding of block codes. Depending on how this stuck-at information is exactly used, this gives rise to a number of different coding (reliable storage) problems. We mention the two most "interesting" ones.

In the first one, the locations of the stuck-at-cells are assumed to be known only at the decoder. These cells then act as erasures. Thus, it makes sense to apply known techniques for decoding block codes with random errors and erasures in this case. We will not go into this problem. The interested reader is refered to [2]. We consider the complementary problem of incorporating stuck-at information in the encoding process.

This last problem was originated by Kusnetsov and Tsybakov in [11]. They consider coding for binary memory units that have a number t of stuck-at cells, where $t \leqq pn, 0 < p < 1$, p fixed. The assumption is that the locations and natures of the defects are known at the encoder but not at the decoder. By allowing the size of the memory unit n to become large, they prove the existence of codes that are capable of storing information without error, for any rate $R < 1 - p$. Moreover, they prove that such codes can be found within the class of additive codes (see [11]). In Section 1.2 we give an outline of this paper. At the end of this section we introduce the related problem of exhaustive test pattern generation. In both problems so-called t-defect-compatible matrices play a very important rôle. Also the equivalence of the notion of t-defect-compatibility and that of t-independence of sets is mentioned. This fact seems to be almost unknown.

In Section 1.3 we prove an upper bound for the largest possible length of a t-defect-compatible matrix with m rows. This bound gives a slight improvement on the one given in [9]. Section 1.4 deals with constructions for additive codes, capable of correcting all word defects of multiplicity t or less and hence by nature, also constructions for exhaustive pattern testing schemes. The constructions described there, in fact generate separable t-defect-compatible matrices.

In [19] Tsybakov introduces the problem of coding for binary memory units with both defects and random errors. Ounce again the locations and natures of the defects are assumed to be known at the encoder but not at the decoder. He introduces the concept of "matched adjacent codes" to solve this problem. In [7] Heegard calls these codes partitioned linear block codes. We will stick to that name. In Section 1.5 we use their ideas and one of our construction methods of Section 1.4 to construct codes that have a better performance than those given in [7]. With these codes the encoding process will take more time, the decoding process on the other hand not.

The problem of determining the capacity of imperfect computer memories when complete or partial defect information is available at the encoder or at the decoder is not studied here. For this problem, we refer the interested reader to [6].

1.2 CODING FOR AN IMPERFECT COMPUTER MEMORY

§1.2.1 An algebraic model

The following figure illustrates the information-transmission (storage) scheme we are concerned with.

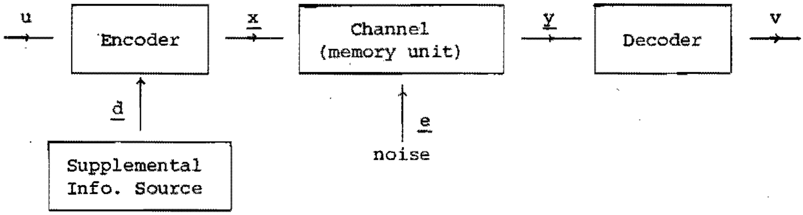


Fig. 1.

We use the following notation:

$u \in \{0,1,\ldots,M-1\}$ = the input message set, $\underline{x} \in \mathbb{F}_2^n$ a channel input word, $\underline{y} \in \mathbb{F}_2^n$ a channel output word, $v \in \{0,1,\ldots,M-1\}$, the output message, $\underline{e} \in \mathbb{F}_2^n$ an error vector describing the noise on the channel and $\underline{d} = \in \{0,1,\delta\}^n$ a word defect describing the states of the memory cells to be used.

The word defect $\underline{d}=(d_1,d_2,\ldots,d_n) \in \{0,1,\delta\}^n$ has to be interpreted as follows:

$$\text{if } d_i = \begin{cases} 0, \text{ then the } i^{th} \text{ cell of the memory unit is stuck-at-0,} \\ 1, \text{ then the } i^{th} \text{ cell of the memory unit is stuck-at-1,} \\ \delta, \text{ then the } i^{th} \text{ cell of the memory unit is defect-free.} \end{cases}$$

The number t of coordinates of $\underline{d}$ equal to 0 or 1 is called the <u>multiplicity</u> of the word defect $\underline{d}$. By D_t^n we denote the set of word defects $\underline{d} \in \{0,1,\delta\}^n$ with multiplicity t or less. Let the "o" operator $\circ : \mathbb{F}_2 \times \{0,1,\delta\} \longrightarrow \mathbb{F}_2$ be defined by

$$x \circ d := \begin{cases} x & \text{if} \quad d = \delta \ , \\ d & \text{if} \quad d \neq \delta \ . \end{cases}$$

The relation between the channel input word $\underline{x}$ and channel output word $\underline{y}$ can then be described by

$$\underline{y} = (\underline{x} \circ \underline{d}) \oplus \underline{e} \ . \tag{1}$$

The errors, described by the error vector $\underline{e}$, occur when reading the memory, so they affect the memory contents of defect-free cells as well as that of stuck-at cells.

<u>EXAMPLE 1</u>. Let $n = 6$, $\underline{x} = (0,0,0,0,0,0)$, $\underline{d} = (\delta,\delta,0,1,1,0)$ and $\underline{e} = (0,1,0,0,1,1)$. Then $\underline{y} = (\underline{x} \circ \underline{d}) \oplus \underline{e} = (0,1,0,1,0,1)$.

§ 1.2.2 The class of additive codes

During the rest of this section we assume that there is no noise on the channel (memory); so $\underline{e} = \underline{0}$ in (1). Furthermore, we assume that the stuck-at cells are randomly distributed over the memory. In [11] Kusnetsov and Tsybakov define a block code of length n for this memory as a partition of $\mathbb{F}_2^n$ into M subsets A_u, $u=0,1,\ldots,M-1$. They use the defect information, known at the encoder, to assign to a message u a channel input word $\underline{x} \in A_u$ in such a way that the stuck-at cells of the memory unit to be used do not alter $\underline{x}$. The decoder, receiving the unaltered $\underline{x}$, recognizes that $\underline{x}$ belongs to the subset A_u and so recovers the message u correctly. The rate R of the code is given by $R = (\log M)/n$. To find suitable partitions of $\mathbb{F}_2^n$, Kusnetsov and Tsybakov make use of so-called separable t-defect-compatible matrices, leading to the introduction of the class of additive codes. We need some definitions.

> A word $x = (x_1,x_2,\ldots,x_n) \in \mathbb{F}_2^n$ is said to be compatible with the word defect $\underline{d} = (d_1,d_2,\ldots,d_n) \in \{0,1,\delta\}^n$ if $\underline{x} = \underline{x} \circ \underline{d}$; so $x_i = d_i$, for all $i \in \{1,2,\ldots,n\}$ with $d_i = 0$ or 1. A binary $m \times n$ matrix C is called a t-defect-compatible matrix, if for any word defect $\underline{d} \in D_t^n$, there is a row of C which is compatible with $\underline{d}$.

We are now ready to define the class of additive codes.

> Let C be a $2^r \times n$ binary matrix in which the first r elements of each row form the binary representation of the number i of that row $(i = 0,1,\ldots,2^r - 1)$. A matrix with this property is said to be a separable matrix. Let, for any $u \in U = \{0,1,\ldots,2^{n-r}-1\}$ and any $\underline{d} \in \{0,1,\delta\}^n$, $\underline{c}(u,\underline{d})$ be a specified row of C. This specification will be made clear later on. For any $u \in U$, the vector $\underline{u} = (u_1,u_2,\ldots,u_n) \in \mathbb{F}_2^n$ is given by
>
> $$u_i = 0 (i = 1,2,\ldots,r) \text{ and } u = \sum_{i=1}^{n-r} u_{r+i} \; 2^{i-1}.$$
>
> The encoding function Φ, $\Phi\colon U \times \{0,1,\delta\}^n \longrightarrow \mathbb{F}_2^n$, is defined by
>
> $$\Phi(u,\underline{d}) := \underline{u} \oplus \underline{c}(u,\underline{d}).$$
>
> The code (partition of $\mathbb{F}_2^n$) defined by C is clearly given by

$$\mathbb{F}_2^n = \bigcup_{u=0}^{2^{n-r}-1} \{\underline{u} \oplus \underline{c} \mid \underline{c} \text{ a row of } C\}.$$

The rate R of this code is equal to $R = (n-r)/n = 1-r/n$. Different separable matrices C define a class of codes, which we call the class of additive codes.

The decoding function Ψ, $\Psi : \mathbb{F}_2^n \to U$, for the additive code is defined by

$$\Psi(\underline{y}) := \sum_{i=1}^{n-r} (y_{r+i} \oplus c_{r+i}) \cdot 2^{i-1},$$

where $\underline{c} = (c_1, c_2, \ldots, c_n)$ is that row of C with

$$c_i = y_i \quad \text{for} \quad i = 1,2,\ldots,r.$$

From the above it will be clear that a necessary and sufficient condition for the additive code, defined by the separable matrix C, to correct all word defects of multiplicity t or less is that C is a separable t-defect-compatible matrix. For any $u \in U$ and $\underline{d} \in D_t^n$ the row $\underline{c}(u,\underline{d})$ from C must then be the (not necessarily unique) row of C which is compatible with the word defect $\underline{d}'$ defined by

$$d'_i := \begin{cases} \delta & \text{if} \quad d_i = \delta, \\ u_i \oplus d_i & \text{if} \quad d_i = 0 \text{ or } 1, \end{cases}$$

where u_i is the i^{th} component of the vector $\underline{u}$ defined above.

<u>EXAMPLE 2.</u> Let $n = 4$, $r = 1$, $t = 1$, $C = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$, $u = 3$ and $\underline{d} = (\delta, \delta, 0, \delta)$.

<u>Encoding</u>: To encode we determine $\underline{u} = (0,1,1,0)$, $\underline{d}' = (\delta,\delta,1,\delta)$. and $\underline{c}(u,\underline{d}) = (1,1,1,1)$. We store

$$\underline{x} = \Phi(u,\underline{d}) = (0,1,1,0) \oplus (1,1,1,1) = (1,0,0,1).$$

Decoding: To decode retrieve from the computer memory the vector $\underline{y} = \underline{x} \circ \underline{d} = (1,0,0,1)$ and from C the row $\underline{c}$ with index $1 \cdot 1 = 1$; so $\underline{c} = (1,1,1,1)$. Now compute the value

$$v = \Psi(\underline{y}) = (0 \oplus 1)1 + (0 \oplus 1)2 + (1 \oplus 1)4 = 3 = u.$$

EXAMPLE 3.

Let $n = 3$, $r=2$, $t = 2$, $C = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$, $u = 1$ and $\underline{d} = (1,\delta,1)$.

Encoding: $\underline{u} = (0,0,1)$, $\underline{d}' = (1,\delta,0)$ and $\underline{c}(u,\underline{d}) = (1,1,0)$. So store the vector

$$\underline{x} = \Phi(u,\underline{d}) = (0,0,1) \oplus (1,1,0) = (1,1,1).$$

Decoding: Retreive $\underline{y} = \underline{x} \circ \underline{d} = (1,1,1)$ and the row $\underline{c}$ of C with index $1 \cdot 1 + 1 \cdot 2 = 3$; so $\underline{c} = (1,1,0)$. Compute

$$v = \Psi(\underline{y}) = 1 \cdot 1 = 1 = u.$$

We define the function $R(n,t)$ by

$R(n,t) :=$ the maximal value of R for which there exists a code with rate R that is capable of correcting all word-defects of multiplicity t or less.

In [11] Kusnetsov and Tsybakov prove the following surprising result.

THEOREM 1. For any $n,t \in \mathbb{N}$, $1 \leq t \leq n$,

$$1 - \frac{t + \lceil \log \ln 2^t \binom{n}{t} \rceil}{n} \leq R(n,t) \leq 1 - \frac{t}{n}. \tag{2}$$

□

The upper bound in Theorem 1 is obvious. The lower bound is a consequence of the existence of separable t-defect-compatible matrices of size $2^r \times n$, with

$$t \leq r \leq t + \lceil \log \ln 2^t \binom{n}{t} \rceil \tag{3}$$

The existence of such matrices is proved by using a probabilistic "counting" argument. Since, for any fixed $p \in [0,1]$,

$$1 - \frac{t + \lceil \log \ln 2^t \binom{n}{t} \rceil}{n} = 1 - p + o(n^{-\frac{1}{2}}), \text{ for } \frac{t}{n} = p \text{ and } n \to \infty \quad ,$$

we have the following consequence of this theorem.

COROLLARY 2. Let p be any fixed number in $[0,1]$ and let $\varepsilon > 0$. Then, for n sufficiently large (depending on p and ε), there exists an additive code of length n that is capable of correcting all word defects of multiplicity np or less, for a rate R, $1 - p - \varepsilon \leqq R \leqq 1 - p$. □

§ 1.2.3 Some related problems

From § 1.2.2, it will be clear that separable t-defect-compatible matrices play an important rôle in the reliable storage of information in an imperfect computer memory with stuck-at defects. Therefore, we define

$r(n,t)$:= the minimal value of r for which there exists a $2^r \times n$ separable t-defect-compatible matrix, $n, t \in \mathbb{N}$, $1 \leqq t \leqq n$.

The functions $R(n,t)$ and $r(n,t)$ are related by

$$nR(n,t) \geqq n - r(n,t).$$

In the conventional approach to logic circuit testing, a set of test vectors, to be applied at the circuit inputs, is derived from an analysis made on the circuit under test. Typical faults one wishes to determine are stuck-at-0 and stuck-at-1 faults at the gate level. Such a test-generation procedure requires a substantial amount of computer time due to the necessary analysis and simulation to be carried out. Due to the growth of the number of logic circuits on a VLSI-chip, the conventional way of logic test generation becomes more and more impractical. Not only the computer time grows excessively, also the single stuck-at fault model becomes more inadequate. A partial solution to this problem is, to use exhaustive pattern testing schemes for testing several logic circuits simultaneously.

In this approach a VLSI-chip is considered to have n binary inputs. Each input may influence many outputs, but due to certain partitioning techniques each output is assumed to depend on atmost t inputs ($t < n$). To test the chip, any set of t or less inputs feeding an output is provided with all possible input patterns. By checking the correctness of the outputs, any single hard fault or combination of hard faults, which results in a permanent alteration of the thruth table,associated with an output function, is noticed. So we are left with the problem of generating a minimal set of test vectors of length n, to provide simultaneouly all input patterns to each of a collection of input subsets of size t or less. From the above, it may be clear that the rows of an $m \times n$ t-defect-compatible matrix form such a set. Therefore, we define

m(n,t) := the minimal value of m for which there exists an $m \times n$ t-defect-compatible matrix.

The relation between the functions R(n,t) and m(n,t) is given by

$$nR(n,t) \leq n - \log m(n,t) .$$

For a more detailed description of the problem of logic circuit testing, the reader is refered to [5,15].

Most authors who work on these two fields of research do not seem to be aware of the fact that the notion of t-defect-compatibility is equivalent to that of t-independence of sets. Consider the i^{th} column of an $m \times n$ t-defect-compatible matrix as the characteristic vector of a subset A_i of the set $A = \{1,2,\ldots,m\}$. Let F denote the collection of subsets A_i of A, $i = 1,2,\ldots,n$, i.e., $F = \{A_1,A_2,\ldots,A_n\}$. The t-defect-compatibility property can then be formulated as

> For any t-tuple of subsets $A_{k_1},A_{k_2},\ldots,A_{k_t}$ from F, all 2^t intersections
>
> $$\bigcap_{i=1}^{t} B_{k_i}$$
>
> are non-empty, where each B_{k_i} can be either A_{k_i}, or $A \setminus A_{k_i}$.

In [9] Kleitman and Spencer call such a collection a t-independent collection of subsets of an m-element set. In [9] a lower bound on the size of such a collection is proved that coincides with the upper bound on $r(n,t)$ given by (3). In Section 1.3 we mention some of their results translated in the terminology of t-defect-compatible matrices.

For later use we give two more definitions. For any $r,m,t \in \mathbb{N}$ we define

$n(r,t)$:= the maximal value of n for which there exists a $2^r \times n$ separable t-defect-compatible matrix, and

$nf(m,t)$:= the maximal value of n for which there exists an $m \times n$ t-defect-compatible matrix.

The relations between $r(n,t)$ and $n(r,t)$ respectively $m(n,t)$ and $nf(m,t)$ are given by

$$n(r_0,t) \geq n_0 \quad \text{iff} \quad r(n_0,t) \leq r_0$$

and

$$nf(m_0,t) \geq n_0 \quad \text{iff} \quad m(n_0,t) \leq m_0 .$$

We conclude this section with a table of known values of r(n,t), m(n,t) and R(n,t), for $t = 0,1,n-1$ and n.

t	r(n,t)	m(n,t)	R(n,t)
0	-	-	1
1	1	2	$1 - 1/n$
$n - 1$	$n - 1$	2^{n-1}	$1/n$
n	n	2^n	0

Table 1.

1.3 UPPER BOUNDS ON nf(m,t)

In [9] Kleitman and Spencer consider the problem of determining the largest size of a t-independent family of subsets of an m-element set. From the previous section we know that this is equivalent with determining the largest value of n, for which there exists an $m \times n$ t-defect-compatible matrix. We have denoted this maximal value by nf(m,t). In [9] Kleitman and Spencer solve this problem for $t = 2$ (see Theorem 3) and give asymptotic upper and lower bounds for nf(m,t), where $t \geq 3$ is fixed and m tends to infinity. Although, from a coding point of view, determination of such bounds is of almost no interest, we found this problem interesting enough to work on. In this section we prove a slight improvement on the upper bound given in [9].

We first give the solution for $t = 2$ in Theorem 3. Because of our interest in seperable t-defect compatible matrices, the value of n(r,2) is also mentioned.

THEOREM 3. [9] For all $m, r \in \mathbb{N}$, $m \geq 4$ and $r \geq 2$ we have

$$nf(m,2) = \binom{m-1}{\lceil \frac{m}{2} \rceil} \text{ and } n(r,2) = \binom{2^r - 1}{2^{r-1}}.$$

$\square$

The values of nf(m,2) and n(r,2) are attained by the following construction.

Construction.

Let $m \in \mathbb{N}$, $m \geq 4$. We define C to be the $m \times \binom{m-1}{\lceil \frac{m}{2} \rceil}$ matrix with as columns all binary vectors of length m and Hamming weight $\lceil \frac{m}{2} \rceil$, of which the first coordinate is equal to zero (see Fig.2. below).

$$\begin{bmatrix} 0&0&0&0&0&0&0&0&0&0 \\ 1&1&1&1&1&1&0&0&0&0 \\ 1&1&1&0&0&0&1&1&1&0 \\ 1&0&0&1&1&0&1&1&0&1 \\ 0&1&0&1&0&1&1&0&1&1 \\ 0&0&1&0&1&1&0&1&1&1 \end{bmatrix}$$

Fig.2. A 6×10, 2-defect-compatible matrix.

From the above figure,it is easy to see that this construction indeed yields a 2-defect-compatible matrix. If $m = 2^r$ the matrix is separable.

As a consequence of Theorem 3 we find the following values for m(n,2), r(n,2) and R(n,2) (see also § 1.2.3). Let, for any $n \in \mathbb{N}$, $m_0 \in \mathbb{N}$ be defined by

$$\binom{m_0 - 2}{\lceil \frac{m_0 - 1}{2} \rceil} < n \leq \binom{m_0 - 1}{\lceil \frac{m_0}{2} \rceil}$$

and take $r_0 = \lceil \log m_0 \rceil$. Then

$$m(n,2) = m_0 \quad ,$$

$$r(n,2) = r_0 \quad \text{and}$$

$$1 - \frac{r_0}{n} \leq R(n,2) \leq 1 - \frac{r_0 - 1}{n} \quad .$$

We now aim our attention at the case $t \geqq 3$. In [9] Kleitman and Spencer prove the following lower and upper bound on $nf(m,t)$.

THEOREM 4. For all $t \in \mathbb{N}$, $t \geqq 3$, t fixed, we have

$$nf(m,t) \geqq 2^{(\ell_t + o(1))m}, \quad m \to \infty,$$

and

$$nf(m,t) \leqq 2^{(u_t + o(1))m}, \quad m \to \infty,$$

where $\ell_t = (\log(1-2^{-t}))/t$ and $u_t = (H(2^{-(t-1)}) - 2^{-(t-2)})/(t-2)$, with $H(p) := -p \log p - (1-p)\log(1-p)$, the well-known binary entropy function. □

Let, as defined in Chapter 0, $A(m,d)$ denote the largest value of M for which there exists a binary (n,M,d) code. The following theorem uses the function $A(m,d)$ to derive an upper bound on $nf(m,t)$, $t \geqq 4$.

THEOREM 5. For any $m, t \in \mathbb{N}$, $4 \leqq t \leqq m$,

$$nf(m,t) \leqq \max_{0 \leqq d \leqq \frac{1}{2}m} \min \{nf(\lfloor \tfrac{d}{2} \rfloor, t-2) + 2, \tfrac{1}{2}A(m,d)\}. \tag{4}$$

PROOF. Let C be an $m \times nf(m,t)$ t-defect compatible matrix. Let A be the binary code with as codewords all the columns of C and $\overline{C}$. From the definition of a t-defect-compatible matrix, it follows that C and $\overline{C}$ have no columns in common. Let d be the minimum distance of A. Then,

$$2nf(m,t) = |A| \leqq A(m,d). \tag{5}$$

Since A has minimum distance d, there are two columns of C, w.l.o.g. the first two columns $\underline{h}_1$ and $\underline{h}_2$, with $d(\underline{h}_1,\underline{h}_2) = d$ or $n-d$. Assume $d(\underline{h}_1,\underline{h}_2) = d$ (the case $d(\underline{h}_1,\underline{h}_2) = n-d$ goes analogously). Consider the matrices C_1 respectively C_2 consisting of those rows of C for which the first entry is equal to 0 and the second entry is equal to 1, respectively the first entry equal to 1 and the second entry equal to 0. From the t-defect-compatibility of C, it follows immediately that the matrices C_1' and C_2', which are formed by deleting the first two columns of C_1 respectively C_2, are (t - 2)- defect-compatible matrices. Let m_i be the number of rows of C_i', $i = 1,2$. Then $m_1 + m_2 = d$. Since both matrices have nf(m,t) - 2 columns, we have

$$\begin{aligned} nf(m,t) - 2 &\leq \min \{ nf(m_1, t-2), nf(m_2, t-2) \} \\ &\leq nf(\lfloor \tfrac{d}{2} \rfloor, t-2). \end{aligned} \tag{6}$$

The last inequality follows from the fact that for fixed t,nf(m,t) is an increasing function of m. Together, (5) and (6) give the desired inequality (4). □

As a consequence of Theorem 5, we have

<u>COROLLARY 6</u>. Let $u_3' = u_3$, $u_4' = u_4$ and let,for $t \geq 5$, u_t' be defined by

$$u_t' := \max_{0 \leq \delta \leq \frac{1}{2}} \min \{ \tfrac{\delta}{2} u_{t-2}', H(\tfrac{1}{2} - \sqrt{\delta(1-\delta)}) \}.$$

Then, for any $t \in \mathbb{N}$, $t \geq 5$,

$$nf(m,t) \leq 2^{(u_t' + o(1))m}, \quad t \text{ fixed and } m \to \infty.$$

<u>PROOF</u>. Use induction on t and the well-known MRRW upper bound [14] as an estimate for A(m,d) in (4). □

Corollary 6 gives a slight improvement on the upper bound on nf(m,t) of Theorem 4, when t is greater than or equal to 4. In Table 2 we list the values of ℓ_t, u_t and u_t' for t = 3,4,5,6,8 and 10.

t	ℓ_t	u_t	u'_t
3	0.0642	0.3112	0.3112
4	0.0232	0.1467	0.1467
5	0.00916	0.0707	0.0643
6	0.00378	0.0345	0.0322
8	$7.058 \cdot 10^{-4}$	$8.382 \cdot 10^{-3}$	$7.635 \cdot 10^{-3}$
10	$1.409 \cdot 10^{-4}$	$2.060 \cdot 10^{-3}$	$1.865 \cdot 10^{-3}$

Table 2.

1.4 CONSTRUCTIONS FOR (SEPARABLE) t-DEFECT-COMPATIBLE MATRICES WITH $t \geqq 3$.

Many authors have considered the problem of constructing (separable) t-defect-compatible matrices [1,3,4,13,18]. In this section we describe two construction methods that, to our knowledge, yield the best results. The first one is due to Busschbach [3]. This construction uses a small t-defect-compatible matrix to generate a larger one. So t stays fixed, while the length n grows. The second construction allows t to grow proportionaly with n and is therefore used to determine a "constructive" asymptotic lower bound on $r(n,np)$ for p fixed, $0 < p \leqq \frac{1}{2}$ and $n \to \infty$. We also use this construction to derive some lower bounds on $n(r,t)$, for $r \leqq 20$ and $3 \leqq t \leqq 10$.

§ 1.4.1 A construction for t-defect-compatible matrices of length n, with $t << n$

In this paragraph we describe the construction method, for t-defect-compatible matrices, found by Busschbach in [3]. The adjustments necessary to make the resulting matrix separable are ours. The construction uses a small t-defect-compatible matrix to generate a larger one. These small matrices can, for instance, be constructed by the method of § 1.4.2.

Construction.

Let A be a $m_0 \times n_0$ t-defect-compatible matrix, where n_0 is a prime power $\geq \frac{t^2}{4}$. Let $\mathcal{B}$ be an n_0-ary linear MDS code with dimension k, $2 \leq k \leq \frac{4n_0}{t^2} + 1$ and length $m = (k-1) \lfloor \frac{t^2}{4} \rfloor + 1$. Since $m \leq n_0 + 1$, these codes are easy to construct (see [14]). Let B be the $m \times n_0^k$ matrix with as columns the codewords of $\mathcal{B}$. Let $\varphi : \mathbb{F}_{n_0} \to \{$columns of A$\}$ be a bijection. Construct the $mm_0 \times n_0^k$ binary matrix C by replacing each entry b of B by $\varphi(b)$.

THEOREM 7. The matrix C, constructed above, is a t-defect-compatible matrix.

PROOF. To prove the t-defect-compatibility of C, let C' be any $mm_0 \times t$ submatrix of C and let $\underline{d}'$ be any binary vector of length t. We have to show that $\underline{d}'$ is contained in the row set of C'. To prove this we go back to the code $\mathcal{B}$. Let $\underline{b}^j = (b_1^j, b_2^j, \ldots, b_m^j)$ be that codeword of $\mathcal{B}$ that corresponds to the j^{th} column of C'. Since every coordinate b_i^j, $1 \leq j \leq t$, is replaced by a column of the t-defect-compatible matrix A, we are done if we can show that there is an i, $1 \leq i \leq m$, such that

$$\{b_i^j \mid d_j' = 0\} \cap \{b_i^j \mid d_j' = 1\} = \emptyset. \qquad (7)$$

From the t-defect-compatibility of A we then have that $\underline{d}'$ is contained in the row set of the submatrix $C'' = (\varphi(b_i^1)\ \varphi(b_i^2) \ldots\ldots \varphi(b_i^t))$ of C'.

So suppose that (7) does not hold for any $i \in \{1,2,\ldots,m\}$; so $\underline{d}' \neq \underline{0}, \underline{1}$. We calculate the sum

$$\sum_{i \mid d_i' = 0} \quad \sum_{j \mid d_j' = 1} d(\underline{b}^i, \underline{b}^j)$$

in two different ways. Firstly, since (7) does not hold for any coordinate $i \in \{1,2,\ldots,m\}$, each coordinate contributes at most $n_0(\underline{d}') \cdot n_1(\underline{d}') - 1$ to the sum, where $n_\lambda(\underline{d}') := |\{i \mid d_i' = \lambda\}|$, $\lambda = 0,1$. Hence

$$\sum_{i \mid d_i' = 0} \quad \sum_{j \mid d_j' = 1} d(\underline{b}^i, \underline{b}^j) \leqq m \cdot (n_0(\underline{d}') \cdot n_1(\underline{d}') - 1).$$

Secondly, since the minimum distance of $\mathcal{B}$ is equal to $m-k+1$ ($\mathcal{B}$ is MDS), we also have

$$\sum_{i \mid d_i' = 0} \quad \sum_{j \mid d_j' = 1} d(\underline{b}^i, \underline{b}^j) \geqq n_0(\underline{d}') \cdot n_1(\underline{d}') \cdot (m-k+1).$$

Thus, we may conclude that

$$n_0(\underline{d}') \cdot n_1(\underline{d}') \cdot (m-k+1) \leqq m \cdot (n_0(\underline{d}') \cdot n_1(\underline{d}') - 1)$$

or equivalently

$$m \leqq (k-1) \cdot n_0(\underline{d}') \cdot n_1(\underline{d}') \leqq (k-1)\left\lfloor \frac{t^2}{4} \right\rfloor.$$

A contradiction with $m = (k-1)\left\lfloor \frac{t^2}{4} \right\rfloor + 1$. □

The bounds on nf(m,t) and n(m,t), that result from this construction are so untransparent that we do not give them here. We confine ourselves to an example for $t = 3$ and refer the interested reader to [3].

EXAMPLE 4. Let A be the 8×4 3-defect-compatible matrix with as rows the codewords of the [4,3,2] binary code. Let $m_i \times n_i$ denote the size of the 3-defect-compatible matrix after i succesive applications of the above construction with maximal k; so $m_0 = 8$ and $n_0 = 4$. Then we find the following values for m_i and n_i, $i = 1,2,3$.

i	1	2	3
m_i	$3\cdot 2^3$	$45\cdot 2^3$	$\geqq 2^{41}$
n_i	2^4	2^{32}	$\geqq 2^{2^{35}}$

We see that the number n_i grows excessively with respect to the number m_i, but never the less, it does not result in a lower bound on $nf(m,3)$ of the form $nf(m,3) \geqq 2^{\alpha m}$, α fixed.

Although, we feel that Busschbach's construction is of little importance (t is too small compared to n) for the construction of additive codes, we adjusted the construction somewhat in order to make it yield (weak) separable t-defect-compatible matrices. A binary $n\times m$ matrix is called weakly separable if there exists a $n \times \lceil \log n \rceil$ submatrix of A that has n different rows. Matrices like this can also be used to define an additive code.

Construction.

Let A be a $2^{r_0} \times n_0$ separable t-defect-compatible matrix, where n_0 is a prime power $\geq \frac{t^2}{4}$. Let $\mathcal{B}$ be a n_0-ary linear MDS code with dimension $2 \leqq k \leqq \frac{4n_0}{t^2} + 1$ and word length $m=(k-1)\lfloor\frac{t^2}{4}\rfloor + 1$ such that $\underline{1} \in \mathcal{B}$. This is not a serious restriction when $m \leqq n_0$. Let B be the $m \times n_0^k$ matrix with as columns the codewords of $\mathcal{B}$. Let the elements of $\mathbb{F}_{n_0}$ be labelled by $\alpha_1, \alpha_2, \ldots, \alpha_{n_0}$ and the columns of A by $\underline{a}_1, \underline{a}_2, \ldots, \underline{a}_{n_0}$ and let $s = \lceil \log m \rceil$. Assume $r_0 + s \leqq n_0$ (this will almost always be the case). For any $v \in \{0,1,\ldots,m-1\}$ we define the vector $\underline{v} \in \mathbb{F}_2^{n_0}$ by

$$\underline{v} := (0,0,\ldots,0,\ v_{r_0+1},\ v_{r_0+2},\ldots,v_{r_0+s},\ 0,0,\ldots,0),$$

where $v = \sum_{i=1}^{s} v_{r_0+i}\ 2^{i-1}$.

And $\varphi_{\underline{v}} : GF(n_o) \to \{\text{columns of } A\}$ by

$$\varphi_{\underline{v}}\ (\alpha_i) = \underline{a}_i \oplus v_i \cdot \underline{1} \quad , \ i = 1,2,\ldots,n_0,$$

where $\underline{1}$ is the all-one vector of length 2^{r_0}. Construct the $m \cdot 2^{r_0} \times n_0^{\ k}$ binary matrix C by replacing each entry b in the v^{th} row of B by $\varphi_{\underline{v}}(b)$, $v = 0,1,\ldots,m-1$.

THEOREM 8. The matrix C, defined above, is a weak separable t-defect-compatible matrix, if $r_0 + s \leqq n_0$.

PROOF. Since the entries in each row of B are mapped on the columns of a t-defect-compatible matrix, the t-defect-compatibility of the matrix C is a direct consequence of the proof of Theorem 7.

To prove the weak separability of C we consider the codewords $\alpha_1 \cdot \underline{1}$, $\alpha_2 \cdot \underline{1}, \ldots, \alpha_{r_0+s} \cdot \underline{1}$ of $\mathcal{B}$. From the separability of A and the definition of $\varphi_{\underline{v}}$, $v \in \{0,1,2,\ldots,m-1\}$,one immediately sees that the $2^{r_0} \cdot m \times (r_0 + s)$ submatrix of C which corresponds with these codewords, consists of $2^{r_0} \cdot m$ differents rows ▯

§ 1.4.2 A generalization of a construction method found by Kusnetsov in [10].

From Corollary 2 we have that, for any $p \in [0,1]$ and n sufficiently large, there exists an additive code of length n that is capable of correcting all word defects of multiplicity np or less, for a rate R very close to $1 - p$. However, the question remains :"how to construct such a code?". In this section we describe a construction method for separable t-defect-compatible matrices that gives a partial solution to this problem.

As a first reaction we and many others with us tried to solve this problem with the help of linear codes. Let $\mathcal{C}$ be a binary $[n,k]$ code for which the dual code has minimum distance $t+1$. Then the $2^k \times n$ matrix C with as rows the codewords of the code $\mathcal{C}$ is easily seen to be a separable t-defect-compatible matrix. From the Gilbert-Varshamov bound one easily derives that this construction yields the following asymptotic upper bound on $r(n,np)$,

$$r(n,np) \leqq n(H(p)+o(1)), \text{ for } p \text{ fixed, } 0<p<\tfrac{1}{2}, \text{ and } n \to \infty \qquad (8)$$

and so

$$R(n,np) \geqq n(1-H(p)+o(1)), \text{ for } p \text{ fixed, } 0<p<\tfrac{1}{2}, \text{ and } n \to \infty \text{ .}$$

However, not only the bound is poor, it is also cheating; no one as yet has found a construction of a family of binary linear codes that realizes the promises of the Gilbert-Varshamov bound. At present we only know that such families of good codes exist and can, for instance, be found within the class of Goppa codes [14, Ch.12].

To our surprise, the following observation shows that it is rather simple to find such a construction for t-defect-compatible matrices; the resulting upper bound for $m(n,np)$ is even sharper than (8). Let C be the matrix with as rows all binary words of length n, which have weight $\lfloor \frac{t}{2} \rfloor$ or weight $n-\lfloor \frac{t-1}{2} \rfloor$. It is clear that C is a t-defect-compatible matrix for any t, $1 \leqq t \leqq \lfloor \frac{2n}{3} \rfloor$. The asymptotic upper bound on $m(n,np)$, $0<p<\frac{2}{3}$, that results from this construction reads

$$m(n,np) \leqq 2^{n(H(\frac{p}{2})+o(1))}, \text{where } p \text{ is fixed, } 0<p<\frac{2}{3} \text{ and } n \to \infty.$$

Although, this bound "improves" (8), it is not really sharp. The resulting t-defect-compatible matrices, however, may be of interest for the generation of exhaustive test patterns; because of the simple structure, the resulting test sets can be effectively implemented (see [17]).

Since we believe that $2^{r(n,t)}$ is about as big as $m(n,t)$, for all values of t and n, $0 < t \leq n$, the simplicity of the above construction convinced us, that it must be possible to find a similar construction method for separable t-defect-compatible matrices. A generalization of a construction method for separable 3-defect-compatible matrices found by Kustnetsov in [10], does the trick.

Construction.

Let A be a binary $[2^r,k,d]$ code with $\underline{1} \in A$ and minimum distance $d \geq \lceil (2^{t-2}-1)2^r/(2^{t-1}-1)\rceil$. Let G be a generator matrix of A with the all-one vector as top-row. Let H be a parity check matrix of an $[n,n-k,2\lceil\frac{t+1}{2}\rceil]$ binary even weight code which has the all-one vector as top-row. We define the $2^{r+1} \times n$ matrix C by

$$C := \left[\begin{array}{c} \overline{G^TH} \\ \hline G^TH \end{array}\right],$$

where $\overline{G^TH}$ is the complementary matrix of G^TH.

THEOREM 9. The matrix C defined above is a t-defect-compatible matrix if $t \geq 3$. If G contains the generator matrix of RM(1,r) as a submatrix, then C can be made separable.

PROOF. To prove the t-defect-compatibility, it suffices to show that for any subset $J \subset \{1,2,\ldots,n\}$ with $|J| = t$ and any $\underline{z} \in \mathbb{F}_2^t$, there is an $i \in \{1,2,\ldots,2^r\}$ such that

$$\underline{e}_i\, G^TH_J = \underline{z} \text{ or } \underline{1} \oplus \underline{z}\ , \tag{9}$$

where H_J is the $k \times t$ matrix that consists of those columns of H which have a column index belonging to J and where $\underline{e}_i$ is the i^{th} basis vector of $\mathbb{F}_2^{2^r}$.

Suppose there is a $J \subset \{1,2,\dots,n\}$, $|J| = t$ and a $\underline{z} \in \mathbb{F}_2^t$ such that (9) does not hold for any $i \in \{1,2,\dots,2^r\}$. Then

$$\sum_{\substack{\underline{x} \in \mathbb{F}_2^t,\ \underline{x} \neq \underline{0},\\ wt(\underline{x}) \equiv 0 \bmod 2}} (\underline{e}_i G^T H_J \oplus \underline{z}, \underline{x}) = 2^{t-2} .$$

So

$$\sum_{i=1}^{2^r} \sum_{\substack{\underline{x} \in \mathbb{F}_2^t,\ \underline{x} \neq \underline{0},\\ wt(\underline{x}) \equiv 0 \bmod 2}} (\underline{e}_i G^T H_J \oplus \underline{z}, \underline{x}) = 2^r \cdot 2^{t-2} . \tag{10}$$

On the otherhand, we have

$$\sum_{i=1}^{2^r} \sum_{\substack{\underline{x} \in \mathbb{F}_2^t,\ \underline{x} \neq \underline{0},\\ wt(\underline{x}) \equiv 0 \bmod 2}} (\underline{e}_i G^T H_J \oplus \underline{z}, \underline{x}) = \sum_{\substack{\underline{x} \in \mathbb{F}_2^t,\ \underline{x} \neq \underline{0},\\ wt(\underline{x}) \equiv 0 \bmod 2}} \sum_{i=1}^{2^r} (\underline{e}_i G^T H_J \oplus \underline{z}, \underline{x}) =$$

$$= \sum_{\substack{\underline{x} \in \mathbb{F}_2^t,\ \underline{x} \neq \underline{0},\\ wt(\underline{x}) \equiv 0 \bmod 2}} \sum_{i=1}^{2^r} \{ (\underline{e}_i G^T H_J, \underline{x}) \oplus (\underline{z}, \underline{x}) \} =$$

$$= \sum_{\substack{\underline{x} \in \mathbb{F}_2^t,\ \underline{x} \neq \underline{0},\\ wt(\underline{x}) \equiv 0 \bmod 2}} wt(\underline{x}\, H_J^T G \oplus (\underline{x}, \underline{z}) \underline{1}) \leq (2^{t-1} - 1) \cdot (2^r - d) . \tag{11}$$

The inequality is consequence of the fact, that for any $\underline{x} \in \mathbb{F}_2^t \setminus \{\underline{0}\}$ with $wt(\underline{x}) \equiv 0 \bmod 2$, the word $\underline{x}\ H_J^T G \oplus (\underline{x}, \underline{z}) \cdot \underline{1} \in A \setminus \{\underline{0}, \underline{1}\}$. For, since H has the all-one vector as top-row, $wt(\underline{x}) \equiv 0 \bmod 2$ and $\underline{x} \neq \underline{0}$, the first coordinate of $\underline{x}H_J^T$ is equal to 0 and $\underline{x}H_J^T \neq \underline{0}$. So, since the top-row of G is also $\underline{1}$, we may conclude that $\underline{x}H_J^T G \oplus A \setminus \{\underline{0}, \underline{1}\}$.

Together (10) and (11) give

$$2^r - d \geq \lceil \frac{2^{t-2} \cdot 2^r}{2^{t-1}-1} \rceil ,$$

or equivalently

$$d \leq 2^r - \lceil \frac{2^{t-2} \cdot 2^r}{2^{t-1}-1} \rceil = \lfloor \frac{(2^{t-2}-1)2^r}{2^{t-1}-1} \rfloor .$$

This is a contradiction with $d = \lceil \frac{(2^{t-2}-1)2^r}{2^{t-1}-1} \rceil$ if $t \geq 3$. So C is a t-defect-compatible matrix if $t \geq 3$.

The separability of C, when G contains the generator matrix of RM(1,r) as a submatrix, is obvious. □

<u>REMARK</u>. For the case $t = 3$, the above construction can somewhat be simplified. Let A be a binary $(m,n,\lceil \frac{m+1}{3} \rceil)$code with the property that for all $\underline{a} \in A$, also $\underline{1} \oplus \underline{a} \in A$. Let A_0 respectively A_1 denote the matrix with as columns the codewords of A of which the first coordinate is equal to 0 respectively equal to 1. Then the $2m \times n$ matrix C defined by

$$C := \left(\frac{A_0}{A_1} \right)$$

is a 3-defect-compatible matrix. When $m = 2^r$ and A contains RM(1,r) as a subcode, the matrix C can be made separable. This is in essence the construction for 3-defect-compatible matrices Kusnetsov gave in [10].

In order to make the above construction work we have to generate the matrices G and H which are mentioned there. The matrix G is the most important one. Suitable candidates for G are the generator matrices of the codes we describe in Theorem 10. For a proof of this theorem and construction of these codes we refer to [14].

<u>THEOREM 10</u>. Let $r = 2\ell + 1$ and let i be any number in the range $1 \leq i \leq \ell$. Then there exists two

$$[2^r, r(\ell - i + 2) + 1, 2^{r-1} - 2^{r-i-1}]$$

subcodes of RM(2 ,r).These subcodes contain RM(1,r) as a subcode.

□

Let A be one of the two $[2^r, r(\ell - i + 2) + 1, 2^{r-1} - 2^{r-i-1}]$ codes of Theorem 10. Then $\underline{1} \in A$, since $\mathrm{RM}(1,r) \subset A$. Since $2^{r-1} - 2^{r-i-1} \geqq \lceil \frac{(2^{t-2}-1)2^r}{2^{t-1}-1} \rceil$, if $t \leqq i+1$, we have, according to Theorem 9, that the existence of an $[n, n-(\ell - i + 2)r - 1, 2\lceil \frac{t+1}{2} \rceil]$ binary code, $3 \leqq t \leqq i+1$, gives rise to the existence of a $2^{r+1} \times n$ separable t-defect-compatible matrix.

In Table 3 we give some lower bounds on $n(r,t)$, for moderate values of r and t, which result from this construction. To generate the matrices G, we did not only use the codes from Theorem 10, but we also used codes that result from Wiseman's construction method, which we described in [16]. For the matrices H we used a table search [14,20]. The letter h in the upper left corner of an entry indicates that the corresponding lower bound on $n(r,t)$ is attained by a linear code whose dual code has minimum distance $t+1$ and dimension r. The letter k indicates that this lower bound is attained by the construction of Kusnetsov [10].

r \ t	3	4	5	6	7	8	9	10
3	h 4	-	-	-	-	-	-	-
4	h 8	h 5	-	-	-	-	-	-
5	2^5 2^7 k	h 6	h 6	-	-	-	-	-
6	2^{10}	h 8	h 7	h 7	-	-	-	-
7	2^{12}	24	h 9	h 8	h 8	-	-	-
8	2^{21}	129	h 12	h 9	h 9	h 9	-	-
9	2^{29} 2^{31} k	272	66	h 10	h 10	h 10	h 10	-
10	2^{36}	2^{13}	513	h 15	h 12	h 11	h 11	h 11
11			2^{10}	36	h 16	h 12	h 12	h 12
12				129	h 24	h 14	h 13	h 13
13				258	68	h 15	h 15	h 14
14				2^{13}	312	h 17	h 16	h 15
15					513	46	h 18	h 17
16					2^{15}	143	h 21	h 18
17						257	74	h 20
18						2^{12}	275	h 23
19							513	64
20							2^{14}	150
21								257
22								2^{12}

Table 3. Lower bounds on n(r,t)

We conclude this section with the promised "constructive" lower bound for R(n,np). Let G be a $((2\ell+1)(\ell-i+2)+1) \times 2^{2\ell+1}$ generator matrix of one of the codes mentioned in Theorem 10 with the all-one vector as top-row. Let H be any $((2\ell+1)(\ell-i+2)+1) \times ((2\ell+1)(\ell-i+2)+1)$ regular binary matrix with the all-one vector as top-row. Then, from Theorem 9, the matrix C defined by

$$C = \begin{bmatrix} \overline{G^T H} \\ \hline G^T H \end{bmatrix} ,$$

is a separable $(i+1)$-defect-compatible matrix of size

$$2^{2\ell+2} \times ((2\ell+1)(\ell-i+2)+1) .$$

Let $n = (2\ell+1)(\ell-i+2)+1$, then the above construction shows

$$r(n,i+1) \leqq 2\ell+2 .$$

Now take $i = \ell - k$, k fixed and let ℓ tend to infinity. Then, since $\lim\limits_{\ell\to\infty} \frac{\ell-k+1}{n} = \lim\limits_{\ell\to\infty} \frac{\ell-k+1}{(2\ell+1)(k+2)} = \frac{1}{2(k+2)}$ we find, for any $k \in \mathbb{N} \cup \{0\}$

$$r(n,\frac{n}{2(k+2)}) \leqq n(\frac{1}{k+2} + o(1)) \text{ , k fixed and } n \to \infty .$$

Since r(n,t) is an increasing function of n if t is fixed, we constructively showed

$$r(n,np) \leqq n(2p + o(1)) \text{ , p fixed , } 0 < p \leqq \tfrac{1}{2} \text{ and } n \to \infty .$$

Hence, for any p, $0 < p \leqq \frac{1}{2}$, the above construction can be used to generate a family of additive codes of length n, $n \in \mathbb{N}$, that are capable of correcting all word defects of multiplicity np or less, for a rate $R(n,p)$, for which

$$\lim_{n \to \infty} R(n,p) = \frac{2p}{n} .$$

1.5 GENERALIZED PARTITIONED LINEAR BLOCK CODES

§ 1.5.1 Partitioned linear block codes

In [19] Tsybakov introduces the problem of coding for binary computer memory units with both defects and random errors. The locations and natures of the defects are assumed to be known at the encoder but not at the decoder. Recall from Section 1.1 that such an n-cell memory unit is defined by

$$\underline{y} = (\underline{x} \circ \underline{d}) \oplus \underline{e},$$

where $\underline{x} \in \mathbb{F}_2^n$ is a channel input word, $\underline{y} \in \mathbb{F}_2^n$ a channel output word, $\underline{d}$ a word defect $\in D_t^n$ and $\underline{e}$ an error vector of weight s or less. To solve this problem Tsybakov uses the codewords of a binary $(n, K, d = 2s+1)$ code $\mathcal{C}$ as channel input words. The code $\mathcal{C}$ is partitioned into a number of subcodes $\mathcal{C}_0, \mathcal{C}_1, \ldots, \mathcal{C}_{M-1}$ each of which forms a t-defect-compatible set. He uses the defect information, known at the encoder, to assign to each message $u \in \{0,1,\ldots,M-1\} = U$ a channel input word $\underline{x} \in \mathcal{C}_u$ which is compatible with $\underline{d}$. The decoder, receiving $\underline{y} = (\underline{x} \circ \underline{d}) \oplus \underline{e} = \underline{x} \oplus \underline{e}$ sees that $d(\underline{y}, \mathcal{C}_u) < d(\underline{y}, \mathcal{C}_v)$, for all $v \neq u$, and so recovers the message u correctly. The rate R is defined by $R = \log M/n$.

Since linear block codes are very suited for this coding strategy, Tsybakov introduces the concept of partitioned linear block codes (in [19] these codes are called matched adjacent).We give a formal definition.

An $[n,k_0,k_1]$ partitioned linear block code is a pair of linear codes $C_0 \subset \mathbb{F}_2^n$, $C_1 \subset \mathbb{F}_2^n$ of dimension k_0 and k_1 respectively such that $C_0 \cap C_1 = \{\underline{0}\}$. The direct sum $C = C_0 \oplus C_1 := \{\underline{c}_0 \oplus \underline{c}_1 | \underline{c}_0 \in C_0, \underline{c}_1 \in C_1\}$ forms the set of channel input words.The partition of C into subcodes is described by

$$C = \bigcup_{\underline{c}_1 \in C_1} \{\underline{c}_1 \oplus C_0\}.$$

The rate R is equal to k_1/n.

To define an encoding Φ, Φ: $U \times \{0,1,\delta\}^n \to C$ and a decoding Ψ, Ψ: $\mathbb{F}_2^n \to U$ we need some more definitions.

Let G_0 and G_1 be generator matrices for C_0 and C_1 respectively. Let H be a parity check matrix for $C = C_0 \oplus C_1$ and let $\tilde{G}_1$ be any $k_1 \times n$ binary matrix such that $G_1 \tilde{G}_1^T = I_{k_1}$ and $G_0 \tilde{G}_1^T = 0_{k_0,k_1}$.

We are ready to define the encoding and decoding functions Φ and Ψ resp..

Take the message set U equal to $\mathbb{F}_2^{k_1}$ and let, for any $\underline{u} \in \mathbb{F}_2^{k_1}$ and any $\underline{d} \in \{0,1,\delta\}^n$, $\underline{z}(\underline{u},\underline{d})$ be a specified vector of $\mathbb{F}_2^{k_0}$ (see the proof of Theorem 11).

The encoding Φ, Φ: $\mathbb{F}_2^{k_1} \times \{0,1,\delta\}^n \to C$, is defined by

$$\Phi(\underline{u},\underline{d}) := \underline{u}\, G_1 \oplus \underline{z}(\underline{u},\underline{d})\; G_0.$$

The decoding Ψ, Ψ : $\mathbb{F}_2^n \to \mathbb{F}_2^{k_1}$, is defined by

$$\Psi(\underline{y}) := (\underline{y} \oplus \underline{\hat{e}})\; \tilde{G}_1^T ,$$

where $\underline{\hat{e}} \in \mathbb{F}_2^n$ is chosen to minimize $\mathrm{wt}(\underline{\hat{e}})$ subject to $\underline{\hat{e}}H^T = \underline{s} := \underline{y}H^T$, the syndrome of $\underline{y}$ with respect to the code C. The vector $\hat{e}$ is an estimate for the error $\underline{e}$ in (1).

For any $[n,k_0,k_1]$ partitioned linear block code (C_0,C_1) a pair of minimum distances $(d_0^\perp,d)$ is adjoined, where d is the minimum of the code C and $d_0^\perp$ is the minimum distance of the dual code $C_0^\perp$ of C_0.

<u>THEOREM 11</u>. Let (C_0,C_1) be an $[n,k_0,k_1]$ partitioned linear block code with minimum distance pair $(d_0^\perp,d)$. Then (C_0,C_1) is capable of correcting all word defects of multiplicity t or less and random errors of weight s or less, if

$$t < d_0^\perp \text{ and } 2s < d.$$

<u>PROOF.</u> For any $\underline{u} \in \mathbb{F}_2^{k_1}$ and $\underline{d} \in D_t^n$ we take $\underline{z}(\underline{u},\underline{d})$ equal to $\underline{z} \in \mathbb{F}_2^{k_0}$ such that $\underline{z}G_0$ is compatible with the word defect $\underline{d}' \in D_t^n$ defined by

$$d_i' := \begin{cases} \delta \text{ if } d_i = \delta, \\ (\underline{u}G_1)_i \oplus d_i \text{ if } d_i = 0 \text{ or } 1, \; i = 1,2,\dots,n. \end{cases}$$

Since any t-columns of G_0 are linearly independent this is possible.

With this choice of $\underline{z}(\underline{u},\underline{d})$ it is clear, from the definitions of Φ and Ψ, that (C_0,C_1) is indeed a t-defect-, s-error-correcting code. $\square$

<u>EXAMPLE 5.</u> Let (C_0,C_1) be the $[7,1,3]$ partitioned linear block code defined by

$$G_0 = (1\ 1\ 1\ 1\ 1\ 1\ 1) \text{ and } G_1 = \begin{bmatrix} 1&0&0&0&1&1&1 \\ 0&1&0&0&0&1&1 \\ 0&0&1&0&1&0&1 \end{bmatrix} .$$

Then (C_0,C_1) has minimum distance pair (2,3). Note that C is the [7,4,3] single error-correcting Hamming code. We can take H and $\tilde{G}_1$ equal to

$$H = \begin{bmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad \tilde{G}_1 = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \end{bmatrix}.$$

Let $\underline{u} = (1,0,1)$ be the message to be stored in a computer memory unit with word defect $\underline{d} = (\delta,\delta,0,\delta,\delta,\delta,\delta)$ and error vector $\underline{e} = (0,0,0,1,0,0,0)$.

Encoding: To store the message $\underline{u} = (1,0,1)$ we first compute the word defect $\underline{d}'$ defined in the proof of Theorem 11 and the vector $\underline{z}(\underline{u},\underline{d})$. We find $\underline{d}' = (\delta,\delta,1,\delta,\delta,\delta,\delta)$ and so $\underline{z}(\underline{u},\underline{d}) = (1)$. Hence

$$\underline{x} = \Phi(\underline{u},\underline{d}) = \underline{u}G_1 \oplus \underline{z}(\underline{u},\underline{d})G_0 = \\ = (1,0,1,0,0,1,0) \oplus (1,1,1,1,1,1,1) = (0,1,0,1,1,0,1).$$

Decoding: To decode retrieve the vecor $\underline{y} = (\underline{x} \circ \underline{d}) \oplus \underline{e} = (0,1,0,0,1,0,1)$ from the memory unit and calculate the syndrome $\underline{s} = \underline{y}H^T$. We find $\underline{s} = (1,0,1)$. Since $\underline{s}$ is equal to the fourth column of H, the decoder estimates $\underline{e}$ by $\hat{\underline{e}} = (0,0,0,1,0,0,0) = \underline{e}$. Hence

$$\Psi(\underline{y}) = (\underline{y} \oplus \underline{e})\tilde{G}_1^T = (1,0,1) = \underline{u}.$$

§ 1.5.2 Generalized partitioned linear block codes

In the coding of an $[n,k_0,k_1]$ partitioned linear block code (C_0,C_1), the entire code C_0 is used for masking the defects of the memory unit. As we have seen in § 1.4.2, this is not always necessary. The class of generalized partitioned linear block codes makes advantage of this observation. We start with a definition.

An $[n,k_0,k_1,k_2]$ generalized linear block code consists of a triple of binary linear codes $C_0, C_1, C_2 \subset \mathbb{F}_2^n$ of dimension k_0, k_1 and k_2 respectively such that $C_i \cap C_j = \{\underline{0}\}$, $i,j \in \{0,1,2\}$, $i \neq j$, and a binary code Z of length $k_0 + k_1$, which is separable on the first k_0 coordinate places. The direct sum $C = C_0 \oplus C_1 \oplus C_2$ forms the set of channel input words. Let G_0, G_1 and G_2 be generator matrices of the codes C_0, C_1 and C_2 respectively. Then C is partitioned into

$$C = \bigcup_{\underline{c} \in C_1 \oplus C_2} \{\underline{c} \oplus \underline{z}\left(\begin{array}{c} G_0 \\ \hline G_1 \end{array}\right) \mid \underline{z} \in Z\}.$$

The rate R is equal to $(k_1 + k_2)/n$.

Let H be a parity check matrix of the direct sum $C = C_1 \oplus C_2 \oplus C_3$. Let $\tilde{G}_0$ be any $k_0 \times n$ matrix such that $G_0 \tilde{G}_0^T = I_{k_0}$ and $\left(\begin{array}{c} G_1 \\ \hline G_2 \end{array}\right) \tilde{G}_0^T = O_{k_1 + k_2, k_0}$ and let $\tilde{G}_{1,2}$ be any $(k_1 + k_2) \times n$ matrix such that $\left(\begin{array}{c} G_1 \\ \hline G_2 \end{array}\right) \tilde{G}_{12}^T = I_{k_1 + k_2}$. Then we can define the encoding Φ, $\Phi\colon \mathbb{F}_2^{k_1 + k_2} \times \{0,1,\delta\}^n \to C$ and the decoding Ψ, $\Psi\colon \mathbb{F}_2^n \to C$ by

$$\Phi(\underline{u},\underline{d}) := \underline{u}\left(\begin{array}{c} G_1 \\ \hline G_2 \end{array}\right) \oplus \underline{z}(\underline{u},\underline{d})\left(\begin{array}{c} G_0 \\ \hline G_1 \end{array}\right),$$

where $\underline{z}(\underline{u},\underline{d})$ is a specified codeword of Z (see the proof of Theorem 12) and

$$\Psi(\underline{y}) := (\underline{y} \oplus \underline{\hat{e}} \oplus \underline{\hat{z}})\ \tilde{G}_{12}^T,$$

where $\underline{\hat{e}}$ is chosen to minimize wt($\underline{\hat{e}}$) subjected to $\underline{\hat{e}}H^T = \underline{s} := \underline{y}H^T$ and $\underline{\hat{z}}$ is that codeword of Z that on the first k_0 coordinate places is equal to the vector $(\underline{y} \oplus \underline{\hat{e}})\ \tilde{G}_0^T$.

<u>THEOREM 12</u>. Let (C_0, C_1, C_2, Z) be a $[n, k_0, k_1, k_2]$ generalized partitioned

linear block code, for which

(i) the direct sum $C = C_0 \oplus C_1 \oplus C_2$ has minimum distance $d = 2s+1$,

(ii) the dual code of $C_0 \oplus C_1$ has minimum distance $d_{01}^{\perp} = 2\lceil \frac{t+1}{2} \rceil$,

(iii) $\underline{1} \in C_0$ and G_0 contains $\underline{1}$ as top row,

(iv) the $(k_0 + k_1) \times 2^{k_0 - 1}$ matrix, with as columns those codewords of Z that have a 1 as first coordinate, is the generator matrix of a binary $[2^{k_0-1}, k_0 + k_1, \lceil \frac{(2^{t-2}-1)}{2^{t-1}-1} 2^{k_0-1} \rceil]$code and

(v) for any $\underline{z} \in Z$ also $\underline{1} \in Z$.

Then (C_0, C_1, C_2, Z) is a t-defect, s-error-correcting code

PROOF. From the properties ii) - (v) and Theorem 9 of § 1.4.2 we have that $\{\underline{z}\begin{pmatrix} G_0 \\ \hline G_1 \end{pmatrix} \mid \underline{z} \in Z\}$ forms a separable t-defect-compatible set. Hence for any $\underline{u} \in \mathbb{F}_2^{k_1+k_2}$ and any $\underline{d} \in D_t^n$ there is a $\underline{z} \in Z$ such that $\underline{z}\begin{pmatrix} G_0 \\ \hline G_1 \end{pmatrix}$ is compatible with the word defect $\underline{d}'$ defined by

$$d_i' := \begin{cases} \delta \text{ if } d_i = \delta, \\ (\underline{u}\begin{pmatrix} G_1 \\ \hline G_2 \end{pmatrix})_i \oplus d_i \text{ if } d_i = 0 \text{ or } 1, \; i = 1,2,\ldots,n. \end{cases}$$

Choose $\underline{z}(\underline{u},\underline{d})$ to be equal to $\underline{z}$. With this choice for $\underline{z}(\underline{u},\underline{d})$ and the definitions of Φ and Ψ, the assertion of Theorem 12 is immediate. □

With the help of primitive binary BCH codes of length $n = 31, 63, 127$ and 255 we constructed the following $[n, k_0, k_1, k_2]$ generalized partitioned t-defect-,s-error-correcting linear block codes listed in Tables 4,5,6 and 7. The rate of such a code is equal to $(k_1 + k_2)/n$. The rate of the corresponding partitioned t-defect-,s-error-correcting code of the same length n, given in [7], is equal to k_2/n or $(k_2+1)/n$. So the gain in rate is at least $(k_1 - 1)/n$.

On the other hand, the encoding process of a generalized partitioned linear block code is more complicated than the encoding process of a partitioned linear block code. In both cases, the determination of the vector $\underline{z}(\underline{u},\underline{d})$ (see Theorem 11 and 12), amounts to solving an equation like

$$\underline{z}\,G' = \underline{d}'' \quad ,$$

where the matrix G' and the vector $\underline{d}''$ are directly determined by the vector $\underline{u}$, the word defect $\underline{d}$ and the code used. However, in the case of a partitioned linear block code any solution $\underline{z}$ will do, while in the case of a generalized partitioned linear block code one has to find a solution $\underline{z}$ of the above equation within the set Z. This will take more time.

The decoding process is in both cases the same.

k_0	k_1	k_2	t	s	k_0	k_1	k_2	t	s	k_0	k_1	k_2	t	s	k_0	k_1	k_2	t	s
5	1	20	3	1	8	3	5	4	3	11	5	5	6	2	17	4	5	9	1
5	1	15	3	2	8	3	0	4	5	11	5	0	6	3	17	4	0	9	2
5	1	10	3	3	9	2	15	5	1	13	3	10	7	1	19	2	5	10	1
5	1	5	3	5	9	2	10	5	2	13	3	5	7	2	19	2	0	10	2
5	1	0	3	7	9	2	5	5	3	13	3	0	7	3					
8	3	15	4	1	9	2	0	5	5	15	6	5	8	1					
8	3	10	4	2	11	5	10	6	1	15	6	0	8	2					

Table 4. Generalized partitioned t-defect, s-error-correcting linear block codes of length $n = 31$.

k_0	k_1	k_2	t	s	k_0	k_1	k_2	t	s	k_0	k_1	k_2	t	s	k_0	k_1	k_2	t	s
6	1	50	3	1	8	5	38	4	2	9	4	11	5	7	13	6	17	7	5
6	1	44	3	2	8	5	32	4	3	12	7	38	6	1	13	6	11	7	6
6	1	38	3	3	8	5	26	4	4	12	7	32	6	2	13	6	5	7	7
6	1	32	3	4	8	5	23	4	5	12	7	26	6	3	16	9	32	8	1
6	1	29	3	5	8	5	17	4	6	12	7	20	6	4	16	9	26	8	2
6	1	23	3	6	8	5	11	4	7	12	7	17	6	5	16	9	20	8	3
6	1	17	3	7	9	4	44	5	1	12	7	11	6	6	17	8	32	9	1
6	1	11	3	10	9	4	38	5	2	12	7	5	6	7	17	8	26	9	2
6	1	9	3	11	9	4	32	5	3	13	6	38	7	1	17	8	20	9	3
6	1	3	3	13	9	4	26	5	4	13	6	32	7	2	19	9	29	10	1
6	1	0	3	15	9	4	23	5	5	13	6	26	7	3	19	9	23	10	2
8	5	44	4	1	9	4	17	5	6	13	6	20	7	4	19	9	17	10	3

Table 5. Generalized partitioned t-defect, s-error-correcting linear block codes of length $n = 63$.

k_0	k_1	k_2	t	s	k_0	k_1	k_2	t	s	k_0	k_1	k_2	t	s	k_0	k_1	k_2	t	s
6	2	112	3	1	8	7	63	4	7	12	10	77	6	4	16	13	91	8	1
6	2	105	3	2	8	7	56	4	9	12	10	70	6	5	16	13	84	8	2
6	2	98	3	3	8	7	49	4	10	12	10	63	6	6	16	13	77	8	3
6	2	91	3	4	8	7	42	4	11	12	10	56	6	7	16	13	70	8	4
6	2	84	3	5	8	7	35	4	13	12	10	49	6	9	16	13	63	8	5
6	2	77	3	6	8	7	28	4	15	12	10	42	6	10	16	13	56	8	6
6	2	70	3	7	8	7	21	4	15	12	10	35	6	11	16	13	49	8	7
6	2	63	3	9	10	5	105	5	1	12	10	28	6	13	18	11	91	9	1
6	2	56	3	10	10	5	98	5	2	12	10	21	6	15	18	11	84	9	2
6	2	49	3	11	10	5	91	5	3	12	10	14	6	15	18	11	77	9	3
6	2	42	3	13	10	5	84	5	4	14	8	98	7	1	18	11	70	9	4
6	2	35	3	15	10	5	77	5	5	14	8	91	7	2	18	11	63	9	5
6	2	28	3	15	10	5	70	5	6	14	8	84	7	3	18	11	56	9	6
6	2	21	3	21	10	5	63	5	7	14	8	77	7	4	18	11	49	9	7
6	2	14	3	23	10	5	56	5	9	14	8	70	7	5	20	16	84	10	1
6	2	7	3	27	10	5	49	5	10	14	8	63	7	6	20	16	77	10	2
6	2	0	3	31	10	5	42	5	11	14	8	56	7	7	20	16	70	10	3
8	7	105	4	1	10	5	35	5	13	14	8	49	7	9	20	16	63	10	4
8	7	98	4	2	10	5	28	5	15	14	8	42	7	10	20	16	56	10	5
8	7	91	4	3	10	5	21	5	15	14	8	35	7	11	20	16	49	10	6
8	7	84	4	4	12	10	98	6	1	14	8	28	7	13	20	16	42	10	7
8	7	77	4	5	12	10	91	6	2	14	8	21	7	15					
8	7	70	4	6	12	10	84	6	3	14	8	14	7	15					

Table 6. Generalized partitioned t-defect, s-error-correcting linear block codes of length $n = 127$.

k_0	k_1	k_2	t	s	k_0	k_1	k_2	t	s	k_0	k_1	k_2	t	s	k_0	k_1	k_2	t	s
6	3	238	3	1	6	3	20	3	47	14	11	222	7	1	18	15	182	9	5
6	3	230	3	2	6	3	12	3	55	14	11	214	7	2	18	15	174	9	6
6	3	222	3	3	6	3	4	3	59	14	11	206	7	3	18	15	166	9	7
6	3	214	3	4	6	3	0	3	63	14	11	198	7	4	18	15	158	9	8
6	3	206	3	5	10	7	230	5	1	14	11	190	7	5	18	15	154	9	9
6	3	198	3	6	10	7	222	5	2	14	11	182	7	6	18	15	146	9	10
6	3	190	3	7	10	7	214	5	3	14	11	174	7	7	18	15	138	9	11
6	3	182	3	8	10	7	206	5	4	14	11	166	7	8	18	15	130	9	12
6	3	178	3	9	10	7	198	5	5	14	11	162	7	9	18	15	122	9	13
6	3	170	3	10	10	7	190	5	6	14	11	154	7	10	18	15	114	9	14
6	3	162	3	11	10	7	182	5	7	14	11	146	7	11	18	15	106	9	15
6	3	154	3	12	10	7	174	5	8	14	11	138	7	12	22	19	206	11	1
6	3	146	3	13	10	7	170	5	9	14	11	130	7	13	22	19	198	11	2
6	3	138	3	14	10	7	162	5	10	14	11	122	7	14	22	19	190	11	3
6	3	130	3	15	10	7	154	5	11	14	11	114	7	15	22	19	182	11	4
6	3	122	3	18	10	7	146	5	12	14	11	106	7	18	22	19	174	11	5
6	3	114	3	19	10	7	138	5	13	14	11	98	7	19	22	19	166	11	6
6	3	106	3	21	10	7	130	5	14	14	11	90	7	21	22	19	158	11	7
6	3	98	3	22	10	7	122	5	15	14	11	82	7	22	22	19	150	11	8
6	3	90	3	23	10	7	114	5	18	14	11	74	7	23	22	19	146	11	9
6	3	82	3	25	10	7	106	5	19	14	11	66	7	25	22	19	138	11	10
6	3	78	3	26	10	7	98	5	21	14	11	62	7	26	22	19	130	11	11
6	3	70	3	27	10	7	90	5	22	14	11	54	7	27	22	19	122	11	12
6	3	62	3	29	10	7	82	5	23	14	11	46	7	29	22	19	114	11	13
6	3	54	3	30	10	7	74	5	25	14	11	38	7	30	22	19	106	11	14
6	3	46	3	31	10	7	70	5	26	18	15	214	9	1	22	19	98	11	15
6	3	38	3	42	10	7	62	5	27	18	15	206	9	2					
6	3	36	3	43	10	7	54	5	29	18	15	198	9	3					
6	3	28	3	45	10	7	46	5	30	18	15	190	9	3					

Table 7. Generalized partitioned t-defect, s-error-correcting linear block codes of length n = 255.

REFERENCES

[1] BELOV, I.B. and SHASHIN, A.M.: *Codes that Correct Triple Defects in Memory*. translated from Prob. Peredach. Info., vol. 13, no 4. Oct.-Dec. 1977, 62-65.

[2] BERLEKAMP, E.R.: *Algebraic Coding Theory*. New York: McGraw-Hill, 1968.

[3] BUSSCHBACH, P.: *Constructive Methods to solve the Problems of: s-surjectivity, conflict resolution, coding in defective memories*. Internal Report, Eindhoven University of Technology, Dec. 1984.

[4] CHEN, C.L.: *Linear Codes for Masking Memory Defects*. preprint.

[5] CHEN, C.L. and TANG, D.T.: *Iterative Exhaustive Pattern Generation for Logic Testing*. IBM J. Res. Develop., vol. 28, March 1984, 212-213.

[6] EL GAMAL, A.A. and HEEGARD, C.: *On the Capacity of Computer memory with Defects*. IEEE Trans. on Info. Theory, vol. IT-29, Sept. 1983, 731-739.

[7] HEEGARD. C.: *Partitioned Linear Block Codes for Computer Memory with "Stuck-at" Defects*. IEEE Trans. on Info. Theory, vol. IT-29, Nov. 1983, 831-842.

[8] KLEITMAN. D.J., *On subsets Contained in a Family of Non-commensurable Subsets of a Finite Set*. J. of Combinatorial Theory, vol. 7, 1969, 181-183.

[9] KLEITMAN, D.J. and SPENCER, J.: *Families of k-independent sets*. Discrete Mathematic, vol. 6, 1973, 255-262.

[10] KUSNETSOV, A.V.: *Private communication*.

[11] KUSNETSOV, A.V. and TSYBAKOV, B.S.: *Coding in a Memory with Defective Cells*. translated from Prob. Peredach. Info., vol. 10, no. 2, April-June, 1974, 52-60.

[12] van LINT, J.H.: *Introduction to Coding Theory*. New York Heidelberg Berlin: Springer, 1982.

[13] LOSEV, V.V., KONOPEL'KO, V.K. and DARYAKIN, Yu.D.: *Double-and Triple-Defect-Correcting Codes*. translated from Prob. Peredach. Info., vol. 14, no. 4, Oct.-Dec. 1978, 98-101.

[14] MacWILLIAMS, F.J. and SLOANE, N.J.A.: *The Theory of Error-correcting Codes*. Amsterdam-New York-Oxford: North Holland, 1977.

[15] MUEHLDORF, E.I. and SAVKAR, A.D.: *LSI Logic Testing-An Overview*. IEEE Trans. on Comput., vol. C-30, 1-7, Jan. 1981.

[16] van PUL, C.L.M.: *On bounds on Codes*. Master's Thesis, Eindhoven University of Technology, 1982.

[17] TANG, D.T. and WOO , L.S. : *Exhaustive Test pattern Generation with Constant Weight Vectors*. IEEE Trans. on Comput., vol. C-32, no.12, Dec. 1983.

[18] TSYBAKOV, B.S.: *Additive Group Codes for Defect Correction*. translated from Prob. Peredach. Info., vol. 11, no. 1, Jan.-March 1975, 111-113.

[19] TSYBAKOV, B.S.: *Defect and Error Correction*. translated from Prob. Peredach. Info. vol. 11, no. 13, July-Sept, 1975, 21-30.

[20] VERHOEFF, T.: *Updating a Table of Bounds on the Minimum Distance of Binary Linear Codes*. EUT Report 85-Wsk-01, Eindhoven University of Technology, 1985.

CHAPTER 2.

TWO CONSTRUCTIONS FOR CONSTANT WEIGHT CODES

2.1 INTRODUCTION

In this chapter we discuss two construction methods for constant weight codes, which improve several of the best known lower bounds on A(n,d,w) in [1,3,6], where A(n,d,w) denotes the maximal cardinality of any binary constant weight code of length n, minimum distance d and constant weight w. Although, our interest in the function A(n,d,w) finds its origin in the fact that this function plays an important rôle in the determination of upper bounds on A(n,d) (e.g.: Johnson bound, linear programming bound), the function A(n,d,w) is also interesting in its own right. Besides the obvious connection with t-designs and Hadamard matrices we would like to mention the application of constant weight codes as a set of protocol sequences for the multiple-acces collison channels without feedback [5,7].

The first construction method, treated in Section 2.2, results from proving a generalization of the well-known Johnson upper bound on A(n,d,w). Unlike the generalization, the resulting construction does improve several of the best known results on A(n,d,w) in [1,3,6]. A table of improved results is given.

In Section 2.3 we treat a construction method for constant weight codes with minimum distance 4. In order to make this construction work, one needs to partition the set V^n_w, $n,w \in \mathbb{N}$ with $w \leq n$, in an as small as possible number of constant weight codes with minimum distance 4. For $n = 6m+1$ or $6m+3$ and $w = 3$ this last problem is equivalent to that of determining a packing of Steiner triple systems of order n. Since the construction method results in many improvements of the lower bounds on

$A(n,4,w)$, $n \leqq 24$, given in [1,3,6], a revised table of the function $A(n,4,w)$, $n \leqq 24$, is included.

2.2 A GENERALIZATION OF THE JOHNSON BOUND FOR CONSTANT WEIGHT CODES

Let $A(n,2\delta,w)$ denote the maximum number of codewords in any binary code of length n, minimum distance 2δ and constant weight w. The following upper bound on $A(n,2\delta,w)$ is well known [4].

THEOREM 1. (Johnson bound)

$$A(n,2\delta,w) \leqq \left\lfloor \frac{n}{w} A(n-1,2\delta,w-1) \right\rfloor \leqq \frac{n}{w} A(n-1,2\delta,w-1). \qquad \square$$

Applying Theorem 1 k times we obtain the following bound:

$$A(n,2\delta,w) \leqq \left\lfloor \frac{n}{w} \left\lfloor \frac{n-1}{w-1} \right\lfloor \ldots \left\lfloor \frac{n-k+1}{w-k+1} A(n-k,2\delta,w-k \right\rfloor \ldots \right\rfloor \right\rfloor \right\rfloor \leq \frac{\binom{n}{k}}{\binom{w}{k}} A(n-k,2\delta,w-k). \quad (1)$$

At the International Workshop "Convolutional Codes Multi-User Communication" Zinoviev [8] presented the following generalization of the Johnson bound (1).

THEOREM 2. For any integers k and ℓ with $0 \leqq \ell \leqq k \leqq n$, the following inequality holds

$$A(n,2\delta,w) \leqq \frac{\binom{n}{k}}{\binom{w}{\ell}\binom{n-w}{k-\ell}} A(n-k,2u,w-\ell),$$

where $u = \delta - \ell$ if $\ell \leqq k/2$ and $u = \delta - k + \ell$ if $\ell > k/2$. $\square$

If we take $\ell = k$ in Theorem 2 we get the Johnson bound (1).

We now give a further improvement of the Johnson bound stated in the next theorem.

THEOREM 3. For any two integers k and ℓ with $0 < \ell \leq k < n$, we have

$$\text{i) } A(n,2\delta,w) \leq \frac{\binom{n}{k}}{\sum_{i=0}^{\ell}\binom{w}{i}\binom{n-w}{k-i}} A(n-k,2(\delta-\ell),w-\ell) \text{ if } \ell \leq k/2$$

and

$$\text{ii) } A(n,2\delta,w) \leq \frac{\binom{n}{k}}{\sum_{i=\ell}^{k}\binom{w}{i}\binom{n-w}{k-i}} A(n-k,2(\delta-k+\ell),w-\ell) \text{ if } \ell > k/2.$$

REMARK. Note that the denominators in Theorem 3 are greater than the corresponding denominators in Theorem 2.

PROOF. Let C be an $(n,2\delta,w)$ binary constant weight code with $|C| = A(n,2\delta,w)$ and let $\ell \leq \min\{\frac{k}{2},w\}$. For every binary vector $\underline{b}$ of length n and weight k (notation $\underline{b} \in V_k^n$) we define the code $C''_{\underline{b}}$ by

$$C''_{\underline{b}} := \{\underline{c} \oplus \underline{c} \circledast \underline{b} \mid \underline{c} \in C\,,\ \mathrm{wt}(\underline{c} \circledast \underline{b}) \leq \ell\},$$

where $\underline{c} \circledast \underline{b} := (c_1b_1, c_2b_2, \ldots, c_nb_n)$. To make things clear we give an example. Let $\underline{b}, \underline{c} \in \mathbb{F}_2^n$ be given by

$$\underline{b} = (\overbrace{1,1,\ldots,1,1,1,\ldots,1}^{k},\overbrace{0,0,\ldots,0,0,0,\ldots,0}^{n-k}),$$

$$\underline{c} = (\overbrace{1,1,\ldots,1}^{\leq \ell},0,0,\ldots,0,1,1,\ldots,1,0,0,\ldots,0).$$

Then

$$\underline{c} \oplus \underline{c} \circledast \underline{b} = (0,0,\ldots,0,0,0,\ldots,0,1,1,\ldots,1,0,0,\ldots,0).$$

From the above it will be clear that, for any two codewords $\underline{c}_1 \oplus \underline{c}_1 \circledast \underline{b}$ and $\underline{c}_2 \oplus \underline{c}_2 \circledast \underline{b}$ $(\underline{c}_1 \neq \underline{c}_2)$ of $C''_{\underline{b}}$ we have

$$d(\underline{c}_1 \oplus \underline{c}_1 \oplus \underline{b}\,,\, \underline{c}_2 \oplus \underline{c}_2 \oplus \underline{b}) = d(\underline{c}_1, \underline{c}_2) - d(\underline{c}_1 \oplus \underline{b}, \underline{c}_2 \oplus \underline{b}) \geq$$

$$\geq 2\delta - \mathrm{wt}(\underline{c}_1 \oplus \underline{b}) - \mathrm{wt}(\underline{c}_2 \oplus \underline{b}) .$$

So if, for every $\underline{c} \oplus \underline{c} \oplus \underline{b} \in C''_{\underline{b}}$, an arbitrary set of $\ell - \mathrm{wt}(\underline{c} \oplus \underline{b})$ coordinates that are one are changed into zeros, we get a code $C'_{\underline{b}}$ with constant weight $w - \ell$ and minimum distance at least $2(\delta - \ell)$.

From the definition of $C''_{\underline{b}}$ one easily sees that all codewords of $C''_{\underline{b}}$ have zeros where $\underline{b}$ has ones. So certainly all codewords of $C'_{\underline{b}}$ have zeros where $\underline{b}$ has ones. This means that we can puncture the code $C'_{\underline{b}}$ k times (delete the k coordinates where $\underline{b}$ has ones) to get a constant weight code $C_{\underline{b}}$ of length $n-k$, minimum distance at least $2(\delta - \ell)$ and constant weight $w - \ell$. Thus

$$|C''_{\underline{b}}| = |C'_{\underline{b}}| = |C_{\underline{b}}| \leq A(n-k, 2(\delta - \ell)\ , w - \ell)\ , \text{ for all } \underline{b} \in V^n_k .$$

We now show that there is a $\underline{b} \in V^n_k$ such that

$$|C_{\underline{b}}| \geq \frac{\sum\limits_{i=0}^{\ell} \binom{w}{i}\binom{n-w}{k-i}}{\binom{n}{k}} A(n, 2\delta, w) .$$

To do this, we calculate the number N of pairs $\{\underline{c},\underline{b}\}$ of the set $\{\{\underline{c},\underline{b}\} \mid \underline{c} \in C, \underline{b} \in V^n_k,\ \mathrm{wt}(\underline{c} \oplus \underline{b}) \leq \ell\}$ in two different ways.

Since, for any $\underline{c} \in C$, there are $\sum\limits_{i=0}^{\ell} \binom{w}{i}\binom{n-w}{k-i}$ vectors in V^n_k that have not more then ℓ ones in common with $\underline{c}$, we have

$$N = \sum_{\underline{c} \in C} \sum_{i=0}^{\ell} \binom{w}{i}\binom{n-w}{k-i} = A(n, 2\delta, w) \sum_{i=0}^{\ell} \binom{w}{i}\binom{n-w}{k-i} .$$

On the otherhand,this number also equals

$$N = \sum_{\underline{b} \in V_k^n} |C_{\underline{b}}|.$$

Hence we have

$$\sum_{\underline{b} \in V_k^n} |C_{\underline{b}}| = A(n,2\delta,w) \sum_{i=0}^{\ell} \binom{w}{i}\binom{n-w}{k-i} .$$

Hence, there is a $\underline{b} \in V_k^n$ with $|C_{\underline{b}}| \geqq (A(n,2\delta,w) / \binom{n}{k}) \sum_{i=0}^{\ell} \binom{w}{i}\binom{n-w}{k-i}$.

Together with $|C_{\underline{b}}| \leqq A(n-k,2\delta,w-\ell)$ this last inequality proves the first part of the theorem.

To prove the second inequality of the theorem, we apply the first one to the complementary code $\bar{C}$ of C, with $\ell' = k - \ell$ and k. This gives

$$A(n,2\delta,w) = A(n,2\delta,n-w) \leqq \frac{\binom{n}{k}}{\sum_{i=0}^{k-\ell} \binom{n-w}{i}\binom{w}{k-i}} \; A(n-k,2(\delta-k+\ell),n-w-k+\ell)$$

$$= \frac{\binom{n}{k}}{\sum_{j=\ell}^{k} \binom{w}{j}\binom{n-w}{k-j}} A(n-k),2(\delta-k+\ell),w-\ell) .$$

□

As in [8] the proof of the above theorem has an immediate consequence, which is stated in the next theorem.

<u>THEOREM 4</u>. Let there exists an $(n,2\delta,w)$ constant weight code with N codewords and let k,ℓ be arbitrary integers with $0 \leqq \ell \leqq k < n$. Then there exists an $(n-k,2u,w-\ell)$ constant weight code with N' codewords, where

$$u = \delta - \ell \text{ and } N' \geq \lceil N \sum_{i=0}^{\ell} \binom{w}{i}\binom{n-w}{k-i} \Big/ \binom{n}{k} \rceil \text{ if } \ell \leq k/2,$$

$$u = \delta - k + \ell \text{ and } N' \geq \lceil N \sum_{i=\ell}^{k} \binom{w}{i}\binom{n-w}{k-i} \Big/ \binom{n}{k} \rceil \text{ if } \ell > k/2.$$

PROOF. One of the codes $C_{\underline{b}}$, $\underline{b} \in V_k^n$ defined in the proof of Theorem 3 does the job. □

The next table contains the improved (according to the table in [6]) lower bounds on A(n,6,w) that follow from Theorem 4, using the constant weight codes formed by the codewords of weight 8 respectively 12 from the [24,12,8] Golay code. For completeness we also mention the lower bounds that can be found in [6] (second column) and the improvements given by A.V. Zinoviev in [8] (third column). Our improved results are stated in the fourth column. The values of k and ℓ, needed to obtain these results, are given in the last column.

A(n,2δ,w)	Upper and lower bounds from [6]	Lower bound from [8]	Lower bound from Th. 4	values of	
				k	ℓ
A(22,6,7)	675-1100	-	682	2	1
A(21,6,7)	465- 828	-	570	3	1
A(20,6,7)	310- 651	320	450	4	1
A(19,6,7)	228- 520	260	338	5	1
A(18,6,7)	160- 349	198	243	6	1
A(17,6,7)	119- 240	141	166	7	1
A(16,6,7)	90- 156	95	108	8	1
A(22,6,11)	1574-5064	-	1960	2	1
A(21,6,10)	1286-2702	-	1288	3	2
A(20,6,9)	736-1362	-	760	4	3
A(19,6,8)	332- 734	360	408	5	4
A(18,6,8)	232- 428	-	239 *	5	4

Table 1. Improved lower bounds on A(n,6,w).

To find the lower bound A(18,6,8) ≥ 239 (*) one uses Theorem 4, starting with the (23,1288,8,12) constant weight code, taking k = 5 and $\ell = 4$.

2.3 LOWER BOUNDS FOR A(n,4,w)

§ 2.3.1 Introduction

In this paper we describe a construction method for constant weight codes with minimum distance 4 that improves several of the known results in [1], [3] and [6]. The method is based on the following observation.

LEMMA 5. Let n_1 and n_2 be two positive integers and let $n = n_1 + n_2$. Then

$$A(n,4,w) \geq \max \left\{ \sum_{i=0}^{\lfloor\frac{w-\delta}{2}\rfloor} T(\delta + 2i , n_1 , w - \delta - 2i , n_2 , 4) \,\middle|\, \delta=0,1 \right\} ,$$

where $T(w_1,n_1,w_2,n_2,4)$ = the maximal number of codewords in any binary code of length n_1+n_2, minimum distance 4, with exactly w_1 1's in the first n_1 coordinate places and exactly w_2 1's in the last n_2 coordinate places (such a code will be denote by an $(n_1,n_2;4;w_1,w_2)$ code).

PROOF. Let $C(w_1,n_1,w_2,n_2)$ denote a binary $(n_1,n_2;4;w_1,w_2)$ code. The lemma is proved if we can show that the codes $C(0)$ and $C(1)$ defined by

$$C(\delta) := \bigcup_{i=0}^{\lfloor\frac{w-\delta}{2}\rfloor} C(2i+\delta , n_1 , w-2i-\delta , n_2) \ ,\delta = 0,1, \qquad (2)$$

both have minimum distance 4. We will prove this for $\delta = 0$.

Let $\underline{u} = (\underline{u}_1 \mid \underline{u}_2)$, $v = (\underline{v}_1 \mid \underline{v}_2)$, $\underline{u}_1$, $\underline{v}_1 \in \{0,1\}^{n_1}$ and $\underline{u}_2$, $\underline{v}_2 \in \{0,1\}^{n_2}$ be two distinct codewords of $C(0)$. Suppose that $wt(\underline{u}_1) = 2i$ and $wt(\underline{v}_1) = 2j$, so $wt(\underline{u}_2) = w-2i$ and $wt(\underline{v}_2) = w-2j$. Then if $i = j$ we have $\underline{u},\underline{v} \in C(2i , n_1 , w-2i , n_2)$ and so $d(\underline{u},\underline{v}) \geq 4$, while if $i \neq j$ we have

$d(\underline{u},\underline{v}) = d(\underline{u}_1,\underline{v}_1) + d(\underline{u}_2,\underline{v}_2) \geq |wt(\underline{u}_1) - wt(\underline{v}_1)| + |wt(\underline{u}_2) - wt(\underline{v}_2)| = |2i - 2j| + |w - 2i - w + 2j| = 4\ |i - j| \geq 4.$ □

From this it is clear that our construction method involves the construction of $(n_1,n_2;4;w_1,w_2)$ binary codes, which we treat in § 2.3.2.

§ 2.3.2 The construction of $(n_1,n_2;4;w_1,w_2)$ codes

Let V_w^n denote the set of all binary vectors of length n and weight w. $\{C^i(w,n)\}_{i=0}^{k-1}$ denotes a partition of V_w^n into k mutually disjoint constant weight codes, each with minimum distance 4 and constant weight w. Assume that the constant weight codes are numbered in such a way that $|C^0(w,n)| \geq |C^1(w,n)| \geq \ldots \geq |C^{k-1}(w,n)|$ holds. The construction of a $(n_1,n_2;4;w_1,w_2)$ code is as follows:

Let $\{C^i(w_1,n_1)\}_{i=0}^{k_1-1}$ and $\{C^i(w_2,n_2)\}_{i=0}^{k_2-1}$ be partitions of $V_{w_1}^{n_1}$ resp. $V_{w_2}^{n_2}$ as we have defined above. The code $C(w_1,n_1,w_2,n_2)$ is then defined by

$$C(w_1,n_1,w_2,n_2) := \bigcup_{i=0}^{\min\{k_1,k_2\}-1} C^i(w_1,n_1) \circledast C^i(w_2,n_2), \tag{3}$$

where $A \circledast B := \{(\underline{a}|\underline{b}) \mid \underline{a} \in A, \underline{b} \in B\}$.

<u>LEMMA 6</u>. The code $C(w_1,n_1,w_2,n_2)$ defined in (3) is a binary $(n_1,n_2;4;w_1,w_2)$ code. The number of codewords is given by

$$|C(w_1,n_1,w_2,n_2)| = \sum_{i=0}^{\min\{k_1,k_2\}-1} |C^i(w_1,n_1)| \cdot |C^i(w_2,n_2)| . \tag{4}$$

PROOF. We only proof that the minimum distance is 4. The rest then follows immediately. Let $\underline{u} = (\underline{u}_1 | \underline{u}_2) \in C^i(w_1,n_1) \circledast C^i(w_2,n_2)$ and $\underline{v} = (\underline{v}_1 | \underline{v}_2) \in C^j(w_1,n_1) \circledast C^j(w_2,n_2)$ be two distinct codewords of $C(w_1,n_1,w_2,n_2)$. Then there are two cases:

i) $i = j$, then $\underline{u}_1, \underline{v}_1 \in C^i(w_1,n_1)$, $\underline{u}_2, v_2 \in C^i(w_2,n_2)$ and $\underline{u}_1 \neq \underline{v}_1$ or $\underline{u}_2 \neq \underline{v}_2$. Hence $d(\underline{u},\underline{v}) = d(\underline{u}_1,\underline{v}_1) + d(\underline{u}_2,\underline{v}_2) \geqq 4$,

ii) $i \neq j$, then $\underline{u}_1 \neq \underline{v}_1$ and $\underline{u}_2 \neq \underline{v}_2$. Hence $d(\underline{u},\underline{v}) = d(\underline{u}_1,\underline{v}_1) + d(\underline{u}_2,\underline{v}_2) \geqq 2 + 2 = 4$. □

REMARK. In (3) we used the codes $C^i(w_1,n_1)$ and $C^i(w_2,n_2)$ to form the direct sum $C^i(w_1,n_2) \circledast C^i(w_2,n_2)$, $i = 0,1,\ldots, \min\{k_1,k_2\} - 1$. Other combinations are possible. However, from (3) and the assumption about the ordering of the codes in a partition, it follows that no other combination gives a larger code $C(w_1,n_1,w_2,n_2)$.

We are left with the problem of finding suitable partitions of V_w^n for $0 < w < n$. One way of solving this problem is to look at the construction method for constant weight codes that Graham and Sloane described in [3]. This method partitions V_w^n into n mutually disjoint constant weight codes with minimum distance 4, which gives a partition $\{C^i(w,n)\}_{i=0}^{n-1}$ of V_w^n $(0 < w < n)$. Using these partitions in (3), we find $(n_1,n_2;4;w_1,w_2)$ codes $C(w_1,n_1,w_2,n_2)$ with $|C(w_1,n_1,w_2,n_2)| \geq \binom{n_1}{w_1}\binom{n_2}{w_2} / n_1$, for every n_1,n_2,w_1 and w_2 with $n_1 \geqq n_2$, $0 < w_1 < n_1$ and $0 < w_2 < n_2$. Taking $n_1 = n_2 = n$ and $0 < w < 2n$ we find codes $C(0)$ and $C(1)$ with $|C(0)| + |C(1)| \geq \binom{2n}{w} / n$, from which we conclude that $A(2n,4,w) \geq \binom{2n}{w} / 2n$. This lower bound was also found by Graham and Sloane [3]. From the above it will be clear that we can expect to find better results if, for instance, we are able to find partitions of V_w^n into fewer than n mutually disjoint constant weight codes. The determination of such partitions is postponed to the appendix.

EXAMPLE 1. Let $n = 16$ and $w = 7$. From [3] we have $A(16,4,7) \geqq 715$. Taking $n_1 = n_2 = 8$ and using the partitions of $V_{w'}^8$, $w' = 0,1,2,\ldots,7$, as determined in the appendix, we find codes $C(2i,8,7-2i,8)$, $i=0,1,2,3$, with

$$|C(0,8,7,8)| = \sum_{i=0}^{0} |C^i(0,8)| \cdot |C^i(7,8)| = 1,$$

$$|C(2,8,5,8)| = \sum_{i=0}^{6} |C^i(2,8)| \cdot |C^i(5,8)| = 7.4.8 = 224,$$

$$|C(4,8,3,8)| = \sum_{i=0}^{5} |C^i(4,8)| \cdot |C^i(3,8)| =$$

$$= 2.14.8 + 2.12.8 + 10.8 + 8.8 = 560 \text{ and}$$

$$|C(6,8,1,8)| = \sum_{i=0}^{6} |C^i(6,8)| \cdot |C^i(1,8)| = 7.4.1 = 28.$$

Hence, for the code $C(0)$ defined in (2) we find

$$|C(0) = \sum_{i=0}^{3} |C(2i,8,7-2i,8)| = 813,$$

giving us the improved lower bound $A(16,4,7) \geqq 813$.

EXAMPLE 2. Let $n=19$ and $w=5$. From [6] we have $A(19,4,5) \geqq 612$. Take $n_1=9$ and $n_2=10$. With the help of the appendix, we find codes $C(2i+1,9,4-2i,10)$, $i=0,1,2$, with

$$|C(1,9,4,10)| = \sum_{i=0}^{8} |C^i(1,9)| \cdot |C^i(4,10)|$$

$$= 3.1.27 + 1.26 + 3.1.25 + 2.1.12 = 206,$$

$$|C(3,9,2,10)| = \sum_{i=0}^{6} |C^i(3,9)| \cdot |C^i(2,10)| = 7.12.5 = 420,$$

$$|C(5,9,0,10)| = \sum_{i=0}^{0} |C^i(5,9)| \cdot |C^i(0,10)| = 16.1 = 16.$$

Thus, for the code $C(1)$ defined by (2) we find

$$|C(1)| = \sum_{i=0}^{2} |C(2i+1,9,4-2i,10)| = 642.$$

Hence, we have $A(19,4,5) \geqq 642$.

However, since any constant weight code A of length 9, minimum distance 4 and constant weight 5, can be transformed into (9,10;4;5,0)-code by adding to each codeword of A a tail of 10 zeros, it is easy to find a (9,10;4;5,0)-code $C'(5,9,0,10)$ with $|C'(5,9,0,10)| = A(9,4,5) = 18$. Replacing $C(5,9,0,10)$ by $C'(5,9,0,10)$ in the above construction, gives

$$A(19,4,5) \geqq 644.$$

We conclude this paragraph with a revised table of lower and upper bounds for the function $A(n,4,w)$ in the range $n \leqq 24$. The entries in this table with an asterisk in the upper left corner are the improved lower bounds found by the above described method, using the partitions given in the appendix. On checking these entries the reader must be aware of the fact that we have used the trick explained in Example 2 several times. The entries without an asterisk are from [1],[3] and [6].

n \ w	2	3	4	5	6	7	8	9	10	11	12
4	2	1	1								
5	2	2	1	1							
6	3	4	3	1	1						
7	3	7	7	3	1	1					
8	4	8	14	8	4	1	1				
9	4	12	18	18	12	4	1	1			
10	5	13	30	36	30	13	5	1	1		
11	5	17	35	66	66	35	17	5	1	1	
12	6	20	51	75-84	132	75-84	51	20	6	1	1
13	6	26	65	118- -132	158- -182	158- -182	118- -132	65	26	6	1
14	7	28	91	169- -182	275- -308	316- -364	275- -308	169- -182	91	28	7
15	7	35	105	222- -271	370- -455	582- -660	582- -660	370- -455	222- -271	105	35
16	8	37	140	305- -336	592- -722	* 813- -1040	1164- -1320	813- -1040	592- -722	305- -336	140
17	8	44	154- -157	424- -476	* 854- -952	* 1320- -1753	1608- -2210	1608- -2210	1320- -1753	854- -952	424- -476
18	9	48	198	504- -565	1260- -1428	* 1936- -2448	* 2760- -3944	* 3150- -4420	2760- -3944	1936- -2448	1260- -1428
19	9	57	228	* 664- -752	* 1496- -1789	* 3024- -3876	* 4330- -5814	* 5490- -8326	5490- -8326	4330- -5814	3024- -3876
20	10	60	285	* 831- -912	* 2120- -2506	* 4103- -5111	* 7112- -9690	* 9197- -12920	* 10536- -16652	9197- -12920	7112- -9690
21	10	70	315	1071- -1197	2856- -3192	* 5708- -7518	* 10045- -13416	* 15143- -22610	* 18057- -27132	18057- -27132	15143- -22610
22	11	73	385	1386	3927- -4389	* 7888- -10032	* 15124- -20674	* 23458- -32794	* 32442- -49742	* 35136- -54264	32442- -49742
23	11	83	416- -419	1771	5313	* 11266- -14421	* 22530- -28842	* 38006- -52833	* 52978- -75426	* 62782- -104006	62782- -104006
24	12	88	498	* 1888- -2011	7084	* 15267- -18216	* 33795- -43263	* 56267- -76912	* 89816- -126799	* 105499- -164565	* 124052- -208012

Table 2. Bounds on A(n,4,w)

§ 2.3.3 An optimal (18,21,8,6)-code

We conclude this chapter with an (18,21,8,6)-code. From [6] we have that $A(18,8,6) \leqq 21$; so the code is optimal. The (18,21,8,6)-code is given by the matrix A below, the rows of which are the codewords.

$$A := \begin{bmatrix} \underline{j}_6 & \underline{0}_6 & \underline{0}_6 \\ \underline{0}_6 & \underline{j}_6 & \underline{0}_6 \\ \underline{0}_6 & \underline{0}_6 & \underline{j}_6 \\ A_1 & A_2 & A_3 \\ A_3 & A_1 & A_2 \\ A_2 & A_3 & A_1 \end{bmatrix} ,$$

where A_i, $1 \leqq i \leqq 3$ are circulant matrices with top row $\underline{a}_i$ given by

$$\underline{a}_1 = (101000) , \ \underline{a}_2 = (110000) \text{ and } \underline{a}_3 = (001100) .$$

It is easy to prove that the rows of A indeed form an (18,21,8,6)-code.

APPENDIX

In this appendix a partition of V_w^n is understood to be a partition of V_w^n into constant weight codes of length n, minimum distance 4 and constant weight w as defined in § 2.3.2. The partitions of V_w^n which we give here, are used in our construction method of Section 2.3 to find the results stated in Table 2.

DEFINITION: Let $\{C^i(w,n)\}_{i=0,1,\dots,k-1}$ be a partition of V_w^n. Then the number vector $\underline{K}_{w,n}$ of $\{C^i(w,n)\}_{i=0,1,\dots,k-1}$ is defined by

$$\underline{K}_{w,n} := \left(|C^0(w,n)|, |C^1(w,n)|, \dots, |C^{k-1}(w,n)|\right) .$$

For the determination of the number of codewords in the codes $\mathcal{C}(0)$ and $\mathcal{C}(1)$ of (2) and hence for the determination of a lower bound on A(n',4,w'), these number vectors are all we have to know. The next lemma gives the number vectors of partitions of V_w^n for $w \leq 2$. The proof is simple and is left to the reader.

LEMMA. For every $w \leq n$ there are partitions of V_w^n with number vectors satisfying:

$$\begin{aligned} &\text{i)}\ \underline{K}_{w,n} = \underline{K}_{n-w,n}, \\ &\text{ii)}\ \underline{K}_{0,n} = (1), \\ &\text{iii)}\ \underline{K}_{1,n} = \underline{1}_n , \\ &\text{iv)}\ \underline{K}_{2,n} = \begin{cases} (\frac{n-1}{2}) \cdot \underline{1}_n, & \text{if n is odd,} \\ (\frac{n}{2}) \cdot \underline{1}_{n-1}, & \text{if n is even.} \end{cases} \end{aligned}$$

□

In the following we give the partitions of V_w^n , $6 \leq n \leq 12$ and $3 \leq w \leq \frac{n}{2}$, of which the number vectors are given in Table 3. That these are indeed partitions of V_w^n as we have defined in § 2.3.2 is straight-forward to check.

In order to limit the amount of writing we frequently use the following notation. Let $\mathcal{C}$ be a binary code of length n with coordinate set $X = \{0,1,\dots,n-1\}$ and let p be a permutation of X. Then we denote by $p(\underline{c})$, $\underline{c} = (c_0, c_1, \dots, c_{n-1}) \in \mathcal{C}$ the vector

$$p(\underline{c}) := (c_{p^{-1}(0)}, c_{p^{-1}(1)}, \dots, c_{p^{-1}(n-1)})$$

and by $p(\mathcal{C})$ the code

$$p(\mathcal{C}) := \{p(\underline{c}) \mid \underline{c} \in \mathcal{C}\}.$$

n	6	k	number vector $\underline{K}_{w,n}$	see
6	3	6	(4,4,4,4,2,2)	A.1
7	3	6	(7,7,6,6,5,4)	A.1
8	3	7	(8,8,8,8,8,8,8)	A.2
8	4	6	(14,14,12,12,10,8)	A.1
9	3	7	(12,12,12,12,12,12,12)	A.2
9	4	8	(16,16,16,16,16,16,16,14)	A.3
10	3	10	(13,13,13,13,13,13,13,13,13,3)	A.4
10	4	10	(27,27,27,26,25,25,25,12,12,4)	A.5
10	5	10	(36,34,32,30,28,26,24,14,14,14)	A.6
11	3	11	(17,17,17,17,17,14,14,14,14,12,12)	A.7
11	4	11	(34,34,34,34,34,28,28,26,26,26,26)	A.7
11	5	11	(66,60,55,50,47,44,41,32,32,30,5)	A.6
12	3	11	(20,20,20,20,20,20,20,20,20,20,20)	A.2
12	4	11	(51,51,51,51,51,40,40,40,40,40,40)	A.7
12	5	12	(70,70,70,70,64,64,64,64,64,64,64,64)	A.7
12	6	11	(92,90,90,90,90,84,80,80,76,76,76)	A.7

Table 3. Number vectors

A.1 Partitions of V_3^6, V_3^7 and V_4^8

The six arrays below form a partition of V_3^7 into six mutually disjoint constant weight codes with minimum distance ≥ 4. The number vector of the partition is $\underline{K}_{3,7} = (7,7,6,6,5,4)$.

$C^0(3,7)$

1	1		1			
	1	1		1		
		1	1		1	
			1	1		1
1				1	1	
	1				1	1
1		1				1

$C^1(3,7)$

1	1				1	
	1	1				1
1		1	1			
	1		1	1		
		1		1	1	
			1		1	1
1				1		1

$C^2(3,7)$

1	1	1				
1					1	1
1			1	1		
	1			1	1	
	1		1			1
		1		1		1

$C^3(3,7)$

	1	1	1			
			1	1	1	
		1			1	1
1			1			1
1		1		1		
	1			1		1

$C^4(3,7)$

		1	1	1		
				1	1	1
1	1			1		
1		1			1	
	1		1		1	

$C^5(3,7)$

1	1					1
		1	1			1
	1	1			1	
1			1		1	

If we consider the vectors that have a zero in the 5th coordinate place we find, after deleting this coordinate, a partition of V_3^6 with number vector $\underline{K}_{3,6} = (4,4,4,4,2,2)$.

Since the distance between two distinct vectors in the same code $C^i(3,7)$ $(i = 0,1,\ldots,5)$ is exactly four, we may adjoin to each $C^i(3,7)$ $(i = 0,1,\ldots,5)$ the complements of its codewords. Adding the overall parity check then yields a partition of V_4^8 with number vector $\underline{K}_{4,8} = (14,14,12,12,10,8)$.

A.2 Partitions of V_3^8, V_3^9 and V_3^{12}

The problem of finding a partition of V_3^n. for $n = 6m + 1$ or $n = 6m + 3$ into a number (as small as possible) of mutually disjoint constant weight codes, is the same as trying to find a packing with Steiner triple systems of order n,(i.e., a partition of the set of triads of n elements into $n - 2$ disjoint Steiner triple systems). In [2] Denniston gives the solution of the above problem for eleven values of n, including $n = 13$. This solution is given on the following page and is used to determine a partition of V_3^{12} with number vector $\underline{K}_{3,12} = (20,20,20,20,20,20,20,20,20,20,20)$. One can also find several references to the above problem in [2].The existence of a packing of order 9 was found by Kirkman (see [2]) and rediscovered several times (also by us). This packing is given below. We use our terminology.

Let C be the constant weight code shown in Fig. 1 and let p_1 be the permutation $(0,1,2,3,4,5,6)(7)(8)$ (so $p_1(i) = i + 1 \pmod 7$, $i = 0,1,2,\ldots,6$, $p_1(7) = 7$ and $p_1(8) = 8$. Then we define the codes $C^i(3,9)$, $i = 0,1,\ldots,6$, by

$$C^i(3,9) := p_1^i(C), \qquad i = 0,1,\ldots,6.$$

These codes form a partition of V_3^9 with number vector $\underline{K}_{3,9} = (12,12,12,12,12,12,12)$.

The codewords with a zero in the last coordinate form, after deleting this coordinate, a partition of V_3^8 with number vector $\underline{K}_{3,8} = (8,8,8,8,8,8,8)$.

0	1	2	3	4	5	6	7	8
1							1	1
				1	1		1	
	1		1				1	
		1				1	1	
		1	1					1
				1		1		1
	1				1			1
1	1					1		
1		1			1			
1			1	1				
	1	1		1				
			1		1	1		

Fig. 1.

Let $\mathcal{D}$ be the code shown in Fig. 2 and let p_2 be the permutation $(0,1,2,3,4,5,6,7,8,9,10)(11)(12)$. Then the codes $C^i(3,13)$, $0 \leq i \leq 10$, defined below form a partiton of V_3^{13}.

$$C^i(3,13) := p_2^i(\mathcal{D}), \qquad i = 0,1,\ldots,10.$$

Shortening these codes give a partition of V_3^{12} with number vector $\underline{K}_{3,12} = (20,20,20,20,20,20,20,20,20,20,20)$.

0	1	2	3	4	5	6	7	8	9	10	11	12
1											1	1
	1	1									1	
				1		1					1	
							1			1	1	
					1				1		1	
			1					1			1	
							1	1				1
	1		1									1
		1			1							1
						1				1		1
				1					1			1
1	1					1						
1		1	1									
			1	1			1					
	1			1	1							
					1	1		1				
		1				1	1					
		1						1	1			
1									1	1		
		1		1						1		
			1		1					1		
1					1		1					
	1						1		1			
	1							1		1		
			1			1			1			
1				1				1				

Fig. 2.

A.3 A partition of V_4^9

Let C be the (9,16,4,4) code shown in Fig. 3 and let p be the permutation $(0,1,2,3,4,5,6)(7)(8)$. Then the partition $\{C^i(4,9)\}_{i=0,1,\ldots,7}$, with number vector $\underline{K}_{4,9} = (16,16,16,16,16,16,16,14)$ is defined by

$$\begin{aligned} \mathcal{C}^{i}\ (4,9) &:= p^{i}(\mathcal{C}), \quad i=0,1,\dots,6, \\ \mathcal{C}^{7}\ (4,9) &:= \{p^{i}((1,1,0,0,0,1,0 \mid 1,0)) \mid i=0,1,\dots,6\} \\ &\quad \cup \{p^{i}((1,1,0,1,0,0,0 \mid 0,1)) \mid i=0,1,\dots,6\}. \end{aligned}$$

$$\mathcal{C} = \begin{array}{ccccccc|cc} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ \hline & 1 & & 1 & & & & 1 & 1 \\ & & 1 & & & & 1 & 1 & 1 \\ & & & & 1 & 1 & & 1 & 1 \\ \hline 1 & 1 & 1 & & & & & 1 & \\ 1 & & & 1 & & & 1 & 1 & \\ & 1 & & & 1 & & 1 & 1 & \\ & & 1 & 1 & & 1 & & 1 & \\ \hline 1 & 1 & & & 1 & & & & 1 \\ 1 & & 1 & & & 1 & & & 1 \\ & & 1 & 1 & 1 & & & & 1 \\ & & & 1 & & 1 & 1 & & 1 \\ \hline 1 & 1 & & & & 1 & 1 & & \\ 1 & & 1 & & 1 & & 1 & & \\ 1 & & & 1 & 1 & 1 & & & \\ & 1 & 1 & & 1 & 1 & & & \\ & 1 & 1 & 1 & & & 1 & & \\ \hline \end{array}$$

Fig. 3.

A.4 A partition of V_3^{10}

We now give a partition of V_3^{10} with number vector $\underline{K}_{3,10} = (13,13,13,13,13,13,13,13,13,3)$. Let p be the permutation (0,1,2,3,4,5,6,7,8)(9) and let $\mathcal{D}_0$, $\mathcal{D}_1$, $\mathcal{D}_2$ and $\mathcal{D}_3$ be the constant weight codes shown in Fig. 4. Then the partition of V_3^{10} is given by

$$\mathcal{C}^{i+3j}(3,10) := p^{i}(\mathcal{D}_j), \quad i=0,1,2 \text{ and } j=0,1,2, \text{ and } \quad \mathcal{C}^{9}(3,10) := \mathcal{D}_3.$$

$$\mathcal{D}_0 = \begin{array}{|ccccccccc|c|}
0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\
\hline
& & & & & 1 & & & 1 & 1 \\
\hline
1 & & & & & & & 1 & & 1 \\
& 1 & & 1 & & & & & & 1 \\
& & & & 1 & & 1 & & & 1 \\
\hline
1 & & 1 & & 1 & & & & & \\
& & & 1 & & 1 & & 1 & & \\
& 1 & & & & & 1 & & 1 & \\
\hline
& & 1 & 1 & & & 1 & & & \\
1 & & & & & 1 & 1 & & & \\
1 & & & 1 & & & & & 1 & \\
\hline
& 1 & 1 & & & & & 1 & & \\
& 1 & & & 1 & 1 & & & & \\
& & & & 1 & & & 1 & 1 & \\
\hline
\end{array}$$

$$\begin{array}{|ccccccccc|c|}
0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\
\hline
& & 1 & & & & & & 1 & 1 \\
\hline
1 & & & & 1 & & & & & 1 \\
& & & 1 & & & & 1 & & 1 \\
& 1 & & & & & 1 & & & 1 \\
\hline
1 & 1 & & & & 1 & & & & \\
& & & 1 & 1 & & & & 1 & \\
& & 1 & & & & 1 & 1 & & \\
\hline
& 1 & 1 & & 1 & & & & & \\
& & & & 1 & 1 & & 1 & & \\
& 1 & & & & & & 1 & 1 & \\
\hline
1 & & & & & & 1 & & 1 & \\
1 & & 1 & 1 & & & & & & \\
& & & 1 & & 1 & 1 & & & \\
\hline
\end{array} = \mathcal{D}_1$$

$$\mathcal{D}_2 = \begin{array}{|ccccccccc|c|}
0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\
\hline
& & 1 & & & 1 & & & & 1 \\
\hline
1 & 1 & & & & & & & & 1 \\
& & & 1 & 1 & & & & & 1 \\
& & & & & & 1 & 1 & & 1 \\
\hline
& 1 & 1 & 1 & & & & & & \\
& & & & 1 & 1 & 1 & & & \\
1 & & & & & & & 1 & 1 & \\
\hline
& & 1 & & 1 & & & 1 & & \\
& 1 & & & & 1 & & 1 & & \\
& 1 & & & 1 & & & & 1 & \\
\hline
1 & & 1 & & & & 1 & & & \\
1 & & & 1 & & 1 & & & & \\
& & & 1 & & & 1 & & 1 & \\
\hline
\end{array}$$

$$\begin{array}{|ccccccccc|c|}
0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\
\hline
1 & & & 1 & & & 1 & & & \\
& 1 & & & 1 & & & 1 & & \\
& & 1 & & & 1 & & & 1 & \\
\hline
\end{array} = \mathcal{D}_3$$

Fig. 4.

A.5 A partition of V_4^{10}

Let C be the (10,27,4,4) constant weight code shown in Fig. 5 and let p be the permutation (0,1,2,3,4,5,6)(7)(8)(9). We define the codes $C^i(4,10)$, $0 \leq i \leq 6$, by ($\underline{c}_1$ and $\underline{c}_2$ are as defined in Fig. 5)

$C^0(4,10) := C$, $C^1(4,10) := p^2(C)$, $C^2(4,10) := p^4(C)$,

$C^3(4,10) := p^6(C) \setminus \{p^6(\underline{c}_2)\}$, $C^4(4,10) := p(C) \setminus \{p(\underline{c}_1), p(\underline{c}_2)\}$,

$C^5(4,10) := p^3(C) \setminus \{p^3(\underline{c}_1), p^3(\underline{c}_2)\}$ and

$C^6(4,10) := p^5(C) \setminus \{p^5(\underline{c}_1), p^5(\underline{c}_2)\}$.

Together with $C^7(4,10)$, $C^8(4,10)$ and $C^9(4,10)$, defined by the arrays shown in Fig. 5a, they define a partition of V_4^{10} with number vector $\underline{K}_{4,10} = (27,27,27,26,25,25,25,12,12,4)$.

$$C = \begin{array}{|ccccccc|ccc|l}
0&1&2&3&4&5&6&7&8&9& \\
\hline
 & & &1& & & &1&1&1& \\
\hline
1& &1& & & & & &1&1& \\
 &1& & &1& & & &1&1& \\
 & & & & &1&1& &1&1& \\
\hline
 & &1& &1& & &1& &1& \\
 &1& & & &1& &1& &1& \\
\hline
1&1& & & & & &1&1& & \\
 & &1& & &1& &1&1& & \\
 & & & &1& &1&1&1& & \\
\hline
1& &1& & & &1&1& & & \\
 &1&1&1& & & &1& & & \\
 & & &1& &1&1&1& & & \\
1& & &1&1& & &1& & & \\
\hline
1& & & &1&1& & &1& & \\
1& & &1& & &1& &1& & \\
 &1&1& & & &1& &1& & \\
 &1& &1& &1& & &1& & \\
 & &1&1&1& & & &1& & \\
\hline
1&1& &1& & & & & &1& \\
 & & &1&1&1& & & &1& \\
 & &1&1& & &1& & &1& \\
1& & & &1& &1& & &1& \\
\hline
1&1& & & &1&1& & & & \\
1&1&1& &1& & & & & & \\
 & &1& &1&1&1& & & & \\
1& &1&1& &1& & & & & =:\underline{c}_1 \\
 &1& &1&1& &1& & & & =:\underline{c}_2 = p(\underline{c}_1) \\
\hline
\end{array}$$

Fig. 5.

$$C^7(4,10) \qquad \begin{array}{|ccccccc|ccc|}
0&1&2&3&4&5&6&7&8&9 \\
\hline
1&1& & & & & &1& &1 \\
 & &1&1& & & &1& &1 \\
 & & & &1&1& &1& &1 \\
\hline
1& &1& &1& & &1& & \\
 &1& &1& &1& &1& & \\
 &1& & &1& &1&1& & \\
\hline
 & &1& &1& &1& & &1 \\
1& & &1& &1& & & &1 \\
 &1& &1& & &1& & &1 \\
\hline
1&1& &1&1& & & & & \\
 &1&1& &1&1& & & & \\
 & &1&1& &1&1& & & \\
\hline
\end{array}$$

$$C^8(4,10) \qquad \begin{array}{|ccccccc|ccc|}
0&1&2&3&4&5&6&7&8&9 \\
\hline
 &1&1& & & & &1& &1 \\
 & & &1&1& & &1& &1 \\
 & & & & &1&1&1& &1 \\
\hline
 & &1& &1& &1&1& & \\
1& &1& & &1& &1& & \\
 &1& &1& & &1&1& & \\
\hline
1& &1& &1& & & & &1 \\
 &1& &1& &1& & & &1 \\
 &1& & &1& &1& & &1 \\
\hline
1& & &1&1& &1& & & \\
1&1& & &1&1& & & & \\
 &1&1& & &1&1& & & \\
\hline
\end{array}$$

$$C^9(4,10) \qquad \begin{array}{|ccccccc|ccc|}
0&1&2&3&4&5&6&7&8&9 \\
\hline
1& & & & & &1&1& &1 \\
\hline
1& & &1& &1& &1& & \\
\hline
1& &1& & &1& & & &1 \\
\hline
1& &1&1& & &1& & & \\
\hline
\end{array}$$

Fig. 5a.

A.6 A partition of V_5^{10} and V_5^{11}

From [6] we have, that the rows of A (see the figure below) and the sums of pairs of rows of A form a (11,66,4,6) constant weight code.

$$A = \begin{array}{c}
\begin{array}{ccccccccccc} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \end{array} \\
\left[\begin{array}{ccccccccccc}
1 & 1 &   & 1 & 1 & 1 &   &   &   & 1 &   \\
  & 1 & 1 &   & 1 & 1 & 1 &   &   &   & 1 \\
1 &   & 1 & 1 &   & 1 & 1 & 1 &   &   &   \\
  & 1 &   & 1 & 1 &   & 1 & 1 & 1 &   &   \\
  &   & 1 &   & 1 & 1 &   & 1 & 1 & 1 &   \\
  &   &   & 1 &   & 1 & 1 &   & 1 & 1 & 1 \\
1 &   &   &   & 1 &   & 1 & 1 &   & 1 & 1 \\
1 & 1 &   &   &   & 1 &   & 1 & 1 &   & 1 \\
1 & 1 & 1 &   &   &   & 1 &   & 1 & 1 &   \\
  & 1 & 1 & 1 &   &   &   & 1 &   & 1 & 1 \\
1 &   & 1 & 1 & 1 &   &   &   & 1 &   & 1
\end{array}\right]
\end{array}$$

Taking the complements of all these codewords and permuting the coordinates $\{0,1,2,\dots,10\}$ with the permutation $(0,10)(1,7)(2,8,4)(3,6)(5,9)$ one finds the (11,66,5)-code $\mathcal{D}$ shown in Fig. 6 (the coordinates are renumbered).

Let $\mathcal{D}'$ be the subcode $\mathcal{D}' := \mathcal{D} \setminus \{\underline{c}_1, \underline{c}_2, \dots, \underline{c}_{28}\}$, where the $\underline{c}_i$, $i = 1,2,\dots,28$, are defined in Fig. 6. Let p be the permutation $(0,1,2,3,4,5,6)(7)(8)(9)(10)$, then we define the following partition of V_5^{11}.

$$\begin{aligned}
C^0(5,11) &:= \mathcal{D} = \mathcal{D}' \cup \{\underline{c}_1, \underline{c}_2, \dots, \underline{c}_{28}\}, \\
C^1(5,11) &:= p^2(\mathcal{D}') \cup \{p^2(\underline{c}_i) \mid i = 1,3,4,5,6,7,8,9,10,12,13,14,16,17, \\
&\qquad 18,19,20,22,23,24,25,27\}, \\
C^2(5,11) &:= p^5(\mathcal{D}') \cup \{p^5(\underline{c}_i) \mid i = 2,3,4,5,8,9,10,11,13,14,15,18, \\
&\qquad 20,21,23,24,26\}, \\
C^3(5,11) &:= p^4(\mathcal{D}') \cup \{p^4(\underline{c}_i) \mid i = 1,3,6,7,9,12,13,16,17,19,22,23\}, \\
C^4(5,11) &:= p^1(\mathcal{D}') \cup \{p(\underline{c}_i) \mid i = 1,2,11,12,15,16,21,22\} \cup \\
&\quad \cup \{(0,1,1,1,0,0,0 | 1,0,0,1)\}, \\
C^5(5,11) &:= p^3(\mathcal{D}') \cup \{p^3(\underline{c}_i) \mid i = 5,8,18,20\} \cup \\
&\quad \cup \{(0,1,1,0,0,1,0 | 0,1,0,1), (1,0,1,0,1,0,0 | 0,0,1,1)\},
\end{aligned}$$

$C^6(5,11) := p^6(\mathcal{D}') \cup \{(1,1,0,0,0,0,1 \mid 1,0,0,1),$
$(0,1,0,0,1,1,0 \mid 0,1,0,1), (1,0,0,1,0,1,0 \mid 0,0,1,1)\},$

$C^7(5,11)$, $C^8(5,11)$, $C^9(5,11)$ and $C^{10}(5,11)$ are defined by the arrays shown in Fig. 6a.

The number vector is $\underline{K}_{5,11} = (66,60,55,50,47,44,41,32,32,30,5)$.

0	1	2	3	4	5	6	7	8	9	10	
			1				1	1	1	1	
1					1			1	1	1	
	1			1				1	1	1	$=:\underline{c}_1$
		1				1		1	1	1	$=:\underline{c}_2$
				1	1		1		1	1	
	1					1	1		1	1	$=:\underline{c}_3$
1		1					1		1	1	$=:\underline{c}_4$
1				1			1	1		1	
	1	1					1	1		1	$=:\underline{c}_5$
					1	1	1	1		1	$=:\underline{c}_6$
1	1						1	1	1		
		1			1		1	1	1		
				1		1	1	1	1		
1		1				1	1	1			
	1			1	1		1	1			
		1	1	1			1	1			
1			1		1		1	1			$=:\underline{c}_7$
	1		1			1	1	1			$=:\underline{c}_8$
1					1	1	1		1		
	1	1		1			1		1		
	1		1		1		1		1		
		1	1			1	1		1		$=:\underline{c}_9$
1			1	1			1		1		$=:\underline{c}_{10}$
1		1		1				1	1		
1			1			1		1	1		
	1				1	1		1	1		
	1	1	1					1	1		$=:\underline{c}_{11}$
			1	1	1			1	1		$=:\underline{c}_{12}$
1	1		1						1	1	
	1	1			1				1	1	
		1	1	1					1	1	
			1		1	1			1	1	$=:\underline{c}_{13}$
1				1		1			1	1	$=:\underline{c}_{14}$

0	1	2	3	4	5	6	7	8	9	10	
1	1					1		1		1	
	1		1		1			1		1	
			1	1		1		1		1	
1		1	1					1		1	$=:\underline{c}_{15}$
		1		1	1			1		1	$=:\underline{c}_{16}$
1			1			1	1			1	
		1	1		1		1			1	
		1		1		1	1			1	
1	1				1		1			1	$=:\underline{c}_{17}$
	1		1	1			1			1	$=:\underline{c}_{18}$
1	1			1		1	1				
1		1		1	1		1				
	1	1			1	1	1				
1	1	1	1				1				$=:\underline{c}_{19}$
			1	1	1	1	1				$=:\underline{c}_{20}$
1	1	1			1			1			
1				1	1	1		1			
	1	1		1		1		1			
1	1		1	1				1			$=:\underline{c}_{21}$
		1	1		1	1		1			$=:\underline{c}_{22}$
1	1	1				1			1		
1	1			1	1				1		
		1		1	1	1			1		
1		1	1		1				1		$=:\underline{c}_{23}$
	1		1	1		1			1		$=:\underline{c}_{24}$
	1	1	1			1				1	
1	1	1		1						1	$=:\underline{c}_{25}$
1			1	1	1					1	$=:\underline{c}_{26}$
	1			1	1	1				1	$=:\underline{c}_{27}$
1		1			1	1				1	$=:\underline{c}_{28}$
1	1		1		1	1					
1		1	1	1		1					
	1	1	1	1	1						

Fig. 6.

$C^7(5,11)$

0	1	2	3	4	5	6	7	8	9	10
				1	1			1	1	1
	1	1						1	1	1
1						1		1	1	1
		1				1	1		1	1
1			1				1		1	1
	1			1			1		1	1
1		1					1	1		1
	1		1				1	1		1
		1		1	1		1	1		
	1	1				1	1	1		
1				1		1	1		1	
1	1				1		1		1	
1		1	1					1	1	
	1		1	1				1	1	
			1		1	1		1	1	
1		1			1				1	1
	1		1		1				1	1
1	1			1				1		1
				1	1	1		1		1
		1	1	1			1			1
1	1		1			1	1			
	1	1	1		1		1			
		1	1	1		1		1		
1		1			1	1		1		
1	1	1		1					1	
1			1	1	1				1	
	1			1	1	1			1	
1	1				1	1				1
1	1	1	1							1
		1	1		1	1				1
	1		1	1		1				1
1		1		1		1				1

$C^8(5,11)$

0	1	2	3	4	5	6	7	8	9	10
1	1							1	1	1
		1	1					1	1	1
					1	1		1	1	1
1				1			1		1	1
	1				1		1		1	1
			1		1		1	1		1
				1		1	1	1		1
1	1				1		1	1		
	1		1	1			1	1		
1		1	1				1		1	
		1		1	1		1		1	
			1		1	1	1		1	
1				1		1		1	1	
	1	1				1		1	1	
	1		1			1			1	1
		1		1		1			1	1
1			1	1				1		1
1					1	1	1			1
1	1	1					1			1
		1	1	1		1	1			
1			1	1	1		1			
	1			1	1	1	1			
1	1		1			1		1		
1	1	1		1				1		
1		1			1	1			1	
	1	1	1		1				1	
	1	1	1	1						1
			1	1	1	1				1
	1	1			1	1				1
1		1	1			1				1
1		1		1	1					1
1	1		1		1					1

$C^9(5,11)$

0	1	2	3	4	5	6	7	8	9	10
			1	1				1	1	1
		1			1		1		1	1
			1			1	1		1	1
1					1		1	1		1
	1					1	1	1		1
		1		1			1	1		1
1		1	1				1	1		
			1		1	1	1	1		
1				1		1	1	1		
	1	1				1	1		1	
	1		1	1			1		1	
1	1				1			1	1	
		1		1	1			1	1	
	1			1		1			1	1
		1			1	1		1		1
1			1			1		1		1
			1	1	1		1			1
1		1			1	1	1			
1	1	1		1			1			
	1	1	1		1			1		
1			1	1	1			1		
	1			1	1	1		1		
1	1		1			1			1	
		1	1	1		1			1	
1				1	1	1				1
1	1	1				1				1
1	1		1	1						1
	1	1		1	1					1
	1		1		1	1				1
1		1	1		1					1

$C^{10}(5,11)$

0	1	2	3	4	5	6	7	8	9	10
		1	1			1		1		1
		1	1	1	1					1
1	1			1	1					1
1			1	1		1				1
	1	1		1		1				1

Fig. 6a.

The codewords with a zero in the 10th coordinate form, after deleting, this coordinate, a partition of V_5^{10} with number vector $\underline{K}_{5,10} = (36,34,32,30,28,26,24,14,14,14)$.

A.7 Partitions found with help of the method described in Section 2.3

Since the partitions in Table 3, with A.7 as reference, are all found in the same way, we only give one,namely that of V_3^{11}. The others are left to the reader.

Let $\{C^i(w,5)\}_{i=0,1,\ldots,k_w-1}$ and $\{C^i(w,6)\}_{i=0,1,\ldots,\ell_w-1}$ be the partitions of V_w^5 $(w=0,1,2,3)$ respectively V_w^6 $(w=0,1,2,3)$ which we found in A.1 or just preceding. Then we define the following partition of V_3^{11},

$$C^i(3,11) := C^i(3,5) \circledast C^0(0,6)$$

$$\cup \left(\bigcup_{j=0}^{4} C^j(1,5) \circledast C^{(j+i) \bmod 5}(2,6) \right), \quad i=0,1,2,3,4,$$

$$C^{i+5}(3,11) := C^0(0,5) \circledast C^i(3,6)$$

$$\cup \left(\bigcup_{j=0}^{4} C^j(2,5) \circledast C^{(j+i) \bmod 6}(1,6) \right), \; i=0,1,\ldots,5.$$

From §2.3.2 it follows that every $C^i(3,11)$, $i=0,1,\ldots,10$, is a constant weight code of length 11, minimum distance $\geqq 4$ and constant weight 3. From this we also have

$$|C^i(3,11)| = 17, \quad i=0,1,2,3,4, \quad |C^i(3,11)| = 14, \quad i=5,6,7,8$$

and

$$|C^i(3,11)| = 12, \quad i=9,10.$$

One further easily sees that all words of weight 3 are different and so $\{C^i(3,11)\}_{i=0,1,\ldots,10}$ is a partition of V_3^{11} with number vector $\underline{K}_{3,11} = (17,17,17,17,17,14,14,14,14,12,12)$.

REFERENCES

[1] BEST, M.R., BROUWER, A.E., MacWILLIAMS, F.J., ODLYZKO, A.M. and SLOANE, N.J.A.: *Bounds for binary codes of length less than 25*. IEEE Trans. Info. Theory, vol. IT-24, Jan. 1978, 81-93.

[2] DENNISTON, R.H.F.: *Some packings with Steiner triple systems*. Discrete Math., vol. 9, 1974, 213-227.

[3] GRAHAM, R.L. and SLOANE, N.J.A.: *Lower bounds for constant weight codes*. IEEE Trans. Info. Theory, vol. IT-26, Jan. 1980, 37-43.

[4] JOHNSON, S.M.: *On upper bounds for unrestricted binary error-correcting codes*. IEEE Trans. Info. Theory, vol. IT-27, 1971, 446-478.

[5] KAUTZ, W.H. and SINGLETON, R.C.: *Nonrandom Binary Superimposed Codes*. IEEE Trans. Info. Theory, vol. IT-10, 1964, 363-377.

[6] MacWILLIAMS, F.J. and SLOANE, N.J.A.: *The Theory of Error-Correcting Codes*. Amsterdam - New York - Oxford : North-Holland, 1977.

[7] NGUYEN QUANG A., GYORFI, L. and MASSEY, J.L.: *Performances of protocol sets for collosion channel without feedback*. Submitted to IEEE Trans. Info. Theory, 1985.

[8] ZINOVIEV, V.A.: *On a generalization of the Johnson bound for constant weight codes*. Proc. of the International Workshop "Convolutional Codes Multi-user Communication", Sochi, 1983, 206-208.

CHAPTER 3

CONSTANT DISTANCE CODE PAIRS

3.1 INTRODUCTION

In this chapter we are concerned with a problem, formulated by Ahlswede, El Gmal and Pang [1] in 1984. They defined a constant distance code pair $(\mathcal{A},\mathcal{B})$ as a pair of binary codes of length n such that, for some $\delta \in \mathbb{N}$, $0 \leq \delta \leq n$,

$$\forall_{\underline{a}\in\mathcal{A}}\,\forall_{\underline{b}\in\mathcal{B}}\,[d(\underline{a},\underline{b})=\delta].$$

If $(\mathcal{A},\mathcal{B}), \mathcal{A},\mathcal{B} \subset \mathbb{F}_2^n$, is a code pair for which the above property holds, we write $\Delta(A,B) = \delta$. They were interested in the following function defined below

$$M(n,\delta) := \max\,\{|\mathcal{A}|\cdot|\mathcal{B}| \;|\; \mathcal{A}\subset\mathbb{F}_2^n,\, \mathcal{B}\subset\mathbb{F}_2^n,\, \Delta(\mathcal{A},\mathcal{B})=\delta\}.$$

In [1] Ahlswede, El Gamal and Pang proved the following upper bound on $M(n,\delta)$.

THEOREM 1.

$$M(n,\delta) \leq 2^{2\lfloor\frac{n}{2}\rfloor}, \text{ for all } n,\delta \in \mathbb{N} \text{ with } 0\leq\delta\leq n.$$

□

They gave the following examples, where equality holds in Theorem 1.

$$A_i := \{(0,0)\ ,\ (1,1)\}^{\lfloor\frac{n}{2}\rfloor} \otimes \{0\}^{\varepsilon}\ ,\quad \text{for } i=0,1,$$
$$B_0 := \{(0,1)\ ,\ (1,0)\}^{\lfloor\frac{n}{2}\rfloor} \otimes \{0\}^{\varepsilon},$$
$$B_1 := \{(0,1)\ ,\ (1,0)\}^{\lfloor\frac{n}{2}\rfloor} \otimes \{1\}^{\varepsilon},$$

where $\varepsilon = 0$ if n is even and $\varepsilon = 1$ if n is odd. One immediately sees that $A_i, B_i \subset \mathbb{F}_2^n$, $\Delta(A_i,B_i) = \lfloor\frac{n}{2}\rfloor + i\varepsilon$ and $|A_i| \cdot |B_i| = 2^{2\lfloor\frac{n}{2}\rfloor}$, $i=0,1$.

In [2] Hall and van Lint proved Theorem 1 using the observation that for an equidistant code pair (A,B), for any $\underline{a} \in A$ and $\underline{b} \in B$, the codes $\underline{a} \oplus A$ and $\underline{b} \oplus B$ are orthogonal even weight codes. Moreover, they proved that essentially the only code pairs for which equality holds in Theorem 1 are the ones given in the example above. To be more precise we need a definition.

Two code pairs (A,B) and (A',B'), $A,B,A',B' \subset \mathbb{F}_2^n$ are called equivalent if there exists a permutation σ of the positions of codewords and an $\underline{x} \in \mathbb{F}_2^n$ such that

$$(\underline{x} \oplus \sigma(A)\ ,\ \underline{x} \oplus \sigma(B)) = (A',B'),$$

where $\sigma(A) = \{(a_{\sigma(1)}, a_{\sigma(2)}, \ldots, a_{\sigma(n)}) \mid (a_1,a_2,\ldots,a_n) \in A\}$. In [2] Hall and van Lint proved that any code pair for which equality holds in Theorem 1 is equivalent to one of the code pairs given in the example above. Since for these examples $\delta = \lfloor\frac{n}{2}\rfloor$ or $\delta = \lceil\frac{n}{2}\rceil$, the question remained:"what is the exact value of $M(n,\delta)$, for $\delta \neq \lfloor\frac{n}{2}\rfloor$ or $\lceil\frac{n}{2}\rceil$?".

In this chapter we will give an answer to this question. In Section 3.2 we determine the exact value of $M(n,\delta)$ for all n and δ with $0 \leqq \delta \leqq n$. In Section 3.3 we additionaly characterize all constant distance code pairs (A,B) of length n and constant distance δ with

$$|A| \cdot |B| = M(n,\delta).$$

3.2 THE EXACT VALUE OF $M(n,\delta)$

From now on $\mathcal{A}$ and $\mathcal{B}$ will always denote two binary codes of length n such that $\Delta(\mathcal{A},\mathcal{B}) = \delta$ and $|\mathcal{A}| \cdot |\mathcal{B}| = M(n,\delta)$. Such a code pair is called optimal. The following lemma shows that without loss of generality we can assume $0 \leqq \delta \leqq \frac{n}{2}$.

LEMMA 2. For any $n,\delta \in \mathbb{N}$, $0 \leqq \delta \leqq n$, we have

$$M(n,\delta) = M(n,n-\delta) \quad .$$

PROOF. Let $(\mathcal{A},\mathcal{B})$ be a constant distance code pair with constant distance δ. Then obviously $\Delta(\underline{1} \oplus \mathcal{A},\mathcal{B}) = n - \delta$. The result follows. □

From now on we assume $0 \leqq \delta \leqq \frac{n}{2}$. The following examples give a lower bound on $M(n,\delta)$.

EXAMPLE 1.

$$A^{i}_{n,\delta} := \{(0,0)\ ,\ (1,1)\}^{i} \circledast \{0\}^{n-2i}$$

$$\mathcal{B}^{i}_{n,\delta} := \{(0,1)\ ,\ (1,0)\}^{i} \circledast V^{n-2i}_{\delta-i}, \text{ for } i = 0,1,2,..,\delta,$$

where $V^{n}_{w} := \{\underline{c} \in \mathbb{F}^{n}_{2} \mid wt(\underline{c}) = w\}$.

We have $A^{i}_{n,\delta}$, $\mathcal{B}^{i}_{n,\delta} \subset \mathbb{F}^{n}_{2}$, $\Delta(A^{i}_{n,\delta},\mathcal{B}^{i}_{n,\delta}) = \delta$ and $|A^{i}_{n,\delta}| \cdot |\mathcal{B}^{i}_{n,\delta}| = 2^{2i}\binom{n-2i}{\delta-i}$.

Hence

$$M(n,\delta) \geqq \max\ \{2^{2i}\binom{n-2i}{\delta-i} | 0 \leqq i \leqq \delta\}\ ,\quad 0 \leqq \delta \leqq \frac{n}{2}\ . \tag{1}$$

The following theorem states that this bound is tight

THEOREM 3. For all $n \in \mathbb{N}$, $\delta \in \mathbb{N}$ with $0 \leq \delta \leq \frac{n}{2}$, we have

$$M(n,\delta) = \max \{2^{2i}\binom{n-2i}{\delta-i} \mid 0 \leq i \leq \delta\}.$$

REMARK. One easily checks that

$$\max \{2^{2i}\binom{n-2i}{\delta-i} \mid 0 \leq i \leq \delta\} = \binom{n}{\delta}, \text{ if } n(n-1) \geq 4\delta(n-\delta).$$

For this reason the inequality $n(n-1) \geq 4\delta(n-\delta)$ will play an important rôle in the proof of Theorem 3.

Before we can prove Theorem 3 we need to do some preliminary work. Looking at the code pairs given in Example 1, it seems more or less natural to consider pairs of positions of codewords. That is why we define, for every i,j with $i,j \in \{1,2,\ldots,n\}$ and $i \neq j$,

$$\alpha_{ij} := \text{the number of pairs } \{\underline{a},\underline{b}\},\ \underline{a} \in A,\ \underline{b} \in B \text{ with } a_i + a_j + b_i + b_j \equiv 1 \bmod 2.$$

The condition $a_i + a_j + b_i + b_j \equiv 1 \bmod 2$ says that the positions i and j contribute exactly 1 to the Hamming distance between $\underline{a}$ and $\underline{b}$, i.e., $d((a_i,a_j),(b_i,b_j)) = 1$. Since $\Delta(A,B) = \delta$, we have

$$\sum_{0 \leq i<j \leq n} \alpha_{ij} = \delta(n-\delta)\ |A| \cdot |B| = \delta(n-\delta)M(n,\delta)\ .$$

It follows that there is an α_{ij} with $\alpha_{ij} \geq 2\delta(n-\delta)\ M(n,\delta)/n(n-1)$. Using a permutation of the positions of codewords we can take care that

$$\alpha_{12} \geq \frac{2\delta(n-\delta)M(n,\delta)}{n(n-1)}\ . \qquad (2)$$

We now try to find an upper bound on α_{12}. For this reason we partition the codes A and B as follows:

$$\mathcal{A} = \{(0,0)\} \otimes \mathcal{A}_{00} \cup \{(1,1)\} \otimes \mathcal{A}_{11} \cup \{(0,1)\} \otimes \mathcal{A}_{01} \cup \{(1,0)\} \otimes \mathcal{A}_{10}$$

and

$$\mathcal{B} = \{(0,0)\} \otimes \mathcal{B}_{00} \cup \{(1,1)\} \otimes \mathcal{B}_{11} \cup \{(0,1)\} \otimes \mathcal{B}_{01} \cup \{(1,0)\} \otimes \mathcal{B}_{10},$$

where $\mathcal{A}_{\varepsilon\mu}, \mathcal{B}_{\varepsilon\mu} \subseteq \mathbb{F}_2^{n-2}$, $\varepsilon, \mu \in \{0,1\}$. Notice that some of the sets $\mathcal{A}_{\varepsilon\mu}$, $\mathcal{B}_{\varepsilon\mu}$ may be empty. With this terminology we can write α_{12} as

$$\alpha_{12} = (|\mathcal{A}_{00}| + |\mathcal{A}_{11}|)(|\mathcal{B}_{01}| + |\mathcal{B}_{10}|) + (|\mathcal{A}_{01}| + |\mathcal{A}_{10}|)(|\mathcal{B}_{00}| + |\mathcal{B}_{11}|).$$

The following two lemmas are useful in finding an upper bound on α_{12}.

<u>LEMMA 4</u>. For every $\varepsilon, \mu \in \{0,1\}$ the following holds:

$$\text{if } \mathcal{A}_{\varepsilon\mu} \cap \mathcal{A}_{\bar{\varepsilon}\bar{\mu}} \neq \emptyset, \text{ then } \mathcal{B}_{\varepsilon\mu} = \mathcal{B}_{\bar{\varepsilon}\bar{\mu}} = \emptyset \text{ and } \mathcal{A}_{\varepsilon\mu} = \mathcal{A}_{\bar{\varepsilon}\bar{\mu}},$$

where $\bar{\varepsilon} \equiv 1 + \varepsilon \bmod 2$ and $\bar{\mu} \equiv 1 + \mu \bmod 2$. By symmetry, the rôles of the $\mathcal{A}_{\varepsilon\mu}$'s and $\mathcal{B}_{\varepsilon\mu}$'s are interchangeable.

<u>PROOF.</u> Without loss of generality we may take ε and μ equal to 0. Since $\mathcal{A}_{00} \cap \mathcal{A}_{11} \neq \emptyset$, there is an $\underline{a}' \in \mathbb{F}_2^{n-2}$ such that $(0,0|\underline{a}')$ and $(1,1|\underline{a}')$ both belong to $\mathcal{A}$. But, for any $\underline{b}' \in \mathbb{F}_2^{n-2}$ we then have

$$d((0,0|\underline{a}'), (0,0|\underline{b}')) = d((1,1|\underline{a}'), (0,0|\underline{b}')) - 2$$

and

$$d((0,0|\underline{a}'), (1,1|\underline{b}')) = d((1,1|\underline{a}'), (1,1|\underline{b}')) + 2.$$

So $\mathcal{B}_{00} = \mathcal{B}_{11} = \emptyset$.

From $\mathcal{B}_{00} = \mathcal{B}_{11} = \emptyset$ and $|\mathcal{A}| \cdot |\mathcal{B}| = M(n,\delta)$, it now easily follows that $(0,0|\underline{c}') \in \mathcal{A} \Leftrightarrow (1,1|\underline{c}') \in \mathcal{A}$. So $\mathcal{A}_{00} = \mathcal{A}_{11}$. □

The following lemma is obvious.

<u>LEMMA 5</u>. The code pairs $(\mathcal{A}_{00} \cup \mathcal{A}_{11}, \mathcal{B}_{01} \cup \mathcal{B}_{10})$ and $(\mathcal{A}_{01} \cup \mathcal{A}_{10}, \mathcal{B}_{00} \cup \mathcal{B}_{11})$

are constant distance code pairs of length $n-2$ and constant distance $\delta-1$. □

We are now ready to give an upper bound on α_{12} and so indirectly an upper bound on $M(n,\delta)$. We have to consider three cases, the first one of which is special.

<u>CASE I:</u> $A_{\varepsilon\mu}\cap A_{\bar{\varepsilon}\bar{\mu}}\neq\emptyset$ and $B_{\varepsilon\bar{\mu}}\cap B_{\bar{\varepsilon}\mu}\neq\emptyset$, for some $\varepsilon,\mu\in\{0,1\}$.
With Lemma 4 we then have

$$M(n,\delta) = |A|\cdot|B| = \alpha_{12} = (|A_{\varepsilon\mu}| + |B_{\bar{\varepsilon}\bar{\mu}}|)(|B_{\varepsilon\bar{\mu}}| + |B_{\bar{\varepsilon}\mu}|)$$

$$= 4|A_{\varepsilon\mu}|\cdot|B_{\varepsilon\bar{\mu}}|.$$

And so with Lemma 5

$$M(n,\delta) = \alpha_{12} \leqq 4\ M(n-2,\delta-1).$$

<u>CASE II:</u> $A_{\varepsilon\mu}\cap A_{\bar{\varepsilon}\bar{\mu}}\neq\emptyset$ and $B_{\varepsilon\bar{\mu}}\cap B_{\bar{\varepsilon}\mu}=\emptyset$, for some $\varepsilon,\mu\in\{0,1\}$. Exchanging the $A_{\varepsilon\mu}$'s and $B_{\varepsilon\mu}$'s gives an equivalent situation.
With Lemma 4 and 5 we then have

$$\alpha_{12} = (|A_{\varepsilon\mu}| + |A_{\bar{\varepsilon}\bar{\mu}}|)(|B_{\varepsilon\bar{\mu}}| + |B_{\bar{\varepsilon}\mu}|) =$$

$$= 2|A_{\varepsilon\mu}|\cdot|B_{\varepsilon\bar{\mu}}\cup B_{\bar{\varepsilon}\mu}| \leqq 2\ M(n-2\ ,\ \delta-1).$$

<u>CASE III:</u> $A_{\varepsilon\mu}\cap A_{\bar{\varepsilon}\bar{\mu}}=\emptyset$ and $B_{\varepsilon\mu}\cap B_{\bar{\varepsilon}\bar{\mu}}=\emptyset$, for all $\varepsilon,\mu\in\{0,1\}$.
Lemma 5 now gives

$$\alpha_{12} = (|A_{00}| + |A_{11}|)(|B_{01}| + |B_{10}|) + (|A_{01}| + |A_{10}|)(|B_{00}| + |B_{11}|) =$$

$$= |A_{00}\cup A_{11}|\cdot|B_{01}\cup B_{10}| + |A_{01}\cup A_{10}|\cdot|B_{00}\cup B_{11}| \leqq 2\ M(n-2\ ,\ \delta-1).$$

Together with (2) case II and case III give

$$M(n,\delta) \leqq \frac{n(n-1)\ M(n-2,\delta-1)}{\delta(n-\delta)} \quad .$$

Hence, we have proved the following inequality

$$M(n,\delta) \leqq \max\ \{4, \frac{n(n-1)}{\delta(n-\delta)}\}\ M(n-2,\delta-1), \quad 1 \leqq \delta \leqq \frac{n}{2} \quad . \qquad (3)$$

An induction argument now completes the proof of Theorem 3.

<u>PROOF OF THEOREM 3</u>. We use induction on δ. First note that the theorem is obviously true for $\delta = 0$ and all $n \in \mathbb{N}$.

Let $\delta \geqq 1$ and suppose that for all $n \in \mathbb{N}$ with $n-2 \geqq 2(\delta-1)$ the following equality holds

$$M(n-2\ ,\ \delta-1) = \max\ \{2^{2i}\binom{n-2-2i}{\delta-1-i} \mid 0 \leqq i \leqq \delta-1\}.$$

From (3) it follows that, for any $n \geqq 2\delta$,

$$M(n,\delta) \leqq \begin{cases} 4M(n-2\ ,\ \delta-1) \ \text{ if } n(n-1) \leqq 4\delta(n-\delta), \\ \\ \dfrac{n(n-1)}{\delta(n-\delta)}\ M(n-2\ ,\ \delta-1) \ \text{ if } n(n-1) \geqq 4\delta(n-\delta). \end{cases}$$

So we have to distinquish two cases.

First suppose $n(n-1) \leqq 4\delta(n-\delta)$. We then have

$$M(n,\delta) \leqq 4\ M(n-2,\delta-1) \leqq 4\ \max\ \{2^{2i}\binom{n-2-2i}{\delta-1-i} \mid 0 \leqq i \leqq \delta-1\}$$

$$= \max\ \{2^{2j}\binom{n-2j}{\delta-j} \mid 1 \leqq j \leqq \delta\} \leqq \max\{2^{2j}\binom{n-2j}{\delta-j} \mid 0 \leqq j \leqq \delta\}$$

Secondly, suppose $n(n-1) \geqq 4\delta(n-\delta)$. Then $(n-2)(n-3) = n(n-1) - 4(n-1) + 1 > 4\delta(n-\delta) - 4(n-1) = 4(\delta-1)(n-1-\delta)$. From the remark directly below Theorem 3 we have

$$\max\ \{2^{2i}\binom{n-2-2i}{\delta-1-i}\,|\,0 \leq i \leq \delta-1\} = \binom{n-2}{\delta-1}.$$

So

$$M(n,\delta) \leq \frac{n(n-1)}{\delta(n-\delta)} M(n-2,\delta-1) = \frac{n(n-1)}{\delta(n-\delta)}\binom{n-2}{\delta-1}$$

$$= \binom{n}{\delta} \leq \max\ \{2^{2i}\binom{n-2i}{\delta-i}\,|\,0 \leq i \leq \delta\}.$$

Together with (1) both cases give

$$M(n,\delta) = \max\ \{2^{2i}\binom{n-2i}{\delta-i}\,|\,0 \leq i \leq \delta\}.$$

□

3.3 OPTIMAL CONSTANT DISTANCE CODE PAIRS

In this section we shall prove that the code pairs of Example 1 are essentially the only optimal constant distance code pairs. The observation at the beginning of Section 3.2 shows us that we only need to consider the case $2\delta \leq n$. So we assume $2\delta \leq n$. In the following (A,B) is an optimal constant distance code pair with constant distance δ. The lemma below deals with a simple case.

<u>LEMMA 6.</u> If $|A| \leq 2$ or $|B| \leq 2$, then (A,B) is equivalent to $(A^0_{n,\delta}, B^0_{n,\delta})$ or $(A^1_{n,\delta}, B^1_{n,\delta})$ defined in Section 3.2.

<u>PROOF.</u> Without loss of generality we may assume that $|A| \leq 2$. If $|A| = 1$ then $B = \underline{a} \oplus V^n_\delta$ with $\{\underline{a}\} = A$ and hence (A,B) is equivalent to $(A^0_{n,\delta}, B^0_{n,\delta})$.

So suppose $|A| = 2$. Then $A = \{\underline{a}_1, \underline{a}_2\}$ with $d(\underline{a}_1, \underline{a}_2) = 2\lambda$, for some $\lambda \in \mathbb{N}$ with $1 \leq \lambda \leq \delta$. Since (A,B) is easily seen to be equivalent to $(A^1_{n,\delta}, B^1_{n,\delta})$ if $d(\underline{a}_1, \underline{a}_2) = 2$, we only need to prove $\lambda = 1$.

Counting the number of words $\underline{b} \in \mathbb{F}^n_2$ with $d(\underline{a}_1, \underline{b}) = d(\underline{a}_2, \underline{b}) = \delta$ we find $|B| = \binom{2\lambda}{\lambda}\binom{n-2\lambda}{\delta-\lambda}$. Hence with Theorem 3

$$\max \{2^{2i}\binom{n-2i}{\delta-i} \mid 0 \leqq i \leqq \delta\} = M(n,\delta) = |A| \cdot |B| = 2\binom{2\lambda}{\lambda}\binom{n-2\lambda}{\delta-\lambda}$$

$$\leqq 2^{2\lambda}\binom{n-2\lambda}{\delta-\lambda} \leqq \max \{2^{2i}\binom{n-2i}{\delta-i} \mid 0 \leqq i \leqq \delta\}.$$

So equality must occur everywhere, which implies $\lambda = 1$. □

As a consequence of Lemma 5 we have:

<u>COROLLARY 7.</u> If $\delta = 1$, then (A,B) is equivalent to either $(A^0_{n,1}, B^0_{n,1})$ or $(A^1_{n,1}, B^1_{n,1})$ defined in Section 3.2.

<u>PROOF</u>. If $\delta = 1$, then one "easily" sees that $|A| \leqq 2$ or $|B| \leqq 2$. □

The following Lemma is very useful in proving Theorem 9.

<u>LEMMA 8</u>. Using the notation of Section 3.2 we have, for every $\varepsilon,\mu \in \{0,1\}$ and $\underline{x} \in \mathbb{F}_2^{n-2}$:

$$\text{if } A_{\varepsilon\mu} \cup A_{\bar{\varepsilon}\bar{\mu}} = \underline{x} \oplus V^{n-2}_{\delta-1}, \text{ then } |B_{\varepsilon\mu}| + |B_{\bar{\varepsilon}\bar{\mu}}| = |B_{\varepsilon\mu} \cup B_{\bar{\varepsilon}\bar{\mu}}| \leqq 1,$$

provided $n(n-1) \geqq 4\delta(n-\delta)$ and $\delta \geqq 2$. The same holds if we interchange the $A_{\varepsilon\mu}$'s and $B_{\varepsilon\mu}$'s.

<u>PROOF</u>. Without loss of generality we may take $\varepsilon = \mu = 0$ and $\underline{x} = \underline{0}$. With Lemma 4 we have $B_{00} \cap B_{11} = \emptyset$. If $B_{00} \cup B_{11} = \emptyset$ there is nothing to be proved. So, let $\underline{b}' \in B_{00} \cup B_{11}$. Then, for any $\underline{a}' \in A_{00} \cup A_{11} = V^{n-2}_{\delta-1}$ we have $d(\underline{a}',\underline{b}') = \delta - 2$ or δ. Since, $n(n-1) \geqq 4\delta(n-\delta)$ implies $n-2 \geqq 2(\delta-1)+2$ the above observation gives us $wt(\underline{b}') = 1$. So $V^{n-2}_{\delta-1}$ is partitioned into A_{00} and A_{11} (by Lemma 4, $A_{00} \cap A_{11} = \emptyset$), where A_{00} and A_{11} are given by

$$A_{00} = \{\underline{a}' \in V^{n-2}_{\delta-1} \mid (\underline{a}',\underline{b}')=1\} \text{ and } A_{11} = \{\underline{a}' \in V^{n-2}_{\delta-1} \mid (\underline{a}',\underline{b}') = 0\} \text{ if } \underline{b}' \in B_{11}$$

or

$$A_{00} = \{\underline{a}' \in V^{n-2}_{\delta-1} \mid (\underline{a}',\underline{b}')=\underline{0}\} \text{ and } A_{11} = \{\underline{a}' \in V^{n-2}_{\delta-1} \mid (\underline{a}',\underline{b}') = 1\} \text{ if } \underline{b}' \in B_{00}.$$

Since any other $\underline{b}'' \in \mathbb{F}_2^{n-2}$ with $\mathrm{wt}(\underline{b}'') = 1$ involves a similar but different partition of $V^{n-2}_{\delta-1}$, we have $|B_{00} \cup B_{11}| \leqq 1$. □

We are now ready to give the characterization of all optimal constant distance code pairs.

THEOREM 9. Any optimal constant distance code pair of length n and constant distance δ, $2\delta \leqq n$, is equivalent to one of the code pairs $(A^i_{n,\delta}, B^i_{n,\delta})$, $i = 0,1,\ldots,\delta$, defined in Section 3.2.

PROOF. We use induction on δ. With Corollary 7 we have that the theorem holds for $\delta \leqq 1$. Suppose the theorem is true for $\delta - 1 \geqq 1$ and let (A,B) be an optimal constant distance code pair of length n and constant distance δ. Without loss of generality we may assume that (see Section 3.2)

$$\alpha_{12} \geqq \frac{2\delta(n-\delta)\ M(n,\delta)}{n(n-1)} \quad .$$

As in the proof of Theorem 3 we consider three cases.

CASE I: $A_{\varepsilon\mu} \cap A_{\bar\varepsilon\bar\mu} \neq \emptyset$ and $B_{\bar\varepsilon\mu} \cap B_{\varepsilon\bar\mu} \neq \emptyset$, for some $\varepsilon,\mu \in \{0,1\}$.
Then $M(n,\delta) = \alpha_{12} = (|A_{\varepsilon\mu}| + |A_{\bar\varepsilon\bar\mu}|)(|B_{\bar\varepsilon\mu}| + |B_{\varepsilon\bar\mu}|) = 4|A_{\varepsilon\mu}| \cdot |B_{\bar\varepsilon\mu}| = 4M(n-2,\delta-1)$.
And so with Lemma 4,5 and the induction hypothesis we have (A,B) is equivalent to $(A^i_{n,\delta}, B^i_{n,\delta})$ for some $i \in \{1,2,\ldots,\delta\}$.

CASE II: $A_{\varepsilon\mu} \cap A_{\bar\varepsilon\bar\mu} \neq \emptyset$ and $B_{\varepsilon\bar\mu} \cap B_{\bar\varepsilon\mu} = \emptyset$, for some $\varepsilon,\mu \in \{0,1\}$.
From Section 3.2 we then have $n(n-1) \geqq 4\delta(n-\delta)$, $A_{\varepsilon\mu} = A_{\bar\varepsilon\bar\mu}$, $B_{\varepsilon\mu} = B_{\bar\varepsilon\bar\mu} = \emptyset$

and $\alpha_{12} = 2|\mathcal{A}_{\varepsilon\mu}| \cdot |\mathcal{B}_{\bar\varepsilon\mu} \cup \mathcal{B}_{\varepsilon\bar\mu}| = 2M(n-2,\delta-1)$. Since, $n(n-1) \geqq 4\delta(n-\delta)$ implies $(n-2)(n-3) \geqq 4(\delta-1)(n-\delta-1)$ Lemma 5 and the induction hypothesis give, either

$$\mathcal{A}_{\varepsilon\mu} = \mathcal{A}_{\bar\varepsilon\bar\mu} = \{\underline{x}\},\ \underline{x} \in \mathbb{F}_2^{n-2} \text{ and } \mathcal{B}_{\varepsilon\bar\mu} \cup \mathcal{B}_{\bar\varepsilon\bar\mu} = \underline{x} \oplus V_{\delta-1}^{n-2} \tag{4}$$

or

$$\mathcal{B}_{\varepsilon\bar\mu} \cup \mathcal{B}_{\bar\varepsilon\mu} = \{\underline{x}\},\ \underline{x} \in \mathbb{F}_2^{n-2} \text{ and } \mathcal{A}_{\varepsilon\mu} = \mathcal{A}_{\bar\varepsilon\bar\mu} = \underline{x} \oplus V_{\delta-1}^{n-2}\ . \tag{5}$$

With Lemma 8, (4) gives $\binom{n}{\delta} = M(n,\delta) = |\mathcal{A}| \cdot |\mathcal{B}| \leq 3\binom{n-2}{\delta-1}$, which contradicts $n(n-1) \geqq 4\delta(n-\delta)$. So (5) must hold. But then $|\mathcal{B}| = 1$ (Lemma 8) and so with Lemma 5, $(\mathcal{A},\mathcal{B})$ equivalent to $(\mathcal{A}^0_{n,\delta},\ \mathcal{B}^0_{n,\delta})$.

<u>CASE III</u>: $\mathcal{A}_{\varepsilon\mu} \cap \mathcal{A}_{\bar\varepsilon\bar\mu} = \emptyset$ and $\mathcal{B}_{\varepsilon\mu} \cap \mathcal{B}_{\bar\varepsilon\bar\mu} = \emptyset$, for all $\varepsilon,\mu \in \{0,1\}$.

From Section 3.2 we have $n(n-1) \geqq 4\delta(n-\delta)$ and $\alpha_{12} = |\mathcal{A}_{00} \cup \mathcal{A}_{11}| \cdot |\mathcal{B}_{01} \cup \mathcal{B}_{10}| + |\mathcal{A}_{01} \cup \mathcal{A}_{10}| \cdot |\mathcal{B}_{00} \cup \mathcal{B}_{11}| = 2M(n-2,\delta-1)$. Lemma 5 and the induction hypothesis now give that $(\mathcal{A}_{00} \cup \mathcal{A}_{11}, \mathcal{B}_{01} \cup \mathcal{B}_{10})$ and $(\mathcal{A}_{01} \cup \mathcal{A}_{10}, \mathcal{B}_{00} \cup \mathcal{B}_{11})$ are equivalent to $(\mathcal{A}^0_{n-2,\delta-1}, \mathcal{B}^0_{n-2,\delta-1})$, $((n-2)(n-3) > 4(\delta-1)(n-\delta-1))$. So without loss of generality we may assume that $\mathcal{A}_{00} \cup \mathcal{A}_{11} = \{\underline{0}\}$ and $\mathcal{B}_{01} \cup \mathcal{B}_{10} = V_{\delta-1}^{n-2}$. But then with Lemma 8, $|\mathcal{A}_{01} \cup \mathcal{A}_{10}| = 1$, so that $|\mathcal{A}| = 2$. Hence, with Lemma 6, $(\mathcal{A},\mathcal{B})$ is equivalent $(\mathcal{A}^1_{n,\delta}, \mathcal{B}^1_{n,\delta})$.

$\square$

REFERENCES

[1] AHLSWEDE, R., EL GAMAL, A. and PANG, K.F.: *A Two-family Extremal Problem in Hamming-space*.Discrete Math. vol. 49,1984, 1-5.

[2] HALL, J.I. and van LINT, J.H.: *Constant Distance Code Pairs.* Proc. Kon. Ned. Akad. v. Wetenschappen vol. A 88, 1985, 41-45.

CHAPTER 4

TOURNAMENT CODES

4.1 INTRODUCTION

In this chapter we discuss a problem which arose in connection with comma-free codes. A q-ary code $\mathcal{D}$ of length n is said to be comma-free if, for every pair of words $(a_1,a_2,\ldots,a_n)$ and $(b_1,b_2,\ldots,b_n)$ of $\mathcal{D}$ the words $(a_k,a_{k+1},\ldots,a_n,b_1,\ldots,b_{k-1})$, $(k=2,3,..,n)$, are not in $\mathcal{D}$. Comma-free codes were first introduced by Grick, Griffith and Orgel [5] as a possible genetic coding scheme for protein synthesis. The general mathematical setting of such codes was presented by Golomb, Gordon and Welch in [3]. They considered the problem of finding the maximal cardinality of such a code.

Let $W_n(q)$ denote the maximal number of codewords in any q-ary comma-free code of length n. From the definition of a comma-free code $\mathcal{D}$ we have that no two codewords of $\mathcal{D}$ are a cyclic permutation of each other and every codeword $\underline{a}=(a_1,a_2,..,a_n)$ of $\mathcal{D}$ is non-periodic, i.e., there is no i, $0<i<n$, such that

$$(a_{i+1},a_{i+2},..,a_n,a_1,..,a_i) = (a_1,a_2,..,a_n).$$

Hence

$$W_n(q) \leqq B_n(q) ,$$

where

$$B_n(q) = \frac{1}{n} \sum_{d|n} \mu(d) q^{n/d} \qquad (1)$$

is the number of non-periodic cyclic equivalence classes of sequences of length n formed from an alphabet of q letters. The summation in (1) is taken over all divisors d of n and $\mu(d)$ is the Möbius function. In [3] Golomb, Gordon and Welch proved that $W_n(q)$ attains the upper bound $B_n(q)$

for arbitrary q if $n=1,3,5,7,9,11,13,15$ and conjectured that this is indeed the case for all odd n. This conjecture was first proved by Eastman [2], who gave a construction for maximal comma-free codes of odd wordlength. A simpler construction for these codes was found by Scholtz [8].

For comma-free codes of even length, the situation becomes surprisingly complicated. It was proved by Golomb, Gordon and Welch [3] that $W_n(q) < B_n(q)$ if $q > 3^{n/3}$. In particular $W_2(q) = \lfloor \frac{q^2}{3} \rfloor < B_2(q) = \binom{q^2}{2}$, $W_4(q) = B_4(q)$ if $q=1,2,3$ and $W_4(q) < B_4(q)$ if $q \geqq 5$. The case $n=4$ and $q=4$ was later solved in [6] by exhaustive computer search, which found $W_4(4) = 57 < B_4(4)$.

An improvement on the relation between k and n such that $W_n(q) < B_n(q)$ for even n was given by Jiggs [6]:

$$W_n(q) < B_n(q) \text{ if } q > 2^{n/2} + n/2.$$

A further improvement based on Jiggs' proof was given by Golomb and Tang [4]:

$$W_n(q) < B_n(q) \text{ if } q > (n/2)^{c \log n/2} + n/2, \ n \geqq 8,$$

where $c = (\ln 2)/0.71$. In Section 4.3 we give a proof of this result for $c = 0.5$. Moreover the proof is much simpler than that of Golomb and Tang [4]. We first present Jiggs' result. The modifications are due to Golomb and Tang.

We consider the simpler problem of finding the maximal cardinality of a q-ary comma-free code $\mathcal{D}'$ of length $n = 2k$ ($k \in \mathbb{N}$) in which every word is a cyclic shift of a word of the form $(a,0,0,\ldots,0,b,0,\ldots,0)$, where a and b are two different symbols of the alphabet separated by $n/2-1$ zeros. So $|\mathcal{D}'| \leqq \binom{q}{2}$. Clearly, if $|\mathcal{D}'| < \binom{q}{2}$, then $W_n(q) < B_n(q)$.

A half-word in $\mathcal{D}'$ is a k-tuple which is either the initial or final half of some word in $\mathcal{D}'$. For each symbol d of the alphabet and $r \in \mathbb{N}$, $1 \leqq r \leqq k$, let $\underline{u}(d,r)$ denote the half-word with d at the r-th position and 0 everywhere else. The half-word $\underline{u}(d,r)$ is called initial resp. final if it equals the initial half resp. final half of some word in $\mathcal{D}'$. To each symbol d we assign a word $\underline{x}^d=(x_1^d,x_2^d,\dots,x_k^d) \in \{0,1,2,*\}^k$, where x_r^d is defined in the following way

$$x_r^d := \begin{cases} 2 \text{ if } \underline{u}(d,r) \text{ is both initial and final,} \\ 1 \text{ if } \underline{u}(d,r) \text{ is final only,} \\ 0 \text{ if } \underline{u}(d,r) \text{ is initial only,} \\ * \text{ if } \underline{u}(d,r) \text{ is neither initial nor final.} \end{cases}$$

<u>EXAMPLE</u>. Let $q=5$, $n=2k=4$ and let $\mathcal{D}'$ be given by

$$\mathcal{D}' = \begin{bmatrix} 1 & 0 & 2 & 0 \\ 1 & 0 & 3 & 0 \\ 1 & 0 & 4 & 0 \\ 1 & 0 & 5 & 0 \\ 2 & 0 & 3 & 0 \\ 2 & 0 & 4 & 0 \\ 2 & 0 & 5 & 0 \\ 0 & 3 & 0 & 4 \\ 0 & 3 & 0 & 5 \\ 0 & 4 & 0 & 5 \end{bmatrix} .$$

Then

$$\underline{x}^1 = (0,*) \ , \ \underline{x}^2 = (2,*) \ , \ \underline{x}^3 = (1,0)$$
$$\underline{x}^4 = (1,2) \ , \ \underline{x}^5 = (1,1) .$$

Jiggs showed that the words $\underline{x}^d$ have the following two properties if $|\mathcal{D}'| = \binom{q}{2}$:

(i) If $d \neq b$, then x_r^d and x_r^b cannot both be 2, for any $1 \leq r \leq k$.

(ii) If $d \neq b$, there exists an r, $1 \leq r \leq k$, such that $(x_r^d,\ x_r^b) = (0,1)$ or $(x_r^d,\ x_r^b) = (0,1)$.

(In particular distinct letters of the alphabet must have distinct words).

The first property implies that the number of distinct words $\underline{x}^d$ containing a 2 is smaller than or equal to k while the second one implies that the number of words $\underline{x}^d$ containing no 2 is smaller than or equal to 2^k. So, if $|\mathcal{D}'| = \binom{q}{2}$, then $q \leq 2^{n/2} + n/2$. Hence $W_n(q) < B_n(q)$ if $q > 2^{n/2} + n/2$.

The improvement of Jiggs' results by Golomb and Tang is a consequence of the following observation.

<u>THEOREM 1</u>. If $|\mathcal{D}'| = \binom{q}{2}$, then for every two different symbols b, d of the alphabet and every two different coordinates r and s we do not have

$$x_r^d = x_s^b = 0 \text{ and } x_r^b = x_s^d = 1.$$

$\square$

This observation led to the definition of a $\{0,1,*\}$ <u>tournament code</u>. A $\{0,1,*\}$ tournament code C of length k is a subset of $\{0,1,*\}^k$ such that for any two distinct codewords $\underline{a},\underline{b} \in C$:

(i) $\delta(\underline{a},\underline{b}) \geq 1$,

(ii) $\forall_{1 \leq i < j \leq k} \left[\begin{pmatrix} a_i & a_j \\ b_i & b_j \end{pmatrix} \notin \left\{ \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right\} \right]$,

where $\delta(\underline{a},\underline{b}) := |\{ i | (a_i, b_i) \in \{(0,1), (1,0)\}\}|$ is the distance between $\underline{a}$ and $\underline{b}$. Let t(k) denote the maximum number of codewords in any $\{0,1,*\}$ tournament code of length k. Then we have $W_n(q) < B_n(q)$ if $q > t(\frac{n}{2}) + \frac{n}{2}$. In [4] Golomb and Tang prove $t(k) \leq k^{c \log k}$, (logarithm to base 2), $k \geq 4$, with $c = (\ln 2)/0.71$; (This upper bound and method for establishing it were suggested by R.L. Graham.). In Section 4.3 we give a simple proof of this upper bound for c=0.5. To be complete we first give some constructions for $\{0,1,*\}$ tournament codes in Section 4.2. We conclude this chapter with the determination of the exact value of t(k) for k=1,2,3,..,9 in Section 4.4.

4.2 A LOWER BOUND

We repeat the definition of a $\{0,1,*\}$ tournament code $\mathcal{C}$ of length k.

DEFINITION 1. A code $\mathcal{C}$ of length k over the alphabet $\{0,1,*\}$ is called a tournament code if, for any two distinct codewords $\underline{a},\underline{b}$ the following two conditions hold:

$$\text{(i)}\quad \delta(\underline{a},\underline{b}) > 1, \qquad (2)$$

$$\text{(ii)}\quad \forall_{1\leq i<j\leq k}\quad \left[\begin{pmatrix} a_i & a_j \\ b_i & b_j \end{pmatrix} \notin \left\{ \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} , \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right\} \right] .$$

Since $\delta(\underline{a},\underline{b}) > 1$, there is at least one coordinate i, $1 \leq i \leq k$, such that $(a_i,b_j) = (0,1)$ or $(1,0)$. If $(a_i,b_i) = (0,1)$, it follows from condition (ii) that $(a_j,b_j) \neq (1,0)$ for all j, $1 \leq j \leq k$, and we shall say $\underline{a} \to \underline{b}$; this defines the tournament.

DEFINITION 2. The maximal value of $|\mathcal{C}|$ over all tournament codes of length k is called t(k).

From now on $\mathcal{C}$ will always denote a $\{0,1,*\}$ tournament code. The matrix with as rows the codewords of $\mathcal{C}$ will be denoted by C . If $|\mathcal{C}| = t(k)$, we call the code optimal.

LEMMA 3. For every $k \in \mathbb{N}$ there is an optimal code $\mathcal{C}$ of length k with $\underline{0} \in \mathcal{C}$, $\underline{1} \in \mathcal{C}$.

PROOF. If $\mathcal{C}$ is optimal and $\underline{0} \notin \mathcal{C}$, then clearly $\mathcal{C}$ must contain a word with distance 0 to $\underline{0}$.Replace this word by $\underline{0}$ to obtain a new optimal code. Similarly for $\underline{1}$.

□

The following lemma is trivial.

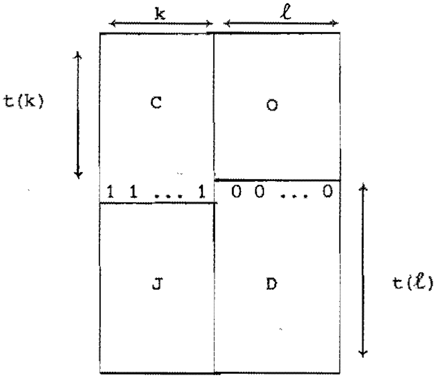
LEMMA 4. If $\mathcal{C}$ is optimal then $\overline{\mathcal{C}}$ is optimal.

□

Until recently Theorem 5 gave the best lower bound on $t(k)$. The proof consists of a construction that produces a long tournament code from two shorter ones.

THEOREM 5. $t(k+\ell) \geqq t(k) + t(\ell) - 1$.

PROOF. Let $\mathcal{C}$ be optimal of length k and let $\underline{0}$ be the top row of $\mathcal{C}$ and $\underline{1}$ the bottom row. Similarly with $\mathcal{D}$ for length ℓ. Consider the code with corresponding matrix,



Clearly, this is a tournament code of length $k+\ell$ and cardinality $t(k) + t(\ell) - 1$.

□

COROLLARY 6. $t(nk) \geq 1 + n(t(k) - 1)$.

□

This shows that $\lim_{k\to\infty} k^{-1} t(k)$ exists (possibly ∞). For a while it was believed that this limit was 2 until Golomb and Tang (1982) found that $t(7) = 16$. The following theorem found by Collins et al. (1984) shows that in fact the limit is ∞.

THEOREM 7. For $n \in \mathbb{N}$ we have $t(n^2+n+1) \geq n(n^2+n+1) + 2$.

PROOF. The following construction for a tournament code C of length n^2+n+1 is due to Collins et al. (1984). The adjustments are from van Lint [7]. We will not prove that the construction indeed gives a tournament code of size $n(n^2+n+1)$; we refer to [7].

The code C, we shall construct consists of $\underline{0}$, $\underline{1}$ and all the cyclic shifts of the words of a set $\{\underline{c}^0, \underline{c}^1, \underline{c}^2, \ldots, \underline{c}^{n-1}\}$. To define these words we number the coordinates with the integers mod n^2+n+1, starting with -1 (i.e., for the first coordinate). The coordinates $\neq -1$ will have their index written in the $(n+1)$-ary system. So (x,y) denotes the coordinate with index $(n+1)x+y$. Therefore $0 \leq x \leq n-1$, $0 \leq y \leq n$. The definition of the words $\underline{c}^i$ is as follows

i) For each i take $\underline{c}^i_{-1} = 1$,

ii) $$\underline{c}^i \text{ has } \begin{cases} 0 \text{ in coordinate place } (x,y) \text{ if } x \geq i \text{ and } x+y \leq n-1, \\ 1 \text{ in coordinate place } (x,y) \text{ if } x \leq i-1 \text{ and } x+y \geq n-1, \\ * \text{ otherwise.} \end{cases}$$

□

To be complete we mention that the first code in this class is the code of length 7 found by Golomb and Tang (1982). The second one of length 13 was, independently of Collins et al., found by Abels, Janse and Verbakel (1984). From their work we copied the following list of tournament codes of length $k = 2,3,\ldots,13$, the first eight of which are optimal (see Section 4.4). Obviously $t(1) = 2$. The optimal $\{0,1,*\}$ tournament code C_1 of length 1 is equal to $C_1 = \{(0),(1)\}$.

$$C_2 = \begin{Bmatrix} 00 \\ 01 \\ 11 \end{Bmatrix} \quad C_3 = \begin{Bmatrix} 000 \\ 1*0 \\ *01 \\ 01* \\ 111 \end{Bmatrix} \quad C_4 = \begin{Bmatrix} 0000 \\ 0*01 \\ 001* \\ 01*0 \\ 0111 \\ 1111 \end{Bmatrix} \quad C_5 = \begin{Bmatrix} 00000 \\ 1*1*0 \\ *0*01 \\ *101* \\ 10*11 \\ 0*11* \\ 0001* \\ 11111 \end{Bmatrix}$$

$$C_6 = \begin{Bmatrix} 000000 \\ 1**1*0 \\ *1**01 \\ **101* \\ 1*0*11 \\ *011*1 \\ 01*11* \\ 0*001* \\ 00*1*0 \\ *00*01 \\ 111111 \end{Bmatrix} \quad C_7 = \begin{Bmatrix} 0000000 \\ **1*110 \\ *1*110* \\ 1*110** \\ *110**1 \\ 110**1* \\ 10**1*1 \\ 0**1*11 \\ 00*0**1 \\ 0*0**10 \\ *0**100 \\ 0**100* \\ **100*0 \\ *100*0* \\ 100*0** \\ 1111111 \end{Bmatrix} \quad C_8 = \begin{Bmatrix} 00000000 \\ *11*1**0 \\ *1*1**01 \\ 1*1**011 \\ *1**011* \\ 1**011*1 \\ 1*011*1* \\ 1011*1** \\ 0***1*11 \\ 0**0*001 \\ 0*0*001* \\ *0*001** \\ 0*001**0 \\ *001**0* \\ *01**0*0 \\ *1**0*00 \\ 1000*0** \\ 11111111 \end{Bmatrix}$$

$$C_9 = \begin{Bmatrix} 000000000 \\ 000111*** \\ *111**1*0 \\ 1*1*1**01 \\ 11***101* \\ ***1*0*11 \\ ****011*1 \\ ***01*11* \\ 1*01111** \\ *01111*1* \\ 01*111**1 \\ 0***00*01 \\ *0*0*001* \\ **000*1*0 \\ 000*010** \\ 00001**0* \\ 0001*0**0 \\ *01****00 \\ 01****0*0 \\ 1*0***00* \\ 111111111 \end{Bmatrix} \quad C_{10} = \begin{Bmatrix} 0000000000 \\ 0**00*0001 \\ **00*0001* \\ *00*0001** \\ 00*0001*** \\ 0*0001***0 \\ *0001***0* \\ 0001***0** \\ 001***0**0 \\ 01***0**00 \\ 1***0**001 \\ 0**0011**1 \\ 011**1***0 \\ 11**1***01 \\ 1**1***011 \\ **1***0111 \\ *1***0111* \\ 1***0111*1 \\ ***0111*11 \\ **0111*11* \\ *0111*11** \\ 0111*11**1 \\ 1111111111 \end{Bmatrix}$$

Fig. 1. List of {0,1,*} tournament codes C_k of length k, k = 2,3,...,10.

$$\begin{array}{c} C_{11} \\ \begin{pmatrix} 00000000000 \\ *0**00*0001 \\ 0**00*0001* \\ **00*0001** \\ *00*0001*** \\ 00*0001***0 \\ 0*0001***0* \\ *0001***0** \\ 0001***0**0 \\ 001***0**00 \\ *1***0**001 \\ 1**1***0**0 \\ 1***0**0011 \\ *0**0011**1 \\ 0**0011**1* \\ 0011**1***0 \\ *11**1***01 \\ 11**1***011 \\ 1**1***0111 \\ **1***0111* \\ *1***0111*1 \\ 1***0111*11 \\ ***0111*11* \\ **0111*11** \\ *0111*11**1 \\ 0111*11**1* \\ 11111111111 \end{pmatrix} \end{array} \qquad \begin{array}{c} C_{12} \\ \begin{pmatrix} 000000000000 \\ **0**00*0001 \\ *0**00*0001* \\ 0**00*0001** \\ **00*0001*** \\ *00*0001***0 \\ 00*0001***0* \\ 0*0001***0** \\ *0001***0**0 \\ 0001***0**00 \\ 001***0**00* \\ 01***0**00*0 \\ 1***0**00*00 \\ *1***0**0011 \\ 1***0**0011* \\ **0**0011**1 \\ *0**0011**1* \\ 0**0011**1** \\ *0011**1***0 \\ 0011**1***0* \\ 11*11**1***0 \\ 1*11**1***01 \\ *11**1***011 \\ 11**1***0111 \\ 1**1***0111* \\ **1***0111*1 \\ *1***0111*11 \\ 1***0111*11* \\ ***0111*11** \\ **0111*11**1 \\ *0111*11**1* \\ 0111*11**1** \\ 111111111111 \end{pmatrix} \end{array} \qquad \begin{array}{c} C_{13} \\ \begin{pmatrix} 0000000000000 \\ ***0**00*0001 \\ **0**00*0001* \\ *0**00*0001** \\ 0**00*0001*** \\ **00*0001***0 \\ *00*0001***0* \\ 00*0001***0** \\ 0*0001***0**0 \\ *0001***0**00 \\ 0001***0**00* \\ 001***0**00*0 \\ 01***0**00*00 \\ 1***0**00*000 \\ 11**1***0**00 \\ 1**1***0**001 \\ **1***0**0011 \\ *1***0**0011* \\ 1***0**0011** \\ ***0**0011**1 \\ **0**0011**1* \\ *0**0011**1** \\ 0**0011**1*** \\ **0011**1***0 \\ *0011**1***0* \\ 0011**1***0** \\ 011**1***0**0 \\ 111*11**1***0 \\ 11*11**1***01 \\ 1*11**1***011 \\ *11**1***0111 \\ 11**1***0111* \\ 1**1***0111*1 \\ **1***0111*11 \\ *1***0111*11* \\ 1***0111*11** \\ ***0111*11**1 \\ **0111*11**1* \\ *0111*11**1** \\ 0111*11**1*** \\ 1111111111111 \end{pmatrix} \end{array}$$

Fig. 2. List of $\{0,1,*\}$ tournament code C_k of length k, $k = 11,12,13$.

4.3 AN UPPER BOUND

In this section we give a simple proof of Grahams upper bound on t(k) mentioned in Section 4.1. We first note that clearly t(k) is stricktly increasing. The following observation essentially proves Grahams result.

Let $\mathcal{C}$ be a tournament code of length k with $\underline{0}$, $\underline{1} \in \mathcal{C}$ (Lemma 3). By permuting rows and columns, the corresponding matrix C can be put in the following "standard form".

$$\begin{array}{cc} & \begin{array}{cc} \hphantom{0} \; \longleftarrow \ell \longrightarrow & \leftarrow k-1-\ell \rightarrow \end{array} \\ C = & \left[\begin{array}{c|c|c} 0 & & \\ \vdots & A & B \\ 0 & & \\ \hline 1 & & \\ \vdots & D & E \\ 1 & & \\ \hline * & & \\ \vdots & F & G \\ * & & \end{array}\right] \end{array} .$$

Fig. 1.

Here, every column of A contains a 1 but no column of B contains a 1.

From definition 1, in particular condition (ii), it follows that no column of D has a 0, while every column of E may have a 0. This shows that $\mathcal{C}$ or $\overline{\mathcal{C}}$ has a standard form with $\ell \leqq \lfloor\frac{k-1}{2}\rfloor$.

THEOREM 8. $t(k) \leqq t(k-1) + t(\ell)$, for some ℓ, $0 \leqq \ell \leqq \lfloor\frac{k-1}{2}\rfloor$.

PROOF. Let $\mathcal{C}$ be an optimal code with C in "standard form" with $\ell \leqq \lfloor\frac{k-1}{2}\rfloor$ (see Fig. 1). By the definition of B the rows of the matrix A form a tournament code of length ℓ. So A has at most $t(\ell)$ rows. Clearly the rows of $\begin{pmatrix} D & E \\ F & G \end{pmatrix}$ form a tournament code of length k-1. The result follows.

□

COROLLARY 9. $t(k) \leqq \lceil\frac{k+3}{2}\rceil\, t(\lfloor\frac{k-1}{2}\rfloor)$.

PROOF. Applying Theorem 8 $\lceil\frac{k+1}{2}\rceil$ times and using the fact that t is a strictly increasing function, we have

$$t(k) \leqq t(k - \lceil\frac{k+1}{2}\rceil) + \sum_{i=1}^{\lceil\frac{k+1}{2}\rceil} t(\lfloor\frac{k-i}{2}\rfloor) =$$

$$= t(\lfloor\frac{k-1}{2}\rfloor) + \sum_{i=1}^{\lceil\frac{k+1}{2}\rceil} t(\lfloor\frac{k-i}{2}\rfloor) \leqq$$

$$\leqq \lceil\frac{k+3}{2}\rceil\, t(\lfloor\frac{k-1}{2}\rfloor) .$$

□

THEOREM 10. $t(k) < k^{0.5 \log k}$ for $k > 7$ (logarithm to base 2).

PROOF. We use induction on k. In Section 4.4 we will show that t(4)=6, t(5)=8, t(6)=11, t(7)=16, t(8)=18 and t(9)=21. With Theorem 8 and the above values of t(k), k=4,5,...,9, one easily checks that the assertion is true for $8 \leqq k \leqq 16$ (see also Table 1 of Section 4.4).

Let $k \geqq 17$, then from Corollary 9 and the induction hypothesis we have

$$t(k) \leq \left\lceil\frac{k+3}{2}\right\rceil t\left(\left\lfloor\frac{k-1}{2}\right\rfloor\right) \leq$$

$$\leq \left\lceil\frac{k+3}{2}\right\rceil \left(\left\lfloor\frac{k-1}{2}\right\rfloor^{0.5 \log\left\lfloor\frac{k-1}{2}\right\rfloor} \right) \leq$$

$$\leq \left(\frac{k+4}{2}\right) \left(\left(\frac{k}{2}\right)^{0.5 \log \left(\frac{k}{2}\right)} \right) \leq$$

$$\left(\frac{k+4}{2}\right) \frac{\sqrt{2}}{k} k^{0.5 \log k} \leq k^{0.5 \log k}$$

□

There is a tremendous gap between the upper bound of this section and the lower bound of Section 4.2. The upper bound is probably not too good but improving it does not look easy. The following section gives an indication.

4.4 THE EXACT VALUE OF t(k) FOR k = 2,3,...,9.

In this section we will prove that the codes of length k, $2 \leqq k \leqq 9$, of Section 4.2 are optimal. It is clear that this is indeed the case for $k = 2$, while the case $k = 3$ follows directly from Theorem 8 and $t(1) = 2$. To prove $t(4)=6$ and $t(5)=8$, we use the following obvious lemma.

<u>LEMMA 11</u>. For any tournament code $\mathcal{C}$ of length k we have,

$$\sum_{\underline{c} \in \mathcal{C}} 2^{n_*(c)} \leq 2^k \quad ,$$

where $n_*(\underline{c}) = |\{i| \; c_i = *\}|$.

PROOF. Since any two codewords from $\mathcal{C}$ have distance greater than or equal to 1, any binary word of length k can have distance 0 to at most one codeword of $\mathcal{C}$. For each codeword $\underline{c}\in\mathcal{C}$ there are clearly $2^{n_*(\underline{c})}$ different binary words of length k having distance 0 to $\underline{c}$. The result follows.

□

We will use this lemma to show that $t(4)\leqq 6$. From Theorem 6 we have $t(4)\leqq t(3) + t(1) = 7$. Assume $t(4) = 7$ and let $\mathcal{C}$ be an optimal code of length 4 with $\underline{0},\underline{1}\in\mathcal{C}$. Since equality holds in Theorem 8 every column of C can contain at most $t(1) + t(2) = 2+3 = 5$ non * elements. Hence C contains at least $4 \times (7-5) = 8$ *'s. From Lemma 11 and $\underline{0},\underline{1}\in\mathcal{C}$ we then have

$$2+\sum_{\substack{\underline{c}\in\mathcal{C},\\ \underline{c}\neq\underline{0},\underline{1}}} 2^{n_*(\underline{c})}\leq 2^4 \quad\text{and}\quad \sum_{\substack{\underline{c}\in\mathcal{C},\\ \underline{c}\neq\underline{0},\underline{1}}} n_*(\underline{c})\geqq 8.$$

This is impossible. Hence $t(4)= 6$. The case k=5 is similar.

The case k=6 and k=7 are again a direct consequence of Theorem 8 and Section 4.2. So we are left with k=8 and k=9. For these cases we need some more machinery.

Let $\mathcal{C}$ be a tournament code of length k and let the coordinates be numbered from 1 up to k. Then we define

$$i<j \;:\Leftrightarrow\; \exists_{\underline{c}\in\mathcal{C}}\,[c_i = 0 \text{ and } c_j = 1]\ .$$

Furthermore we define the vectors $\underline{a}^r$, $\underline{b}^r$, $\in\{0,1,*\}^k$, $r = 1,2,..,k$, by:

$$\underline{a}^r \text{ has } \begin{cases} 0 \text{ in coordinate place } r, \\ 1 \text{ in coordinate place } i \text{ if } r< i, \\ * \text{ otherwise} \end{cases}$$

and

$$\underline{b}^r \text{ has } \begin{cases} 1 \text{ in coordinate place } r, \\ 0 \text{ in coordinate place } i \text{ if } i< r, \\ * \text{ otherwise.} \end{cases}$$

REMARK. The words $\underline{a}^r$, $\underline{b}^r$, $r = 1,2,..,k$ satisfy condition (ii) of (2).

LEMMA 12. let C be an optimal tournament code of length k, for which $\sum_{\underline{c} \in C} n_*(\underline{c})$ is minimal among all optimal codes of length k and let $\underline{a}^r$ and $\underline{b}^r$, $r = 1,2..,k$, be defined as above. Then the set of words $\{\underline{a}^r \mid r = 1,2,...,k\} \cup \{\underline{b}^r \mid r = 1,2,...,k\} \cup C$ satisfies condition (ii) of (2). Furthermore, for every $r = 1,2,...,k$ there is a unique word $\underline{c} \in C$ with distance 0 to a^r. Similarly for $\underline{b}^r$.

PROOF. The first assertion of Lemma 12 is a direct consequence of the definitions of $\underline{a}^r$ respectively $\underline{b}^r$, $r = 1,2,..,k$. So we only have to prove the second one.

Since C is optimal, there is at least one word of C that has distance 0 to a^r. Assume there are two different codewords $\underline{c}$ and $\underline{d}$ in C that have distance 0 to $\underline{a}^r$. Since $\underline{c}$ and $\underline{d}$ have distance greater than or equal to 1, there is an s, $1 \leq s \leq k$, where $c_s=0$ and $d_s=1$ say. Since $\underline{c}$ and $\underline{d}$ both have distance 0 to $\underline{a}^r$, $a_s^r = *$. From the definition of $\underline{a}^r$ we then have $r \neq s$ and $r \not< s$. So $d_r=*$. Now define $\underline{d}' \in \{0,1,*\}^k$ by

$$d_i := \begin{cases} 1 & \text{if } i = r\ , \\ d_i & \text{if } i \neq r\ . \end{cases}$$

Since, for all i with $d_i' = 0$, also $d_i = 0$ and so $a_i^r = *$, it follows $r \not< i$. So the words of $\{d'\} \cup C$ satisfy condition (ii) of (2).

But then $C' := \{\underline{d}'\} \cup C \setminus \{\underline{d}\}$ is an optimal tournament of length k with $\sum_{\underline{c}' \in C'} n_*(\underline{c}') = \sum_{\underline{c} \in C} n_*(\underline{c}) - 1$. A contradiction.

□

The following lemma is "trivial". The words $\underline{a}^r, \underline{b}^r$, $r = 1,2,..,k$, are as defined above.

LEMMA 13. A codeword $\underline{c} \in C$, $\underline{c} \neq \underline{0},\underline{1}$ that has distance greater than or equal to 1 to all the words of $\{\underline{a}^r \mid r = 1,2,..,k\} \cup \{\underline{b}^r \mid r = 1,2,..,k\}$ has at least three coordinates equal to 0 and at least three coordinates equal to $\underline{1}$.

□

For any $\lambda \in \{0,1,*\}$ and $\underline{c} \in \{0,1,*\}^k$, let $n_\lambda(\underline{c})$ denote the number of coordinate places i with $c_i = \lambda$. The following lemma is obvious.

LEMMA 14. Let C be an optimal code of length k. Then for all $\underline{c} \in C$

$$t(k) \leq t(k - n_0(\underline{c})) + t(k - n_1(\underline{c})) - 1.$$

PROOF. By permuting rows and columns one can achieve that C looks like

$$C = \left[\begin{array}{c|c|c} \leftarrow n_0(\underline{c}) \rightarrow & \leftarrow n_1(\underline{c}) \rightarrow & \\ 0\;0\ldots0 & 1\;1\ldots1 & *\;*\ldots* \\ \hline A & B & D \\ \hline E & F & G \end{array}\right] \begin{array}{l} \\ \leftarrow \underline{c} \\ \\ \\ \end{array},$$

where each row of A contains at least one $\underline{1}$. It follows that B is a matrix with all entries equal to 1 or * and E is a matrix with all entries equal to 0 or *. Hence the rows of $\left(\frac{0..0}{A} \middle| \frac{*..*}{D}\right)$ respectively $\left(\frac{1..1}{F} \middle| \frac{*..*}{G}\right)$ form a tournament code of length $n-n_1(\underline{c})$ respectively $n-n_0(\underline{c})$. The result follows.

□

COROLLARLY 15. $t(8) = 18$ and $t(9) = 21$.

PROOF. From Section 4.2 we already have $t(8) \geq 18$.

Let C be an optimal code of length 8 for which the sum $\sum_{\underline{c} \in C} n_*(\underline{c})$ is minimal among all optimal codes of length 8. Let the words $\underline{a}^r$ and $\underline{b}^r$, $r = 1,2,..,8$, be defined as above. From Lemma 14 and $t(8) \geq 18$ it follows that there is no codeword $\underline{c} \in C$ with both $n_0(\underline{c}) \geq 3$ and $n_1(\underline{c}) \geq 3$. Hence with Lemma 12 and 13, $t(8) \leq |\{ \underline{a}^r \mid r = 1,2,..,8\}| + |\{ \underline{b}^r \mid r = 1,2,..,8\}| + 2 \leq 8 + 8 + 2 = 18$.

The proof of $t(9) = 21$ is similar to that of $t(8) = 18$.

□

We conclude this section with a small table of lower and upper bounds on t(k) for k=10,11,..,21. In the last column of Table 1 we indicate the theorems and lemmas we used to derive the upper bound. The lower bounds are from Abels, Janse and Verbakel [1].

k	Lower bound on t(k)	Upper bound on t(k)	Comment
10	23	27	Th 8
11	27	33	Th 8 + L 11
12	33	40	Th 8 + L 11
13	41	48	Th 8 + L 11
14	43	57	Th 8
15	46	73	Th 8
16	48	81	Th 8 + L 11
17	54	92	Th 8 + L 11
18	59	108	Th 8 + L 11
19	66	124	Th 8 + L 11
20	75	141	Th 8 + L 11
21	86	159	Th 8 + L 11

Table. 1.

REFERENCES

[1] ABELS, F., JANSE, W. and VERBAKEL, J.: *Opdracht Discrete Wiskunde II*. Internal Report Technological University Eindhoven, 1984.

[2] EASTMAN, W.L.: *On the Construction of Comma-Free Codes*. IEEE Trans. Info. Theory, vol. IT-11, Apr. 1965, 263-266.

[3] GOLOMB, S.W., GORDON, B., WELCH, L.R.: *Comma-Free Codes*. Can. J. Math., vol. 10, 1958, 202-209.

[4] GOLOMB, S.W. and TANG, B.: *A New Result on Comma-Free Codes of Even Word-length*. To appear.

[5] GRICK, F.H.C., GRIFFITH, J.S. and ORGEL, L.E.: *Codes without Commas*. Proc. Nat. Acad. Sci., vol. 43, 1957, 416-421.

[6] JIGGS, B.H.: *Recent Results in Comma-Free Codes*. Can. J. Math., vol. 15, 1963, 178-187.

[7] Van LINT, J.H.:"$\{0,1,*\}$ *Distance Problems in Combinatorics*". Internal Report Technological University Eindhoven.

[8] SCHOLTZ. R.A.: *Maximal and Variable Word-length Comma-Free Codes*. IEEE Trans. Info. Theory, vol. IT-15, Mar. 1969, 300-306.

SAMENVATTING

In dit proefschrift worden vier problemen uit de coderingstheorie behandeld, met als voornaamste doel het bepalen van bovengrenzen voor de maximale cardinaliteit van en het vinden van goede constructie methoden voor de betreffende codes. Het begrip afstand speelt hierbij een belangrijke rol. Om de lezer enigszins vertrouwd te maken met de in dit proefschrift gebruikte terminologie, geven we in hoofdstuk 0 een korte inleiding in de coderingstheorie.

In hoofdstuk 1 houden wij ons voornamelijk bezig met het bepalen van blok codes voor de betrouwbare opslag van informatie in computer geheugens met defecten en random errors. De hier behandelde constructie methoden doen een stevig beroep op reeds bestaande constructies voor lineaire en niet-lineaire codes voor het binair symmetrisch kanaal.

In hoofdstuk 2 behandelen we twee constructie methoden voor constante gewichts codes. Vooral de tweede constructie geeft scherpe verbeteringen op reeds bestaande ondergrenzen voor $A(n,4,w)$. Het bepalen van partities van $V^n_w = \{\underline{c} \in \mathbb{F}^n_2 \mid wt(\underline{c}) = w\}$ in zo weinig mogelijk constant gewicht codes met minimum afstand 4 is in deze constructie van cruciaal belang.

In hoofdstuk 3 geven we de volledige classificatie van alle optimale code paren van lengte n en constante afstand δ, $n,\delta \in \mathbb{N}$, $0 \leq \delta \leq n$. Daartoe bepalen we eerst de waarde van $M(n,\delta)$,

$$M(n,\delta) = \{ |A| \cdot |B| \mid A,B \subset \mathbb{F}^n_2 \; , \; \Delta(A,B) = \delta \}.$$

In het laatste hoofdstuk houden we ons bezig met het bepalen van bovengrenzen voor de maximale cardinaliteit $t(k)$ van $\{0,1,*\}$ tournament codes van lengte k, $k \in \mathbb{N}$. We geven een verscherping van Graham's bovengrens voor $t(k)$, $k > 7$, en bepalen vervolgens de exacte waarden van $t(k)$ voor $k = 1,2,\ldots,9$.

CURRICULUM VITEA

De schrijver van dit proefschrift werd geboren op 29 april 1953 te Rotterdam. Na in 1969 het Mulo-A diploma behaald te hebben, volgde hijde HBS-B opleiding aan het Moller-lyceum in Bergen op Zoom, die hij in 1972 beëindigde. Hierna leverde hij een verplichte bijdrage van 1½ jaar aan 's lands verdediging. Vanaf 1974 studeerde hij wiskunde aan de Tachnische Hogeschool in Eindhoven, waar hij in 1982 zijn doctoraal examen aflegde. Tijdens zijn studie was hij gedurende 2 jaar student-assistent.

Van september 1982 tot september 1986 was hij wetenschappelijk assistent bij de onderafdeling der wiskunde en informatica van de bovengenoemde hogeschool. Thans is hij werkzaam bij Philips-Usfa, Eindhoven.

STELLINGEN

1 Zij $\mathcal{C}$ een binaire lineaire code met woord lengte 26, dimensie k en minimum afstand 8. Dan geldt: $k < 13$, en dus

$$B(26,8) = 2^{12}$$

HELGERT, H.J. and STINAFF, R.D.: *Minimum Distance Bounds for Binary Linear Codes*. IEEE. Trans. on Info. Theory, vol. IT-19, May 1973, 344-356.

2 Er bestaat een uniek decodeerbaar code paar $(\mathcal{C},\mathcal{D})$ voor het two-acces binary adder channel met woord lengte 6 waarvan de som rate $R_1 + R_2$ gelijk is aan

$$R_1 + R_2 = 0.59749 + 0.72032 = 1.31781$$

$(|\mathcal{C}| = 12$ en $|\mathcal{D}| = 20)$. Dit is een nieuw record.
COEBERGH van den BRAAK, P.A.B.M. and van TILBORG, H.C.A.: *A Family of Good Uniquely Decodable Code Pairs for the Two-Acces Binary Adder Channel*. IEEE Trans. on Info. Theory, vol. IT-31, Jan. 1985, 3-9.

3 Zij $\mathcal{C}$ een $\{0,1,*\}$ tournament code van lengte 10. Dan geldt $|\mathcal{C}| \leq 23$. Dus volgt uit tabel 1 van hoofdstuk 4 dat

$$t(10) = 23.$$

4 Zij $\mathcal{C}$ een constant gewicht code van lengte 17, minimum afstand 8 en constant gewicht 8. Dan geldt: $|\mathcal{C}| \leq 34$, en dus

$$A(17,8,8) = 34.$$

MacWILLIAMS, F.J. and SLOANE, N.J.A.: *The Theory of Error-correcting Codes*. Amsterdam - New York - Oxford: North Holland, 1977.

5 Zij $M_k(n,\delta) := \max\{|A|\cdot|B| \mid A,B \in \{0,1,\dots,k-1\}^n,\ \Delta(A,B) = \delta\}$. Vermoedelijk is de waarde van $M_k(n,\delta)$ gelijk aan

$$M_k(n,\delta) = \begin{cases} \max \{18^i \binom{n-3i}{\delta-2i} 2^{\delta-2i} \mid 0 \leq i \leq \min \{\frac{\delta}{2}, \frac{n}{3}\} \} & \text{als } k=3, \\ \max \{ (\lceil\frac{k}{2}\rceil\lfloor\frac{k}{2}\rfloor)^i \binom{n-i}{\delta-i} (k-1)^{\delta-i} \mid 0 \leq i \leq \delta\} & \text{als } k \geq 4. \end{cases}$$

Voor $k = 4,5$ is dit bewezen door Ahlswede (to appear) en voor $k \geq 10$ door van Pul (unpublished).

6 De lineaire programmeringsgrens voor binaire codes is een generalisatie van de Plotkin grens.

7 Zij C een binaire $[n,k,d]$-code met overdekkingsstraal ρ. Zij $C^i_\varepsilon := \{\underline{c} \in C \mid c_1 = \varepsilon\}$, $i = 1,2,\ldots,n$ en $\varepsilon = 0,1$. De norm N van de code C wordt gedefinieërd door

$$N := \min_{1 \leq i \leq n} \max_{\underline{x} \in \mathbb{F}_2^n} \{d(\underline{x}, C_0^i) + d(\underline{x}, C_1^i)\}.$$

Dan geldt

$$N \leq 2\rho - 1 + \lceil\frac{d}{2}\rceil.$$

Als $N \leq 2\rho + 1$, dan noemen we de code C normaal. Uit bovenstaande ongelijkheid volgt dat iedere binaire lineaire code met minimum afstand ≤ 4 normaal is. Vermoedelijk zijn alle binaire lineaire codes normaal. GRAHAM, R.L. and SLOANE, N.J.A.: *On the Covering Radius of Codes*. IEEE Trans. on Info. Theory, vol. IT-31, May 1985, 385-401.

8 Standaardisatie van cryptosystemen leidt tot diversificatie.

9 Ondanks de resultaten van Tsfasman, Vlădut en Zink, is het vermoedelijk waar dat de Gilbert-Varshamov grens voor binaire codes scherp is. TSFASMAN, M.A., VLĂDUT, S.G. and ZINK, Th.: *Modular curves,Shimura curves and Goppa codes, better than Varshamov-Gilbert bound*. Math. Nachr. vol. 104, 1982, 13-28.

10 De AIO-regeling voor jonge onderzoekers is equivalent met de BKR-regeling voor beeldende kunstenaars.