ON CODES WITH MULTI-LEVEL ERROR-CORRECTION CAPABILITIES

March 1, 1987

Technical Report

to

NASA
Goddard Space Flight Center
Greenbelt, Maryland

Grant Number NAG 5-407 SA-1
July 1, 1985 - June 30, 1986

Shu Lin
Principal Investigator
Department of Electrical Engineering
University of Hawaii at Manoa
Honolulu, Hawaii 96822

ON CODES WITH MULTI-LEVEL ERROR-CORRECTION CAPABILITIES*

Mao-Chao Lin and Shu Lin

University of Hawaii

Department of Electrical Engineering

Honolulu, Hawaii 96822

ABSTRACT

In conventional coding for error control, all the information symbols of a message are regarded equally significant, and hence codes are devised to provide equal protection for each information symbol against channel errors. However, in some occasions, some information symbols in a message are more significant than the other symbols. As a result, it is desired to devise codes with multi-level error-correcting capabilities. Another situation where codes with multi-level error-correcting capabilities are desired is in broadcast communication systems. An m-user broadcast channel has one input and m outputs. The single input and each output form a component channel. The component channels may have different noise levels, and hence the messages transmitted over the component channels require different levels of protection against errors. In this research, we investigate block codes with multi-level error-correcting capabilities, which are also are known as unequal error protection (UEP) codes. Structural properties of these codes are derived. Based on these structural properties, two classes of UEP codes are constructed. A subclass of codes

I. INTRODUCTION

In conventional channel coding, all the information symbols of a message are regarded equally significant, and hence redundant (or parity-check) symbols are added to provide equal protection for each information symbol against channel errors. However, on some occasions, some information symbols in a message are more significant than other information symbols in the same message. Therefore, it is desirable to devise coding schemes which provide higher protection for the more significant information symbols and lower protection for the less significant information symbols. Suppose a message from an information source consists of m parts, each has a different level of significance and requires a different level of protection against channel errors. An obvious way to accomplish this is to use a separate code for each message part and then time share the codes. The redundant symbols of each code are designed to provide an appropriate level of error-correcting capability for the corresponding message part. This coding scheme requires a separate encoder and decoder pair for each code. A more efficient way is to devise a single code for all the message parts. The redundant symbols are designed to provide m levels of error protection for the m parts of a message. It has been proved that a single code with m levels of error-correcting capability usually requires less redundant symbols than that required by time-sharing m separate codes with the same m levels of error-correcting capability [1-9]. Moreover, a single code requires only one encoder and one decoder. This may be desirable

in many situations. A code with multi-level error-correcting capabilities is known as an unequal error protection (UEP) code. UEP codes were first studied by Masnick and Wolf [1], then by other coding theorists [6,7,10-18]. Another situation where codes with multi-level error-correcting capabilities are desired is in a broadcast channel communication system as shown in Figure 1, in which m independent information sources attempt to transmit information to m separate users through a single transmitter. Only message $\bar{x}_i$ emanating from the i-th source is intended to be recovered by the i-th decoder (or user). The m messages emanating from the m sources are encoded by a single encoder into a single codeword $\bar{v}(\bar{x}_1,\bar{x}_2,\ldots,\bar{x}_m)$. This codeword is then transmitted to the m users over a broadcast channel which has a single input and m outputs. Each output of the channel is connected to a decoder for the corresponding user. Each decoder receives a vector which is a corrupted version of the transmitted codeword $\bar{v}(\bar{x}_1,\bar{x}_2,\ldots,\bar{x}_m)$. For $1\leq i\leq m$ let $\bar{r}_i$ be the vector received by the i-th decoder. Then, the i-th decoder decodes $\bar{r}_i$ into $\bar{x}_i^*$ which is an estimate of the message $\bar{x}_i$ produced by the i-th source. The decoders do not collaborate with each other. The broadcast channel actually consists of m component channels, where the i-th component channel consists of the input terminal and the i-th output terminal of the broadcast channel. These m component channels may have different noise levels, and hence the m messages transmitted over the component channels require different levels of protection against errors. Consequently,

codes with multi-level error-correcting capabilities are desired. Coding for broadcast channels has recently been studied by Heegard, dePedro and Wolf [9], Dowey and Karlof [19], Bassalygo, et. al., [7], and Kasami, et. al. [8].

In this paper we investigate codes with multi-level error-correction capabilities. We intend to unify the concepts that have been separately developed for the single user communications and the multi-user broadcast communications. Two classes of multi-level UEP codes are presented. In this paper we use the terms, multi-level error-correction codes and multi-level UEP codes, interchangeably.

II. BASIC CONCEPTS

A. Cloud Structures of Block Codes and the Associated Separation Vectors

Let $A_1, A_2, \ldots, A_m$ be m message spaces. A message from A_i is denoted by $\bar{x}_i$. Consider the following set of m-tuples:

$$A = \{ (\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_m) : \bar{x}_i \in A_i \text{ for } 1 \le i \le m \} \quad (1)$$

The set A is called the product of $A_1, A_2, \ldots, A_m$, and A_i is called the i-th component message space of the message space A. Accordingly, $\bar{x}_i$ is called the i-th component message of the message $(\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_m)$ from A. Let $|S|$ denote the cardinality of a set S. Then

$$|A| = |A_1| \times |A_2| \times \ldots \times |A_m|.$$

A special case is that, for $1 \le i \le m$, the i-th component message space A_i consists of all the 2^{k_i} k_i-tuples over GF(2). In this

case, each message in A is a k-tuple over GF(2), where

$$k = k_1+k_2+\ldots+k_m.$$

In a single-user communication system, A is the message space for the single information source with every message in A being partitioned into m parts. For a multi-user communication system, A_i is simply the message space for the i-th information source of the system. Without loss of generality, we assume that messages from A_1 have the highest level of significance, messages from A_2 have the second highest level of significance, ... , and the messages from A_m have the lowest level of significance.

Let n be a positive integer such that

$$n \geq \lceil \log_2 |A| \rceil,$$

where $\lceil q \rceil$ denotes the smallest integer greater than or equal to the number q. Let C be a binary block code of length n for the message space A. Then C is a subset of $\{0,1\}^n$, the vector space of all n-tuples over GF(2). If C is a subspace of $\{0,1\}^n$, then C is a linear block code for A. The codeword which corresponds to the message $(\bar{x}_1,\bar{x}_2, \ldots ,\bar{x}_m)$ is denoted by $\bar{v}(\bar{x}_1,\bar{x}_2, \ldots ,\bar{x}_m)$.

Let $\bar{v}$ and $\bar{w}$ be two n-tuples in $\{0,1\}^n$. The Hamming distance between $\bar{v}$ and $\bar{w}$, denoted by $d(\bar{v},\bar{w})$ is defined as the number of places where $\bar{v}$ and $\bar{w}$ differ. The minimum distance of C is defined as

$$d_{min} = \min \{d(\bar{v},\bar{w}) : \bar{v}, \bar{w} \in C, \bar{v} \neq \bar{w}\}. \qquad (2)$$

In conventional coding for a single user, a code is designed to provide uniform (or equal) error protection for every component message of a message. The error correction capability is determined by the minumum distance d_{min} of the code. Every

component message can be correctly decoded if there are

$$t = \lfloor (d_{min}-1)/2 \rfloor$$

or fewer errors in the received word, where $\lfloor q \rfloor$ denotes the largest integer less than or equal to the number q.

However, for designing codes with multi-level error-correction capabilities, a different distance measure is needed. Let V and W be two subsets of vectors in $(0,1)^n$. We define the separation between V and W, denoted by d(V,W), as follows:

$$d(V,W) = \min\{d(\bar{v},\bar{w}) : \bar{v} \in V \text{ and } \bar{w} \in W\}. \quad (3)$$

Let C be a block code for the product message space $A = A_1 x A_2 x \ldots x A_m$. Let $\bar{a}$ be a specific message in A_i. Consider the following subset of codewords in C,

$$Q_i(\bar{a}) = \{\bar{v}(\bar{x}_1, \ldots, \bar{x}_{i-1}, \bar{a}, \bar{x}_{i+1}, \ldots, \bar{x}_m) : \bar{x}_j \in A_j \text{ for } 1 \le j \le m \text{ and } j \neq i\}. \quad (4)$$

Clearly, there are

$$|Q_i(\bar{a})| = \prod_{\substack{j=1 \\ j \neq i}}^{m} |A_j|$$

codewords in $Q_i(\bar{a})$. We call the set $Q_i(\bar{a})$ an i-cloud of C corresponding to the message $\bar{a}$ in A_i. There are $|A_i|$ i-clouds in C corresponding to $|A_i|$ messages in A_i. These i-clouds form a disjoint partition of C, i.e.,

$$C = \bigcup_{\bar{a} \in A_i} Q_i(\bar{a}) \quad \text{and} \quad Q_i(\bar{a}) \cap Q_i(\bar{b}) = \phi$$

for $\bar{a} \neq \bar{b}$. The codewords in an i-cloud are called satellites. Consider two distinct i-clouds, $Q_i(\bar{a})$ and $Q_i(\bar{b})$. The separation(or distance) between $Q_i(\bar{a})$ and $Q_i(\bar{b})$ is

$d(Q_i(\bar{a}), Q_i(\bar{b}))$. Then, the minimum separation of the i-clouds is defined as

$$s_i = \min\{d(Q_i(\bar{a}), Q_i(\bar{b})) : \bar{a}, \bar{b} \in A_i \text{ and } \bar{a} \neq \bar{b}\}. \quad (5)$$

Geometrically, we may view the code C as partitioned into $|A_i|$ i-clouds, where any two i-clouds are separated by a distance of at least s_i. From (4) and (5), it is clear that

$$s_i = \min\{d(\bar{v}(\bar{x}_1, \ldots, \bar{x}_i, \ldots, \bar{x}_m), \bar{v}(\bar{x}_1', \ldots, \bar{x}_i', \ldots, \bar{x}_m')) : \bar{x}_\ell, \bar{x}_\ell' \in A_\ell \text{ for } 1 \leq \ell \leq m \text{ and } \bar{x}_i \neq \bar{x}_i'\}. \quad (6)$$

The m-tuple

$$\bar{s} = (s_1, s_2, \ldots, s_m)$$

is called the separation vector of code C. It follows from (2) and (6) that the minimum distance d_{min} of the code is equal to the minimum component of the separation vector $\bar{s}$, i.e.,

$$d_{min} = \min\{s_i : 1 \leq i \leq m\}. \quad (7)$$

In the following we will show that the minimum separation s_i of the i-clouds indicates the level of error protection for the i-th component message $\bar{x}_i$.

Lemma 1: Let V and W be two subsets of $\{0,1\}^n$. For any arbitrary vector $\bar{r}$ in $\{0,1\}^n$, the following inequality holds,

$$d(\{\bar{r}\},V) + d(\{\bar{r}\},W) \geq d(V,W). \quad (8)$$

Proof: See Appendix A.

ΔΔ

Now we devise a decoding algorithm for C for which each component message $\bar{x}_i \in A_i$ is decoded independently. Suppose

some codeword $\bar{v}$ is transmitted. Let $\bar{r}$ be the received vector. To decode the i-th component message, we need to compute the distance $d(\{\bar{r}\},Q_i(\bar{x}_i))$ between $\bar{r}$ and each i-cloud $Q_i(\bar{x}_i)$. Let $Q_i(\bar{a})$ be the i-cloud such that $d(\{\bar{r}\},Q_i(\bar{a}))$ is the smallest, i.e.

$$d(\{\bar{r}\},Q_i(\bar{a})) < d(\{\bar{r}\},Q_i(\bar{x}_i))$$

for $\bar{x}_i \neq \bar{a}$. Then the i-th component message is decoded into $\bar{a}$. The i-th component message will be decoded correctly provided that there are

$$\lfloor (s_i-1)/2 \rfloor$$

or fewer transmission errors in the received vector $\bar{r}$. To see this, let $\bar{v} = \bar{v}(\bar{x}_1,\ldots,\bar{x}_i,\ldots,\bar{x}_m)$ be the transmitted codeword. Let $\bar{x}_i' \neq \bar{x}_i$. It follows from Lemma 1 that

$$d(\{\bar{r}\},Q_i(\bar{x}_i))+d(\{\bar{r}\},Q_i(\bar{x}_i')) \geq d(Q_i(\bar{x}_i),Q_i(\bar{x}_i'))$$

Since

$$d(Q_i(\bar{x}_i),Q(\bar{x}_i')) \geq s_i, \tag{9}$$

we have

$$d(\{\bar{r}\},Q_i(\bar{x}_i')) \geq s_i - d(\{\bar{r}\},Q_i(\bar{x}_i)). \tag{10}$$

However,

$$d(\bar{r},\bar{v}) \geq d(\{\bar{r}\},Q_i(\bar{x}_i)). \tag{11}$$

From (10) and (11), we obtain the following inequality,

$$d(\{\bar{r}\},Q(\bar{x}_i')) \geq s_i - d(\bar{r},\bar{v}). \tag{12}$$

If there are $t_i=\lfloor (s_i-1)/2 \rfloor$ or fewer transmission errors in $\bar{r}$, then

$$d(\bar{r},\bar{v}) \leq t_i. \tag{13}$$

It follows from (11) to (13) that

$$d(\{\bar{r}\},Q_i(\bar{x}_i)) \leq t_i,$$

and

$$d((\bar{r}),Q_i(\bar{x}_i')) > t_i.$$

Hence,

$$d((\bar{r}),Q_i(\bar{x}_i)) < d((\bar{r}),Q_i(\bar{x}_i')) \qquad (14)$$

for $\bar{x}_i' \neq \bar{x}_i$. Based on the decoding algorithm described above, the i-th component message is decoded into $\bar{x}_i$. This results in a correct decoding.

We have shown that the minimum separation s_i of the i-clouds of a code determines the level of protection for the i-th component message $\bar{x}_i$. Summarizing the above results, we have Theorem 1.

Theorem 1: Let C be a block code for the product of m message spaces, A_1, A_2, ... , A_m. Let $\bar{s} = (s_1,s_2,\ldots,s_m)$ be the separation vector of C. Then, for $1 \leq i \leq m$, the i-th component message $\bar{x}_i$ contained in a received word can be correctly decoded provided that the number of transmission errors in the recieved word is $\lfloor (s_i-1)/2 \rfloor$ or less.

ΔΔ

Suppose $s_i > s_j$. We see readily that if there are $\lfloor (s_i-1)/2 \rfloor$ or fewer transmission errors in a received word, the i-th component message $\bar{x}_i$ can always be decoded correctly but the j-th component message $\bar{x}_j$ may not be decoded correctly. However, if there are $\lfloor (s_j-1)/2 \rfloor$ or fewer transmission errors, both component messages, $\bar{x}_i$ and $\bar{x}_j$, can be decoded correctly. The parameter

$$t_i = \lfloor (s_i-1)/2 \rfloor$$

is referred to as the level of error protection for the i-th component message. A code C with a separation vector $\bar{s} = (s_1,s_2,\ldots,s_m)$ is called a $(t_1,t_2,\ldots,t_m)$-error-correcting code

with $t_i=\lfloor(s_i-1)/2\rfloor$ for $1\le i\le m$. If not all the t_i's are equal, code C provides unequal error protection for the component messages in the product message space $A = A_1xA_2x...xA_m$. If all the t_i's are different, then C provides m distinct levels of error protection, one for each component message. We call C an m-level UEP code or m-level error-correction code. For the case where $t_1=t_2=...=t_m$, the code provides equal error protection for all the component messages. Then C becomes a conventional error-correcting code.

Without loss of generality, we assume that $s_1\ge s_2\ge...\ge s_m$. In a single-user communication system, we simply regard that the first component message $\bar{x}_1$ is most significant, and hence it requires the highest level of error protection. The m-th component message $\bar{x}_m$ is least significant, and hence it requires the least protection. In a broadcast communication system with m information sources as shown in Figure 1, the first component channel is regarded as the noisiest channel. Hence, a word received by user-1 contains the most errors. Therefore, the first component message $\bar{x}_1$ needs more error protection than other component messages.

In this paper we only consider multi-level UEP codes for either the single-user binary symmetric channel(BSC) or the multi-user binary symmetric broadcast channel(BSBC). For an m-user BSBC, each component channel is a BSC with certain transition probability.

Linear unequal error protection codes were first studied by

Masnick and Wolf[1]. The concept of separation vector for unequal error protection codes was first introduced by Dunning and Robbins [13]. The separation vector defined in this paper is a generalized version of Dunning and Robbins', which applies for linear or nonlinear codes, single user or multi-user coding.

Note that the minimum separation s_i for the i-th clouds depends on how a code is partitioned into the i-th clouds. Different encodings (or mappings) of A onto C yields different partitions of C. As a result, the separation vector of C depends on the encoding mapping. This is best illustrated by an example.

Example 1: Consider the product A of two component message spaces, $A_1=A_2=\{0,1\}$. Hence, $A=\{0,1\}^2$ and each message $\bar{u}$ in A is of the form (u_1,u_2) with $u_1 \in A_1$ and $u_2 \in A_2$. Let C={(0000), (1111), (1110), (0001)} be a linear block code for A. Consider the two encoding mappings shown in Tables 1-(a) and 1-(b).

Table 1

Encoding (a)

message (u_1,u_2)	codewords $\bar{v}(u_1,u_2)$
0 0	0 0 0 0
1 0	1 1 1 1
0 1	0 0 0 1
1 1	1 1 1 0

Encoding (b)

message (u_1,u_2)	codewords $\bar{v}(u_1,u_2)$
0 0	0 0 0 0
1 0	1 1 1 1
0 1	1 1 1 0
1 1	0 0 0 1

For the encoding mapping (a), the 1-clouds are:

$Q_1(0)$={(0000), (0001)},

$Q_1(1)=\{(1111), (1110)\}$.

The 2-clouds are:

$Q_2(0)=\{(0000), (1111)\}$,

$Q_2(1)=\{(0001), (1110)\}$.

We see that

$s_1=d(Q_1(0),Q_1(1))=3$,

$s_2=d(Q_2(0),Q_2(1))=1$.

Hence, the separation vector of C based on decoding (a) is $\bar{s}=(3,1)$. In this case, the message bit u_1 will be decoded correctly provided there is no more than one error in the received word. The second message bit u_2 has no error protection. The code is a (1,0)-error-correcting code.

For the encoding mapping (b), the 1-clouds and 2-clouds are

$Q_1(0)=\{(0000), (1110)\}$,

$Q_1(1)=\{(1111), (0001)\}$,

$Q_2(0)=\{(0000), (1111)\}$,

$Q_2(1)=\{(1110), (0001)\}$.

Note that

$s_1=d(Q_1(0),Q_1(1))=1$,

$s_2=d(Q_1(0),Q_2(1))=1$.

Hence, for the encoding mapping (b), the code has a separation vector

$\bar{s} = (1,1)$.

In this case, the code provides no error protection for either u_1 or u_2.

ΔΔ

B. Direct-Sum Codes for Unequal Error Protection

For $1 \le i \le m$, let

$$C_i = \{\bar{v}(\bar{x}_i): \bar{x}_i \in A_i\}$$

be a block code of length n for the i-th component message space A_i. We assume that codes, $C_1, C_2, \ldots, C_m$, satisfies the following conditions:

(1) For $i \neq j$, $C_i \cap C_j = \{\bar{0}\}$, where $\bar{0}$ is the all zero vector in $\{0,1\}^n$.

(2) $\bar{v}(\bar{x}_1)+\bar{v}(\bar{x}_2)+\ldots+\bar{v}(\bar{x}_m) = \bar{v}(\bar{x}_1')+\bar{v}(\bar{x}_2')+\ldots+\bar{v}(\bar{x}_m')$ if and only if $\bar{x}_i=\bar{x}_i'$ for $i=1,2,\ldots,m$.

The first condition implies that every code contains the all-zero vector. Now we consider the following set of vectors:

$$C = \{\bar{v}(\bar{x}_1)+\bar{v}(\bar{x}_2)+\ldots+\bar{v}(\bar{x}_m): \bar{v}(\bar{x}_i) \in C_i \text{ for } 1 \le i \le m\}$$

The set C is called the direct sum of $C_1, C_2, \ldots, C_m$, denoted

$$C = C_1 \oplus C_2 \oplus \ldots \oplus C_m.$$

Now we use C as a code for the product message space A. For any message $(\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_m)$ in A, the corresponding codeword $\bar{v}(\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_m)$ is simply the following direct sum:

$$\bar{v}(\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_m) = \bar{v}(\bar{x}_1)+\bar{v}(\bar{x}_2)+\ldots+\bar{v}(\bar{x}_m).$$

Let $\{j_1, j_2, \ldots, j_\ell\}$ be a subset of $\{1,2,3,\ldots,m\}$. Let

$$C(j_1, j_2, \ldots, j_\ell) = C_{j_1} \oplus C_{j_2} \oplus \ldots \oplus C_{j_\ell}$$

Then $C(j_1, j_2, \ldots, j_\ell)$ is a subcode of C. The i-cloud of C for the component message $\bar{x}_i$ is simply the following set:

$$Q_i(\bar{x}_i) = \bar{v}(\bar{x}_i) \oplus C(1,\ldots,i-1,i+1,\ldots,m) \qquad (15)$$

Since $\bar{0}$ is a vector in $C(1,\ldots,i-1,i+1,\ldots,m)$, the vector $\bar{v}(\bar{x}_i)$ is in the i-cloud $Q_i(\bar{x}_i)$. The vector $\bar{v}(\bar{x}_i)$ is called the center

of $Q_i(\bar{x}_i)$. A satellite in $Q_i(\bar{x}_i)$ is of the form,

$$\bar{v}(\bar{x}_i)+\bar{w},$$

where $\bar{w} \in C(1,\ldots,i-1,i+1,\ldots,m)$.

Let $\bar{s}=(s_1,s_2,\ldots,s_m)$ be the separation vector of C. Suppose the codeword

$$\bar{v} = \bar{v}(\bar{x}_1)+\bar{v}(\bar{x}_2)+\ldots+\bar{v}(\bar{x}_m)$$

is transmitted. It follows from Theorem 1 that, if there are $\lfloor(s_i-1)/2\rfloor$ or fewer errors in the received vector, the i-cloud $Q_i(\bar{x}_i)$ which contains $\bar{v}$ can be identified, and hence the center $\bar{v}(\bar{x}_i)$ and the message $\bar{x}_i$ can be recovered.

Theorem 2: Let C be the direct sum of $C_1, C_2,\ldots,C_m$. Let $\bar{e}$ be an error pattern with $\lfloor(s_i-1)/2\rfloor$ or fewer errors, i.e. the Hamming weight of $\bar{e}$, $w(\bar{e})$, is $\lfloor(s_i-1)/2\rfloor$ or less. Then, the subcode $C(1,2,\ldots,i)$ is capable of correcting any error pattern of the following form,

$$\bar{e}+\bar{z},$$

with $\bar{z} \in C(i+1,i+2,\ldots,m)$.

Proof: Let $\bar{y}$ be a codeword in the subcode $C(1,2,\ldots,i)$. Then

$$\bar{y} = \bar{v}(\bar{x}_1)+\bar{v}(\bar{x}_2)+\ldots+\bar{v}(\bar{x}_i)$$

for some $\bar{x}_1 \in A_1$, $\bar{x}_2 \in A_2$, ..., $\bar{x}_i \in A_i$. Suppose $\bar{y}$ is transmitted and corrupted by the error pattern $\bar{e}+\bar{z}$. Then, the recieved vector is

$$\bar{r} = \bar{y}+\bar{e}+\bar{z}.$$

Note that $\bar{y}+\bar{z}=\bar{v}$ is a codeword in C. Thus, $\bar{r}=\bar{e}+\bar{v}$. Let

$$\bar{z} = \bar{v}(\bar{x}_{i+1})+\bar{v}(\bar{x}_{i+2})+\ldots+\bar{v}(\bar{x}_m).$$

Since $w(\bar{e}) \leq \lfloor(s_i-1)/2\rfloor$ and $s_1 \geq s_2 \geq \ldots \geq s_i$, it follows Theorem 1

that $\bar{v}(\bar{x}_1)$, $\bar{v}(\bar{x}_2)$, ..., $\bar{v}(\bar{x}_i)$ can be decoded correctly, i.e. $\bar{y} = \bar{v}(\bar{x}_1)+\bar{v}(\bar{x}_2)+...+\bar{v}(\bar{x}_i)$ can be decoded correctly. Therefore, $\bar{e}+\bar{z}$ is a correctable error pattern for the subcode $C(1,2,...,i)$.

Q.E.D.

Encoding of a direct-sum code can be done easily. Each component message $\bar{x}_i$ is encoded into a codeword $\bar{v}(\bar{x}_i)$ based on its corresponding code C_i. Then the m component codewords are added to form the codeword for the entire message $(\bar{x}_1,\bar{x}_2,...,\bar{x}_m)$.

Decoding of a direct-sum code can be carried out in m steps. Suppose the codeword

$$\bar{v} = \bar{v}(\bar{x}_1)+\bar{v}(\bar{x}_2)+...+\bar{v}(\bar{x}_m)$$

is transmitted and

$$\bar{r}_1 = \bar{v}+\bar{e}$$

is recieved where $\bar{e}$ is the error pattern. At the first step, we decode $\bar{x}_1$ based on the m-level error-protection code $C = C_1 \circledast C_2 \circledast ... \circledast C_m$. If $w(\bar{e}) \leq \lfloor (s_1-1)/2 \rfloor$, $\bar{x}_1$ and $\bar{v}(\bar{x}_1)$ can be correctly recovered. Then, we subtract $\bar{v}(\bar{x}_1)$ from $\bar{r}$. This results in the following vector

$$\bar{r}_2 = \bar{v}(\bar{x}_2)+...+\bar{v}(\bar{x}_m)+\bar{e}.$$

At the second step, we decode $\bar{x}_2$ based on the (m-1)-level error protection code $C(2,3,...,m)$. If $w(\bar{e}) \leq \lfloor (s_2-1)/2 \rfloor$, $\bar{x}_2$, and $\bar{v}(\bar{x}_2)$ can be recovered correctly. Subtracting $\bar{v}(\bar{x}_2)$ from $\bar{r}_2$, we obtain

$$\bar{r}_3 = \bar{v}(\bar{x}_3)+...+\bar{v}(\bar{x}_m)+\bar{e}.$$

Repeating the above process, we decode the rest of component messages. Each subsequent component message is decoded based on a

smaller code. If $w(\bar{e}) \leq \lfloor (s_i-1)/2 \rfloor$, $\bar{x}_1$, $\bar{x}_2$, ..., $\bar{x}_i$ will be decoded correctly.

At each step of the above m-step decoding procedure for a direct-sum code, two approaches can be applied to decode the component message. Suppose that $\bar{v}(\bar{x}_1)$, $\bar{v}(\bar{x}_2),\ldots,\bar{v}(\bar{x}_{i-1})$ have been correctly decoded. Then, we have

$$\bar{r}_i = \bar{v}(\bar{x}_i)+\bar{v}(\bar{x}_{i+1})+\ldots+\bar{v}(\bar{x}_m)+\bar{e}.$$

At the i-th step, we need to decode $\bar{x}_i$ and $\bar{v}(\bar{x}_i)$ from $\bar{r}_i$. For the first approach, we view $\bar{r}_i$ as an error corrupted version of a codeword $\bar{v}(\bar{x}_i)+\bar{v}(\bar{x}_{i+1})+\ldots+\bar{v}(\bar{x}_m)$ in $C(i,i+1,\ldots,m)$. Then, we can apply the basic nearest-neighbor decoding method, i.e., searching for the i-cloud nearest to $\bar{r}_i$ and using the center of the i-cloud as an estimate of $\bar{v}(\bar{x}_i)$. Clearly, the estimate of $\bar{v}(\bar{x}_i)$ is correct if $w(\bar{e}) \leq \lfloor (s_i-1)/2 \rfloor$. Then, we can find the component message $\bar{x}_i$ corresponding to $\bar{v}(\bar{x}_i)$. For the second approach, we view $\bar{r}_i$ as an error corrupted version of a codeword $\bar{v}(\bar{x}_i)$ in the component code C_i. Then, we decode $\bar{v}(\bar{x}_i)$ based on the decoding algorithm of C_i. Suppose that $w(\bar{e}) \leq \lfloor (s_i-1)/2 \rfloor$. It follows from Theorem 2 that

$$\bar{r}_{i+1} = \bar{v}(\bar{x}_{i+1})+\bar{v}(\bar{x}_{i+2})+\ldots+\bar{v}(\bar{x}_m)+\bar{e}$$

is a correctable error pattern for C_i. Thus, $\bar{v}(\bar{x}_i)$ and $\bar{x}_i$ can be correctly decoded.

There is an example for which the second approach can be applied. For some $i=1,2,\ldots,m$, suppose that the i-th component code C_i is a linear code with parity check matrix H_i. Note that other component codes may or may not be linear. At the i-th step

of decoding, we can apply the second approach for which the decoding algorithm of C_i is the syndrome decoding. We compute the syndrome for $\bar{r}_i$ based on H_i, i.e.

$$\bar{S}_i = \bar{r}_i \cdot H_i^T.$$

From $\bar{S}_i$, we identify the correctable error pattern (a coset leader with respect to C_i) which corresponds to $\bar{S}_i$. If $w(\bar{e}) \leq \lfloor (s_i-1)/2 \rfloor$, then the corresponding error pattern is

$$\bar{r}_{i+1} = \bar{v}(\bar{x}_{i+1})+\bar{v}(\bar{x}_{i+2})+\ldots+\bar{v}(\bar{x}_m)+\bar{e}.$$

Subtracting $\bar{r}_{i+1}$ from $\bar{r}_i$, we obtain $\bar{v}(\bar{x}_i)$. Then, we can find the component message $\bar{x}_i$ corresponding to $\bar{v}(\bar{x}_i)$.

C. Hamming Bound for Systematic UEP Codes

An m-level unequal error protection code C is said to be systematic if the codeword for the message $(\bar{x}_1,\bar{x}_2,\ldots,\bar{x}_m)$ has the following form:

$$\bar{v}(\bar{x}_1,\bar{x}_2,\ldots,\bar{x}_m) = (\bar{x}_1,\bar{x}_2,\ldots,\bar{x}_m,\bar{p})$$

where $\bar{p}$ represents the n-k redundant digits. Now we are going to derive a lower bound on the number of parity-check digits of an m-level linear systematic unequal error protection code with a separation vector $\bar{s} = (d_1,d_2,\ldots,d_m)$. Let $\bar{y}=(y_1,y_2,\ldots,y_n)$ be a binary n-tuple in $\{0,1\}^n$. For $1\leq j\leq n$, define

$$\bar{y}^*(j) = (y_j,y_{j+1},\ldots,y_n).$$

Note that $\bar{y}^*(j)$ is simply a suffix of $\bar{y}$. Define the following set of n-tuples:

$$Y = \{\bar{y} : \bar{y}\in\{0,1\}^n \text{ and the number of nonzero components in } \bar{y}^*(\lambda_{i-1}+1) \text{ is at most } t_i \text{ for } 1\leq i\leq m\} \quad (16)$$

where $\lambda_0=0$, $t_i=\lfloor (d_i-1)/2 \rfloor$ and $\lambda_i=k_1+k_2+\ldots+k_i$.

Lemma 2: Let $\bar{y}$ and $\bar{y}'$ be two n-tuples in Y. Let $\bar{v} = \bar{v}(\bar{x}_1,\bar{x}_2,\ldots,\bar{x}_m)$ and $\bar{v}' = \bar{v}(\bar{x}_1',\bar{x}_2',\ldots,\bar{x}_m')$ be two codewords in C. Then

$$\bar{y} + \bar{v} = \bar{y}' + \bar{v}'$$

if, and only if, $\bar{y} = \bar{y}'$ and $\bar{v} = \bar{v}'$.

proof: The if part of the lemma is obvious. Consider the only if part. Suppose $\bar{y} + \bar{v} = \bar{y}' + \bar{v}'$. Then

$$\bar{y} + \bar{y}' = \bar{v} + \bar{v}'. \tag{17}$$

From the definition of the set Y, we see that the number of nonzero components in the last $n-\lambda_{i-1}$ positions of $\bar{y}+\bar{y}'$ is at most $2t_i$ for $1\leq i\leq m$. Assume that $\bar{x}_1\neq\bar{x}_1'$. Since the separation vector of C is $(d_1,d_2,\ldots,d_m)$, we have

$$d(\bar{v},\bar{v}') = w(\bar{v}+\bar{v}') \geq d_1 \geq 2t_1+1. \tag{18}$$

However, from (16), we have

$$w(\bar{v}+\bar{v}') = w(\bar{y}+\bar{y}') \leq 2t_1. \tag{19}$$

The condition given by (18) contradicts the condition given by (19). Hence the hypothesis that $\bar{x}_1\neq\bar{x}_1'$ is invalid. As a result, we must have $\bar{x}_1=\bar{x}_1'$. Since C is systematic, it follows from (17) that the first λ_1 components of $\bar{y}+\bar{y}'$ are zero.

Now we assume that $\bar{x}_2\neq\bar{x}_2'$. Then

$$d(\bar{v},\bar{v}') = w(\bar{v}+\bar{v}') \geq d_2 \geq 2t_2+1. \tag{20}$$

However, it follows from (17) and the fact $\bar{x}_1=\bar{x}_1'$ that

$$w(\bar{v}+\bar{v}')=w(\bar{y}+\bar{y}') \leq 2t_2. \tag{21}$$

Equation (20) contradicts Equation (21). Hence our hypothesis that $\bar{x}_2\neq\bar{x}_2'$ is invalid.

Since $\bar{x}_1=\bar{x}_1'$ and $\bar{x}_2=\bar{x}_2'$, the first λ_2 components of $\bar{y}+\bar{y}'$ are

zero. Repeat the above argument, we can prove that $\bar{x}_3=\bar{x}'_3,\ldots,\bar{x}_m=\bar{x}'_m$. Consequently we must have $\bar{v}=\bar{v}'$ and $\bar{y}=\bar{y}'$.

Q.E.D.

Based on the conditions on Y, we can readily find that the number of elements in Y is

$$|Y| = \sum_{s=0}^{t_1} \binom{n}{s} - \sum_{e=1}^{m-1} \sum_{\ell=t_{e+1}+1}^{t_e} \binom{n-\lambda_e}{\ell} \qquad (22)$$

Next we will prove that the elements in Y are correctable error patterns for the code C.

Theorem 3: Let C be an m-level (n,k) systematic unequal error protection code with a separation vector $(d_1,d_2,\ldots,d_m)$. Then the n-tuples in Y defined by (16) are correctable error patterns for C.

Proof: For every $\bar{v} \in C$, we form the set

$$\{\bar{v}+\bar{y} : \bar{y} \in Y\}.$$

It follows from Lemma 2 that, for $\bar{v}$, $\bar{v}' \in C$ and $\bar{v} \neq \bar{v}'$,

$$\{\bar{v}+\bar{y} : \bar{y} \in Y\} \cap \{\bar{v}'+\bar{y} : \bar{y} \in Y\} = \phi.$$

We can use $\{\bar{v}+\bar{y} : \bar{y} \in Y\}$ as the decoding region for $\bar{v}$. If the received vector $\bar{r}$ is in $\{\bar{v}+\bar{y} : \bar{y} \in Y\}$, we decode $\bar{r}$ into $\bar{v}$. Hence, if the error pattern during the transmission of a codeword $\bar{v}$ is a member in Y, then the received word $\bar{r}$ will be in $\{\bar{v}+\bar{y} : \bar{y} \in Y\}$ and the decoding would be correct. Hence the elements in Y are correctable error patterns for C.

Q.E.D.

Note that the total number of codewords in C is 2^k. We must have

$$2^n \geq 2^k \cdot |Y|. \qquad (23)$$

From (22) and (23), we have the following lower bound on n-k,

$$n-k \geq \log_2 \left\{ \sum_{s=0}^{t_1} \binom{n}{s} - \sum_{e=1}^{m-1} \sum_{\ell=t_{e+1}+1}^{t_e} \binom{n-\lambda_e}{\ell} \right\} \qquad (24)$$

The bound given by (24) is equivalent to the well known Hamming bound [20] for the single-level error correcting code. For m=1, (24) recudes to

$$n-k \geq \log_2 \left\{ \sum_{s=0}^{t_1} \binom{n}{s} \right\},$$

which is the Hamming bound for the single-level error correcting code. Different versions of Hamming bound for multi-level linear unequal error protection code were proved by Masnick and Wolf[1], and Van Gils[23]. Note that our version of Hamming bound applies to either linear or nonlinear systematic UEP code.

D. Linear Unequal Error Protection Codes

Suppose the component code C_i is linear for i = 1,2,...,m. Then, $C = C_1 \oplus C_2 \oplus \ldots \oplus C_m$ is a linear code of length n for the product message space $A = A_1 x A_2 x \ldots x A_m$, where the i-th component message space A_i consists of all the k_i-tuples over GF(2), i.e. $A_i = \{0,1\}^{k_i}$ for $1 \leq i \leq m$. Hence C is an (n,k) code with

$$k = k_1 + k_2 + \ldots + k_m.$$

Every i-cloud $Q_i(\bar{x}_i)$ of C consists of 2^{k-k_i} codewords. The i-cloud $Q_i(\bar{x}_i=\bar{0})$ is a $(k-k_i)$-dimemsional subcode of C, and any i-cloud for which $\bar{x}_i \neq \bar{0}$ is simply a coset of $Q_i(\bar{x}_i=\bar{0})$. Since $d(\bar{u},\bar{v}) = w(\bar{u}+\bar{v})$, it follows from (3) to (6) that, for a linear code C, the minimum separation of i-clouds is

$$s_i = \min_{\substack{\bar{x}_i \in A_i \\ \bar{x}_i \neq \bar{0}}} \{ \min \{w(\bar{a}) : \bar{a} \in Q_i(\bar{x}_i)\}\}$$

$$= \min \{w[\bar{v}(\bar{x}_1,\ldots,\bar{x}_i,\ldots,\bar{x}_k)] : \bar{x}_i \neq \bar{0}\} \qquad (25)$$

Theorem 4: Let C_i be an (n,k_i) linear code of length n, where i =1,2. Consider the (n,k_1+k_2) code C which is the direct sum of C_1 and C_2. C is a two-level error-correcting code with separation vector $\bar{s} = (d_1,d_2)$. if the following condition are satisfied:

(i) The minimum distance of C_2 is d_2.

(ii) The minimum distance of $C-C_2$ is d_1 and $d_1 \geq d_2$.

Then, for any message, the first k_1 message symbols are protected against $t_1 = \lfloor (d_1-1)/2 \rfloor$ or fewer errors and the next k_2 message symbols are protected against $t_2 = \lfloor (d_2-1)/2 \rfloor$ or fewer errors.

Proof: Note that the message space A is the product of A_1 and A_2, where $A_1=\{0,1\}^{k_1}$ and $A_2=\{0,1\}^{k_2}$. Each message $\bar{x}=(\bar{x}_1,\bar{x}_2)$ consists of two parts, $\bar{x}_1$ and $\bar{x}_2$, where $\bar{x}_1$ is a k_1-bit component message and $\bar{x}_2$ is a k_2-bit component message. The codeword for the message is

$$\bar{v}(\bar{x}_1,\bar{x}_2) = \bar{v}(\bar{x}_1)+\bar{v}(\bar{x}_2),$$

where $\bar{v}(\bar{x}_1) \in C_1$ and $\bar{v}(\bar{x}_2) \in C_2$. The 1-cloud of the code for $\bar{x}_1=\bar{0}$, $Q_1(\bar{x}_1=\bar{0})$, is simply the subcode C_2. It follows from (25) and the given condition that

$$s_1 = \min_{\substack{\bar{x}_1 \in A_1 \\ \bar{x}_1 \neq \bar{0}}} \{ \min \{w(\bar{v}) : \bar{v} \in Q_1(\bar{x}_1)\}\}$$

$$= \min \{w(\bar{v}) : \bar{v} \in C-C_2\}$$

$$= d_1.$$

The 2-cloud of C for $\bar{x}_2=\bar{0}$ is simply the subcode C_1. Then, it follows from (25) that

$$s_2 = \min \{\bar{v} : \bar{v} \in C\text{-}C_1\}. \qquad (26)$$

Note that $C\text{-}C_1$ contains all the nonzero codewords of C_2. The minimum weight of nonzero vectors in C_2 is d_2. A codeword in $C\text{-}C_1$ but not in $C_2\text{-}\{\bar{0}\}$ has weight at least d_1. Since $d_1 \geq d_2$, it follows from (26) that

$$s_2 = d_2.$$

Q.E.D.

A direct generalization of Theorem 4 is Theorem 5.

Theorem 5: Consider an (n,k) linear code C which is the direct sum of codes C_1, C_2, ..., and C_m, where C_i is an (n,k_i) linear code. Let $C(i,i+1,\ldots,m)=C_i \oplus C_{i+1} \oplus \ldots \oplus C_m$. Let d_m be a lower bound on the minimum distance of C_m. If the minimum weight of codewords in $C\text{-}C(i,i+1,\ldots,m)$ is at least d_{i-1} and $d_1 \geq d_2 \geq \ldots \geq d_m$, then C is an m-level error correcting code for the product message space $A = A_1 x A_2 x \ldots x A_m$ with separation vector

$$\bar{s} = (s_1, s_2, \ldots, s_m)$$

where A_i is the component message space for C_i and $s_i \geq d_i$ for $i = 1,2, \ldots ,m$.

Proof: Similar to the proof of Theorem 4.

Q.E.D.

Theorem 5 actually describes a method for constructing a multi-level error-correcting code by taking the direct sum of component codes. With this method, we are able to construct codes which are presented in the rest of this paper.

III. CONSTRUCTION OF LINEAR MULTI-LEVEL UEP CODES BY COMBINING SHORTER CODES

A. Construction of Linear Multi-Level UEP codes by Combining Generator Matrices of Shorter Codes

We first present a construction method based on generator matrices. Let G_{aa} and

$$G_a = \begin{bmatrix} G_{aa} \\ G_{ab} \end{bmatrix}$$

be the generator matrices of an (n_a, k_a) linear code C_{aa} and an $(n_a, k_a+\lambda)$ linear code C_a respectively. Clearly C_{aa} is a subcode of C_a and G_{ab} is a $\lambda x n_a$ binary matrix. Let d_{aa} and d_a be the minimum distances of C_{aa} and C_a respectively. Then $d_{aa} \geq d_a$. Let G_{bb} and

$$G_b = \begin{bmatrix} G_{bb} \\ G_{ba} \end{bmatrix}$$

be the generator matrices of an (n_b, k_b) linear code C_{bb} and an $(n_b, k_b+\lambda)$ linear code C_b respectively. Note that C_{bb} is a subcode of C_b and G_{ba} is a $\lambda x n_b$ binary matrix. The submatrices G_{ab} and G_{ba} have the same dimension (number of rows) λ. Let d_{bb} and d_b be the minimum distances of C_{bb} and C_b respectively. Then $d_{bb} \geq d_b$.

We assume that the following condition holds:

$$d_a + d_b \geq d_{aa} \geq d_{bb}.$$

Now we form the following $(k_a+k_b+\lambda)x(n_a+n_b)$ matrix:

$$G = \begin{bmatrix} G_{ab} & G_{ba} \\ G_{aa} & 0_{ab} \\ 0_{ba} & G_{bb} \end{bmatrix} \qquad (27)$$

where 0_{ab} and 0_{ba} are a $k_a x n_b$ and a $k_b x n_a$ zero matrices. The matrix G generates an $(n_a+n_b,\ k_a+k_b+\lambda)$ linear code C. Let C_1, C_2 and C_3 be three subcodes of C generated by matrices, $[G_{ab}\ G_{ba}]$, $[G_{aa}\ 0_{ab}]$, and $[0_{ba}\ G_{bb}]$ respectively. We readily see that the minimum distance of C_1 is at least d_a+d_b, the minimum distance of C_2 is d_{aa}, and the minimum distance of C_3 is d_{bb}. Code C is actually the direct-sum of C_1, C_2 and C_3, i.e.,

$$C = C_1 \oplus C_2 \oplus C_3.$$

Note that C_1, C_2 and C_3 are codes for message spaces $A_1 = \{0,1\}^{\lambda}$, $A_2 = \{0,1\}^{k_a}$ and $A_3 = \{0,1\}^{k_b}$ respectively. Hence C is a code for the product message space $A=A_1 x A_2 x A_3$.

Now we examine the distance structure of $C=C_1 \oplus C_2 \oplus C_3$. Let $C(2,3) = C_2 \oplus C_3$. First we note that a codeword in C-C(2,3) is the concatenation of a nonzero codeword in C_a and a nonzero codeword in C_b. Hence a codeword in C-C(2,3) has weight at least d_a+d_b. Next we note that a codeword in $C-C_3$ is either the concatenation of a nonzero codeword in C_a and a nonzero codeword in C_b, or a codeword in C_2. Thus a codeword in $C-C_3$ has weight at least $\min\{d_a+d_b, d_{aa}\} = d_{aa}$. In fact the minimum weight of $C-C_3$ is d_{aa}. It is easy to check that the minimum distance of C is d_{bb}. In summary, C has the following distance (or weight) structure:

(1) the minimum weight of codewords in C-C(2,3) is at least d_a+d_b;

(2) the minimum weight of codewords in $C-C_3$ is d_{aa}.

(3) the minimum weight of C is d_{bb}.

It follows from Theorem 5 that the separation vector of C is $\bar{s}=(s_1,s_2,s_3)$ where $s_1 \geq d_a+d_b$, $s_2 \geq d_{aa}$ and $s_3=d_{bb}$.

Example 2: Let α be a primitive element in $GF(2^5)$. Let C_{bb} be the (31,21) BCH code over GF(2) whose generator polynomial has α and α^3 as roots. Let C_b be the (31,26) Hamming code over GF(2). The minimum weights of C_{bb} and C_b are 5 and 3 respectively, and C_{bb} is a subcode of C_b. Let G_{bb} and

$$G_b = \begin{bmatrix} G_{bb} \\ G_{ba} \end{bmatrix}$$

be the generator matrices of C_{bb} and C_b respectively. Then G_{ba} is a 5x31 matrix. Let C_{aa} be the (32,21) code obtained by adding an overall parity-check bit to each codeword in C_{bb}. Then the minimum weight of C_{aa} is 6. Let C_a be the (32,26) code obtained by adding an overall parity-check bit to every codeword in C_b. Then the minimum weight of C_a is 4, and C_{aa} is a subcode of C_a. Let G_{aa} and

$$G_a = \begin{bmatrix} G_{aa} \\ G_{ab} \end{bmatrix}$$

be the generator matrices of C_{aa} and C_a respectively where G_{ab} is a 5x32 matrix. Then the code C generated by the generator matrix G of (27) is a (63,47) code with a separation vector $\bar{s}=(s_1,s_2,s_3)$ where $s_1 \geq 7$, $s_2 \geq 6$ and $s_3=5$. We may divide a message $\bar{x}$ of 47 bits into two parts, $\bar{x}_1$ and $\bar{x}_2$, where $\bar{x}_1$ consists of the first 5 bits of $\bar{x}$ and $\bar{x}_2$ consists of the next 42 bits of $\bar{x}$. Then all five message bits in $\bar{x}_1$ are protected against 3 or fewer random errors, and the 42 bits in $\bar{x}_2$ are protected against two or fewer random errors. Hence C is a two-level UEP code. Note that there is a single-level double-error-correcting (63,51)

BCH code and a single-level triple-error-correcting (63,45) BCH code[20,21].

Consider the special case for which $k_b=0$ and $G_b=G_{ba}$. Then the matrix G of (27) reduces to the following form:

$$G = \begin{bmatrix} G_{ab} & G_{ba} \\ G_{aa} & 0_{ab} \end{bmatrix} \qquad (28)$$

If $d_a+d_b \geq d_{aa}$, the code generated by G of (28) is then an $(n_a+n_b, k_a+\lambda)$ code with a separation vector $\bar{s}=(s_1,s_2)$ where $s_1 \geq d_a+d_b$ and $s_2=d_{aa}$. This special case was first presented by Boyarinov [17].

B. Construction of Linear Multi-Level UEP Codes by Combining Parity-Check Matrices of Shorter Codes

Let H_{aa} and

$$H_a = \begin{bmatrix} H_{aa} \\ H_{ab} \end{bmatrix}$$

be the parity-check matrices of an (n_a,k_a) linear code C_{aa} and an (n_a,k_a-r) linear code C_a respectively, where H_{aa} is an $(n_a-k_a)xn_a$ matrix, H_{ab} is a rxn_a matrix and H_a is an $(n_a-k_a+r)xn_a$ matrix. It is clear that C_a is a subcode of C_{aa}. Let d_a and d_{aa} be the minimum distances of C_a and C_{aa} respectively. Then

$$d_a \geq d_{aa}.$$

Let H_{bb} and

$$H_b = \begin{bmatrix} H_{bb} \\ H_{ba} \end{bmatrix}$$

be the parity-check matrices of an (n_b,k_b) linear code C_{bb} and an (n_b,k_b-r) linear code C_b, where H_{bb} is an $(n_b-k_b)xn_b$ matrix, H_{ba} is a rxn_b matrix, and H_b is an $(n_b-k_b+r)xn_b$ matrix. Note that C_b

is a subcode of C_{bb}. Let d_b and d_{bb} be the minimum distances of C_b and C_{bb}. Then

$$d_b \geq d_{bb}.$$

Consider the $(n_a+n_b,\ k_a+k_b-r)$ linear code C with the following parity-check matrix

$$H = \begin{bmatrix} H_{aa} & 0_{ab} \\ H_{ab} & H_{ba} \\ 0_{ba} & H_{bb} \end{bmatrix} \tag{29}$$

where 0_{ab} is an $(n_a-k_a)xn_b$ zero matrix and 0_{ba} is an $(n_b-k_b)xn_a$ zero matrix. Let C_2 be the (n_a+n_b,k_a-r) subcode of C such that each codeword in C_2 is a concatenation of a codeword in C_a and the all-zero n_b-tuple. Clearly the minimum weight of C_2 is d_a. Let C_3 be the (n_a+n_b,k_b-r) subcode of C such that every codeword in C_3 is a concatenation of the all-zero n_b-tuple and a codeword in C_b. The minimum weight of C_3 is d_b. The direct-sum of C_2 and C_3, denoted $C(2,3)=C_2 \oplus C_3$, is an (n_a+n_b,k_a+k_b-2r) subcode of C. Hence there must exist r linearly independent codewords in C-C(2,3). These r linearly independent codewords span an (n_a+n_b,r) linear subcode C_1 of C. We readily see that C is the direct-sum of C_1,C_2 and C_3, i.e., $C=C_1 \oplus C_2 \oplus C_3$.

Suppose $d_{aa}+d_{bb} \geq d_a \geq d_b$. Now we examine the distance structure of C. Any codeword $\bar{v}$ in C can be expressed as

$$\bar{v} = (\bar{v}_a, \bar{v}_b)$$

where $\bar{v}_a$ is an n_a-tuple and $\bar{v}_b$ is an n_b-tuple. Then

$$(\bar{v}_a, \bar{v}_b) \cdot H^T = \bar{0}.$$

This implies that $\bar{v}_a \cdot H_{aa}{}^T = \bar{0}$ and $\bar{v}_b \cdot H_{bb}{}^T = \bar{0}$. Consider a codeword $(\bar{v}_a,\bar{v}_b)$ in C-C(2,3). Then, $\bar{v}_a \neq \bar{0}$ and $\bar{v}_b \neq \bar{0}$. For $\bar{v}_a \neq \bar{0}$,

the weight of $\bar{v}_a$ is at least d_{aa}. This follows from the fact that any d_{aa}-1 or fewer columns of H_{aa} are linearly independent. Similarly, for $\bar{v}_b \neq \bar{0}$, the weight of $\bar{v}_b$ is at least d_{bb}. Hence, for any codeword $(\bar{v}_a, \bar{v}_b)$ in C-C(2,3), the weight of $(\bar{v}_a, \bar{v}_b)$ is at least $d_{aa}+d_{bb}$. Therefore, the minimum weight of codewords in C-C(2,3) is at least $d_{aa}+d_{bb}$. For any codeword $(\bar{v}_a, \bar{v}_b)$ in C-C_3, either it is in C_2, or both $\bar{v}_a$ and $\bar{v}_b$ are not zero. For the former case, the weight of the codeword is at least d_a. For the latter case, the weight of the codeword is at least $d_{aa}+d_{bb}$. Since $d_{aa}+d_{bb} \geq d_a$, the minimum weight of codewords in C-C_3 is d_a. Since $d_{aa}+d_{bb} \geq d_a \geq d_b$, the minimum weight of C is d_b. In summary, the code C genereated by the parity-check matrix H of (29) has the following distance structure:

(1) the minimum weight of codewords in C-C(2,3) is at least $d_{aa}+d_{bb}$;

(2) the minimum weight of codewords in C-C_3 is d_a; and

(3) the minimum weight of C is d_b.

It follows from Theorem 5 that, for $d_{aa}+d_{bb} \geq d_a \geq d_b$, the code C generated by the parity-check matrix H of (29) is a linear block code for the product message space $A=A_1 x A_2 x A_3$ where $A_1=\{0,1\}^r$, $A_2=\{0,1\}^{k_a-r}$ and $A_3=\{0,1\}^{k_b-r}$. The separation vector of C is $\bar{s} = (s_1, s_2, s_3)$ where $s_1 \geq d_{aa}+d_{bb}$, $s_2 \geq d_a$ and $s_3=d_b$.

Now we shall present several classes of linear UEP codes with parity-check matrices of the form given by (29).

Let α be a primitive element from the Galois field $GF(2^m)$. Every nonzero element in $GF(2^m)$ can be expressed as a power of α and can be represented by a nonzero m-tuple over GF(2) (in column

form). For any nonnegative integer ℓ, let

$$\beta_1, \beta_2, \cdots, \beta_{2^{m+\ell}-2^m}$$

represent all the $(m+\ell)$-tuples over GF(2) (in column form) for which the last ℓ components are not all zero. Consider the binary code C generated by the following parity-check matrix:

$$H = \left[\begin{array}{cccc|c} 1 & \alpha & \alpha^2 & \cdots & \alpha^{2^m-2} \\ 1 & \alpha^3 & \alpha^6 & \cdots & \alpha^{3(2^m-2)} \\ \hline 0_\ell & 0_\ell & 0_\ell & \cdots & 0_\ell \end{array} \;\middle|\; \begin{array}{c} 0_m \; 0_m \cdots 0_m \\ \hline \beta_1 \; \beta_2 \cdots \beta_{2^{m+\ell}-2^m} \end{array}\right] \qquad (30)$$

where each power of α is represented by an m-tuple, 0_ℓ is a column of ℓ zeros and 0_m is a column of m zeros. The matrix H consists of $2m+\ell$ rows and $2^{m+\ell}-1$ columns, and hence the code C generated by H is a $(2^{m+\ell}-1,\ 2^{m+\ell}-2m-\ell-1)$ linear code with $2m+\ell$ parity-check bits.

Note that the H matrix has the form given by (29) where

$$H_{aa} = [1 \quad \alpha \quad \alpha^2 \quad \cdots \quad \alpha^{2^m-2}]$$

$$H_a = \begin{bmatrix} H_{aa} \\ H_{ab} \end{bmatrix} = \begin{bmatrix} 1 & \alpha & \alpha^3 & \cdots & \alpha^{2^m-2} \\ 1 & \alpha^3 & \alpha^6 & \cdots & \alpha^{3(2^m-2)} \end{bmatrix}$$

$$H_b = \begin{bmatrix} H_{ba} \\ H_{bb} \end{bmatrix} = [\beta_1 \; \beta_2 \cdots \beta_{2^{m+\ell}-2^m}]$$

H_{bb} = some ℓ x $(2^{m+\ell}-2^m)$ matrix for which any column is not a zero column.

The codes, C_{aa} and C_a, generated by the parity-check matrices H_{aa} and H_a are simply primitive single-error-correcting and double-error-correcting BCH codes of length 2^m-1 respectively [20]. Code C_{aa} has minimum distance 3, and C_a has minimum distance 5. It is also known that the dimensions of H_{aa} and H_a are m and 2m respectively. The code C_b generated by parity-check matrix H_b is a shortened Hamming code with minimum weight 3. The code C_{bb} generated by parity-check matrix H_{bb} has minimum distance 2. As a result, C is a code for the product message space $A=A_1xA_2xA_3$ where $A_1=\{0,1\}^m$, $A_2=\{0,1\}^{2^m-2m-1}$ and $A_3=\{0,1\}^{2^{m+\ell}-2^m-m-\ell}$.

The separation vector of C is

$$\bar{s} = (s_1, s_2, s_3)$$

where $s_1 \geq d_{aa}+d_{bb}=3+2=5$, $s_2 \geq d_a=5$, and $s_3=d_b=3$.

For this code, the first 2^m-m-1 message bits of a message are protected against up to 2 random errors while the next $2^{m+\ell}-2^m-m-\ell$ message bits against any single error. Hence it is a (2,1)-error-correcting code.

For m=0, C becomes a conventional single-error-correcting Hamming code [20] of length $2^\ell-1$. For $\ell=0$, C reduces to a primitive double-error-correcting BCH code of length 2^m-1. For $m=\ell$, C is equivalent to a Boyarinov-Katsman UEP code [16]. The code C can be transformed into systematic form with identical two-level error correcting capability. The proof is given in Appendix B.

Consider the number of parity-check bits required of a two-level UEP code with the following parameters:

$$n = 2^{m+\ell}-1,$$

$$\lambda_1 = 2^m - m - 1,$$

$$t_1 = 2,$$

$$t_2 = 1.$$

It follows from the Hamming bound given by (24) that

$$\begin{aligned} 2^{n-k} &\geq 1+(2^{m+\ell}-1) + \binom{2^{m+\ell}-1}{2} - \binom{2^{m+\ell}-2^m+m}{2} \\ &= 2^{-1}\cdot\{2^{2m+\ell+1} - (2m)\cdot 2^{m+\ell} - (2^m-2m+1)\cdot 2^m - (m^2-m)+2\} \\ &= 2^{-1}\cdot\{2^{2m+\ell} + [2^{m+\ell}(2^{m-1}-2m) + 2^{2m}(2^{\ell-1}-1) + (2m- \\ &\quad 2)\cdot 2^m + (2^m - m^2 + m + 2)]\}. \end{aligned} \tag{31}$$

Let $\Delta = 2^{m+\ell}(2^{m-1}-2m) + 2^{2m}(2^{\ell-1}-1) + (2m-2)\cdot 2^m + (2^m - m^2 + m + 2)$. (32)

From (31) and (32), we have

$$2^{n-k} \geq 2^{2m+\ell-1} + \Delta/2. \tag{33}$$

For either $m=3$ and $\ell=3$ or $m\geq 4$ and $\ell\geq 1$, the number Δ is strictly greater than zero, i.e.,

$$\Delta > 0. \tag{34}$$

Hence, it follows from (33) and (34) that

$$n-k > 2m+\ell-1. \tag{35}$$

This is to say that the number of parity-check symbols required for a two-level linear systematic UEP code with parameters, $n=2^{m+\ell}-1$, $\lambda_1=2^m-m-1$, $t_1=2$ and $t_2=1$ is at least $2m+\ell$. The two-level UEP code given by the parity-check matrix H of (30) has exactly $2m+\ell$ parity-check symbols. Hence, under the condition that $m=3$ $=\ell=3$, or $m\geq 4$ and $\ell\geq 1$, the code meets the Hamming bound of (24) and is optimal. A list of codes of length 31, 63, 127 and 255 is given in Table 2 for various m and ℓ, where $k_1=2^m-m-1$ and $k_2=2^{m+\ell}-2^m-m-\ell$ and $k=k_1+k_2$ if $\ell\neq 0$. From the table, we see that

there is a (63,52) code which protects 26 message bits against two or fewer errors and 26 other message bits against any single error. Later we shall present a decoding scheme for any code with parity check matrix of form (29). By that time, we can make a more thorough comparison between the (63,52) code and the time sharing of conventional single-level codes based on their information rates and decoding complexities.

Table 2

Codes of length 31

m	ℓ	k	k_1	k_2
0	5	26	0	26
2	3	24	1	23
3	2	23	4	19
4	1	22	11	11
5	0	21	21	0

Codes of length 63

m	ℓ	k	k_1	k_2
0	6	57	0	57
2	4	55	1	54
3	3	54	4	50
4	2	53	11	42
5	1	52	26	26
6	0	51	51	0

Codes of length 127

m	ℓ	k	k_1	k_2
0	7	120	0	120
2	5	118	1	117
3	4	117	4	113
4	3	116	11	105
5	2	115	26	89
6	1	114	57	57
7	0	113	113	0

Codes of length 255

m	ℓ	k	k_1	k_2
0	8	247	0	247
2	6	245	1	244
3	5	244	4	240
4	4	243	11	232
5	3	242	26	216
6	2	241	57	184
7	1	240	120	120
8	0	239	239	0

The class of two-level UEP codes given above can be generalized in a straight forward manner. Consider the binary code C with the following parity-check matrix:

$$H = \left[\begin{array}{cccc|c} 1 & \alpha & \cdots & \alpha^{2^m-2} & 0_m \cdots 0_m \\ 1 & \alpha^3 & \cdots & (\alpha^3)^{2^m-2} & 0_m \cdots 0_m \\ \vdots & & & & \\ 1 & \alpha^{2t-3} & \cdots & (\alpha^{2t-3})^{2^m-2} & 0_m \cdots 0_m \\ \hline 1 & \alpha^{2t-3} & \cdots & (\alpha^{2t-1})^{2^m-2} & \\ \hline 0_\ell & 0_\ell & \cdots & 0_\ell & \beta_1 \cdots \beta_{2^{m+\ell}-2^m} \end{array}\right] \quad (36)$$

The code C generated by the parity-check matrix H of (36) has length $n=2^{m+\ell}-1$ and at most $mt+\ell$ parity-check bits. It can be easily proved that the code is a two-level UEP code with a separation vector $\bar{s}=(2t+1,3)$. The code provides protection of at least $\lambda_1=2^m-m(t-1)-1$ message bits against t or fewer errors and protection of other message bits against any single error.

There is another class of linear UEP codes with parity-check matrices of the form given by (29). The submatrices are given below:

$$H_{aa} = \left[\begin{array}{cccccc} 1 & 1 & 1 & \cdots & 1 \\ 0_m & 1 & \alpha & \cdots & \alpha^{2^m-2} \\ 0_m & 1 & \alpha^3 & \cdots & (\alpha^3)^{2^m-2} \\ \vdots & \vdots & \vdots & & \vdots \\ 0_m & 1 & \alpha^{2t-3} & \cdots & (\alpha^{2t-3})^{2^m-2} \end{array}\right] \quad (37)$$

$$H_{ab} = [0_m \; 1 \; \alpha^{2t-1} \; \ldots \; (\alpha^{2t-1})^{2^m-2}] \tag{38}$$

$$H_{ba} = [1 \; \alpha^{2s-1} \; \ldots \; (\alpha^{2s-1})^{2^m-2}] \tag{39}$$

$$H_{bb} = \begin{bmatrix} 1 & \alpha^{2s-3} & \ldots & (\alpha^{2s-3})^{2^m-2} \\ . & . & \ldots & . \\ . & . & \ldots & . \\ . & . & \ldots & . \\ 1 & \alpha^3 & \ldots & (\alpha^3)^{2^m-2} \\ 1 & \alpha & \ldots & \alpha^{2^m-2} \end{bmatrix} \tag{40}$$

where $s \leq t$.

Note that H_{aa} and $H_a = [H_{aa}^T \; H_{ab}^T]^T$ generate an extended (t-1)-error-correcting and an extended t-error-correcting primitive BCH codes of length 2^m respectively. The dimensions of H_{aa} and H_a are at most $m(t-1)$ and mt respectively. The parity-check matrices H_{bb} and $H_b = [H_{bb}^T \; H_{ba}^T]^T$ generate an (s-1)-error-correcting and an s-error-correcting primitive BCH codes of length 2^m-1 respectively. We require that H_{ab} and H_{ba} have the same dimension, i.e. α^{2t-1} and α^{2s-2} from the same subfield of $GF(2^m)$.

It follows from the argument given for codes with parity matrix of form (29) that the code generated by H with submatrices given by (37) to (40) is a linear block code with a separation vector $\bar{s}=(s_1,s_2,s_3)$ where

$$s_1 \geq 2(t+s)-1, \; s_2 \geq 2t+2, \; s_3 = 2s+1.$$

The code has at most $m(t+s-1)+1$ parity-check symbols. It protects the first $k_1=m$ message bits against s+t-1 or fewer errors, the next $k_2=2^m-mt-1$ message bits against t or fewer errors, and the

other message bits against s or fewer errors.

Example 3 : Let m=5 and t=s=2. Let α be a primitive element in $GF(2^5)$. Consider the code generated by the following parity-check matrix:

$$H = \begin{bmatrix} 1 & 1 & 1 & 1 & \cdots & 1 & 0_1 & 0_1 & 0_1 & \cdots & 0_1 \\ 0_5 & 1 & \alpha & \alpha^2 & \cdots & \alpha^{30} & 0_5 & 0_5 & 0_5 & \cdots & 0_5 \\ 0_5 & 1 & \alpha^3 & \alpha^6 & \cdots & \alpha^{90} & 1 & \alpha^3 & \alpha^6 & \cdots & \alpha^{90} \\ 0_5 & 0_5 & 0_5 & 0_5 & \cdots & 0_5 & 1 & \alpha & \alpha^2 & \cdots & \alpha^{30} \end{bmatrix}$$

Note that α^3 is also a primitive element in $GF(2^5)$. The code C generated by H has 3x5+1=16 parity-check bits. It is a (63,47) UEP code with separation vector $\bar{s}$ at least (7,6,5). This code is the same code given in Example 2.

C. Decoding

Now we consider the decoding of linear UEP codes generated by matrices of the form given by (29). Since the error-correcting capability of a UEP code depends on the encoding scheme, we need to know the corresponding generator matrix. Theorem 6 gives the generator matrix which correspond to the parity-check matrix of (29).

Theorem 6 : A linear code C with a parity-check matrix

$$H = \begin{bmatrix} H_{aa} & 0_{ab} \\ H_{ab} & H_{ba} \\ 0_{ba} & H_{bb} \end{bmatrix}$$

has a generator matrix of the following form:

$$G = \begin{bmatrix} G_{ab} & G_{ba} \\ G_{aa} & 0'_{ab} \\ 0'_{ba} & G_{bb} \end{bmatrix} \qquad (41)$$

where

(1) 0_{ab} is an $(n_a-k_a)xn_a$ zero matrix, 0_{ba} is an $(n_b-k_b)xn_b$ zero matrix, $0'_{ab}$ is an $(k_a-r)xn_b$ zero matrix, $0'_{ba}$ is an $(k_b-r)xn_a$ zero matrix.

(2) G_{aa} is a generator matrix of the (n_a,k_a-r) code C_a generated by the parity-check matrix $[H_{aa}{}^T\ H_{ab}{}^T]^T$.

(3) G_{bb} is a generator matrix of the (n_b,k_b-r) code C_b generated by the parity-check matrix $[H_{bb}{}^T\ H_{ba}{}^T]^T$.

(4) $[G_{aa}{}^T\ G_{ab}{}^T]^T$ is a generator matrix of the (n_a,k_a) code C_{aa} generated by the parity-check matrix H_{aa}.

(5) $[G_{bb}{}^T\ G_{ba}{}^T]^T$ is a generator matrix of the (n_b,k_b) code C_{bb} generated by the parity-check matrix H_{bb}.

Proof: See Appendix C.

ΔΔ

From the above theorem, we note that both G_{ab} and G_{ba} have r rows (or dimension r). The matrix G of (41) is of the same form of (27).

Now we present a decoding procedure for UEP codes with parity-check matrices of the form given by (29). Each message $\bar{x}$ consists of three parts $\bar{x}_1$, $\bar{x}_2$ and $\bar{x}_3$, i.e.,

$$\bar{x} = (\bar{x}_1, \bar{x}_2, \bar{x}_3)$$

where $\bar{x}_1$ is a binary r-tuple, $\bar{x}_2$ is a binary (k_a-r)-tuple, and $\bar{x}_3$ is a binary (k_b-r)-tuple. The codeword for message $\bar{x}$ is

$$\bar{v}(\bar{x}_1, \bar{x}_2, \bar{x}_3) = \bar{x}G \qquad (42)$$

where G is given by (41). For simplicity, we use $\bar{v}$ to represent $\bar{v}(\bar{x}_1,\bar{x}_2,\bar{x}_3)$. Every (n_a+n_b)-tuple $\bar{u}$ can be divided into two parts, $\bar{u}_a$ and $\bar{u}_b$, such that

$$\bar{u} = (\bar{u}_a, \bar{u}_b)$$

where $\bar{u}_a$ is an n_a-tuple and $\bar{u}_b$ is an n_b-tuple. Then $\bar{v} = (\bar{v}_a,\bar{v}_b)$. It follows from (41) and (42) that

$$\bar{v}_a = (\bar{x}_1,\bar{x}_2) \begin{bmatrix} G_{ab} \\ G_{aa} \end{bmatrix}$$

and

$$\bar{v}_b = (\bar{x}_1,\bar{x}_3) \begin{bmatrix} G_{ba} \\ G_{bb} \end{bmatrix}.$$

Suppose a codeword $\bar{v} = \bar{v}(\bar{x}_1,\bar{x}_2,\bar{x}_3)$ is transmitted and a word $\bar{r}$ is received. Let $\bar{e}$ be the error pattern. Then

$$\bar{r} = \bar{v} + \bar{e}.$$

Express $\bar{r}=(\bar{r}_a,\bar{r}_b)$ and $\bar{e}=(\bar{e}_a,\bar{e}_b)$. Then $\bar{r}_a=\bar{v}_a+\bar{e}_a$ and $\bar{r}_b=\bar{v}_b+\bar{e}_b$. Let $w(\bar{e})$ denote the weight of $\bar{e}$. The decoding of $\bar{r}$ consists of the following steps:

(1) Based on code C_{aa} (with parity-check matrix H_{aa} and generator matrix $[G_{ab}{}^T\ G_{aa}{}^T]^T$), we decode $\bar{r}_a$ into a codeword $\bar{v}_a^*$ in C_{aa}, which is a temporary estimate of $\bar{v}_a$. Based on code C_{bb} (with parity-check matrix H_{bb} and generator matrix $[G_{ba}{}^T\ G_{bb}{}^T]^T$), we decode $\bar{r}_b$ into a codeword $\bar{v}_b^{**}$ in C_{bb}, which is a temporary estimate of $\bar{v}_b$. Later we will show that either $\bar{v}_a^*$ or $\bar{v}_b^{**}$ is a correct estimate if $w(\bar{e})\le\lfloor(d_{aa}+d_{bb}-$

$1)/2\rfloor$. The decodings of $\bar{r}_a$ and $\bar{r}_b$ are based on the best available decoding schemes for C_{aa} and C_{bb}.

(2) Find $\bar{x}_1^*$ and $\bar{x}_2^*$ such that

$$(\bar{x}_1^*, \bar{x}_2^*) \begin{bmatrix} G_{ab} \\ G_{aa} \end{bmatrix} = \bar{v}_a^*.$$

Then $\bar{x}_1^*$ and $\bar{x}_2^*$ are estimates of message components $\bar{x}_1$ and $\bar{x}_2$. Also, find $\bar{x}_1^{**}$ and $\bar{x}_3^{**}$ such that

$$(\bar{x}_1^{**}, \bar{x}_3^{**}) \begin{bmatrix} G_{ba} \\ G_{aa} \end{bmatrix} = \bar{v}_b^{**}.$$

Then $\bar{x}_1^{**}$ and $\bar{x}_3^{**}$ are estimates of $\bar{x}_1$ and $\bar{x}_3$. Note that $\bar{x}_1^*$ and $\bar{x}_1^{**}$ are two estimates of $\bar{x}_1$. If $w(\bar{e}) \leq \lfloor (d_{aa}+d_{bb}-1)/2 \rfloor$, at least one of these two estimates is identical to $\bar{x}_1$.

(3) Form $\bar{w}^* = (\bar{x}_1^*, \bar{x}_2^*, \bar{0}_3)G$ and $\bar{u}^{**} = (\bar{x}_1^{**}, \bar{0}_2, \bar{x}_3^{**})G$ where $\bar{0}_3$ and $\bar{0}_2$ are a zero (k_b-r)-tuple and a zero (k_a-r)-tuple respectively. Note that $\bar{w}^*$ and $\bar{u}^{**}$ are codewords in C.

(4) Compute $\bar{r}^* = \bar{r} + \bar{w}^*$ and $\bar{r}^{**} = \bar{r} + \bar{u}^{**}$.

Note that

$$\bar{r}^* = \bar{v} + \bar{e} + \bar{w}^* = \bar{e} + (\bar{x}_1+\bar{x}_1^*, \bar{x}_2+\bar{x}_2^*, \bar{x}_3)\cdot G,$$

$$\bar{r}^{**} = \bar{v} + \bar{e} + \bar{u}^* = \bar{e} + (\bar{x}_1+\bar{x}_1^{**}, \bar{x}_2, \bar{x}_3+\bar{x}_3^{**})\cdot G,$$

$$\bar{r}_b^* = \bar{e}_b + (\bar{x}_1+\bar{x}_1^*, \bar{x}_3) \begin{bmatrix} G_{ba} \\ G_{bb} \end{bmatrix}, \qquad (43)$$

$$\bar{r}_a^{**} = \bar{e}_a + (\bar{x}_1+\bar{x}_1^{**}, \bar{x}_2) \begin{bmatrix} G_{ab} \\ G_{aa} \end{bmatrix}. \qquad (44)$$

(5) Based on code C_b (with parity-check matrix $[H_{ba}{}^T \; H_{bb}{}^T]^T$

and generator matrix G_{bb}), decode $\bar{r}_b^*$ into a codeword $\bar{z}_b^*$ in C_b. Based on code C_a (with parity-check matrix $[H_{aa}{}^T\ H_{ab}{}^T]^T$ and generator matrix G_{aa}), decode $\bar{r}_a^{**}$ into a code word $\bar{z}_a^{**}$ in C_a. The decoding algorithms for C_a and C_b at this step must be nearest neighbor decodings.

(6) Find $\bar{x}_3^*$ and $\bar{x}_2^{**}$ such that $\bar{x}_3^* \cdot G_{bb} = \bar{z}_b^*$ and $\bar{x}_2^{**} \cdot G_{aa} = \bar{z}_a^{**}$. Note that $\bar{x}_3^*$ and $\bar{x}_2^{**}$ are estimates of $\bar{x}_3$ and $\bar{x}_2$ respectively.

(7) Form $\bar{v}^* = \bar{w}^* + (\bar{0}_a, \bar{z}_b^*)$ and $\bar{v}^{**} = \bar{u}^{**} + (\bar{z}_a^{**}, \bar{0}_b)$ where $\bar{0}_a$ and $\bar{0}_b$ are a zero n_a-tuple and a zero n_b-tuple respectively. Note that

$$\bar{v}^* = (\bar{x}_1^*, \bar{x}_2^*, \bar{x}_3^*) \cdot G, \tag{45}$$

$$\bar{v}^{**} = (\bar{x}_1^{**}, \bar{x}_2^{**}, \bar{x}_3^{**}) \cdot G. \tag{46}$$

From (45) and (46), we see that $\bar{v}^*$ and $\bar{v}^{**}$ are estimates of the transmitted codeword $\bar{v}$.

(8) Compute the distances $d(\bar{r}, \bar{v}^*)$ and $d(\bar{r}, \bar{v}^{**})$. If

$$d(\bar{r}, \bar{v}^*) \leq d(\bar{r}, \bar{v}^{**}),$$

we decode $\bar{r}$ into $\bar{v}^*$. Then

$$(\bar{x}_1^*, \bar{x}_2^*, \bar{x}_3^*)$$

is the decoded message. On the other hand, if

$$d(\bar{r}, \bar{v}^*) > d(\bar{r}, \bar{v}^{**}),$$

we decode $\bar{r}$ into $\bar{v}^{**}$, and

$$(\bar{x}_1^{**}, \bar{x}_2^{**}, \bar{x}_3^{**})$$

is chosen as the decoded message.

Now we need to show that, using the above decoding procedure,

the following are true:

(1) If $w(\bar{e})\leq\lfloor(d_{aa}+d_{bb}-1)/2\rfloor$, the message component $\bar{x}_1$ will be correctly decoded;

(2) If $w(\bar{e})\leq\lfloor(d_a-1)/2\rfloor$, both the message components, $\bar{x}_1$ and $\bar{x}_2$, will be decoded correctly; and

(3) If $w(\bar{e})\leq\lfloor(d_b-1)/2\rfloor$, all the three message components will be decoded correctly.

Consider the first case for which $w(\bar{e})\leq\lfloor(d_{aa}+d_{bb}-1)/2\rfloor$. Then either $w(\bar{e}_a)\leq\lfloor(d_{aa}-1)/2\rfloor$ or $w(\bar{e}_b)\leq\lfloor(d_{bb}-1)/2\rfloor$. Thus at least one of the estimates, $\bar{v}_a^*$ and $\bar{v}_b^{**}$, at step 1 of the decoding procedure is correct, i.e., either $\bar{v}_a^*=\bar{v}_a$ or $\bar{v}_b^{**}=\bar{v}_b$. Suppose $w(\bar{e}_a)\leq\lfloor(d_{aa}-1)/2\rfloor$. Then $\bar{v}_a^*$ is the correct estimate of $\bar{v}_a$ and $\bar{v}_a^*=\bar{v}_a$. Also $\bar{x}_1^*=\bar{x}_1$ and $\bar{x}_2^*=\bar{x}_2$. Hence, $\bar{w}^*=(\bar{x}_1^*,\bar{x}_2^*,\bar{0}_3)\cdot G=(\bar{x}_1,\bar{x}_2,\bar{0}_3)\cdot G$. Note that

$$d(\bar{r}_a,\bar{v}_a^*) = d(\bar{r}_a,\bar{v}_a) = w(\bar{e}_a). \tag{47}$$

Let $\bar{z}_b=\bar{x}_3\cdot G_{bb}$. Then $\bar{z}_b$ is a codeword in C_b. Recall that, at step 5, $\bar{r}_b^*$ is decoded into $\bar{z}_b^*=\bar{x}_3^*\cdot G_{bb}$. Based on the nearest neighbor decoding, we have that

$$d(\bar{r}_b^*,\bar{z}_b^*) \leq d(\bar{r}_b^*,\bar{z}_b). \tag{48}$$

Now consider

$$\begin{aligned} d(\bar{r}_b,\bar{v}_b^*) &= d(\bar{r}_b,\bar{w}_b^*+\bar{z}_b^*) \\ &= d(\bar{r}_b^*+\bar{w}_b^*,\bar{w}_b^*+\bar{z}_b^*) \\ &= d(\bar{r}_b^*,\bar{z}_b^*). \end{aligned} \tag{49}$$

From (48) and (49), we have

$$\begin{aligned} d(\bar{r}_b,\bar{v}_b^*) &\leq d(\bar{r}_b^*,\bar{z}_b) \\ &= d(\bar{r}_b^*+\bar{w}_b^*,\bar{z}_b+\bar{w}_b^*). \end{aligned} \tag{50}$$

Note that $\bar{w}^* = (\bar{x}_1, \bar{x}_2, \bar{0}_3) \cdot G$, and $(\bar{0}_a, \bar{z}_b) = (\bar{0}_1, \bar{0}_2, \bar{x}_3) \cdot G$, where $\bar{0}_1$ is a zero r-tuple. Thus,

$$\bar{w}^* + (\bar{0}_a, \bar{z}_b) = \bar{v}$$

and $\bar{w}_b^* + \bar{z}_b = \bar{v}_b$.

It follows from (50) that

$$\begin{aligned} d(\bar{r}_b, \bar{v}_b^*) &\leq d(\bar{r}_b, \bar{v}_b) \\ &= w(\bar{r}_b + \bar{v}_b) \\ &= w(\bar{e}_b). \end{aligned} \tag{51}$$

It follows from (47) and (51) that

$$d(\bar{r}, \bar{v}^*) \leq w(\bar{e}) \leq \lfloor (d_{aa} + d_{bb} - 1)/2 \rfloor. \tag{52}$$

Similarly, we can show that, if $w(\bar{e}_b) \leq \lfloor (d_{bb} - 1)/2 \rfloor$, then

$$d(\bar{r}, \bar{v}^{**}) \leq w(\bar{e}) \leq \lfloor (d_{aa} + d_{bb} - 1)/2 \rfloor. \tag{53}$$

Hence we conclude that, for $w(\bar{e}) \leq \lfloor (d_{aa} + d_{bb} - 1)/2 \rfloor$, the distance between the received word $\bar{r}$ and the estimate of $\bar{v}$ (either $\bar{v}^*$ or $\bar{v}^{**}$) is no greater then $\lfloor (d_{aa} + d_{bb} - 1)/2 \rfloor$ if and only if the corresponding estimate of $\bar{x}_1$ is correct. Consequently, the smaller one of $d(\bar{r}, \bar{v}^*)$ and $d(\bar{r}, \bar{v}^{**})$ is no greater than $\lfloor (d_{aa} + d_{bb} - 1)/2 \rfloor$. Hence, the decoding rule at step 8 ensures the correct decoding of message component $\bar{x}_1$.

Next we consider the case for which the error pattern $\bar{e}$ contains $\lfloor (d_a - 1)/2 \rfloor$ or fewer errors, (i.e., $w(\bar{e}) \leq \lfloor (d_a - 1)/2 \rfloor$) where d_a is the minimum distance of code C_a. Since $\lfloor (d_a - 1)/2 \rfloor \leq \lfloor (d_{aa} + d_{bb} - 1)/2 \rfloor$, it follows from the above argument that $\bar{x}_1$ is decoded correctly. In fact, at least one of the two estimates, $(\bar{x}_1^*, \bar{x}_2^*)$ and $(\bar{x}_1^{**}, \bar{x}_3^{**})$, at the step 2 is correct. If $(\bar{x}_1^*, \bar{x}_2^*) = (\bar{x}_1, \bar{x}_2)$, it follows from the same argument as above from (47)

to (51) that

$$d(\bar{r},\bar{v}^*) \le w(\bar{e}) \le \lfloor (d_a-1)/2 \rfloor .$$

If $(\bar{x}_1^{**},\bar{x}_3^{**})=(\bar{x}_1,\bar{x}_3)$, it follows from (44) that

$$\bar{r}_a^{**} = \bar{e}_a + \bar{x}_2 \cdot G_{aa} .$$

Since $w(\bar{e}_a) \le w(\bar{e}) \le \lfloor (d_a-1)/2 \rfloor$, steps 5 and 6 will give the correct message component $\bar{x}_2$. Again we can show that

$$d(\bar{r},\bar{v}^{**}) \le \lfloor (d_a-1)/2 \rfloor .$$

Hence, for $w(\bar{e}) \le \lfloor (d_a-1)/2 \rfloor$, the distance between $\bar{r}$ and the estimate of $\bar{v}$ is no greater than $\lfloor (d_a-1)/2 \rfloor$ if and only if the corresponding estimate of $\bar{x}_2$ is correct. Thus the decoding rule at step 8 ensures the correct decoding of $\bar{x}_2$.

The last case is that $w(\bar{e}) \le \lfloor (d_b-1)/2 \rfloor$. By an argument similar to the one above, we can show that all three message components, $\bar{x}_1$, $\bar{x}_2$, and $\bar{x}_3$, will be decoded correctly. Either step 2 or 6 gives the correct estimate of $\bar{x}_3$.

Now, we can compare the (63,52) code listed in Table 2 to the time sharing of a (31,26) Hamming code and a (31,21) double-error-correcting BCH code. We see that the (63,52) code is superior considering information rate but inferior considering decoding complexity. We can also compare this (63,52) code to the time sharing of a (63,57) Hamming code and a (63,51) double-error-correcting BCH code. We see that the (63,52) code is inferior considering information rate but is superior considering decoding complexity. In general, the UEP code with parity check matrix of form (29) provides a tradeoff for coding designs considering information rate and decoding complexity.

IV. DIRECT SUMS OF PRODUCTS CODES

Let V be an (N,K) linear code with minimum distance D and W be an (n,k) linear code with minimum distance d. Let V⊗W denote the product of V and W [21]. Then V⊗W is an (Nn,Kk) linear code with minimum distance Dd. A codeword in V⊗W can be arranged as an nxN array in which every row is a codeword in V and every column is a codeword in W. For a nonzero code array in V⊗W, there are at least D nonzero columns and each nonzero column has at least d nonzero components. Hence, the weight of any nonzero code array in V⊗W is at least Dd. Product codes are capable of correcting both random and burst errors[21]. Now we consider direct sums of certain product codes which provide burst error protection in addition to the two-level random error protection.

Let V_1 and V_2 be (N,K_1) and (N,K_2) linear codes with minimum distances D_1 and D_2 respectively. The intersection of V_1 and V_2, denote $V_1 \cap V_2$, is a linear subcode of both V_1 and V_2. Let $\hat{D}$ be the minimum distance of $V_1 \cap V_2$. It is clear that $\hat{D} \geq D_1$ and $\hat{D} \geq D_2$. Let V_1+V_2 denote the set,

$$\{\bar{v} : \bar{v}=\bar{v}_1+\bar{v}_2 \text{ with } \bar{v}_1 \in V_1 \text{ and } \bar{v}_2 \in V_2\}.$$

V_1+V_2 is also a linear code and is a supercode of both V_1 and V_2. If $V_1 \cap V_2=\{\bar{0}\}$, then V_1+V_2 is equal to the direct sum $V_1 \oplus V_2$. Let D be minimum distance of V_1+V_2. Then $D \leq D_1, D_2$. Therefore, we have

$$\hat{D} \geq D_1, D_2 \geq D.$$

Let W_1 and W_2 be an (n,k_1) and an (n,k_2) linear codes with minimum distances d_1 and d_2 respectively. We assume that

$W_1 \cap W_2 = \{\bar{0}\}$

Then the direct sum W of W_1 and W_2 is an (n, k_1+k_2) linear code. Let d be the minimum distance of W. Then $d \leq d_1, d_2$.

For i=1 and 2, the product $V_i \otimes W_i$ is an $(Nn, K_i k_i)$ linear code with minimum distance $D_i d_i$. Since $W_1 \cap W_2 = \{\bar{0}\}$, $V_1 \otimes W_1$ and $V_2 \otimes W_2$ have only the zero code array in common. Let C be the direct sum of $V_1 \otimes W_1$ and $V_2 \otimes W_2$. Then C is an $(Nn, K_1 k_1 + K_2 k_2)$ linear code. A code array $\bar{c}$ in C is the sum of a code array $\bar{c}_1$ in $V_1 \otimes W_1$ and a code array $\bar{c}_2$ in $V_2 \otimes W_2$, i.e.,

$$\bar{c} = \bar{c}_1 + \bar{c}_2.$$

Each row in array $\bar{c}$ is a code word in V_1+V_2, and each column in $\bar{c}$ is a codeword in $W_1 \otimes W_2$.

Now we consider the weight of a nonzero code array $\bar{c}$ in $C = V_1 \otimes W_1 \oplus V_2 \otimes W_2$. If $\bar{c} \in V_1 \otimes W_1$, then the weight of $\bar{c}$, denoted $w(\bar{c})$, is at least $D_1 d_1$. If $\bar{c} \in V_2 \otimes W_2$, then $w(\bar{c}) \geq D_2 d_2$. If $\bar{c}$ is neither in $V_1 \otimes W_1$ nor in $V_2 \otimes W_2$, then $\bar{c}$ is the sum of a nonzero code array $\bar{c}_1$ in $V_1 \otimes W_2$ and a nonzero code array $\bar{c}_2$ in $V_2 \otimes W_2$. To determine the weight of $\bar{c} = \bar{c}_1 + \bar{c}_2$, there are four cases to be considered.

Case I: Suppose that all the nonzero rows in $\bar{c}_1$ and $\bar{c}_2$ are alike and identical to a certain vector $\bar{v}$. Then $\bar{v}$ must be a codeword in $V_1 \cap V_2$. Thus, $w(\bar{v}) \geq \hat{D}$. This implies that there are at least $\hat{D}$ nonzero columns in array $\bar{c}_1$ and at least $\hat{D}$ nonzero columns in array $\bar{c}_2$. Since $W_1 \cap W_2 = \{\bar{0}\}$, the sum of a nonzero column in $\bar{c}_1$ and a nonzero column in $\bar{c}_2$ is a nonzero codeword in $W_1 \otimes W_2$. Thus, there are at least $\hat{D}$ nonzero columns in array $\bar{c} = \bar{c}_1 + \bar{c}_2$, and each of these columns has weight at least d. Therefore, $w(\bar{c}) \geq \hat{D}d$.

Case II: Suppose that all the nonzero rows in $\bar{c}_1$ are identical to some codeword $\bar{v}_1$ in V_1 and all the nonzero rows in $\bar{c}_2$ are identical to some codeword $\bar{v}_2$ in V_2, where $\bar{v}_1 \neq \bar{v}_2$. Then $\bar{v}_1+\bar{v}_2$ is a nonzero codeword in V_1+V_2 and has weight at least D. Note that $w(\bar{v}_1) \geq D_1$ and $w(\bar{v}_2) \geq D_2$. There are two types of nonzero columns in $\bar{c}$. The first type is that each column is either the sum of a zero column in $\bar{c}_1$ and a nonzero column in $\bar{c}_2$ or the sum of a nonzero column in $\bar{c}_1$ and a zero column in $\bar{c}_2$. Such a column is either a nonzero codeword in W_2 or a nonzero codeword in W_1. Therefore, a nonzero column of the first type in $\bar{c}$ has weight at least $\min\{d_1,d_2\}$. The second type of nonzero columns in $\bar{c}$ is that each column is the sum of a nonzero column in $\bar{c}_1$ and a nonzero column in $\bar{c}_2$. Such a column is a nonzero codeword in $W_1 \oplus W_2$ and has weight at least d. The fact that $w(\bar{v}_1+\bar{v}_2) \geq D$ implies that there are at least D type-1 nonzero columns in $\bar{c}$. Let f be the number of type-1 nonzero columns in $\bar{c}$ where $f \geq D$. Then there are at least $\lceil (D_1+D_2-f)/2 \rceil$ type-2 nonzero columns in $\bar{c}$. Hence a lower bound on the weight of $\bar{c}$ is

$$\min_{f \geq D} \{f \cdot \min\{d_1,d_2\} + \lceil (D_1+D_2-f)/2 \rceil \cdot d\}$$
$$= D \cdot \min\{d_1,d_2\} + \lceil (D_1+D_2-D)/2 \rceil \cdot d.$$

Case III: Suppose that there are two nonzero rows $\bar{v}_1$ and $\bar{v}_1'$ in $\bar{c}_1$ such that $\bar{v}_1 \neq \bar{v}_1'$. Then there are at least $D_1+\lceil D_1/2 \rceil$ nonzero columns in $\bar{c}_1$. This implies that there are at least $D_1+\lceil D_1/2 \rceil$ nonzero columns in $\bar{c}$. Each of these nonzero columns is a nonzero codeword in $W_1 \oplus W_2$ and has weight at least d. Thus the weight of $\bar{c}$ is at least $\{D_1+\lceil D_1/2 \rceil\} \cdot d$.

Case IV: Suppose that there are two nonzero rows, $\bar{v}_2$ and $\bar{v}_2'$, in

$\bar{c}_2$. It follows from the same argument as that in Case III that $w(\bar{c}) \geq \{D_2 + \lceil D_2/2 \rceil\} \cdot d$.

Denote $D \cdot \min\{d_1, d_2\} + \lceil (D_1 + D_2 - D)/2 \rceil \cdot d$ by λ, $\{D_1 + \lceil D_1/2 \rceil\} \cdot d$ by λ_1 and $\{D_2 + \lceil D_2/2 \rceil\} \cdot d$ by λ_2. Summarizing the above results we have the following weight structure of a nonzero code array $\bar{c}$ in $C = V_1 \otimes W_1 \oplus V_2 \otimes W_2$:

(1) For $\bar{c} \in V_1 \otimes W_1$, $w(\bar{c}) \geq D_1 d_1$;

(2) For $\bar{c} \in V_2 \otimes W_2$, $w(\bar{c}) \geq D_2 d_2$; and

(3) For $\bar{c} \notin V_1 \otimes W_1$ and $\bar{c} \notin V_2 \otimes W_2$,

$$w(c) \geq \min \{\hat{D}d, \lambda, \lambda_1, \lambda_2\}.$$

From the above weight distribution, we see that the weight of a nonzero code array $\bar{c}$ in $V_1 \otimes W_1 \oplus V_2 \otimes W_2$ is at least $\min \{D_1 d_1, D_2 d_2, \hat{D}d, \lambda, \lambda_1, \lambda_2\}$.

Suppose $\min \{D_1 d_1, \hat{D}d, \lambda, \lambda_1, \lambda_2\} \geq D_2 d_2$.

Then we have the following weight structure of a nonzero code array $\bar{c}$ in $V_1 \otimes W_1 \oplus V_2 \otimes W_2$:

(1) For $\bar{c} \in V_2 \otimes W_2$, $w(\bar{c}) \geq D_2 d_2$.

(2) For $\bar{c} \in V_1 \otimes W_1 \oplus V_2 \otimes W_2 - V_2 \otimes W_2$,

$$w(\bar{c}) \geq \min \{D_1 d_1, \hat{D}d, \lambda, \lambda_1, \lambda_2\}.$$

It follows from Theorem 5 that $C = V_1 \otimes W_1 \oplus V_2 \otimes W_2$ is linear block code with a separation vector $\bar{s} = (s_1, s_2)$ where

$s_1 \geq \min \{D_1 d_1, \hat{D}d, \lambda, \lambda_1, \lambda_2\}$,

$s_2 \geq D_2 d_2$.

The message space A for C is the product of $A_1 = \{0,1\}^{K_1 k_1}$ and $A_2 = \{0,1\}^{K_2 k_2}$

Example 4: Let V_1 and V_2 be two equivalent (7,4) Hamming codes.

Let W_1 and W_2 be the (7,1) and (7,3) BCH codes over GF(2) respectively. Then $W_1 \otimes W_2$ is a (7,4) Hamming code. The minimum distances of V_1 and V_2 are $D_1=3$ and $D_2=3$ respectively. The minimum distances of W_1, W_2, and $W_1 \otimes W_2$ are $d_1=7$, $d_2=4$, and $d=3$ respectively. Note that $V_1 \cap V_2$ is the (7,1) binary code with minimum distance $\hat{D}=7$ while V_1+V_2 is the (7,7) binary code with minimum distance $D=1$. Thus, $\lambda = D \cdot \min\{d_1,d_2\} + \lceil (D_1+D_2-D)/2 \rceil \cdot d = 13$, $\lambda_1 = \{D_1 + \lceil D_1/2 \rceil\} \cdot d = 15$, $\lambda_2 = \{D_2 + \lceil D_2/2 \rceil\} \cdot d = 15$, $\hat{D}d=21$, $D_1d_1=21$, and $D_2d_2=12$. Note that $N=7$, $K_1=K_2=4$, $n=7$, $k_1=1$, $k_2=3$. Since $\min\{D_1d_1, \hat{D}d, \lambda, \lambda_1, \lambda_2\} = 13 \geq D_2d_2 = 12$, we see that $V_1 \otimes W_1 \oplus V_2 \otimes W_2$ is a two-level UEP (49,16) binary linear code for the message space $A=A_1 \times A_2$ with separation vector $\bar{s}=(s_1,s_2)$, where $A_1=\{0,1\}^4$, $A_2=\{0,1\}^{12}$, $s_1 \geq 13$, $s_2 \geq 12$. Thus, 4 message bits of a message are protected against up to 6 random errors, while 12 other message bits of the same message are protected against up to 5 random errors. We may compare this code to the product code of two (7,4) BCH codes with minimum distance 3, which is a (49,16) binary linear code with minimum distance 9.

$\Delta\Delta$

A special case for the above direct sums of product codes is that

$$V_1 \cap V_2 = \{\bar{0}\}.$$

For this case, if $\min\{D_1d_1, \lambda, \lambda_1, \lambda_2\} \geq D_2d_2$, then a nonzero code array $\bar{c}$ in $V_1 \otimes W_1 \oplus V_2 \otimes W_2$ has the following weight structure:

(1) For $\bar{c} \in V_2 \otimes W_2$, $w(\bar{c}) \geq D_2d_2$;

(2) For $\bar{c} \in V_1 \otimes W_1 \oplus V_2 \otimes W_2 - V_2 \otimes W_2$,

$$w(\bar{c}) \geq \min\{D_1d_1, \lambda, \lambda_1, \lambda_2\}.$$

Then the code $V_1 \otimes W_1 \oplus V_2 \otimes W_2$ is a linear block code with separation

vector $\bar{s}=(s_1,s_2)$ where

$$s_1 \geq \min\{D_1 d_1, \lambda, \lambda_1, \lambda_2\},$$

$$s_2 \geq D_2 d_2.$$

A. A class of Direct Sums of Product Codes

Now we present a specific class of direct sums of product codes. Let α and β be two different primitive N-th roots of unity. Let V_1 be an (N,K_1) binary cyclic code which has α, $\alpha^2,\ldots,\alpha^{2t}$ and their conjugates as zeros. Let V_2 be an (N,K_2) binary cyclic code which has $\beta,\beta^2,\ldots,\beta^{2t}$ and their conjugates as zeros. Clearly, V_1 and V_2 are equivalent codes. Hence, $K_1=K_2=K$ and $D_1=D_2\geq 2t+1$, where D_1 is the minimum distance of V_1 and D_2 is the minimum distance of V_2. If the set $\{(\beta^i)^{2^m} : i=1,2,\ldots,2t,\ m$ is an integer$\}$ contains $\{\alpha^{2t+1},\alpha^{2t+2},\ldots,\alpha^{2t+2s}\}$ as a subset, then $V_1 \cap V_2$ includes $\alpha,\alpha^2,\ldots,\alpha^{2t+2s}$ as zeros. Thus, either the minimum distance $\hat{D}$ of $V_1 \cap V_2$ is at least $2t+2s+1$ or $V_1 \cap V_2 = \{\bar{0}\}$ which is the case that $V_1 \cap V_2$ contains all the α^i's as zeros. If the set $\{(\alpha^i)^{2^m} : i=1,2,\ldots,2t$ and m is an integer$\}$ contains $\{\beta,\beta^2,\ldots,\beta^{2u}\}$ as a subset, then V_1+V_2 contains $\beta,\beta^2,\ldots,\beta^{2u}$ as zeros. Thus, D, the minimum distance of V_1+V_2 is at least $2u+1$.

With the above V_1 and V_2, if $\min\{(2t+1)d_1,(2t+2s+1)d,\lambda,\lambda_1,\lambda_2\} \geq (2t+1)d_2$, the direct sum $V_1 \otimes W_1 \oplus V_2 \otimes W_2$ is an $(Nn,K(k_1+k_2))$ code with separation vector $\bar{s} = (s_1,s_2)$ where

$$s_1 \geq \min\{(2t+1)d_1,(2t+2s+1)d,\lambda,\lambda_1,\lambda_2\},$$

$$s_2 \geq (2t+1)d_2,$$

$$\lambda = (2u+1)\cdot\min\{d_1,d_2\}+(2t-u+1)d,$$

$$\lambda_1 = \lambda_2 = (3t+2)d.$$

Example 5: Let α be a primitive element in $GF(2^5)$. Let V_1 be a (31,21) BCH code with minimum distance $D_1=5$, which contains α, α^3 and their conjugates as zeros. Let V_2 be a (31,21) BCH code with minimum distance $D_2=5$, which contains α^3, $(\alpha^3)^3$, and their conjugates as zeros. Since α^9 is a conjugate of α^5, $V_1 \cap V_2$ includes $\alpha, \alpha^3, \alpha^5$, and their conjugates as zeros. Since $V_1 \cap V_2 \neq \{\bar{0}\}$, the minimum distance $\hat{D}$ of $V_1 \cap V_2$ is at least 7. Furthermore, the minimum distance D of V_1+V_2 is at least 3 since α^3 is a zero for both V_1 and V_2. Let W_1 and W_2 be (7,1) and (7,3) BCH codes over GF(2). Thus, the minimum distance of W_1 is $d_1=7$ and the minimum distance of W_2 is $d_2=4$. Furthermore, $W_1 \circledast W_2$ is a (7,4) BCH code over GF(2) with minimum distance d=3. Thus, t=2, s=1, u=1, $\lambda = (2u+1)\cdot\min\{d_1,d_2\}+(2t-u+1)\cdot d = 24$, $\lambda_1 = \lambda_2 = (3t+2)\cdot d = 24$, $\hat{D}d \geq (2t+2s+1)\cdot d=21$, $D_1d_1=(2t+1)\cdot d_1=35$, and $D_2d_2 = (2t+1)\cdot d_2=20$. Note that N=31, n=7, $k_1=1$, $k_2=3$, $K_1=K_2=21$. Since $\min\{D_1d_1, \hat{D}d, \lambda, \lambda_1, \lambda_2\} \geq 21 \geq D_2d_2 = 20$, $V_1 \circledast W_1 \oplus V_2 \circledast W_2$ is a (217,84) binary two-level UEP linear code for the message space $A=A_1 \times A_2$ with separation vector $\bar{s}=(s_1,s_2)$ where $A_1=\{0,1\}^{21}$, $A_2=\{0,1\}^{63}$, $s_1 \geq 21$, and $s_2 \geq 20$. Note that the product code of a (7,4) Hamming code with minimum distance 3 and a (31,21) BCH code with minimum distance 5 has minimum distance 15.

B. Burst Error Correction

So far, we have studied the multi-level error-correcting capabilities of block codes through their separation vectors. However, the separation vector of a block code only specified its

multi-level random-error-correcting capability. Now we want to show that the direct sum of product codes inherits the burst-error-correcting capability from their component product codes. If an nxN code array $\bar{c}$ in $V_1 \otimes W_1 \oplus V_2 \otimes W_2$ is transmitted row by row, any error burst of length $N \cdot \lfloor (d-1)/2 \rfloor$ can affect at most $\lfloor (d-1)/2 \rfloor$ components in each column of $\bar{c}$. Hence, every column of $\bar{c}$ can be correctly recovered. That means that any error burst of length up to $N \cdot \lfloor (d-1)/2 \rfloor$ can be corrected. Thus, in addition to the random-error-correcting capability, $V_1 \otimes W_1 \oplus V_2 \otimes W_2$ has burst-error-correcting capability. Suppose that $V_1 \otimes W_1 \oplus V_2 \otimes W_2$ is a code for the message space $A = A_1 \times A_2$ with separation vector $\bar{s} = (s_1, s_2)$, where $s_1 \geq s_2$. Let $t_1 = \lfloor (s_1-1)/2 \rfloor$, $t_2 = \lfloor (s_2-1)/2 \rfloor$. We shall show that

(1) Any component message from A_1 is protected against up to t_1 random errors and any error burst of length up to $N \cdot \lfloor (d-1)/2 \rfloor$ (not the combination of both random errors and error burst).

(2) Any component message from A_2 is protected against up to t_2 random errors and any error burst of length up to $N \cdot \lfloor (d-1)/2 \rfloor$.

For $i=1,2$, let $\bar{e}_r^{(i)}$ be an nxN array with at most t_i nonzero components. Let $\bar{e}_b$ be an nxN array with a burst of length at most $N \cdot \lfloor (d-1)/2 \rfloor$. To justify property (1), we need to show that both $\bar{e}_r^{(1)} + \bar{c}_2$ and $\bar{e}_b + \bar{c}_2'$ are correctable error patterns for $V_1 \otimes W_1$, where $\bar{c}_2$ and $\bar{c}_2'$ are two arbitrary code arrays in $V_2 \otimes W_2$. Equivalently, we need to show that $\bar{e}_r^{(1)} + \bar{c}_2$ and $\bar{e}_b + \bar{c}_2'$ can not be in the same coset of the standard array for $V_1 \otimes W_1$ if

$\bar{e}_r^{(1)}+\bar{c}_2 \neq \bar{e}_b+\bar{c}_2'$. To justify property (2), we need to show that both $\bar{e}_r^{(2)}$ and $\bar{e}_b$ are correctable patterns for $V_1 \otimes W_1 \otimes V_2 \otimes W_2$. Equivalently, we need to show that $\bar{e}_r^{(2)}$ and $\bar{e}_b$ can not be in the same coset of the standard array for $V_1 \otimes W_1 \otimes V_2 \otimes W_2$ if $\bar{e}_r^{(2)} \neq \bar{e}_b$.

Suppose that $\bar{e}_r^{(1)}+\bar{c}_2$ and $\bar{e}_b+\bar{c}_2'$ are in the same coset of the standard array for $V_1 \otimes W_1$, where $\bar{c}_2$ and $\bar{c}_2'$ are two arbitrary codewords of $V_2 \otimes W_2$. The sum of $\bar{e}_r^{(1)}+\bar{c}_2$ and $\bar{e}_b+\bar{c}_2'$ must be equal to some codeword $\bar{c}_1$ in $V_1 \otimes W_1$. Then, we have $\bar{e}_r^{(1)}+\bar{e}_b=\bar{c}_1+\bar{c}_2+\bar{c}_2'$. If $\bar{c}_1=0$, then $\bar{e}_r^{(1)}+\bar{c}_2=\bar{e}_b+\bar{c}_2'$. We only have to consider the case for which $\bar{e}_r^{(1)}+\bar{c}_2 \neq \bar{e}_b+\bar{c}_2'$. Hence, $\bar{c}_1 \neq \bar{0}$. Thus, the weight of $\bar{e}_r^{(1)}+\bar{e}_b$ is at least s_1. Consider a nonzero column of $\bar{e}_r^{(1)}+\bar{e}_b$, which is a nonzero codeword of W. Thus, this column has at least d nonzero components. Note that there are at most $t=\lfloor (d-1)/2 \rfloor$ nonzero components in each column of $\bar{e}_b$. Thus, a nonzero column of $\bar{e}_r^{(1)}+\bar{e}_b$ is composed of at most t nonzero components from $\bar{e}_b$ and at least d-t components from $\bar{e}_r^{(1)}$. This implies that a nonzero column of $\bar{e}_r^{(1)}$ has at least d-t nonzero components. Since there are at most t_1 nonzero components in $\bar{e}_r^{(1)}$, there are at most $\lfloor t_1/(d-t) \rfloor$ nonzero columns in $\bar{e}_r^{(1)}$. This implies that there are at most $\lfloor t_1/(d-t) \rfloor$ nonzero columns in $\bar{e}_b$. Therefore, $\bar{e}_b$ has at most $\lfloor t_1/(d-t) \rfloor \cdot t$ nonzero components. Then, we see that $\bar{e}_r^{(1)}+\bar{e}_b$ contains at most $\lfloor t_1/(d-t) \rfloor \cdot t+t_1$ nonzero components. However,

$$\lfloor t_1/(d-t) \rfloor \cdot t+t_1 \leq t_1\{[t/(d-t)]+1\}$$
$$< 2t_1$$

$< s_1$.

This contradicts the previous result which requires $w(\bar{e}_r^{(1)}+\bar{e}_b)$ to be no less than s_1. Thus, we have proved property (1).

Suppose that $\bar{e}_r^{(2)}$ and $\bar{e}_b$ are in the same coset of $V_1 \otimes W_1 \otimes V_2 \otimes W_2$. Then, $\bar{e}_r^{(2)}+\bar{e}_b=\bar{c}$ for some nonzero codeword $\bar{c}$ in $V_1 \otimes W_1 \otimes V_2 \otimes W_2$. Thus, $w(\bar{e}_r^{(2)}+\bar{e}_b) \geq s_2$. By an argument similar to that for property (1), we find that there are at most $\lfloor t_2/(d-t) \rfloor$ nonzero columns in $\bar{e}_b$. Then, $\bar{e}_r^{(2)}+\bar{e}_b$ contains at most

$$\lfloor t_2/(d-t) \rfloor \cdot t + t_2 \leq t_2\{[t/(d-t)]+1\}$$
$$< 2t_2$$
$$< s_2$$

nonzero components, which leads to a contradiction. Thus, we have proved property (2).

Consider the (49,16) binary code illustrated in Example 4. For this code, 4 message bits of a message are protected against up to 6 random errors and any error burst of length up to 7, while the other 12 message bits of the same message are protected against up to 5 random errors and any error burst of length up to 7.

Consider the (217,84) binary linear code illustrated in Example 5. For this code, 21 message bits of a message are protected against up to 10 random errors and any error burst of length up to 31, while the other 63 message bits of the same message are protected against up to 9 random errors and any error burst of length up to 31.

If an nxN code array $\bar{c}$ in $V_1 \otimes W_1 \otimes V_2 \otimes W_2$ is transmitted column

by column, any error burst of length $n \cdot \lfloor (D-1)/2 \rfloor$ can affect at most $\lfloor (D-1)/2 \rfloor$ components in each row of $\bar{c}$. Hence, every row of $\bar{c}$ can be recovered. Therefore, any error burst of length up to $n \cdot \lfloor (D-1)/2 \rfloor$ can be recovered. With an argument similar to the case for which a codeword is transmitted row by row, we can show that

(1) Any component message from A_1 is protected against up to t_1 random errors and any error burst of length up to $n \cdot \lfloor (D-1)/2 \rfloor$.

(2) Any component message from A_2 is protected against up to t_2 tandom errors and any error burst of length up to $n \cdot \lfloor (D-1)/2 \rfloor$.

V. CONCLUSION

This research is concerned with coding for unequal error protection. The basic idea is that it is possible to achieve multi-level error-correcting capability of a block code by partitioning the code into disjoint groups (clouds). For a linear direct-sum code, if a partition yields a proper weight structure, then the code has multi-level error-correcting capability and hence is a UEP code. By studying the weight structures of various linear codes, we presented the following UEP codes:

(1) A class of UEP codes for which the generator matrices (or parity check matrices) are certain combinations of generator matrices (or parity check matrices) of shorter codes. Especially, there is a class of system-

atic codes which meet the Hamming bound for systematic UEP codes.

(2) A class of direct sums of product codes which are UEP codes and have greater minimum distance than the simple product codes of comparable dimensions. Besides, the direct sums of product codes still retain the burst-error-correcting capabilities of simple product codes.

We have also constructed two classes of UEP cyclic codes which are not presented in this paper due to limited space[22]. From the results of our research, we believe that, by our approach, i.e., studying the weight structure of block codes, more classes of powerful UEP codes can be constructed in the future.

APPENDIX A

Proof of Lemma 1

Let $\bar{v}_0$ and $\bar{w}_0$ be two vectors in V and W respectively such that

$$d(\{\bar{r}\},V) = d(\bar{r},\bar{v}_0),$$

and $$d(\{\bar{r}\},W) = d(\bar{r},\bar{w}_0).$$

Since Hamming distance satisfies triangluar inequality, we have

$$d(\{\bar{r}\},V)+d(\{\bar{r}\},W) = d(\bar{r},\bar{v}_0)+d(\bar{r},\bar{w}_0) \geq d(\bar{v}_0,\bar{w}_0).$$

However, it follows from the definition of d(V,W) given by (3) that

$$d(\bar{v}_0,\bar{w}_0) \geq d(V,W).$$

Combining the above results, we obtain the inequality,

$$d(\{\bar{r}\},V)+d(\{\bar{r}\},W) \geq d(V,W).$$

APPENDIX B

Systematic Equivalent Code of The Code with Parity Check Matrix Given by (30)

Now, we will show that the code C with parity check matrix H given by (30) can be transformed into a systematic code with identical two-level error correcting capability.

Let $H(2^m-1)$ be the submatrix of H which consists of the first 2^m-1 columns of H. Note that a linear combination of less than 5 columns from H with at least one column from $H(2^m-1)$ can not be zero. This implies that a codeword of C with at least one nonzero component at the first 2^m-1 positions has weight at least 5. By row operations, H can be transformed into the following form:

$$H' = \begin{bmatrix} I_1 & P & 0_{12} & P' \\ 0_{21} & & I_2 & \end{bmatrix}$$

where I_1 is an mxm identity matrix, I_2 is an $(m+\ell)x(m+\ell)$ identity matrix, 0_{21} is the zero $(m+\ell)x$ m matrix, 0_{12} is the zero $mx(m+\ell)$ matrix, P is some $(2m+\ell)$ x (2^m-1-m) matrix, and P' is some $(2m+\ell)$ x $(2^{m+\ell}-2^m-m-\ell)$ matrix. Let $k_1=2^m-m-1$ and $k_2=2^{m+\ell}-2^m-m-\ell$. Let $\bar{x}_1$ be a component message from $A_1=\{0,1\}^{k_1}$ and $\bar{x}_2$ be a component message from $A_2=\{0,1\}^{k_2}$. Thus, $\bar{x}_1$ and $\bar{x}_2$ are k_1-tuple and k_2-tuple respectively. From H', we see that any codeword $\bar{v}(\bar{x}_1,\bar{x}_2)$ of C can be written as

$$[\bar{p}\ \ \bar{x}_1\ \ \bar{p}'\ \ \bar{x}_2],$$

where $\bar{p}$ and $\bar{p}'$ are some m-tuple and some $(m+\ell)$-tuple respectively which represent the $(2m+\ell)$ redundant digits [20].

Regardless of the order of redundant digits and message digits, the expression of $\bar{v}(\bar{x}_1,\bar{x}_2)=[\bar{p}\ \bar{x}_1\ \bar{p}'\ \bar{x}_2]$ is in fact in systematic form. Note that the message digits in $\bar{x}_1$ are located within the first 2^m-1 positions of $\bar{v}(\bar{x}_1,\bar{x}_2)$. From the result at the beginning of this paragraph and (25), we have

$$s_1 = \min \{w(\bar{v}(\bar{x}_1,\bar{x}_2) : \bar{x}_1 \in A_1 \text{ and } \bar{x}_1 \neq 0\} \geq 5.$$

Clearly, $s_2 = \min \{w(\bar{v}(\bar{x}_1,\bar{x}_2) : \bar{x}_2 \in A_2 \text{ and } \bar{x}_2 \neq 0\} = 3$.

Thus, C is in systematic form with 2^m-m-1 message bits protected against any 2 or fewer random errors, while the other $2^{m+\ell}-2^m-m-\ell$ message bits protected against any single error.

APPENDIX C

Proof of Theorem 6

Pick an arbitrary generator matrix G_{aa} of the (n_a,k_a-r) code C_a generated by the parity check matrix $[H_{aa}{}^T\ H_{ab}{}^T]^T$. It is easy to check that $[G_{aa}\ 0'_{ab}]\cdot H^T=0$. Hence, the subcode C_2 generated by the generator matrix $[G_{aa}\ 0'_{ab}]$ is a k_a-r dimensional subcode of C. Pick an arbitrary generator matrix G_{bb} of the $(n_b,\ k_b-r)$ code C_b generated by the parity check matrix $[H_{bb}{}^T\ H_{ba}{}^T]^T$. We see that $[0'_{ba}\ G_{bb}]\cdot H^T=\bar{0}$. Hence, the subcode C_3 generated by the generator matrix $[0'_{ba}\ G_{bb}]$ is a k_b-r dimensional subcode of C. Since $C_2 \cap C_3 = \{\bar{0}\}$, the direct sum of C_2 and C_3 forms a k_a+k_b-2r dimensional subcode of C. There must exist an r dimensional subcode C_1 such that C is the direct sum of C_1, C_2, and C_3. Pick an arbitrary generator matrix of C_1 which is expressed as $[G_{ab}\ G_{ba}]$ where G_{ab} is an rxn_a matrix and G_{ba} is an rxn_b matrix. Thus, the matrix G of (41) is the generator matrix of C. Note that $G_{ab}\cdot H_{aa}{}^T=\bar{0}$ and $G_{ba}\cdot H_{bb}{}^T=\bar{0}$. To prove that $[G_{aa}{}^T\ G_{ab}{}^T]$ is a generator matrix of the $(n_a,\ k_a)$ code C_{aa} generated by the parity check matrix H_{aa}, we need to show that G_{ab} generates an r dimensional subcode C_{ab} of C_{aa}, for which the only common codeword with C_a is the zero n_a-tuple. The fact that $G_{ab}\cdot H_{aa}{}^T=\bar{0}$ implies that G_{ab} generates a subcode of C_{aa}. Assume that the rank of G_{ab} is less than r. Since the rank of $[G_{ab}\ G_{ba}]$ is r, there exists a nonzero codeword $\bar{v}$ in C_1 for which the first n_a positions are all zero. This implies that $\bar{v}$ is in C_3 which contradicts the fact that C is the direct sum of C_1, C_2, and C_3. Thus, the rank of G_{ab} is r and G_{ab} generates an r

dimensional subcode C_{ab} of C_{aa}. Assume that the code C_{ab} and C_a have a nonzero common codeword $\bar{v}_a$. Let $\bar{v}_1=[\bar{v}_a\ \bar{v}_b]$ be a codeword of C_1 where $\bar{v}_b$ is some nonzero n_b-tuple. Note that $\bar{v}_2=[\bar{v}_a\ \bar{0}_b]$ is a codeword of C_2, where $\bar{0}_b$ is the zero n_b-tuple. Then, $\bar{v}_1+\bar{v}_2=[\bar{0}_a\ \bar{v}_b]$, where $\bar{0}_a$ is the zero n_a-tuple. Thus, $[\bar{0}_a\ \bar{v}_b]$ is in C_3 which again leads to a contradiction. Hence, C_{ab} and C_a have only zero n_a-tuple as common codeword. Thus, we have shown that $[G_{aa}{}^T\ G_{ab}{}^T]^T$ is a generator matrix of the $(n_a,\ k_a)$ code c_{aa} generated by the parity check matrix H_{aa}. We can similarly prove that $[G_{bb}{}^T\ G_{ba}{}^T]^T$ is a generator matrix of the (n_b,k_b) code C_{bb} generated by the parity check matrix H_{bb}.

REFERENCES

1. B. Masnick and J. Wolf, "On Linear Unequal Error PRotection Codes," IEEE Trans. on Information Theory, IT-13, No. 4, pp. 600-607, July, 1967.

2. T. Cover, "Broadcast Channels," IEEE Trans. on Information Theory, IT-18, pp. 2-14, Jan. 1972.

3. P.P. Bergmans, "Random Coding Theorem for Broadcast Channels with Degraded Components," IEEE Trans. on Information Theory IT-19, pp. 197-207, Mar. 1973.

4. A.D. Wyner, "A Theorem on the Entropy of Certain Binary Sequences and Application: Part II," IEEE Trans. on Informations Theory, IT-19, pp. 772-777, Nov., 1973.

5. R.G. Gallager, "Capacity and Coding for Degraded Channels," Problemy Peredachi Informatsii, Vol. 10, No. 3, pp. 3-14, 1974.

6. L.A. Bassalygo, et.al. "Bounds for Codes with Unequal Protection of Two Sets of Messages," Problemy Peredachi Informatsii, Vol. 15, No. 3, pp. 40-49, July-September, 1979.

7. G.L. Katsman, "Bounds on Volume of Linear Codes with Unequal Information Symbol Protection," Problemy Peredachi Informatsii, Vol. 16, No. 2, pp. 25-32, April-June 1980.

8. T. Kasami, S. Lin, V.K. Wei, S. Yamamura, "Coding for the Binary Symmetric Broadcast Channel with Two Receivers," IEEE Trans. on Information Theory, Vol. IT-31, No. 5, pp. 616-625, September, 1985.

9. C. Heagard, H. dePedro, and J. Wolf, "Permutation Codes for the Gaussian Broadcast Channel with Two Receivers," IEEE Trans. on Information Theory, Vol. IT-24, No. 5, pp. 569-578, September, 1978.

10. W.C. Gore and C.C. Kilgus, "Cyclic Codes with Unequal Error Protection," IEEE Trans. on Information Theory, IT-17, No. 2, pp. 214-215.

11. D. Mandelbaum, "Unequal-Error-Protection Codes Derived from Difference Sets," IEEE Trans. on Information Theory, IT-18, No. 5, pp. 686-687, September, 1972.

12. C.C. Kilgus and W.C., Gore, "A Class of Cyclic Unequal-Error-Protection Codes," IEEE Trans. on Information Theory, IT-18, No. 5, pp. 687-690, September, 1972.

13. L.A. Dunning and W.E. Robbins, "Optimal Encoding of Linear Block Codes for Unequal Error Protection," Information and

Control 37, pp. 150-177, 1978.

14. V.N. Dynkin and V.A. Togonidze, "Cyclic Codes with Unequal Symbol Protection," Problemy Peredachi Informatsii, Vol. 12, No. 1, pp. 24-28, January-March, 1976.

15. V.A. Zinovev and V.V. Zyablov, "Codes with Unequal Protection of Information Symbols," Problemy Peredachi Informatsii, Vol. 15, No. 3, pp. 50-60, July-September, 1979.

16. I.M. Boyarinov and G.L. Katsman, "Linear Unequal Error Protection Codes," IEEE Trans. on Information Theory, Vol. IT-27, No. 2, pp. 168-175, March 1981.

17. I.M. Boyarinov, "Combined Decoding Methods for linear Codes with Unequal Protection of Information Symbols," Problemy Peredachi Informatsii, Vol. 19, No. 1, pp 17-25, January-March, 1983.

18. W.J. Van Gils, "Two Topics on Linear Unequal Error Protection Protection Codes: Bounds on Their Length and Cyclic Code Classes," IEEE Trans. on Information Theory, Vol. IT-29, No. 6, November, 1983.

19. C.P. Downey and J.K. Karlof, "Group Codes for the Gaussian Broadcast Channel with Two Receivers," IEEE Trans. on Information Theory, Vol. IT-26, No. 4, pp. 406-412, July, 1980.

20. W.W. Peterson and E.J. Weldon Jr, "Error Correcting Codes" The MIT Press, Cambridge, Massachussetts, 1972.

21. S. Lin and D.J. Costello,Jr, "Error Control Coding: Fundamentals and Applications," Prentice Hall, New Jersey, 1983.

22. M.C. Lin, "Coding for Unequal Error Protection", Ph.D. dissertation, University of Hawaii, 1986.

23. W.J. Van Gils, "On Linear Unequal Error Protection Codes", Report, Research Laboratories, N.V. Philips' Gloeilampenfabrieken, Eindhoven-Netherlands.

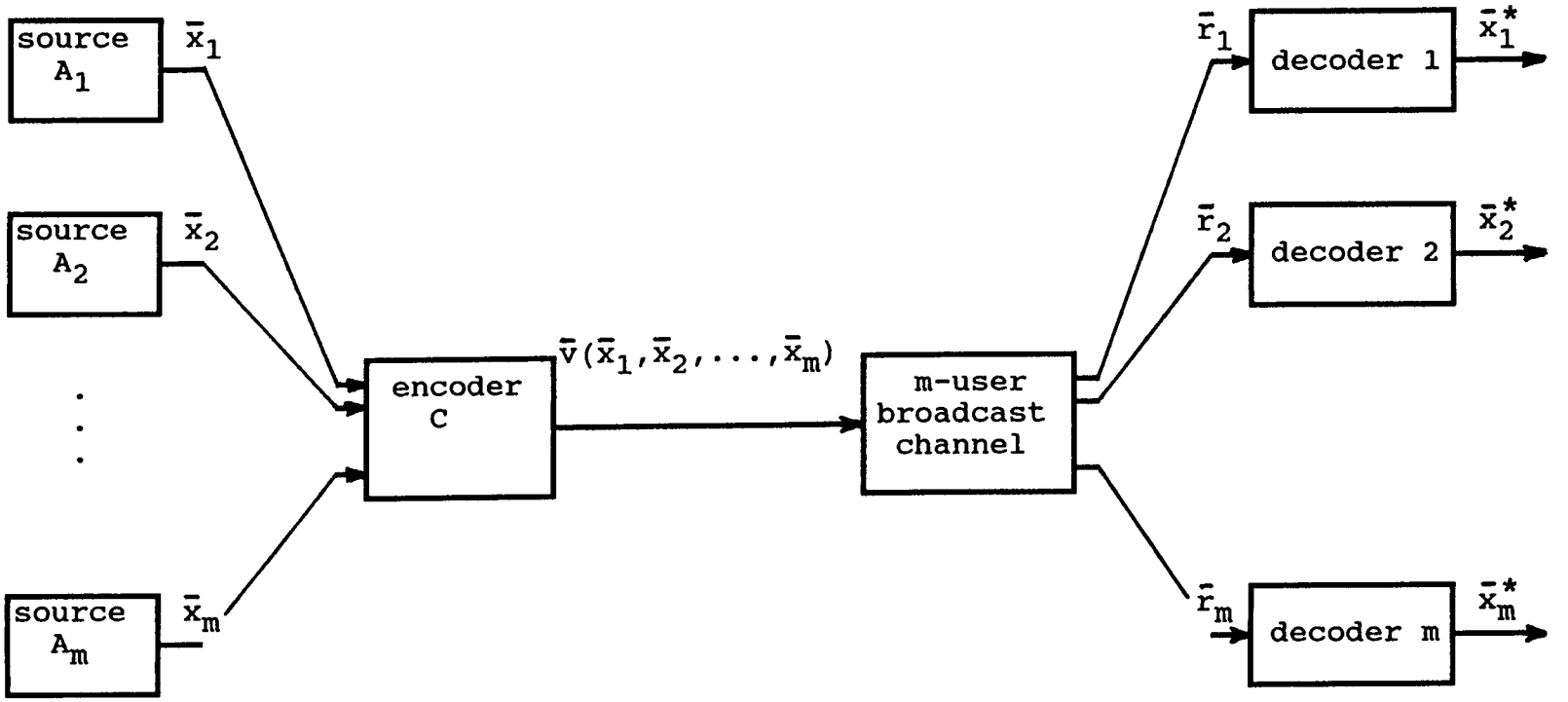

Figure 1. An m-user broadcast communication system.