Partitioned Linear Block Codes for Computer Memory with "Stuck-at" Defects

CHRIS HEEGARD, MEMBER, IEEE

Abstract—**Linear block codes are studied for improving the reliability of message storage in computer memory with stuck-at defects and noise. The case when the side information about the state of the defects is available to the decoder or to the encoder is considered. In the former case, stuck-at cells act as erasures so that techniques for decoding linear block codes for erasures and errors can be directly applied. We concentrate on the complimentary problem of incorporating stuck-at information in the encoding of linear block codes. An algebraic model for stuck-at defects and additive errors is presented. The notion of a "partitioned" linear block code is introduced to mask defects known at the encoder and to correct random errors at the decoder. The defect and error correction capability of partitioned linear block codes is characterized in terms of minimum distances. A class of partitioned cyclic codes is introduced. A BCH-type bound for these cyclic codes is derived and employed to construct partitioned linear block codes with specified bounds on the minimum distances. Finally, a probabilistic model for the generation of stuck-at cells is presented. It is shown that partitioned linear block codes achieve the Shannon capacity for a computer memory with symmetric defects and errors.**

Manuscript received November 16, 1981; revised April 18, 1983. This work was supported under NSF Grant ECS78-23334 and DARPA Contract MDA 903-79-C-0680. This work was presented at the IEEE International Symposium on Information Theory, Les Arcs, France, June 1982.

The author was with the Department of Electrical Engineering, Stanford University, CA 94305. He is now with the School of Electrical Engineering, Cornell University, Ithaca, NY 14853.

I. INTRODUCTION

LINEAR block codes can be used to improve the reliability of message storage in an imperfect computer memory. We consider a memory that is composed of n cells. Each cell is expected to store one of q symbols. We

0018-9448/83/1100-0831$01.00

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. For example, some of the cells of a binary memory may be stuck at 0, and 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 intermittent. Often the terms hard and soft errors, respectively, are used to describe these sources of error.

By testing the memory it may be possible to determine the locations of the stuck-at cells. The side information that describes the state of the defects can be incorporated into the decoding or into the encoding of linear block codes. When the locations of the stuck-at defects are known at the decoder, these cells act as erasures. Thus techniques for decoding linear block codes with random errors and erasures can be directly applied as in Stiffler [13]. In this paper we consider the complimentary problem of incorporating stuck-at information in the encoding of linear block codes.

The origin of this problem is a paper by Kusnetsov and Tsybakov [1]. They consider coding for binary memories that have a fixed fraction p of stuck-at cells. The assumption is that the location and nature of the defects are available to the encoder and not to the decoder.

They define a code as a partition of the set of all binary sequences of length n into 2^k disjoint subsets $\{A_0, A_1, \cdots, A_{2^k-1}\}$. A message is associated with each subset. It is desired that when a k-bit message $w \in \{0,1,\cdots,2^k-1\}$ is given to the encoder, along with a description of the stuck-at cells of the memory, a sequence $x \in A_w$ can be found that agrees with all of the defective cells. This compatible sequence can then be stored without alteration. The decoder, recognizing that the sequence belong to subset A_w, can infer that message w was stored.

A binary linear block code can be used to define such a partition code. Let G and H be the $l \times n$ generator and $k \times n$ parity-check matrix for a binary $[n, l]$ linear block code ($k + l = n$). Each matrix has full rank and $GH^t = 0$. For each k-bit message vector w define

$$A_w = \{ y \mid yH^t = w \}.$$

These sets are known as the cosets of the code and they partition the binary n-tuples into 2^k subsets of equal size (2^l).

The decoder for these linear block codes takes the retrieved vector y and sets $\hat{w} = yH^t$. Suppose that a memory has u defects at locations $1 \leqslant i_1 < i_2 < \cdots < i_u \leqslant n$ and these cells are stuck at $s_1, s_2, \cdots, s_u \in \{0,1\}$. Let x be any vector with $xH^t = w$ (i.e., any vector that would be decoded as w). The encoder tries to solve the matrix equation

$$dG' = \left[s_1 - x_{i_1}, s_2 - x_{i_2}, \cdots, s_u - x_{i_u} \right]$$

for d an l vector, where

$$G' = \left[g_{i_1}, g_{i_2}, \cdots, g_{i_u} \right]$$

is the $l \times u$ submatrix of G that involves the u defects. If a solution is found then $y = x + dG$ is stored. Note that $y_{i_1} = s_1$, $y_{i_2} = s_2, \cdots, y_{i_u} = s_u$ and $yH^t = w$ so that it will be correctly decoded. In order to guarantee that every message be correctly decoded for every memory with u or fewer defects, it is necessary and sufficient that every submatrix G' have rank u, i.e., the columns must be linearly independent. Thus u can be as large as the minimum distance, minus one, of the dual code which is generated by H. For example, the $[n, 1]$ repetition code with $G = [1 \ 1 \ \cdots \ 1]$ can store $n - 1$ bits in any memory with one stuck-at cell. The $[n = 2^l - 1, l]$ binary simplex codes (dual of the Hamming codes) can store $k = 2^l - 1 - l$ bits in a memory with two stuck-at cells.

Codes that are capable of correcting both defects and random errors are obtained by partitioning random error correcting codes. Let $\mathcal{C}$ be a binary error-correcting code of length n and size $|\mathcal{C}| = 2^{k+l}$. Partition $\mathcal{C}$ into 2^k subsets $\{A_0, A_1, \cdots, A_{2^k-1}\}$. Then given $w \in \{0,1,\cdots,2^k-1\}$ and a description of the defects of the memory, the encoder selects a sequence $x \in A_w$ that is compatible with the stuck-at cells. The decoder observes $y = x + z$, a noisy version of x. By using the error correction capability of the code, an estimate $\hat{x} \in \mathcal{C}$ is obtained. Finally the message is retrieved by identifying the subset of $\mathcal{C}$ to which $\hat{x}$ belongs.

A binary $[n, k, l]$ partitioned linear block code is defined in terms of an $[n, k + l]$ linear block code with an $[n, l]$ subcode. Let G_0, G_1, H, and $\tilde{G}_1$ be full rank binary matrices of sizes $l \times n$, $k \times n$, $r \times n$, and $k \times n$ respectively ($k + l + r = n$). Assume that the $(k + l) \times n$ matrix $G = [G_1^t, G_0^t]^t$ and the matrix H are the generator and parity-check matrices for an $[n, k + l]$ linear block code $\mathcal{C}$ (i.e., $\operatorname{rank}(G) = k + l$, $GH^t = 0$), $G_0\tilde{G}_1^t = 0$ and $G_1\tilde{G}_1^t = I$. Then

$$A_w = \left\{ y \mid yH^t = \mathbf{0}, y\tilde{G}_1^t = w \right\}$$

partitions $\mathcal{C}$ into 2^k subsets each of size 2^l. The following example illustrates such a code.

Example: Suppose we would like to store 3 bits of information ($k = 3$) in a binary memory of block length $n = 7$ and protect this information against a single stuck-at cell, known at the encoder, and a single random error.

We can define an $[n = 7, k = 3, l = 1]$ partitioned linear block code by

$$G_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$$

$$G_0 = [1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1]$$

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

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

The matrix $G = [G_1^t, G_0^t]^t$ is a generator for the [7], [4] single error correcting Hamming code. Suppose, for exam-

ple, we would like to store the message $w = [1\ \ 1\ \ 0]$ and the memory has a stuck-at-1 defect in the third position and a random error in the fourth position. The encoder first computes $wG_1 = [1\ \ 1\ \ 0\ \ 0\ \ 1\ \ 0\ \ 0]$. Since this codeword has a 0 in the third position and the encoder is told that this cell is stuck-at-1, the encoder stores $x = wG_1 + G_0 = [0\ \ 0\ \ 1\ \ 1\ \ 0\ \ 1\ \ 1]$. Note that this sequence agrees with the defect, thus avoiding a defect-induced error. The decoder then retrieves

$$y = [0\ \ 0\ \ 1\ \ 0\ \ 0\ \ 1\ \ 1]$$

which has an error in the fourth position. Computing $s = yH^t = [1\ \ 1\ \ 0]$, the decoder corrects the fourth position by recoginizing the syndrome as the fourth row of H^t. Finally, we get a correct estimate of the message $\hat{w} = \hat{x}\tilde{G}_1^t = [1\ \ 1\ \ 0]$.

Kusnetsov and Tsybakov [1] show that for binary memory with defects, it is sufficient to inform the encoder of the location and nature of the defects. By allowing the size of the memory n to become large, they prove the existence of codes that are capable of storing messages, without error, for any rate $R \triangleq k/n < 1 - p$ (p = fraction of defects). This paper initiated the search for codes that are capable of correcting a fixed number of defects [2]–[4]. In [5], Tsybakov looks at linear block codes for the defect problem and shows that the asymptotic rate for the best linear block code, under the guaranteed correction criterion, satisfies

$$1 - h(p) \leqslant R \leqslant 1 - h\left(\left(1 - \sqrt{1 - 2p}\right)/2\right)$$

for $0 \leqslant p \leqslant 1/2$; $h(p)$ is the binary entropy function. Since it is generally true that $1 - p > 1 - h((1 - \sqrt{1 - 2p})/2)$ we see that these codes are suboptimal under this criterion. Tsybakov [6] goes on to introduce the problem of coding for binary memory with both defects and random errors. He again considers the asymptotic rates achievable by linear block codes under a guaranteed correction criterion. This paper led to [7] where a scheme for correcting defects and random errors is described. Many of these results are summarized in [8].

In an attempt to further understand the problem of defective memory, Gel'fand and Pinsker [9] put forth a probabilistic model for the existence of defects and random errors in computer memory. They define the capacity to be the largest rate R for which there exist codes, for some (possibly large) block length n, that exhibit an arbitrarily small average error probability. They determine the capacity for these memories when the state of the defects is available only to the encoder. In [10], Heegard and El Gamal develop a similar probabilistic model for defective memory. They determine a lower bound to the capacity of these memories when complete or partial defect information is available to the encoder or to the decoder. This bound yields the capacity for several cases including all cases involving complete description of the defects. In this paper we are primarily interested only in the case when the encoder is told the state of the defects.

In Section II we introduce an algebraic model for stuck-at defects and additive errors. This model assumes that q, the alphabet size of the memory, is a power of a prime. To find codes that efficiently incorporate the defect information in the encoding process we define the class of partitioned linear block codes (PLBC's); In [6], these codes are called matched adjacent codes. The guaranteed defect and error correction capability of these codes is described.

Section III introduces the class of partitioned cyclic codes. The BCH bound for these codes is derived and used to construct partitioned linear block codes with guaranteed defect-and-error-correction capability. Two examples are given of the code construction procedure, one for $q = 2$ and one for $q = 3$. Parameters of binary codes for several block lengths are tabulated.

In Section IV, we turn our attention to the question of reliable storage of messages using partitioned linear block codes. The capacity of a class of q-ary symmetric memory cells is derived using the information-theoretic model [9], [10]. We show that the class of partitioned linear block codes achieve capacity for these memories.

The fact that partitioned linear block codes achieve capacity means that with minimum distance encoding and decoding we can achieve arbitrarily small error probability (for long block lengths) for any rate less than capacity. On the other hand, under a guaranteed defect and error correction criterion, the rate of the best PLBC's will necessarily be smaller than the capacity [6]. Thus PLBC's with suboptimal encoding and decoding, constrained to encode and decode only for defects and errors within the guaranteed correction capability of the code, will result in an asymptotic rate less than the capacity. We can conclude that while the class of PLBC's contain "good" codes in the sense of error probability, we may need to make a sacrifice in the storage rate in order to apply suboptimal encoding and decoding algorithms.

Before proceeding we note the analogy between information storage and information transmission. Natural similarities exist between the notions of space and time, memory and channel, data storage and data transmission, data retrieval and data reception. We can measure information in bits per cell or bits per second. Information and coding theory need not differentiate these notions. Thus the choice of terminology is a function of the nature of the problem to which the theory is applied. In this paper we have chosen the language of information storage as we feel that the codes that are discussed will find more application in this arena.

II. Partitioned Linear Block Codes

A. Model of a Defective Memory with Errors

We begin by developing an algebraic model for a computer memory with q-ary inputs and outputs. This model assumes both stuck-at defects and additive errors.

Let q be a power of a prime and F_q be the field with q elements. Let F_q^n denote the set of all n-tuples over F_q. (F_q^n

is an n-dimensional vector space over F_q.) Define an additional character "λ", $\tilde{F}_q = F_q \cup \{\lambda\}$ and define the "$\circ$" operator $\circ: F_q \times \tilde{F}_q \to F_q$ by

$$x \circ s = \begin{cases} x, & \text{if } s = \lambda; \\ s, & \text{if } s \neq \lambda. \end{cases}$$

An n-cell memory with defects and errors is defined by

$$y = (x \circ s) + z, \tag{2.1}$$

where x is the stored vector, z is the error vector and s is the defect vector. The addition "+" is defined over the field F_q and both + and $\circ$ operate on the vectors componentwise. It may seem more general to allow both input and output errors, e.g., $y = ((x + z_1) \circ s) + z_2)$. However, we can always redefine the error vector $z \triangleq y - ((x + z_1) \circ s) - z_2$ and obtain (2.1).

Example: Let $q = 2$, $n = 6$, $x = [0\ 0\ 0\ 0\ 0\ 0]$, $z = [0\ 0\ 0\ 1\ 1\ 1]$, and $s = [\lambda\ 0\ 1\ \lambda\ 0\ 1]$. Then $y = [0\ 0\ 1\ 1\ 1\ 0]$.

The number of defects u is equal to the number of non-λ components in s or

$$u = \max_{x \in F_q^n} \|(x \circ s) - x\|,$$

where $\|\cdot\|$ is the Hamming weight of the vector. The number of errors is defined by $t = \|z\|$. In the above example, $u = 4$, $t = 3$.

B. Block Encoders and Decoders

An (n, k) block code consists of a set of q^k messages, an encoder function that maps the message and the state vector s into an input vector x

$$f_e: \{0, 1, \cdots, q^k - 1\} \times \tilde{F}_q^n \to F_q^n$$

and a decoder function that estimates the message from the output vector y

$$f_d: F_q^n \to \{0, 1, \cdots, q^k - 1\}.$$

A code is said to be a u-defect, t-error correcting code if for every message $w \in \{0, 1, \cdots, q^k - 1\}$ and any memory with u or fewer defects and t or fewer errors

$$f_d((f_e(w, s) \circ s) + z) = w.$$

C. Definition of Partitioned Linear Block Codes

For linear block codes we take the messages as vectors in the set F_q^k. An $[n, k]$ linear block code $\mathscr{C} \subset F_q^n$ is a k-dimensional linear subspace since it forms a group under vector addition and is closed under scalar multiplication. The cardinality $|\mathscr{C}| = q^k$. Any k linearly independent vectors from the code $\mathscr{C}$ (all vectors are taken as row vectors) can be used to obtain a $k \times n$ generator matrix G. Then

$$\mathscr{C} = \{x \in F_q^n | x = wG;\ w \in F_q^k\}.$$

A parity-check matrix H is any $k \times r$ matrix $(k + r = n)$ of rank r over F_q such that $GH^t = 0_{k,r}$, the $k \times r$ zero matrix. Then

$$\mathscr{C} = \{x \in F_q^n | xH^t = \mathbf{0}\}.$$

The dual code is defined as the $[n, r]$ LBC generated by H.

An $[n, k, l]$ partitioned linear block code (PLBC) is a pair of linear subspaces $\mathscr{C}_1 \subset F_q^n$, $\mathscr{C}_0 \subset F_q^n$ of dimension k and l such that $\mathscr{C}_1 \cap \mathscr{C}_0 = \{\mathbf{0}\}$. Then the direct sum

$$\mathscr{C} \triangleq \mathscr{C}_1 + \mathscr{C}_0 \triangleq \{x = y + z | y \in \mathscr{C}_1, z \in \mathscr{C}_0\}$$

is an $[n, k + l]$ LBC with a generator matrix $G = [G_1^t, G_0^t]^t$, where G_1 generates $\mathscr{C}_1$ and G_0 generates $\mathscr{C}_0$, and $r \times n$ parity-check matrix H with $k + l + r = n$. A message-inverse matrix $\tilde{G}_1$ is defined as any $k \times n$ matrix with $G_1\tilde{G}_1^t = I_k$ (the k-dimensional identity), and $G_0\tilde{G}_1^t = 0_{l,k}$. Since a PLBC involves two generator matrices G_0 and G_1 and a parity-check matrix H, several dual codes can be defined. The (l, r)-dual code is defined as the $[n, k, r]$ PLBC with generator matrix $[\tilde{G}_1^t, G_0^t]^t$ for $\mathscr{C}$. For the (l, r)-dual code we may take G_0 as the parity-check matrix and G_1 as the message-inverse matrix.

D. Encoding and Decoding PLBC's

The following minimum distance encoding and decoding algorithms minimize probability of error for the memories which are modeled in Section IV in this paper.

Choose an $[n, k, l]$ PLBC and fix G_1, G_0, H, and $\tilde{G}_1$.

Encoding: To encode $w \in F_q^k$ store $x = wG_1 + dG_0$ where $d \in F_q^l$ is chosen to minimize $\|(x \circ s) - x\|$. Note that $xG_1^t = w$.

Decoding: Retrieve $y = (x \circ s) + z$. Compute the syndrome $s = yH^t$; note that $s = z'H^t$ where $z' = y - x$. Choose $\hat{z} \in F_q^n$ that minimizes $\|\hat{z}\|$ subject to $\hat{z}H^t = s$. Then $\hat{w} = \hat{x}\hat{G}_1^t$ where $\hat{x} = y - \hat{z}$.

E. Systematic Form

An $[n, k, l]$ PLBC is said to be in systematic form if we can find generators of the form $G_1 = [I_k\ \ 0_{k,l}\ \ P]$ and $G_0 = [R\ \ I_l\ \ Q]$ where P is a $k \times r$ matrix, R is an $l \times k$ matrix and Q is an $l \times r$ matrix. Thus for systematic generators, the message w with a "cover" sequence dR added to it forms the first k symbols of the stored vector x and the "cover message" d forms the $k + 1$ to $k + l$ components of x. In this form we may take $H = [-P^t\ \ -(Q + RP)^tI_r]$ and $G_1 = [I_k\ \ -R^t\ \ 0_{k,r}]$.

F. Correction Capability of PLBC's

The distance profile of an $[n, k]$ linear block code is defined as the set of Hamming weight enumerators $\{A_i\}$ where

$$A_i = |\{x | xH^t = \mathbf{0}, \|x\| = i\}|.$$

Given any x in the code, A_i is the number of codewords that are distance i from x. The minimum distance d for the

code is defined by

$$d = \min_{\substack{x \neq 0 \\ xH^t = 0}} \|x\|.$$

Thus d is the smallest number of linearly dependent columns of H. Note that $A_0 = 1$, $A_i = 0$ for $0 < i < d$, and $A_d > 0$.

The distance profile of an $[n, k, l]$ PLBC is defined as a pair of sets $(\{A_i\}, \{B_i\})$, where $\{A_i\}$ is the distance profile of the $[n, k + l]$ linear block code with parity-check matrix H and $\{B_i\}$ is the distance profile of the $[n, k + r]$ linear block code with parity-check matrix G_0. The distance profile for the (l, r)-dual PLBC is the pair $(\{B_i\}, \{A_i\})$. A PLBC has a pair of minimum distances (d_1, d_0) where

$$d_1 = \min_{\substack{\tilde{x}G_1^t \neq 0 \\ xH^t = 0}} \|x\|$$

and

$$d_0 = \min_{\substack{x \neq 0 \\ xG_0^t = 0}} \|x\|.$$

Note that d_1 is greater than or equal to the minimum distance of the $[n, k + l]$ linear block with parity-check matrix H, while d_0 is the minimum distance of the $[n, k + r]$ code with parity-check matrix G_0. Thus $A_0 = B_0 = 1$, $B_i = 0$ for $0 < i < d_0$, $A_{d_1} > 0$ and $B_{d_0} > 0$, yet it may be possible that $A_i > 0$ for $0 < i < d_1$.

Theorem 1: An $[n, k, l]$ with minimum distances (d_1, d_0) is a u-defect, t-error correcting code if

$$u < d_0 \quad \text{and} \quad 2t < d_1$$

or

$$u \geq d_0 \quad \text{and} \quad 2(u + t + 1 - d_0) < d_1.$$

Given an $[n, k, l]$ PLBC with distance profile $(\{A_i\}, \{B_i\})$, we can find an $[n, k, l]$ PLBC with systematic generators and the same distance profile.

A proof of this theorem can be found in the Appendix.

Example: Let $q = 2$, $n = 15$, $k = 6$, $u = 3$, and $t = 1$. We can generate a 3-defect, 1-error correcting (15, 6, 5) PLBC by

$$G_1 = \begin{bmatrix} 100000 & 00000 & 1100 \\ 010000 & 00000 & 0110 \\ 001000 & 00000 & 0011 \\ 000100 & 00000 & 1101 \\ 000010 & 00000 & 1010 \\ 000001 & 00000 & 0101 \end{bmatrix}$$

$$G_0 = \begin{bmatrix} 110010 & 10000 & 1110 \\ 011001 & 01000 & 0111 \\ 011110 & 00100 & 1101 \\ 101111 & 00010 & 0110 \\ 100101 & 00001 & 1101 \end{bmatrix}$$

$$H = \begin{bmatrix} 100110 & 10111 & 1000 \\ 110101 & 11100 & 0100 \\ 011010 & 11110 & 0010 \\ 001101 & 01111 & 0001 \end{bmatrix}$$

and

$$\tilde{G}_1 = \begin{bmatrix} 100000 & 10011 & 0000 \\ 010000 & 11100 & 0000 \\ 001000 & 01110 & 0000 \\ 000100 & 00111 & 0000 \\ 000010 & 10110 & 0000 \\ 000001 & 01011 & 0000 \end{bmatrix}$$

which has $(d_1, d_0) = (3, 4)$.

III. The Class of Partitioned Cyclic Codes

We now consider a more restrictive class of PLBC's, the class of partitioned cyclic codes. The additional structure imposed by these codes may be useful both in the implementation of encoders and decoders and in the construction of codes with specified bounds on the correction capability.

A. Polynomials

In discussing cyclic codes it is useful to consider a codeword as a polynomial over F_q of degree less than n.

Define the set of polynomials over F_q,

$$F_q(x) = \left\{ f(x) | f(x) = \sum_{i=0}^{m} f_i x^i, f_i \in F_q \right\}.$$

The set of polynomials of degree less than n is denoted by $F_q^n(x)$. We may define an isomorphism between the n-dimensional vector space and the set of polynomials of degree less than n, ψ: $F_q^n \to F_q^n(x)$ by

$$\psi: c = [c_0, c_1, \cdots, c_{n-1}] \to c(x) = \sum_{i=0}^{n-1} c_i x_i.$$

$F_q^n(x)$ forms a ring under polynomial multiplication modulo $x^n - 1$; thus the isomorphism induces a ring structure on F_q^n.

B. Cyclic Codes

An $[n, k]$ LBC $\mathscr{C}$ is a cyclic code if and only if every cyclic shift of a codeword is also a codeword. A cyclic LBC is an ideal in the ring $F_q^n(x)$ of polynomials modulo $x^n - 1$. It is known that every cyclic code has a unique monic polynomial $g(x)$ of degree r, such that

$$\mathscr{C} = \left\{ c(x) \in F_q^n(x) | c(x) = w(x)g(x) \right. \quad \left. \text{for some } w(x) \in F_q^k(x) \right\}.$$

The generator polynomial $g(x)$ must divide $x^n - 1$, thus we may define a parity-check polynomial $h(x) = (x^n - 1)/g(x)$, of degree k. Then an equivalent definition for a cyclic code $\mathscr{C}$ is

$$\mathscr{C} = \left\{ c(x) \in F_q^n(x) | c(x)h(x) = 0 \bmod x^n - 1 \right\}.$$

An $[n, k, l]$ partitioned cyclic code (PCC) is an $[n, k, l]$ PLBC such that both $\mathscr{C}$ and $\mathscr{C}_0$ are cyclic LBC's. Note that $\mathscr{C} = \mathscr{C}_1 + \mathscr{C}_0$ does not imply that $\mathscr{C}_1$ is a cyclic code. An $[n, k, l]$ PCC has two generators, $g(x)$ of degree r ($n = k$

$+ l + r)$ and $g_0(x)$ of degree $k + r$ such that $g(x)|g_0(x)$ and $g_0(x)|x^n - 1$. We might take

$$\mathscr{C}_1 = \{c(x) \in F_q^{k+r}(x) | c(x) = w(x)g(x) \text{ for some } w(x) \in F_q^k(x)\},$$

since specifying $\mathscr{C}$ and $\mathscr{C}_0$ does not uniquely define $\mathscr{C}_1$. In this case, we may take

$$G_1 = \begin{bmatrix} g(x) \\ xg(x) \\ x^2g(x) \\ \vdots \\ x^{k-1}g(x) \end{bmatrix}, \quad G_0 = \begin{bmatrix} g_0(x) \\ xg_0(x) \\ x^2g_0(x) \\ \vdots \\ x^{l-1}g_0(x) \end{bmatrix},$$

and

$$H = \begin{bmatrix} h^-(x) \\ xh^-(x) \\ x^2h^-(x) \\ \vdots \\ x^{r-1}h^-(x) \end{bmatrix},$$

where $h^-(x) = x^{k+l}h(x^{-1})$ and $h(x) = (x^n - 1)/g(x)$.

Define $h_0(x) = (x^n - 1)/g_0(x)$, the parity-check polynomial for $\mathscr{C}_0$. The (l, r)-dual code is an $[n, k, r]$ PCC with $h_0(x)$ as the generator polynomial for $\mathscr{C}$ and $h(x) = (x^n - 1)/g(s)$ as the generator polynomial for $\mathscr{C}_0$.

C. Encoding and Decoding of Partitioned Cyclic Codes

Choose an $[n, k, l]$ PCC and define $g(x)$, $g_0(x)$, $h(x)$, and $h_0(x)$.

Encoding: Store $c(x) = w(x)g(x) + d(x)g_0(x)$, where $d(x) \in F_q^l(x)$ is chosen to minimize $\|(c(x) \circ s(x)) - c(x)\|$.

Decoding: Retrieve $y(x) = (c(x) \circ s(x)) + z(x)$. Compute the syndrome $S(x) = y(x) \bmod g(x)$. Choose $\hat{z}(x) \in F_q^n(x)$ which minimizes $\|\hat{z}(x)\|$ subject to $\hat{z}(x) \bmod g(x) = S(x)$. Then

$$w(x) = \frac{(y(x) - \hat{z}(x)) \bmod g_0(x)}{g(x)}.$$

D. Factors of $x^n - 1$

In order to find a bound on the minimum distance of cyclic codes, it is useful to characterize the factors of $x^n - 1$ over both F_q and the splitting field of $x^n - 1$. We borrow our notation from MacWilliams and Sloane [11].

Assume that n and q are relatively prime, and let m be the smallest positive integer such that $n|q^m - 1$. Then F_{q^m} is the splitting field of $x^n - 1$ and

$$x^n - 1 = \prod_{i=0}^{n-1}(x - \alpha^i) = \prod_{i \in I} M^{(i)}(x),$$

where $\alpha \in F_q m$ is a primitive nth root of unity, $\{M^{(i)}(x)\}$ are the irreducible factors of $x^n - 1$ over F_q, and $I \subset \{0, n - 1\}$ is a set of coset representatives for the cyclotomic cosets modulo n over F_q. That is, if $i \in I$, then

$$M^{(i)}(\alpha^j) \begin{cases} = 0, & \text{if } j = iq^s \bmod n \text{ for some } s; \\ \neq 0, & \text{otherwise.} \end{cases}$$

E. Partitioned BCH Codes

We now introduce the class of partitioned BCH codes. A BCH-type bound for PCC's will be derived. This bound can then be used to construct codes with specified lower bounds on the minimum distances. Examples will be given.

Theorem 2: Let $g(x)$ and $g_0(x)$ be the generator polynomials for an $[n, k, l]$ PCC and let $h_0(x)$ be the parity-check polynomial for $g_0(x)$. If there exists an i, j, δ_1, and δ_0 such that

$$g(\alpha^i) = g(\alpha^{i+1}) = \cdots = g(\alpha^{i+\delta_1-2}) = 0,$$

and

$$h_0(\alpha^j) = h_0(\alpha^{j+1}) = \cdots = h_0(\alpha^{j+\delta_0-2}) = 0,$$

(i.e., $g(x)$ has a string of $\delta_1 - 1$ consecutive powers of α as zeros, etc.) then

$$d_1 \geq \delta_1,$$

and

$$d_0 \geq \delta_0.$$

The proof of Theorem 2 is given in the Appendix.

An $[n, k, l]$ partitioned BCH code of designed distances (δ_1, δ_0) is defined by

$$g(x) = \operatorname{lcm}\{M^{(i)}(x), M^{(i+1)}(x), \cdots, M^{(i+\delta_1-2)}(x)\}$$

and

$$h_0(x) = \operatorname{lcm}\{M^{(j)}(x), M^{(j-1)}(x), \cdots, M^{(j-\delta_0+2)}(x)\}$$

for some (i, j) under the restriction that $g(x)h_0(x)|x^n - 1$, (i.e., $g(x)$ and $h_0(x)$ share no common roots). Then $r \leq m(\delta_1 - 1)$, $l \leq m(\delta_0 - 1)$, $k \geq n - m(\delta_1 + \delta_0 - 2)$, $d_1 \geq \delta_1$, and $d_0 \geq \delta_0$.

For $q = 2$, $M^{(i)}(x) = M^{(2i)}(x)$; thus we have

$$r \leq m\left\lceil \frac{\delta_1 - 1}{2} \right\rceil, \quad l \leq m\left\lceil \frac{\delta_0 - 1}{2} \right\rceil$$

and

$$k \geq n - m\left(\left\lceil \frac{\delta_1 - 1}{2} \right\rceil + \left\lceil \frac{\delta_0 - 1}{2} \right\rceil\right).$$

Examples: Let $q = 2$, $n = 15$. Then

$$\begin{aligned} x^{15} + 1 &= (x + 1)(x^4 + x + 1)(x^4 + x^3 + x^2 + x + 1) \\ &\quad \cdot (x^2 + x + 1)(x^4 + x^3 + 1) \\ &= M^{(0)}(x)M^{(1)}(x)M^{(3)}(x)M^{(5)}(x)M^{(7)}(x) \end{aligned}$$

where $\alpha^4 + \alpha + 1 = 0$. Then we can construct the codes given in Table I.

Let $q = 3$, $n = 8$. Then

$$\begin{aligned} x^8 - 1 &= (x - 1)(x^2 - x - 1) \\ &\quad \cdot (x^2 + 1)(x + 1)(x^2 + x - 1) \\ &= M^{(0)}(x)M^{(1)}(x)M^{(2)}(x)M^{(4)}(x)M^{(5)}(x) \end{aligned}$$

TABLE I
PARTITIONED BINARY BCH CODES FOR $n = 15$

k	l	r	δ_0	$h_0(x)$	δ_1	$g(x)$
10	1	4	2	$M^{(0)}(x)$	3	$M^{(1)}(x)$
7	4	4	3	$M^{(7)}(x)$	3	"
6	1	8	2	$M^{(0)}(x)$	5	$M^{(1)}(x)\,M^{(3)}(x)$
6	4	5	3	$M^{(7)}(x)$	4	$M^{(0)}(x)\,M^{(1)}(x)$
4	1	10	2	$M^{(0)}(x)$	7	$M^{(1)}(x)\,M^{(3)}(x)\,M^{(5)}(x)$
3	4	8	3	$M^{(7)}(x)$	5	$M^{(1)}(x)\,M^{(3)}(x)$
2	4	9	3	"	6	$M^{(0)}(x)\,M^{(1)}(x)\,M^{(3)}(x)$
2	5	8	4	$M^{(0)}(x)\,M^{(7)}(x)$	5	$M^{(1)}(x)\,M^{(3)}(x)$
1	4	10	3	$M^{(7)}(x)$	7	$M^{(1)}(x)\,M^{(3)}(x)\,M^{(5)}(x)$

TABLE II
PARTITIONED TERNARY BCH CODES FOR $n = 8$

k	l	r	δ_0	$h_0(x)$	δ_1	$g(x)$
6	1	1	2	$M^{(4)}(x)$	2	$M^{(0)}(x)$
4	1	3	2	"	3	$M^{(0)}(x)\,M^{(1)}(x)$
3	1	4	2	"	4	$M^{(1)}(x)\,M^{(2)}(x)$
2	1	5	2	"	5	$M^{(0)}(x)\,M^{(1)}(x)\,M^{(2)}(x)$
2	3	3	3	$M^{(4)}(x)\,M^{(5)}(x)$	3	$M^{(0)}(x)\,M^{(1)}(x)$
1	3	4	3	"	4	$M^{(1)}(x)\,M^{(2)}(x)$

TABLE III
PARTITIONED BINARY BCH CODES FOR $n = 7$

k	l	r	δ_0	δ_1
3	1	3	2	3
1	3	3	3	3

TABLE IV
PARTITIONED BINARY BCH CODES FOR $n = 31$

k	l	r	δ_0	δ_1	k	l	r	δ_0	δ_1	k	l	r	δ_0	δ_1
25	1	5	2	3	15	5	11	3	6	10	10	11	5	6
21	5	5	3	3	15	6	10	4	5	6	10	15	5	7
20	1	10	2	5	11	5	15	3	7	5	10	16	5	8
20	5	6	3	4	11	10	10	5	5	5	11	15	6	7
16	5	10	3	5	10	5	16	3	8	1	15	15	7	7
15	1	15	2	7	10	6	15	4	7					

where $\alpha^2 - \alpha - 1 = 0$. Then we can construct the codes in Table II.

Tables III–IX list some partitioned binary BCH codes for block lengths 7, 31, 63, 127, 255, 511, and 1023. These codes are defined by using the construction procedure with $(i, j) = (0, n-1)$, $(1, 0)$, or $(1, n-1)$, and $l \leqslant r$. Note that for an $[n, k, l]$ partitioned BCH code of designed distance (δ_1, δ_0), the (l, r)-dual code is an $[n, k, r]$ partitioned BCH code with designed distance (δ_0, δ_1).

TABLE V
PARTITIONED BINARY BCH CODES FOR $n = 63$

k	l	r	δ_0	δ_1	k	l	r	δ_0	δ_1	k	l	r	δ_0	δ_1
58	1	6	2	3	44	7	12	4	5	35	1	27	2	11
51	6	6	3	3	39	6	18	3	7	33	6	24	3	9
50	1	12	2	5	39	12	12	5	5	33	12	18	5	7
50	6	7	3	4	38	1	24	2	9	32	6	25	3	10
45	6	12	3	5	38	6	19	3	8	32	7	24	4	9
44	1	18	2	7	38	7	18	4	7	32	12	19	5	8
44	6	13	3	6	38	12	13	5	6	32	13	18	6	7

TABLE VI
PARTITIONED BINARY BCH CODES FOR $n = 127$

k	l	r	δ_0	δ_1	k	l	r	δ_0	δ_1	k	l	r	δ_0	δ_1
119	1	7	2	3	91	14	22	5	8	77	21	29	7	10
113	7	7	3	3	91	15	21	6	7	77	22	28	8	9
112	1	14	2	5	85	7	35	3	11	71	7	49	3	15
112	7	8	3	4	85	14	28	5	9	71	14	42	5	13
108	7	14	3	5	85	21	21	7	7	71	21	35	7	11
105	1	21	2	7	84	1	42	2	13	71	28	28	9	9
105	7	15	3	6	84	7	36	3	12	70	1	56	2	19
105	8	14	4	5	84	8	35	4	11	70	7	50	3	16
99	7	21	3	7	84	14	29	5	10	70	8	49	4	15
99	14	14	5	5	84	15	28	6	9	70	14	43	5	14
98	1	28	2	9	84	21	22	7	8	70	15	42	6	13
98	7	22	3	8	78	7	42	3	13	70	21	36	7	12
98	8	21	4	7	78	14	35	5	11	70	22	35	8	11
98	14	15	5	6	78	21	28	7	9	70	28	29	9	10
92	7	28	3	9	77	1	49	2	15	64	7	56	3	19
92	14	21	5	7	77	7	43	3	14	64	14	49	5	15
91	1	35	2	11	77	8	42	4	13	64	21	42	7	13
91	7	29	3	10	77	14	36	5	12	64	28	35	9	11
91	8	28	4	9	77	15	35	6	11					

TABLE VII
PARTITIONED BINARY BCH CODES FOR $n = 255$

k	l	r	δ_0	δ_1	k	l	r	δ_0	δ_1	k	l	r	δ_0	δ_1
248	1	8	2	3	215	8	32	3	9	206	17	32	6	9
239	8	8	3	3	215	16	24	5	7	206	24	25	7	8
238	1	16	2	5	214	1	40	2	11	199	8	48	3	13
238	8	9	3	4	214	8	33	3	10	199	16	40	5	11
231	8	16	3	5	214	9	32	4	9	199	24	32	7	9
230	1	24	2	7	214	16	25	5	8	198	1	56	2	15
230	8	17	3	6	214	17	24	6	7	198	8	49	3	14
230	9	16	4	5	207	8	40	3	11	198	9	48	4	13
223	8	24	3	7	207	16	32	5	9	198	16	41	5	12
223	16	16	5	5	207	24	24	7	7	198	17	40	6	11
222	1	32	2	9	206	1	48	2	13	198	24	33	7	10
222	8	25	3	8	206	8	41	3	12	198	25	32	8	9
222	9	24	4	7	206	9	40	4	11					
222	16	17	5	6	206	16	33	5	10					

TABLE VIII
PARTITIONED BINARY BCH CODES FOR $n = 511$

k	l	r	δ_0	δ_1	k	l	r	δ_0	δ_1	k	l	r	δ_0	δ_1
501	1	9	2	3	474	9	28	3	8	457	18	36	5	9
493	9	9	3	3	474	10	27	4	7	457	27	27	7	7
492	1	18	2	5	474	18	19	5	6	456	1	54	2	13
492	9	10	3	4	466	9	36	3	9	456	9	46	3	12
484	9	18	3	5	466	18	27	5	7	456	10	45	4	11
483	1	27	2	7	465	1	45	2	11	456	18	37	5	10
483	9	19	3	6	465	9	37	3	10	456	19	36	6	9
483	10	18	4	5	465	10	36	4	9	456	27	28	7	8
475	9	27	3	7	465	18	28	5	8	448	9	54	3	13
475	18	18	5	5	465	19	27	6	7	448	18	45	5	11
474	1	36	2	9	457	9	45	3	11	448	27	36	7	9

TABLE IX
PARTITIONED BINARY BCH CODES FOR $n = 1023$

k	l	r	δ_0	δ_1	k	l	r	δ_0	δ_1	k	l	r	δ_0	δ_1
1012	1	10	2	3	982	1	40	2	9	972	21	30	6	7
1003	10	10	3	3	982	10	31	3	8	963	10	50	3	11
1002	1	20	2	5	982	11	30	4	7	963	20	40	5	9
1002	10	11	3	4	982	20	21	5	6	963	30	30	7	7
993	10	20	3	5	973	10	40	3	9	962	1	60	2	13
992	1	30	2	7	973	20	30	5	7	962	10	51	3	12
992	10	21	3	6	972	1	50	2	11	962	11	50	4	11
992	11	20	4	5	972	10	41	3	10	962	20	41	5	10
983	10	30	3	7	972	11	40	4	9	962	21	40	6	9
983	20	20	5	5	972	20	31	5	8	962	30	31	7	8

IV. PLBC'S ACHIEVE CAPACITY

A. A Random Model for Defects and Errors

A stochastic model for the generation of defects and errors in a computer memory cell is obtained by assigning probabilities to the defect and error events. The (p, ϵ, γ) q-symmetric discrete memoryless memory cell (q-SDMMC) is defined by the equation $Y = (x \circ S) + Z$, where $x, Y, Z \in F_q$, $S \in \tilde{F}_q$,

$$P(S = s) = \begin{cases} 1 - p, & s = \lambda; \\ \dfrac{p}{q}, & s \neq \lambda, \end{cases}$$

$$P(Z = z | S = \lambda) = \begin{cases} 1 - \epsilon, & z = 0; \\ \dfrac{\epsilon}{q-1}, & z \neq 0, \end{cases}$$

and

$$P(Z = z | S \neq \lambda) = \begin{cases} 1 - \gamma, & z = 0; \\ \dfrac{\gamma}{q-1}, & z \neq 0. \end{cases}$$

The probability of a defect is $0 \leqslant p \leqslant 1$, the symmetric error probability on a nondefective cell is $0 \leqslant \epsilon \leqslant (q-1)/q$, and the symmetric error probability on a stuck-at cell is $0 \leqslant \gamma \leqslant (q-1)/q$. This model allows for errors at both the input and the output. For example, $\gamma = 0$ implies only noise at the input while $\gamma = \epsilon$ implies output noise only.

B. Capacity of the q-SDMMC

Assume that a memory is composed of n statistically independent and identically distributed q-SDMMC's. The rate of an (n, k) partitioned block code is defined as the ratio $R \triangleq k/n$, and

$$P_e = P(\hat{W} \neq W) = q^{-k} \sum_{w=0}^{q^k - 1} P(f_d(\boldsymbol{Y}) \neq w | \boldsymbol{x} = f_e(w, \boldsymbol{S}))$$

is the probability of error averaged over the message W, the defect vector $\boldsymbol{S}$ and the error vector $\boldsymbol{Z}$. A rate R is said to be achievable if and only if there exist codes of that rate that exhibit arbitrarily small probability of error. That is, for any $\delta > 0$ there exists an (n, k) partitioned block code with $P_e < \delta$. The capacity C of the memory is defined as the least upper bound on the set of achievable rates.

Theorem 3: The capacity of the q-SDMMC is

$$C = 1 - p + \mathrm{ph}(\alpha) - h(\beta) + (p\alpha - \beta)\log(q-1), \tag{4.1}$$

where $h(x) = -x \log x - (1-x)\log(1-x)$, $\beta = (1-p)\epsilon + p(\alpha + \gamma - \alpha\gamma q/q - 1)$, and $0 < \alpha < 1$ is the root

$$\log \frac{\alpha}{1-\alpha} + \left(1 - \frac{\gamma q}{q-1}\right) \log \frac{1-\beta}{\beta} - \frac{\gamma q}{q-1} \log(q-1) = 0. \tag{4.2}$$

All logs are to base q.

The proof of Theorem 3 can be found in the Appendix. For $\gamma = 0$ and $\alpha = \beta = \epsilon$, $C = (1-p)(1 - h(\epsilon) - \epsilon \log(q-1))$.

To prove that PLBC's achieve C we need the following lemma.

Lemma: Fix $\delta > 0$, $0 < \epsilon < (q-1)/q$ and $R = k/n$. Let G be a random matrix chosen uniformly over the set of $k \times n$ matrices over the field F_q.

a) Let $\boldsymbol{Z} \in F_q^n$ be a random vector with independent and identically distributed components chosen according to

$$P(Z = z) = \begin{cases} 1 - \epsilon, & z = 0; \\ \dfrac{\epsilon}{q-1}, & z \neq 0. \end{cases} \tag{4.3}$$

Let $\boldsymbol{Y} = wG + \boldsymbol{Z}$ for any $w \in F_q^k$. If $R < 1 - h(\epsilon) - \epsilon \log(q-1)$ then there exists an $n_0(\delta)$ such that for every $n \geqslant n_0$

$$P\left(\min_{\substack{w' \in F_q^k \\ w' \neq w}} \|\boldsymbol{Y} - w'G\| \leqslant \|\boldsymbol{Y} - wG\| \right) < \delta.$$

b) Let $\boldsymbol{Y}$ be a random vector chosen uniformly over F_q^n. If $R > 1 - h(\epsilon) - \epsilon \log(q-1)$ then there exists an $n_0(\delta)$

such that for every $n \geq n_0$

$$P\left(\min_{w \in F_q^k} \|Y - wG\| > n\epsilon \right) < \delta.$$

This lemma is proved in the Appendix.

The first part of the lemma pertains to a channel coding problem. The channel is the q-symmetric discrete memoryless channel (q − SDMC) with parameter ϵ. The q-SDMC is defined by the equation $Y = x + Z$ where $Y, x, Z \in F_q$, and Z has the symmetric distribution (4.3). The capacity of this channel is $1 - h(\epsilon) - \epsilon \log(q-1)$. The lemma implies that linear block codes achieve capacity since for any rate R less than capacity, the expected probability of error for a randomly chosen generator matrix with minimum distance decoding, can be made arbitrarily small by increasing the block length n.

The second half of the lemma relates to a source coding problem. The q-symmetric discrete memoryless source (q-SDMS) is an independent and identically distributed source with output letters that are uniformly distributed over F_q. For the q-SDMS, the rate distortion function with Hamming distortion ϵ is given by $R(\epsilon) = 1 - h(\epsilon) - \epsilon \log(q-1)$. This is a lower bound on the rate needed to describe the q-SDMS with expected Hamming distortion at most ϵ. Again, the lemma implies that linear block codes achieve the rate distortion bound, since the expected distortion for a randomly chosen generator matrix with rate greater than $R(\epsilon)$ can be made arbitrarily close to ϵ by increasing the block length n.

Theorem 4: Given a (p, ϵ, γ) q-SDMMC and $\delta > 0$ let $\beta = (1-p)\epsilon + p(\alpha + \gamma - \alpha\gamma q/q - 1)$ for some $0 < \alpha < 1$. Fix rates $R = k/n$ and $R' = l/n$ such that

$$R + R' < 1 - h(\beta) - \beta \log(q-1)$$

and

$$R' > p(1 - h(\alpha) - \alpha \log(q-1)).$$

Then there exists an $[n, k, l]$ PLBC such that for any $W \in F_q^k$

$$P_e = P(\hat{W} \neq W) < \delta.$$

The proof of this theorem is found in the Appendix.

Theorem 4 shows that PLBC's achieve capacity. By combining the bounds on the rates, any rate R satisfying

$$\begin{aligned} R &< 1 - h(\beta) - \beta \log(q-1) - R' \\ &< 1 - p + \mathrm{ph}(\alpha) - h(\beta) + (p\alpha - \beta)\log(q-1) \end{aligned}$$

is achievable. By choosing α as the root of (4.2) we see that this bound is the capacity (4.1).

V. Closing Remarks

We conclude this paper with a short discussion of two related issues.

In some computer memories the defects may occur in "clusters" or "bursts." The presence of one defective cell may indicate that adjacent cells of the memory are also defective. A u-burst defect-correcting code must correct any u (or fewer) adjacent defects. An $[n, k, l]$ modified cyclic code is well suited to the task of correcting a single burst defect. In fact the redundancy l required by these codes is equal to the burst length u. The reason for this is the fact that any l (cyclically) consecutive columns of the generator matrix G_0 are linearly independent.

The second issue involves the splitting of defect information between the encoder and the decoder. We have assumed throughout that only the encoder or the decoder, but not both, is given the locations of the defects. It is better, in some instances, to tell the encoder the location of some defects and to tell the decoder the locations of other defects. The following example demonstrates this.

Example: Suppose that we store 10 bits ($k = 10$) in a binary memeory ($q = 2$) and protect this information against one random error and two stuck-at defects. If we use an $[n, k]$ LBC and correct both defects at the decoder, then a minimum distance $d = 5$ is required. The Hamming bound for binary codes (see [11, p. 19]),

$$2^{n-k} \geq \sum_{i=0}^{\left\lfloor \frac{d-1}{2} \right\rfloor} \binom{n}{i},$$

requires that $n \geq 18$ for $k = 10$ and $d = 5$. If we correct both defects at the encoder using an $[n, k, l]$ PLBC then minimum distances $d_1 = d_0 = 3$ are required. The Singleton bound for binary $[n, k]$ linear block codes with minimum distance d ([11, p. 33]), $n - k \geq d - 1$, can be combined with the Hamming bound to obtain

$$2^{n-k} \geq \max\left\{ 2^{d_0 - 1} \sum_{1=0}^{\left\lfloor \frac{d_1-1}{2} \right\rfloor} \binom{n}{i}, 2^{d_1 - 1} \sum_{i=0}^{\left\lfloor \frac{d_0-1}{2} \right\rfloor} \binom{n}{i} \right\}$$

for $[n, k, l]$ PLBC's with minimum distances d_1, d_0. For $k = 10$ and $d_1 = d_0 = 3$, this bound requires $n \geq 17$.

However, we can correct one defect at the encoder and one defect at the decoder by using an $[n, k, l]$ PLBC with minimum distances $d_1 = 4$ and $d_0 = 2$. The following [16, 10, 1] modified linear block code with

$$G_1 = \begin{bmatrix} 1000000000011100 \\ 0100000000011010 \\ 0010000000011001 \\ 0001000000010110 \\ 0000100000010101 \\ 0000010000010011 \\ 0000001000001110 \\ 0000000100001101 \\ 0000000010001011 \\ 0000000001000111 \end{bmatrix}$$

$$G_0 = [1111111111111111]$$

$$H = \begin{bmatrix} 1111110000110000 \\ 1110001110101000 \\ 1001101101100100 \\ 0101011011100010 \\ 0010110111100001 \end{bmatrix}$$

is such a code. Note that the block length of 16 is smaller

than the required block length when the defect information is not split.

Appendix

Proof of Theorem 1

Let G_1, G_0 be the generator matrices for an $[n, k, l]$ PLBC with minimum distances (d_1, d_0). Assume that $2(t + \max(0, u + 1 - d_0)) < d_1$ and let s, z be a u-defect, t-error memory.

To encode, let $i_1, i_2, \cdots, i_u$ be the locations of s where $s_i \neq \lambda$. Let $m = \min(d_0 - 1, u)$, $s' = [s_{i_1}, s_{i_2}, \cdots, s_{i_m}]$,

$$G_1' = \left[\boldsymbol{g}_{1,i_1}\boldsymbol{g}_{1,i_2}\cdots\boldsymbol{g}_{1,i_m}\right] \quad \text{and} \quad G_0' = \left[\boldsymbol{g}_{0,i_1}\boldsymbol{g}_{0,i_2}\cdots\boldsymbol{g}_{0,i_m}\right],$$

where $\boldsymbol{g}_{1,i}$ is the ith column of G_1, etc. Then $\operatorname{rank}(G_0') = m$ and there exists a solution $\boldsymbol{d}$ (not necessarily unique) to the equation $\boldsymbol{d}G_0' = s' + wG_1'$. For this solution, $\|(x \circ s) - x\| \leqslant \max(0, u + 1 - d_0)$.

For the decoder let $x = wG_1 + \boldsymbol{d}G_0$, $x' = w'G_1 + \boldsymbol{d}'G_0$, where $w \neq w'$, and $y = (x \circ s) + z$. Then

$$\|y - x\| = \|(x \circ s) - x + z\| \leqslant t + \max(0, u + 1 - d_0)$$

and

$$\|y - x'\| = \|(x \circ s) - x' + z\| \geqslant d_1 - t - \max(0, u + 1 - d_0).$$

Since $t + \max(0, u + 1 - d_0) < d_1 - t - \max(0, u + 1 - d_0)$, we conclude $\|y - x\| < \|y - x'\|$.

For a given pair of generators G_1 and G_0 we can find invertible matrices A_0, A_1, and B and a permutation matrix π such that $G_0' = A_0G_0\Pi$ and $G_1' = A_1(G_1\Pi - BG_0')$ are of the form $G_0' = [RI_lQ]$ and $G_1' = [I_k 0_{k,l} P]$, without disturbing the distance profile of the code.

Proof of Theorem 2

The proof of this theorem follows directly from the BCH bound for cyclic codes (see [11, p. 201]) and the following facts:

a) The minimum distance d_0 is equal to the minimum distance of the linear block code generated by $h_0(x)$.
b) The minimum distance d_1 satisfies

$$d_1 \geqslant \min_{\substack{xH^t = 0 \\ x \neq 0}} \|x\|.$$

Thus d_1 is bounded by the minimum distance of the linear block code generated by $g(x)$.

Proof of Theorem 3

The capacity C for a $(\mathcal{S}, p(s), \mathcal{X}, p(y|x, s), \mathcal{Y})$ discrete memoryless memory cell [9], [10] is given by

$$C = \max_{p(u, x|s)} I(U; Y) - I(U; S).$$

For the q-SDMMC, let $U = X$, $\bar{\alpha} = 1 - \alpha$, and

$$p(u|s) = \begin{cases} \dfrac{1}{q}, & s = \lambda; \\ \bar{\alpha}, & u = s \neq \lambda; \\ \dfrac{\alpha}{q-1}, & u \neq s \neq \lambda. \end{cases}$$

Then

$$\begin{aligned} I(U; Y) - I(U; S) &= H(U|S) - H(U|Y) \\ &= \bar{p}\log(q) + p(h(\alpha) + \alpha\log(q-1)) \\ &\quad - h(\beta) - \beta\log(q-1) \\ &= \bar{p} + \mathrm{ph}(\alpha) - h(\beta) + (p\alpha - \beta)\log(q-1) \end{aligned}$$

where $\bar{p} = 1 - p$, $h(x) = -x\log(x) - x\log(x)$ and $\beta = \bar{p}\epsilon + p(\alpha + \gamma - \alpha\gamma(q/(q-1)))$. Taking the derivative with respect to α we get

$$p\log\left(\frac{\bar{\alpha}}{\alpha}\right) - p\left(1 - \frac{\gamma q}{q-1}\right)\log\left(\frac{\bar{\beta}}{\beta}\right) + (p\gamma q/q - 1)\log(q-1).$$

Setting the derivative equal to zero is equivalent to finding the root of

$$\log\left(\frac{\alpha}{\bar{\alpha}}\right) + \left(1 - \frac{\gamma q}{q-1}\right)\log\left(\frac{\bar{\beta}}{\beta}\right) - \frac{\gamma q}{q-1}\log(q-1) = 0.$$

Proof of Lemma

a) Fix $0 < \gamma < (q - 1/q) - \epsilon$ such that $\boldsymbol{R} < 1 - h(\epsilon + \gamma) - (\epsilon + \gamma)\log(q-1)$, with base q logs.

Since $\boldsymbol{Y} - w\boldsymbol{G} = \boldsymbol{Z}$ and $\boldsymbol{Y} - w'\boldsymbol{G} = \boldsymbol{Z} - (w' - w)G$,

$$\begin{aligned} &P\left[\min_{\substack{w' \in F_q^k \\ w' \neq w}} \|\boldsymbol{Y} - w'G\| \leqslant \|\boldsymbol{Y} - w\boldsymbol{G}\|\right] \\ &= P\left[\min_{\substack{w \in F_q^k \\ w \neq \boldsymbol{0}}} \|\boldsymbol{Z} - w\boldsymbol{G}\| \leqslant \|\boldsymbol{Z}\|\right] \\ &\leqslant P[\|\boldsymbol{Z}\| \geqslant n(\epsilon + \gamma)] + \sum_{w \in F_q^k} [P\|\boldsymbol{Z} - w\boldsymbol{G}\| \leqslant n(\epsilon + \gamma)] \\ &\leqslant P[\|\boldsymbol{Z}\| \geqslant n(\epsilon + \gamma)] + q^{k-n}\sum_{i=0}^{n(\epsilon+\gamma)} \binom{n}{i}(q-1)^i \\ &\leqslant P[\|\boldsymbol{Z}\| \geqslant n(\epsilon + \gamma)] + q^{k-n}q^{nh(\epsilon+\gamma)}(q-1)^{n(\epsilon+\gamma)} \\ &= P[\|\boldsymbol{Z}\| \geqslant n(\epsilon + \gamma)] + q^{n(R-1+h(\epsilon+\gamma)+(\epsilon+\gamma)\log(q-1))}. \end{aligned}$$

By the law of large numbers, there exists an n_1 such that for all $n \geqslant n_1$

$$P(\|Z\| \geqslant n(\epsilon + \gamma)) \leqslant \frac{\delta}{2}.$$

Since $R < 1 - h(\epsilon + \gamma) - (\epsilon + \gamma)\log(q-1)$, there exists an n_2 such that for all $n \geqslant n_2$

$$q^{n(R-1+h(\epsilon+\gamma)+(\epsilon+\gamma)\log(q-1))} \leqslant \frac{\delta}{2}.$$

Thus we may take $n_0 = \max(n_1, n_2)$.

b) This proof is suggested by a technique due to El Gamal [12]. Fix $0 \leqslant \gamma \leqslant 1$, and define

$$\mathcal{C} = \left\{x \in F_q^n | wG = x \text{ for some } w \in F_q^k\right\}$$

and

$$A_Y = \left\{z \in F_q^n | Y + z \in \mathcal{C}, \|z\| \leqslant n\epsilon\right\}.$$

Then

$$P\left[\min_{w \in F_q^l} \|\boldsymbol{Y} - w\boldsymbol{G}\| > n\epsilon\right] = P[|A_Y| = 0].$$

Using Chebyshev's inequality, one gets

$$P[|A_Y| = 0] \leqslant P\left[\,\big||A_Y| - E|A_Y|\big| \geqslant \gamma E|A_Y|\right] \leqslant \frac{E[|A_Y| - E|A_Y|]^2}{\gamma^2[E|A_Y|]^2} = \frac{\sigma^2[|A_Y|]}{\gamma^2[E|A_Y|]^2}. \quad \text{(A1)}$$

We may write

$$A_Y = \sum_{\substack{x \in F_q^n \\ \|x\| \leqslant n\epsilon}} 1_c(Y + z) = \sum_{i=1}^{N} 1_c(Y + z_i)$$

where

$$1_c(x) \triangleq \begin{cases} 1, & x \in \mathscr{C}; \\ 0, & \text{otherwise}, \end{cases}$$

$$N = \sum_{i=0}^{n\epsilon} \binom{n}{i}(q-1)^i,$$

and $z_1, z_2, \cdots, z_N$ is a lexicographical ordering of all length n sequences with Hamming weight $\leqslant n\epsilon$.

Now

$$E|A_Y| = \sum_{i=1}^{N} E[1_c(Y + z_i)] = \sum_{i=1}^{N} P[Y + z_i \in \mathscr{C}]$$

and

$$\begin{aligned} P[Y + z_i \in \mathscr{C}] &\geqslant P[\operatorname{rank}(G) = k]\,P[Y + z_i \in \mathscr{C} | \operatorname{rank}(G) = k] \\ &= [1 - P[\operatorname{rank}(G) < k]]\,q^{k-n} \\ &= \left[1 - P\left[\bigcup_{\substack{w \in F_q^k \\ w \neq 0}} \{wG = 0\}\right]\right] q^{k-n} \\ &\geqslant [1 - q^{k-n}]\,q^{k-n} \end{aligned}$$

where the last step follows from the union of events bound. Thus

$$E|A_Y| \geqslant Nq^{k-n}[1 - q^{k-n}]$$

and

$$[E|A_Y|]^2 \geqslant N^2 q^{2(k-n)}[1 - 2q^{k-n}]. \quad \text{(A2)}$$

Next,

$$\begin{aligned} E(|A_Y|)^2 &= \sum_{i=0}^{N} E[1_c(Y + z_i)] \\ &\quad + \sum_{i=0}^{N} \sum_{\substack{j=0 \\ j \neq i}}^{N} E[1_c(Y + z_i)1_c(Y + z_j)] \\ &= \sum_{i=0}^{N} P[Y + z_i \in \mathscr{C}] \\ &\quad + \sum_{i=0}^{N} \sum_{\substack{j=0 \\ j \neq i}}^{N} P[Y + z_i \in \mathscr{C}, Y + z_j \in \mathscr{C}] \end{aligned}$$

and

$$P[Y + z_i \in \mathscr{C}] = P\left[\bigcup_{w \in F_q^k} \{wG = Y + z_i\}\right] \leqslant q^{k-n}.$$

Also,

$$\begin{aligned} &P[Y + z_i \in \mathscr{C}, Y + z_j \in \mathscr{C}] \\ &\quad = P\left[\bigcup_{x \in F_q^k} \bigcup_{w \in F_q^k} \{xG = Y + z_i, wG = Y + z_j\}\right] \\ &\quad \leqslant \sum_{x \in F_q^k} \sum_{w \in F_q^k} P[(x - w)G = z_i - z_j] \\ &\qquad \cdot P[xG = Y + z_i | (x - w)G = z_i - z_j]. \end{aligned}$$

Since Y is independent of G,

$$P[xG = Y + z_i | (x - w)G = z_i - z_j] = P[Y = xG - z_i] = q^{-n},$$

and so we have

$$P[Y + z_i \in \mathscr{C}, Y + z_j \in \mathscr{C}] \leqslant q^{2(k-n)}$$

and

$$E(|A_Y|)^2 \leqslant Nq^{k-n} + N^2 q^{2(k-n)}. \quad \text{(A3)}$$

Combining (A2) and A3) we have

$$\begin{aligned} \sigma^2[|A_Y|] &\leqslant Nq^{k-n} + 2N^2 q^{3(k-n)} \\ &= N^2 q^{2(k-n)}[N^{-1}q^{n-k} + 2q^{k-n}]. \end{aligned}$$

Thus from (A1)

$$P(|A_Y| = 0) \leqslant \frac{[N^{-1}q^{n-k} + 2q^{k-n}]}{\gamma^2[1 - 2q^{k-n}]}.$$

Since

$$N = \sum_{i=0}^{n\epsilon} \binom{n}{i}(q-1)^i \geqslant \frac{q^{nh(\epsilon)}(q-1)^{n\epsilon}}{\sqrt{cn}},$$

where $c = 8\epsilon(1 - \epsilon)$([11, p. 310]),

$$\begin{aligned} N^{-1}q^{n-k} &\leqslant \sqrt{cn}\,q^{n-k-nh(\epsilon)}(q-1)^{-n\epsilon} \\ &= q^{n(1 - R - h(\epsilon) - \epsilon\log(q-1) + (1/2n)\log(cn))}, \end{aligned}$$

which approaches zero for large n since $R > 1 - h(\epsilon) - \epsilon\log(q - 1)$. Similarly, $q^{k-n} = q^{n(R-1)}$ goes to zero since $R < 1$, and we conclude that there exists an n_0 such that $P(|A_Y| = 0) \leqslant \delta$ for all $n \geqslant n_0$.

Proof of Theorem 4

Fix $w \in F_q^k$ and small $\lambda > 0$ such that

$$R' > (p + \lambda)(1 - h(\alpha) > \alpha\log(q-1)),$$

$$R + R' < 1 - h(\beta') - \beta'\log(q-1),$$

where $\beta' \triangleq (1 - \lambda)\beta + 2\lambda((q-1)/q)$ and $\beta' < (q-1)/q$.

Define the two random variables

$$T = \max_{x \in F_q^n} \|(x \circ S) - x\|$$

$$Q = \min_{d \in F_q^l} \|((wG_1 + dG_0) \circ S) - wG_1 - dG_0\|.$$

Let $x = wG_1 + dG_0$ where $\|(x \circ S) - x\| = Q$ and $Y = (x \circ S) + Z$. Then

$$P_e = P(\hat{W} \neq W)$$

$$P\left[\min_{\substack{w' \in F_q^k \\ w' \neq w}} \min_{d' \in F_q} \|Y - x'\| \leqslant \|Y - x\|\right]$$

where $x' = w'G_1 + d'G_0$. Thus

$$P_e \leqslant P[|T - np| > n\lambda] + P[|T - np| \leqslant n\lambda, Q > n(p+\lambda)\alpha]$$
$$+ P\left[\min_{\substack{w' \in F_q^k \\ w' \neq w}} \min_{d' \in F_q} \|Y - x'\| \leqslant \|Y - x\|, \text{ for } |T - np| \leqslant n\lambda, \right.$$
$$\left. Q \leqslant n(p+\lambda)\alpha\right].$$

By the law of large numbers, there exists an n_1 such that for all $n \geqslant n_1$

$$P[|T - np| > n\lambda] < \frac{\delta}{3}.$$

By applying the proof of the lemma (part a)), we see that if we randomly chose G_0 there exists an n_2 such that for all $n \geqslant n_2$

$$P[|T - np| \leqslant n\lambda, Q > n(p+\lambda)\alpha] < \frac{\delta}{3}$$

since $R' > (p+\lambda)(1 - h(\alpha) - \alpha \log(q-1))$.

Finally, on the set

$$\{|T - np| \leqslant n, Q \leqslant n(p+\lambda)\alpha\}$$

$\tilde{Z} = Y - x$ is an independent identically distributed random vector with components chosen according to

$$P(\tilde{Z} = z) = \begin{cases} 1 - \rho, & z = 0; \\ \dfrac{\rho}{q-1}, & z \neq 0, \end{cases}$$

where $\rho \leqslant \beta'$. From the proof of the lemma (part b)), if we randomly choose G_1 and G_0, there exists an n_3 such that for all $n \geqslant n_3$,

$$P\left(\min_{\substack{w' \in F_q^k \\ w' \neq w}} \min_{d' \in F_q} \|Y - x'\| \leqslant \|Y - x\|, \text{ for } |T - np| \leqslant n\lambda, \right.$$
$$\left. Q \leqslant n(p+\lambda)\alpha\right) < \frac{\delta}{3}$$

since $R + R' < 1 - h(\beta') - \beta' \log(q-1)$.

By letting $n > \max(n_1, n_2, n_3)$ we see that there exist PLBC's with $P_e < \delta$, concluding the proof of the theorem.

Acknowledgment

The author would like to thank Thomas Cover and Abbas El Gamal for their encouragement and support during the course of this research. The author also would like to thank the reviewers for their suggested improvements to the presentation.

References

[1] A. V. Kusnetsov and B. S. Tsybakov, "Coding in a memory with defective cells," *Problemy Peredach. Informatsii*, vol. 10, no. 2, pp. 52–60, April–June 1974.

[2] I. B. Belov and A. M. Shashin, "Codes that correct triple defects in memory," *Problemy Peredach. Informatsii*, vol. 13, no. 4, pp. 62–65, Oct.–Dec. 1977.

[3] V. V. Losev, V. K. Konopel'ko, and Yu.D. Daryakin, "Double-and-triple-defect-correcting codes," *Problemy Peredachi. Informatsii*, vol. 14, no. 4, pp. 98–101, Oct.–Dec. 1978.

[4] A. V. Kuznetsov, "Masking triple fixed defects in memory," preprint Institute for Problems of Information Transmission, USSR Academy of Sciences.

[5] B. S. Tsybakov, "Additive group codes for defect correction," *Problemy Peredach. Informatsii*, vol. 11, no. 1, pp. 111–113, Jan.–Mar. 1975.

[6] ——, "Defect and error correction," *Problemy Peredach. Informatsii*, vol. 11, no. 3, pp. 21–30, July–Sept. 1975.

[7] A. V. Kuznetsov, T. Kasami, and S. Tamamura, "An error correcting scheme for defective memory," *IEEE Trans. Inform. Theory*, vol. IT 24, no. 6, pp. 712–718, Nov. 1978.

[8] B. S. Tsybakov, S. I. Gel'fand, A. V. Kuznetsov, and S. I. Ortyukov, "Reliable computation and reliable storage of information," in *Proc. 1975 IEEE–USSR Joint Workshop on Inform. Theory*, pp. 215–226, Dec. 15–19, 1975.

[9] S. I. Gel'fand and M. S. Pinsker, "Coding for channel with random parameters," *Probl. Contr. and Inform. Theory*, vol. 9 (1), pp. 19–31, 1980.

[10] C. Heegard and A. El Gamal, "On the Capacity of Computer Memory with Defects," *IEEE Trans. Inform. Theory*, vol. IT-29, pp. 731–739, Sept. 1983.

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

[12] A. El Gamal and E. van der Meulen, "A proof of Marton's coding theorem for the discrete memoryless broadcast channel," *IEEE Trans. Inform. Theory*, vol. IT-27, Jan. 1981.

[13] J. J. Stiffler, "Coding for random-access memories," *IEEE Trans. Comput.*, vol. C-27, no. 6, June 1978.