CODES WITH MULTI-LEVEL ERROR-CORRECTING CAPABILITIES*

Mao-Chao LIN
National Taiwan University, Taipei, Taiwan, ROC

Shu LIN
University of Hawaii, Honolulu, Hawaii 96822, USA

Received 29 December 1987
Revised 29 August 1988

In conventional channel coding, 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 circumstances, some information symbols in a message are more significant than the other symbols. As a result, it is desirable to devise codes with multi-level error-correcting capabilities. In this paper, we investigate block codes with multi-level error-correcting capabilities, which are also known as unequal error protection (UEP) codes. Several classes of UEP codes are constructed. One class of codes satisfies the Hamming bound on the number of parity-check symbols for systematic linear UEP codes and hence is optimal.

1. 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, in some occasions, some information symbols in a message are more significant than the 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. 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 encoding 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 m parts of a message. It has been proved that a single code with m levels of

* This research was supported by NSF Grant DCI-8418248-01 and NASA Grant NAG 5-407.

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 [1–8]. Moreover, a single code requires only one encoder and one decoder. This may be desirable in many situations. A code with multi-levels of error-correcting capabilities is known as an unequal error protection (UEP) code. UEP codes were first studied by Masnick and Wolf [9], than by other coding theorists [5, 6, 10–20].

In this paper, we investigate codes with multi-level error-correcting capabilities. Two classes of multi-level UEP codes are presented. Each code in the first class is obtained by combining codes of shorter lengths. We find that a subclass of such codes meets the Hamming bound on the parity-check symbols for systematic linear UEP codes. Each of the second class of codes is achieved by taking direct sums of product codes. The minimum distances of such codes are greater than those for the simple product codes of comparable dimensions, besides, some message bits have extra error protection.

2. Cloud structure and the separation vector of a block code

Let $\{0,1\}^n$ denote the vector space of all n-tuples over the binary field GF(2). Let V and W be two subsets of $\{0,1\}^n$. Let $\boldsymbol{v}$ and $\boldsymbol{w}$ denote two vectors from V and W respectively. We define the separation between V and W, denoted $d(V, W)$, as follows:

$$d(V, W) \triangleq \min\{d(\boldsymbol{v}, \boldsymbol{w}) : \boldsymbol{v} \in V \text{ and } \boldsymbol{w} \in W\}, \tag{1}$$

where $d(\boldsymbol{v}, \boldsymbol{w})$ denotes the Hamming distance between $\boldsymbol{v}$ and $\boldsymbol{w}$. Clearly the separation $d(V, W)$ between V and W is simply a measure of distance between the two sets, V and W. Let $\boldsymbol{r}$ be a vector in $\{0,1\}^n$. Then it is easy to show that the separations between $\{\boldsymbol{r}\}$, V and W satisfy the following triangle inequality,

$$d[\{\boldsymbol{r}\}, V] + d[\{\boldsymbol{r}\}, W] \geq d(V, W). \tag{2}$$

Consider a message space M which is the product of m component message spaces, $M_1, M_2, \ldots, M_m$. For $1 \leq i \leq m$, let $\boldsymbol{x}_i$ denote a message from the message space M_i. Then the product space M consists of the following set of m-tuples,

$$M = \{(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m) : \boldsymbol{x}_i \in M_i \text{ for } 1 \leq i \leq m\}. \tag{3}$$

Let C be a binary block code of length n for the product message space M. Let $\boldsymbol{v}(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m)$ denote the codeword for the message $(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m)$ from M. Let $\boldsymbol{a}$ be a specific message in M_i. Consider the following subset of codewords in C,

$$Q_i(\boldsymbol{a}) = \{\boldsymbol{v}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{i-1}, \boldsymbol{a}, \boldsymbol{x}_{i+1}, \ldots, \boldsymbol{x}_m) : \boldsymbol{x}_j \in M_j \text{ for } 1 \leq j \leq m \text{ and } j \neq i\}. \tag{4}$$

This set $Q_i(\boldsymbol{a})$ is called an *i-cloud* of C corresponding to the message $\boldsymbol{a}$ in M_i.

There are $|M_i|$ i-clouds in C corresponding to $|M_i|$ messages in M_i. These i-clouds form a partition of C. For two distinct i-clouds, $Q_i(\boldsymbol{a})$ and $Q_i(\boldsymbol{b})$, the separation between them is $d(Q_i(\boldsymbol{a}), Q_i(\boldsymbol{b}))$. Then we define the minimum separation among the i-clouds of C as follows:

$$s_i \triangleq \min\{dQ_i(\boldsymbol{a}), Q_i(\boldsymbol{b})) : \boldsymbol{a}, \boldsymbol{b} \in M_i \text{ and } \boldsymbol{a} \neq \boldsymbol{b}\}. \tag{5}$$

It follows from (1), (4) and (5) that

$$s_i = \min\{d[\boldsymbol{v}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_i, \ldots, \boldsymbol{x}_m), \boldsymbol{v}(\boldsymbol{x}_1', \ldots, \boldsymbol{x}_i', \ldots, \boldsymbol{x}_m')] :$$
$$\boldsymbol{x}_l, \boldsymbol{x}_l' \in M_l \text{ for } 1 \leqslant l \leqslant m \text{ and } x_i \neq x_i'\}.$$

Geometrically, we may view that the code C consists of $|M_i|$ i-clouds, where any two i-clouds are separated by a distance at least s_i. This distance structure of i-clouds determines the level of error protection for component message $\boldsymbol{x}_i$. The m-tuple,

$$\boldsymbol{s} \triangleq (s_1, s_2, \ldots, s_m),$$

is called the *separation vector* of the block code C for the product space $M = M_1 \times M_2 \times \cdots \times M_m$. This separation vector determines the levels of error protection for the m component messages, $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m$. We readily see that the minimum Hamming distance of C is $d_{\min} = \min\{s_i : 1 \leqslant i \leqslant m\}$.

Now we are ready to show that the minimum separation s_i of the i-clouds of a block code C determines the level of error protection (or error correction) for the ith component message $\boldsymbol{x}_i$ from M_i. To do this we devise a nearest cloud decoding algorithm for which each component message is decoded independently. Suppose a codeword $\boldsymbol{v}$ is transmitted and a vector $\boldsymbol{r}$ is received. To decode the ith component message, we compute the separation between $\{\boldsymbol{r}\}$ and every i-cloud. Let $Q_i(\boldsymbol{a})$ be the i-cloud such that

$$d[\{\boldsymbol{r}\}, Q_i(\boldsymbol{a})] < d[\{\boldsymbol{r}\}, Q_i(\boldsymbol{x}_i)]$$

for any $\boldsymbol{x}_i \in M_i$ and $\boldsymbol{x}_i \neq \boldsymbol{a}$. Then the ith component message is decoded into $\boldsymbol{a}$. The ith component message contained in $\boldsymbol{r}$ will be decoded correctly provided that there are $\lfloor (s_i - 1)/2 \rfloor$ or fewer transmission errors in $\boldsymbol{r}$. To see this, let $\boldsymbol{v} = \boldsymbol{v}(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m)$ be the transmitted codeword. For $\boldsymbol{x}_i' \neq \boldsymbol{x}_i$, it follows from (2) that

$$d[\{\boldsymbol{r}\}, Q_i(\boldsymbol{x}_i)] + d[\{\boldsymbol{r}\}, Q_i(\boldsymbol{x}_i')] \geqslant d[Q_i(\boldsymbol{x}_i), Q_i(\boldsymbol{x}_i')]. \tag{6}$$

Since $d[Q_i(\boldsymbol{x}_i), Q_i(\boldsymbol{x}_i')] \geqslant s_i$ and $d(\boldsymbol{r}, \boldsymbol{v}) \geqslant d[\{\boldsymbol{r}\}, Q_i(\boldsymbol{x}_i)]$, we have

$$d[\{\boldsymbol{r}\}, Q_i(\boldsymbol{x}_i')] \geqslant s_i - d(\boldsymbol{r}, \boldsymbol{v}). \tag{7}$$

If there are $t_i = \lfloor (s_i - 1)/2 \rfloor$ or fewer transmission errors in $\boldsymbol{r}$, then $d(\boldsymbol{r}, \boldsymbol{v}) \leqslant t_i$. It follows from (6) and (7) that $d[\{\boldsymbol{r}\}, Q_i(\boldsymbol{x})] \leqslant t_i$ and $d[\{\boldsymbol{r}\}, Q_i(\boldsymbol{x}_i')] > t_i$. Hence, $d[\{\boldsymbol{r}\}, Q_i(\boldsymbol{x}_i)] < d[\{\boldsymbol{r}\}, Q_i(\boldsymbol{x}_i')]$ for $\boldsymbol{x}_i \neq \boldsymbol{x}_i'$. Thus, the decoding algorithm described above results in the correct i-cloud, $Q_i(\boldsymbol{x}_i)$, and hence the correct component message $\boldsymbol{x}_i$. However, if there are more than t_i errors in the received vector $\boldsymbol{r}$, the inequality $d[\{\boldsymbol{r}\}, Q_i(\boldsymbol{x}_i)] < d[\{\boldsymbol{r}\}, Q_i(\boldsymbol{x}_i')]$ for $\boldsymbol{x}_i \neq \boldsymbol{x}_i'$ may not hold. As a result, the ith component message is decoded incorrectly into some $\boldsymbol{x}_i' \neq \boldsymbol{x}_i$. Theorem 1 characterizes the multi-level error-correcting capabilities of a block code.

Theorem 1. *Let C be a block code for the product of m message spaces, $M_1, M_2, \ldots, M_m$. Let $\boldsymbol{s} = (s_1, s_2, \ldots, s_m)$ be the separation vector of C. Then, for $1 \leq i \leq m$, the* i*th component message contained in a received vector can be correctly decoded provided that the number of transmission errors in the received vector is $\lfloor (s_i - 1)/2 \rfloor$ or less.*

A code C with a separation vector $\boldsymbol{s} = (s_1, s_2, \ldots, s_m)$ is called a $(t_1, t_2, \ldots, t_m)$-error-correcting code where $t_i = \lfloor (s_i - 1)/2 \rfloor$ for $1 \leq i \leq m$ and is the error correcting capability of the code for the ith component message $\boldsymbol{x}_i$. If $t_1, t_2, \ldots, t_m$ are all distinct, then C provides m levels of error-correcting capabilities, one for each component message. In this case, C is called a m-level error-correcting code or a m-level UEP code. Without loss of generality, we assume that $s_1 \geq s_2 \geq \cdots \geq s_m$ throughout of this paper.

The concept of separation vector was first introduced by Dunning and Robbins [13]. The separation vector defined in this paper is a generalization of Dunning and Robbins', which applies for either linear or nonlinear codes. Note that the minimum separation s_i for the i-clouds depends on how a code is partitioned into the i-clouds. Different encoding (or mapping) of M onto C yields different partitions of C. As a result, the separation vector of C depends on the encoding mapping.

3. Direct-sum codes for unequal error protection

An approach for constructing multi-level UEP codes is to take direct-sums of linear component codes. For $1 \leq i \leq m$, let C_i be a binary (n, k_i) linear block code for the message space $M_i = \{0, 1\}^{k_i}$. For $i \neq j$, we require that $C_i \cap C_j$ contains only the all-zero n-tuple $\boldsymbol{0}$. Let $\boldsymbol{v}(\boldsymbol{x}_i)$ denote the codeword in C_i for the message $\boldsymbol{x}_i \in M_i$. Let C be the direct-sum of $C_1, C_2, \ldots, C_m$, denoted $C = C_1 \oplus C_2 \oplus \cdots \oplus C_m$. Then C is an (n, k) linear code for the product message space $M = M_1 \times M_2 \times \cdots \times M_m$ where $k = k_1 + k_2 + \cdots + k_m$. For any message $(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m)$ in M, the corresponding codeword is

$$\boldsymbol{v}(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m) = \boldsymbol{v}(\boldsymbol{x}_1) + \boldsymbol{v}(\boldsymbol{x}_2) + \cdots + \boldsymbol{v}(\boldsymbol{x}_m). \tag{8}$$

Let $\{j_1, j_2, \ldots, j_l\}$ be a subset of $\{1, 2, \ldots, m\}$. Consider the direct-sum,

$$C(j_1, j_2, \ldots, j_l) = C_{j_1} \oplus C_{j_2} \oplus \cdots \oplus C_{j_l}.$$

Then $C(j_1, j_2, \ldots, j_l)$ is a subcode of C. An i-cloud of C for the component message $\boldsymbol{x}_i$ from M_i is simply the following set:

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

The vector $\boldsymbol{v}(\boldsymbol{x}_i)$ is in the i-cloud $Q_i(\boldsymbol{x}_i)$ and is called the center of $Q_i(\boldsymbol{x}_i)$. A vector in $Q_i(\boldsymbol{x}_i)$ is of the form $\boldsymbol{v}(\boldsymbol{x}_i) + \boldsymbol{w}$, where $\boldsymbol{w} \in C(1, \ldots, i-1, i+1, \ldots, m)$.

Let $w(\boldsymbol{v})$ denote the Hamming weight of the vector $\boldsymbol{v}$. Since $d(\boldsymbol{v}, \boldsymbol{u}) = w(\boldsymbol{v} + \boldsymbol{u})$, the minimum separation of the i-clouds of C is

$$s_i = \min\{w[\boldsymbol{v}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_i, \ldots, \boldsymbol{x}_m)] : \boldsymbol{x}_i \neq 0\}. \tag{9}$$

Theorem 2. *Consider an (n, k) linear code C which is the direct sum of codes $C_1, C_2, \ldots, C_m$, where C_i is an (n, k_i) linear code for the component message space $M_i = \{0, 1\}^{k_i}$ for $1 \leq i \leq m$. If the minimum weight of codewords in $C - C(i+1, i+2, \ldots, m)$ is at least d_i and $d_1 \geq d_2 \geq \cdots \geq d_m$, then C is an m-level error-correcting code for the product message space $M = M_1 \times M_2 \times \cdots M_m$ with separation vector $\boldsymbol{s} = (s_1, s_2, \ldots, s_m)$, where $s_i \geq d_i$ for $i = 1, 2, \ldots, m$.*

Proof. Note that for each codeword in $C(i+1, i+2, \ldots, m)$, the corresponding component message $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_i$ are all zero. Each codeword of C, $\boldsymbol{v}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_i, \ldots, \boldsymbol{x}_m)$ with $\boldsymbol{x}_i \neq 0$, is not in $C(i+1, i+2, \ldots, m)$ and hence has weight at least d_i. The proof then follows from (9). □

Theorem 2 describes a method of constructing multi-level UEP codes by taking direct sums of linear component codes. With this method, we are able to construct two classes of UEP codes.

4. Construction of linear multi-level UEP codes by combining shorter codes

Let $\boldsymbol{H}_{aa}$ and $\boldsymbol{H}_a = [\boldsymbol{H}_{aa}^{\mathrm{T}} \boldsymbol{H}_{ab}^{\mathrm{T}}]^{\mathrm{T}}$ be the parity-check matrices of an $(n_a, k_a + r)$ linear code C_{aa} and an (n_a, k_a) linear code C_a respectively, where $\boldsymbol{H}_{aa}$ is an $(n_a - k_a - r) \times n_a$ matrix, $\boldsymbol{H}_{ab}$ is a $r \times n_a$ matrix, $\boldsymbol{H}_a$ is an $(n_a - k_a) \times n_a$ matrix and T denotes the transpose operation. Let $\boldsymbol{H}_{bb}$ and $\boldsymbol{H}_b = [\boldsymbol{H}_{bb}^{\mathrm{T}} \boldsymbol{H}_{bb}^{\mathrm{T}}]^{\mathrm{T}}$ be the parity-check matrices of an $(n_b, k_b + r)$ linear code C_{bb} and an (n_b, k_b) linear code C_b respectively, where $\boldsymbol{H}_{bb}$ is an $(n_b - k_b - r) \times n_b$ matrix, $\boldsymbol{H}_{ba}$ is a $r \times n_b$ matrix and $\boldsymbol{H}_b$ is an $(n_b - k_b) \times n_b$ matrix. Consider the $(n_a + n_b, k_a + k_b + r)$ linear code C with the following parity-check matrix,

$$\boldsymbol{H} = \begin{bmatrix} \boldsymbol{H}_{aa} & \boldsymbol{0} \\ \boldsymbol{H}_{ab} & \boldsymbol{H}_{ba} \\ \boldsymbol{0} & \boldsymbol{H}_{bb} \end{bmatrix}, \tag{10}$$

where $\mathbf{0}$ represents a zero matrix of proper dimension. Let C_2 be the $(n_a + n_b, k_a)$ subcode of C such that each codeword in C_2 is the concatenation of a codeword in C_a and the all-zero n_b-tuple. Let C_3 be the $(n_a + n_b, k_b)$ subcode of C such that every codeword in C_3 is the concatenation of the all-zero n_a-tuple and a codeword in C_b. Since $C_2 \oplus C_3 = C(2, 3)$ is an $(n_a + n_b, k_a + k_b)$ subcode of C, there must exist r linear independent codewords in $C - C(2, 3)$. These r linear 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. $C = C_1 \oplus C_2 \oplus C_3$.

Let d_{aa}, d_{bb}, d_a and d_b be the minimum distances of C_{aa}, C_{bb}, C_a and C_b respectively. Suppose $d_{aa} + d_{bb} \geqslant d_a \geqslant d_b$. Now we examine the distance structure of C. Any codeword $\boldsymbol{v}$ in C can be expressed as $\boldsymbol{v} = (\boldsymbol{v}_a, \boldsymbol{v}_b)$ where $\boldsymbol{v}_a$ is an n_a-tuple and $\boldsymbol{v}_b$ is an n_b-tuple. Then $(\boldsymbol{v}_a, \boldsymbol{v}_b) \cdot \boldsymbol{H}^{\mathrm{T}} = 0$. This implies that $\boldsymbol{v}_a \cdot \boldsymbol{H}_{aa}^{\mathrm{T}} = 0$ and $\boldsymbol{v}_b \cdot \boldsymbol{H}_{bb}^{\mathrm{T}} = 0$. Thus, $\boldsymbol{v}_a$ is a codeword in C_{aa} and $\boldsymbol{v}_b$ is a codeword in C_{bb}. Consider a codeword $(\boldsymbol{v}_a, \boldsymbol{v}_b)$ in $C - C(2, 3)$. Then, $\boldsymbol{v}_a \neq 0$ and $\boldsymbol{v}_b \neq 0$. Hence, the weight of any codeword $\boldsymbol{v}$ in $C - C(2, 3)$ is at least $d_{aa} + d_{bb}$. For any codeword $(\boldsymbol{v}_a, \boldsymbol{v}_b)$ in $C - C_3$, either it is in C_2, or both $\boldsymbol{v}_a$ and $\boldsymbol{v}_b$ 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 $d_{aa} + d_{bb} \geqslant d_a$, the minimum weight of codewords in $C - C_3$ is d_a. Since $d_{aa} + d_{bb} \geqslant d_a \geqslant d_b$, we can easily see that the minimum weight of C is d_b. It follows from Theorem 2 that, for $d_{aa} + d_{bb} \geqslant d_a \geqslant d_b$, the code C with the parity-check matrix $\boldsymbol{H}$ of (10) is a linear block code for the product message space $M = \{0, 1\}^r \times \{0, 1\}^{k_a} \times \{0, 1\}^{k_b}$ with separation vector $\boldsymbol{s} = (s_1, s_2, s_3)$ where $s_1 \geqslant d_{aa} + d_{bb}$, $s_2 \geqslant d_a$ and $s_3 = d_b$.

A generator matrix for the code C with a parity-check matrix of the form given by (10) can be formed easily. Let $\boldsymbol{G}_a$ and $\boldsymbol{G}_b$ be the generator matrices for the (n_a, k_a) code C_a and (n_b, k_b) code C_b respectively. Let $[\boldsymbol{G}_a^{\mathrm{T}} \boldsymbol{G}_{aa}^{\mathrm{T}}]^{\mathrm{T}}$ and $[\boldsymbol{G}_b^{\mathrm{T}} \boldsymbol{G}_{bb}^{\mathrm{T}}]^{\mathrm{T}}$ be generator matrices for the $(n_a, k_a + r)$ code C_{aa} and the $(n_b, k_b + r)$ code C_{bb} respectively. Then the following $(k_a + k_b + r) \times (n_a + n_b)$ matrix,

$$\boldsymbol{G} = \begin{bmatrix} \boldsymbol{G}_{aa} & \boldsymbol{G}_{bb} \\ \boldsymbol{G}_a & \mathbf{0} \\ \mathbf{0} & \boldsymbol{G}_b \end{bmatrix}, \tag{11}$$

is a generator matrix for C where $[\boldsymbol{G}_{aa} \boldsymbol{G}_{bb}]$, $[\boldsymbol{G}_a \mathbf{0}]$ and $[\mathbf{0} \boldsymbol{G}_b]$ generate the $(n_a + n_b, r)$ code C_1, the $(n_a + n_b, k_a)$ code C_2 and the $(n_a + n_b, k_b)$ code C_3 respectively.

In the following, we present two special classes of linear *UEP* codes with parity-check matrices of the form given by (10). Let α be a primitive element in $\mathrm{GF}(2^m)$. Every nonzero element in $\mathrm{GF}(2^m)$ 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 $\beta_1, \beta_2, \ldots, \beta_{2^{m+l}-2^m}$ represent all the $(m + l)$-tuples over GF(2) (in column form) for which the last l components are not all zero.

Consider the binary code C associated with the following parity-check matrix:

$$\boldsymbol{H}=\left[\begin{array}{ccccc:cccc} 1 & \alpha & \alpha^2 & \cdots & \alpha^{2^m-2} & 0_m & 0_m & \cdots & 0_m \\ 1 & \alpha^3 & \alpha^6 & \cdots & \alpha^{3(2^m-2)} & \cdots & \cdots & \cdots & \cdots \\ 0_l & 0_l & 0_l & \cdots & 0_l & \beta_1 & \beta_2 & \cdots & \beta_{2^{m+1}-2^m} \end{array}\right], \tag{12}$$

where each power of α is represented by an m-tuple in column form, 0_l is a column of l zeros and 0_m is a column of m zeros. The matrix $\boldsymbol{H}$ consists of $2m+l$ rows and $2^{m+l}-1$ columns, and hence the code C associated with $\boldsymbol{H}$ is a $(2^{m+l}-1,\ 2^{m+l}-2m-l-1)$ linear code over GF(2). Note that the $\boldsymbol{H}$ matrix has the form given by (10) where

$$\boldsymbol{H}_a=\begin{bmatrix}\boldsymbol{H}_{aa}\\ \boldsymbol{H}_{ab}\end{bmatrix}=\begin{bmatrix}1 & \alpha & \alpha^3 & \cdots & \alpha^{2^m-2}\\ 1 & \alpha^3 & \alpha^6 & \cdots & \alpha^{3(2^m-2)}\end{bmatrix}$$

$$\boldsymbol{H}_b=\begin{bmatrix}\boldsymbol{H}_{ba}\\ \boldsymbol{H}_{bb}\end{bmatrix}=[\beta_1\ \ \beta_2\ \ \cdots\ \ \beta_{2^{m+l}-2^m}]$$

$$\boldsymbol{H}_{aa}=[1\ \ \alpha\ \ \alpha^3\ \ \cdots\ \ \alpha^{2^m-2}]$$

$\boldsymbol{H}_{bb}=$ some $l\times(2^{m+l}-2^m)$ matrix which has no zero column.

The codes, C_{aa} and C_a, generated by parity-check matrices $\boldsymbol{H}_{aa}$ and $\boldsymbol{H}_a$ are simply the Hamming and double-error-correcting BCH codes of length 2^m-1 respectively. Hence, $d_{aa}=3$ and $d_a=5$. The code C_b generated by the parity-check matrix $\boldsymbol{H}_b$ is a shortened Hamming code with minimum distance $d_b=3$, and the code C_{bb} generated by the parity-check matrix $\boldsymbol{H}_{bb}$ has minimum distance $d_{bb}=2$. As a result, C is a code for the product message space $M=M_1\times M_2\times M_3$ where $M_1=\{0,1\}^m$, $M_2=\{0,1\}^{2^m-2m-1}$ and $M_3=\{0,1\}^{2^{m+l}-2^m-m-l}$. The separation vector of C is $\boldsymbol{s}=(s_1,s_2,s_3)$ where $s_1\geqslant d_{aa}+d_{bb}=5$, $s_2\geqslant d_a=5$, 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 $2^{m+l}-2^m-m-l$ 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 of length 2^l-1. For $l=0$, C reduces to a primitive double-error-correcting BCH code of length 2^m-1. For $m=l$, C is equivalent to Boyarinov–Katsman UEP code [16]. The code C can be transformed into systematic form with identical two-level error-correcting capability.

A lower bound (equivalent to the Hamming bound for single-level error-correcting codes) on the number of parity-check bits for systematic linear UEP codes has been derived by Masnick and Wolf [9], and van Gils [20]. It follows from Theorem 2 of [9] that, for a two-level (t_1, t_2)-error-correcting code of length n, the number of parity-check bits satisfies the following inequality:

$$2^{n-k}\geqslant 1+\sum_{i=1}^{t_2}\binom{n}{i}+\sum_{j=t_2+1}^{t_1}\sum_{i=0}^{t_2}\binom{n-k_1}{i}\binom{k_1}{j-i}. \tag{13}$$

Consider a two-level UEP code with the following parameters: $n = 2^{m+l} - 1$, $k_1 = 2^m - m - 1$, $t_1 = 2$, $t_2 = 1$. It follows from the Hamming bound given by (13) that

$$\begin{aligned} 2^{n-k} &\geqslant 2^{-1}\cdot\{2^{2m+l+1} - (2m)\cdot 2^{m+l} - (2^m - 2m + 1)\cdot 2^m - (m^2 - m) + 2\} \\ &= 2^{-1}\cdot\{2^{2m+l} + \Delta\}, \end{aligned} \tag{14}$$

where $\Delta = 2^{m+l}(2^{m-1} - 2m) + 2^{2m}(2^{l-1} - 1) + (2m - 2)\cdot 2^m + (2^m - m^2 + m + 2)$. For either $m = 3$ and $l = 3$, or $m \geqslant 4$ and $l \geqslant 1$, the number Δ is greater than zero. Hence, it follows from (14) that $n - k > 2m + l - 1$. This is to say that the number of parity-check symbols required for a two-level linear systematic UEP code with parameters, $n = 2^{m+l} - 1$, $k_1 = 2^m - m - 1$, $t_1 = 2$ and $t_2 = 1$ is at least $2m + l$. The two-level UEP code given by the parity-check matrix $\boldsymbol{H}$ of (12) has exactly $2m + l$ parity-check symbols. Hence, under the condition that $m = 3$, $l = 3$, or $m \geqslant 4$ and $l \geqslant 1$, the code meets the Hamming bound of (13) and hence is optimal. A list of codes with lengths 63 and 127 is given in *Table 1* for various m and l, where $k_1 = 2^m - m - 1$ and $k_2 = 2^{m+l} - 2^m - m - l$ and $k = k_1 + k_2$. For example, there is a (63, 52) code which provides protection for the first 26 message bits against up to 2 random errors and protection for the next 26 message bits against any single error.

The second class of linear UEP codes with parity-check matrices of the form given by (10) is specified by the following submatrices:

$$\begin{aligned} \boldsymbol{H}_{aa} &= \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 0_m & 1 & \alpha & \cdots & \alpha^{2^m-2} \\ 0_m & 1 & \alpha^3 & \cdots & (\alpha^3)^{2^m-2} \\ \vdots & \vdots & \vdots & & \vdots \\ 0_m & 1 & \alpha^{2t-3} & \cdots & (\alpha^{2t-3})^{2^m-2} \end{bmatrix} \\ \boldsymbol{H}_{ab} &= [0_m \quad 1 \quad \alpha^{2t-1} \quad \cdots \quad (\alpha^{2t-1})^{2^m-2}] \\ \boldsymbol{H}_{ba} &= [1 \quad \alpha^{2s-1} \quad \cdots \quad (\alpha^{2s-1})^{2^m-2}] \\ \boldsymbol{H}_{bb} &= \begin{bmatrix} 1 & \alpha^{2s-3} & \cdots & (\alpha^{2s-3})^{2^m-2} \\ \vdots & \vdots & & \vdots \\ 1 & \alpha^3 & \cdots & (\alpha^3)^{2^m-2} \\ 1 & \alpha & \cdots & \alpha^{2^m-2} \end{bmatrix} \end{aligned} \tag{15}$$

where $s \leqslant t$. Note that $\boldsymbol{H}_{aa}$ and $\boldsymbol{H}_a = [\boldsymbol{H}_{aa}^{\mathrm{T}}\boldsymbol{H}_{ab}^{\mathrm{T}}]^{\mathrm{T}}$ are parity-check matrices of an extended $(t-1)$-error-correcting and an extended t-error-correcting primitive BCH codes of length 2^m respectively. Also note that $\boldsymbol{H}_{bb}$ and $\boldsymbol{H}_b = [\boldsymbol{H}_{bb}^{\mathrm{T}}\boldsymbol{H}_{ba}^{\mathrm{T}}]^{\mathrm{T}}$ are parity-check matrices of an $(s-1)$-error-correcting and an s-error-correcting primitive BCH codes of length $2^m - 1$ respectively. We require that $\boldsymbol{H}_{ab}$ and $\boldsymbol{H}_{ba}$ have the same dimension, i.e. α^{2t-1} and α^{2s-1} from the same subfield of GF(2^m). The submatrices of (15) arranged in the form of (10) generate a linear UEP code with a separation vector $\boldsymbol{s} = (s_1, s_2, s_3)$ where $s_1 \geqslant 2(t+s) - 1$, $s_2 \geqslant 2t + 2$ and

Table 1. A list of (2, 1)-error-correcting codes.

Codes of length 63					Codes of length 127				
m	l	k	k_1	k_2	m	l	k	k_1	k_2
0	6	57	0	57	0	7	120	0	120
2	4	55	1	54	2	4	118	1	117
3	3	54	4	50	3	4	117	4	113
4	2	53	11	42	4	3	116	11	105
5	1	52	26	26	5	2	115	26	89
6	0	51	51	0	6	1	114	57	57
					7	0	113	113	0

$s_3 = 2s + 1$. 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 1. 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:

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

It is a (63, 47) UEP code for the message space $M = \{0, 1\}^5 \times \{0, 1\}^{21} \times \{0, 1\}^{21}$ with separation vector at least (7, 6, 5). Note that there is a (63, 45) triple-error-correcting BCH code and a (63, 51) double-error-correcting BCH code.

5. Direct sums of product 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 $V \otimes W$ denote the product of V and W [21]. Then, $V \otimes W$ is an (Nn, Nk) linear code with minimum distance Dd. A codeword in $V \otimes W$ can be arranged as an $n \times N$ array in which every row is a codeword in V and every column is a codeword in W. For a nonzero code array in $V \otimes W$, 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 $V \otimes W$ is at least Dd. Product codes are capable of correcting both random and burst errors [21]. Now, we consider direct sums of certain binary 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) binary 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 following set,

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

Clearly $V_1 + V_2$ is linear and a supercode of both V_1 and V_2. If $V_1 \cap V_2 = \{0\}$, then $V_1 + V_2$ is the direct sum of V_1 and V_2. Let D be the minimum distance of $V_1 + V_2$.

Let W_1 and W_2 be an (n, k_1) and an (n, k_2) binary linear codes with minimum distances d_1 and d_2 respectively. We assume that $W_1 \cap W_2 = \{0\}$. Then, the direct sum of W_1 and W_2, denoted $W = W_1 \oplus W_2$, is an $(n, k_1 + k_2)$ linear code. Let d be the minimum distance of W.

For $i = 1, 2$, the product $V_i \otimes W_i$ is an $(Nn, K_i k_i)$ linear code with minimum distance $D_i d_i$. Since $W_1 \otimes W_2 = \{0\}$, $V_1 \otimes W_1$ 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 $\boldsymbol{c}$ in C is sum of a code array $\boldsymbol{c}_1$ in $V_1 \otimes W_1$ and a code array $\boldsymbol{c}_2$ in $V_2 \otimes W_2$, i.e. $\boldsymbol{c} = \boldsymbol{c}_1 + \boldsymbol{c}_2$. Each row in array $\boldsymbol{c}$ is a codeword in $V_1 + V_2$, and each column in $\boldsymbol{c}$ is a codeword in $W_1 \otimes W_2$.

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

Case I. Suppose all the nonzero rows in $\boldsymbol{c}_1$ and $\boldsymbol{c}_2$ are alike and identical to a certain vector $\boldsymbol{v}$. Then $\boldsymbol{v}$ must be a codeword in $V_1 \cap V_2$. Thus, $w(\boldsymbol{v}) \geq \hat{D}$. This implies that there are at least $\hat{D}$ nonzero columns in array $\boldsymbol{c}$. Since each of these columns has weight at least d. As a result, $w(\boldsymbol{c}) \geq \hat{D}d$.

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

$$\begin{aligned}&\min_{f \geqslant D}\{f \cdot \min\{d_1, d_2\} + \lceil (D_1 + D_2 - f)/2 \rceil \cdot d\}\\&\quad = D \cdot \min\{d_1, d_2\} + \lceil (D_1 + D_2 - f)/2 \rceil \cdot d.\end{aligned}$$

Case III. Suppose that there are two nonzero rows $\boldsymbol{v}_1$ and $\boldsymbol{v}_1'$ in $\boldsymbol{c}_1$ such that $\boldsymbol{v}_1 \neq \boldsymbol{v}_1'$. Then there are at least $D_1 + \lceil D_1/2 \rceil$ nonzero columns in $\boldsymbol{c}_1$. This implies that there are at least $D_1 + \lceil D_1/2 \rceil$ nonzero columns in $\boldsymbol{c}$. 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 $\boldsymbol{c}$ is at least $\{D_1 + \lceil D_1/2 \rceil\} \cdot d$.

Case IV. Suppose that there are two nonzero rows, $\boldsymbol{v}_2$ and $\boldsymbol{v}_2'$ in $\boldsymbol{c}_2$ such that $\boldsymbol{v}_2 \neq \boldsymbol{v}_2'$. It follows the same argument as that in Case III that $w(\boldsymbol{c}) \geqslant \{D_2 + \lceil D_2/2 \rceil\} \cdot d$.

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 $\boldsymbol{c}$ in $C = V_1 \otimes W_1 \oplus V_2 \otimes W_2$:

(1) For $\boldsymbol{c} \in V_1 \otimes W_2$, $w(\boldsymbol{c}) \geqslant D_1 d_1$.

(2) For $\boldsymbol{c} \in V_2 \otimes W_2$, $w(\boldsymbol{c}) \geqslant D_2 d_2$.

(3) For $\boldsymbol{c} \notin V_1 \otimes W_1$ and $\boldsymbol{c} \notin V_1 \otimes W_2$, $w(\boldsymbol{c}) \geqslant \min\{\hat{D}d, \lambda, \lambda_1, \lambda_2\}$.

From the above weight distribution, we see that the weight of a nonzero code array $\boldsymbol{c}$ 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 $\min\{D_1 d_1, \hat{D}d, \lambda, \lambda_1, \lambda_2\} \geqslant D_2 d_2$. Then we have the following weight structure of a nonzero code array $\boldsymbol{c}$ in $V_1 \otimes W_1 \oplus V_2 \otimes W_2$:

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

(2) For $\boldsymbol{c} \in V_1 \otimes W_1 \oplus V_2 \otimes W_2 - V_2 \otimes W_2$, $w(\boldsymbol{c}) \geqslant \min\{D_1 d_1, \tilde{D}d, \lambda, \lambda_1, \lambda_2\}$.

It follows from Theorem 2 that $C = V_1 \otimes W_1 \oplus V_2 \otimes W_2$ is a linear block code with a separation vector $\boldsymbol{s} = (s_1, s_2)$, where

$$\begin{aligned}&s_1 \geqslant \min\{D_1 d_1, \hat{D}d, \lambda, \lambda_1, \lambda_2\},\\&s_2 \geqslant D_2 d_2.\end{aligned} \tag{16}$$

The message space for C is $M = \{0, 1\}^{K_1 k_1} \times \{0, 1\}^{K_2 k_2}$.

Example 2. 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, $W_1 \oplus W_2$ 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 $W_1 \oplus W_2$ 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, $\lambda = 13$, $\lambda_1 = 15$, $\lambda_2 = 15$, $\hat{D}d = 21$, $D_1 d_1 = 21$ and $D_2 d_2 = 12$. Note that $N = 7$, $K_1 = K_2 = 4$, $n = 7$, $k_1 = 1$, $k_2 = 3$. Since $\min\{D_1 d_1, \hat{D}d, \lambda, \lambda_1, \lambda_2\} = 13 \geqslant D_2 d_2 = 12$, we see that $V_1 \otimes W_1 \oplus V_2 \otimes W_2$ is a two-level UEP (49, 16) binary code for the message space $M = M_1 \times M_2$ with separation vector $s = (s_1, s_2)$, where $M_1 = \{0,1\}^4$, $M_2 = \{0, 1\}^{12}$, $s_1 \geqslant 13$, $s_2 \geqslant 12$. We may compare this code with the product code of two (7, 4) BCH codes with minimum distance 3, which is a (49, 16) single-level binary code with minimum distance only 9.

Direct sums of product codes for unequal error protection was first studied by Boyarinov and Katsman [16]. The lower bound on the separation vector given by (16) is better than the bound derived by Boyarinov and Katsman.

Now we present a special class of direct sums of product codes. Let α and β be two different primitive Nth roots of unity. Let V_1 be an (N, K_1) binary cyclic code which has $\alpha, \alpha^2, \ldots, \alpha^{2t}$ 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 $D_1 = D_2 \geqslant 2t + 1$, where D_1 is the minimum distance of V_1 and D_2 is the minimum distance of V_2. If the set $\{(\beta^i)^{2^m} : i = 1, 2, \ldots, 2t,\ m \text{ is an integer}\}$ 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 $V_1 \cap V_2 = \{0\}$ which is the case that $V_1 \cap V_2$ contains all the α^i's as zeros. If the set $\{(\alpha^i)^{2^m} : i = 1, 2, \ldots, 2t \text{ and } m \text{ is an integer}\}$ 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 codes V_1 and V_2, if $\min\{(2t + 1)d_1, (2t + 2s + 1)d, \lambda, \lambda_1, \lambda_2\} \geqslant (2t + 1)d_2$, 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 $s = (s_1, s_2)$ where

$$\begin{aligned} &s_1 \geqslant \min\{(2t + 1)d_1, (2t + 2s + 1)d, \lambda, \lambda_1, \lambda_2\}, \\ &s_2 \geqslant (2t + 1)d_2, \\ &\lambda = (2u + 1) \cdot \min\{d_1, d_2\} + (2t - u + 1)d, \\ &\lambda_1 = \lambda_2 = (3t + 2)d. \end{aligned}$$

Example 3. 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 α, α^3, α^5, and their conjugates as zeros. Since $V_1 \cap V_2 \neq \{0\}$, 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* code 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, $W_1 \oplus W_2$ is a (7, 4) BCH code over GF(2) with minimum distance $d = 3$. Thus, $t = 2$, $s = 1$, $u = 1$, $\lambda = 24$, $\lambda_1 = \lambda_2 = 24$, $\hat{D}d \geqslant 21$, $D_1d_1 = 35$, and $D_2d_2 = 20$. Note that $N = 31$, $n = 7$, $k_1 = 1$, $k_2 = 3$, $K_1 = K_2 = 21$. Since $\min\{D_1d_1, \hat{D}d, \lambda, \lambda_1, \lambda_2\} \geqslant 21 \geqslant D_2d_2 = 20$, $V_1 \otimes W_1 \oplus V_2 \otimes W_2$ is a (217, 84) binary two-level UEP for the

Table 2. Some direct sums of product codes for unequal error protection

N	K_i	D_i	D	$\hat{D}$	n	k_1	k_2	d_1	d_2	d	Nn	K_1k_1	K_2k_2	s_1	s_2	t_B
15	10	4	2	6	17	1	8	17	6	5	255	10	80	27	24	30
15	10	4	2	6	7	1	3	7	4	3	105	10	30	17	16	15
15	10	4	2	6	15	1	10	15	4	3	225	10	100	17	16	15
15	10	4	2	6	15	1	6	15	6	5	225	10	60	27	24	30
15	10	4	2	6	15	2	5	10	7	5	225	20	50	29	28	30
31	21	5	3	7	15	1	10	15	4	3	465	21	210	21	20	31
31	21	5	3	7	15	1	6	15	6	5	465	21	126	35	30	62
31	21	5	3	7	15	2	5	10	7	5	465	42	105	35	35	62

message space $M = M_1 \times M_2$ with separation vector $s = (s_1, s_2)$ where $M_1 = \{0, 1\}^{21}$, $M_2 = \{0, 1\}^{63}$, $s_1 \geqslant 21$, and $s_2 \geqslant 20$. 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 is a (217, 84) linear code with minimum distance 15 which is inferior to the (217, 84) direct-sum code.

Suppose we transmit each code array in $V_1 \otimes W_1 \oplus V_2 \otimes W_2$ row by row. By a proof similar to that for simple product codes [21], it can be shown that the code $V_1 \otimes W_1 \oplus V_2 \otimes W_2$ can correct any error-burst of length up to $N \cdot \lfloor (d-1)/2 \rfloor$ in addition to the random-error-correcting capabilities represented by its separation vector. Consider the (217, 84) binary code illustrated in Example 3. 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 of length up to 31.

Some direct-sums of product codes are listed in *Table 2* where t_B denotes the maximum length of correctable burst errors. Let α and β be primitive elements in GF(2^4) and GF(2^5) respectively. For $N = 15$, V_1 and V_2 are binary BCH codes with $\{1, \alpha, \alpha^2, \alpha^4, \alpha^8\}$ and $\{1, \alpha^7, \alpha^{14}, \alpha^{13}, \alpha^{11}\}$ as zeros respectively. For $N = 31$, V_1 and V_2 are binary BCH codes with $\{\beta, \beta^2, \beta^4, \beta^8, \beta^{16}, \beta^3, \beta^6, \beta^{12}, \beta^{24}, \beta^{17}\}$ and $\{\beta^3, \beta^6, \beta^{12}, \beta^{24}, \beta^{17}, \beta^9, \beta^{18}, \beta^5, \beta^{10}, \beta^{20}\}$ as zeros respectively. All W_1 and W_2 are binary BCH codes.

References

[1] T. Cover, Broadcast channels, IEEE Trans. Inform. Theory IT-18 (Jan. 1972) 2–14.

[2] P.P. Bergmans, Random coding theorem for broadcast channels with degraded components, IEEE Trans. Inform. Theory IT-19 (Mar. 1973) 197–207.

[3] A.D. Wyner, A theorem on the entropy of certain binary sequences and application: Part II, IEEE Trans. Inform. Theory IT-19 (Nov. 1973) 772–777.

[4] R.G. Gallagher, Capacity and coding for degraded channels, Problemy Peredachi Informatsii 10 (3) (1974) 3–14.

[5] L.A. Bassalygo et al., Bounds for codes with unequal protection of two sets of message, Problemy Peredachi Informatsii, 15 (3), (July–September 1979) 40–49.

[6] G.L. Katsman, Bounds on volume of linear codes with unequal information symbol protection, Problemy Peredachi Informatsii 16 (2) (April–June, 1980) 25–32.

[7] T. Kasami, S. Lin, V.K. Wei and S. Yamamura, Coding for the binary symmetric broadcast channel with two receivers, IEEE Trans. Inform. Theory IT-31 (5) (September 1985) 616–625.

[8] C. Heagard, H. Depedro and J. Wolf, Permutation codes for the Gaussian broadcast channel with two receivers, IEEE Trans. Inform. Theory IT-24 (5) (September, 1978) 569–578.

[9] B. Masnick and J. Wolf, On linear unequal error protection codes, IEEE Trans. Inform. Theory. IT-13 (4) (July, 1967) 600–607.

[10] W.C. Gore and C.C. Kilgus, Cyclic codes with unequal error protection, IEEE Trans. Inform. Theory IT-17 (2) (1971) 214–215.

[11] D. Mandelbaum, Unequal-error-protection codes derived from difference sets, IEEE Trans. Inform. Theory IT-18 (5) (September, 1972) 686–687.

[12] C.C. Kilgus and W.C. Gore, A class of cyclic unequal-error-protection codes, IEEE Trans. Inform. Theory IT-18 (5) (September, 1972) 687–690.

[13] L.A. Dunning and W.E. Robbins, Optimal encoding of linear block codes for unequal error protection, Information and Control 37 (1978) 150–177.

[14] V.N. Dynkin and V.A. Togonidze, Cyclic codes with unequal symbol protection, Problemy Peredachi Informatsii 12 (1) (January–March, 1976) 24–28.

[15] V.A. Zinovev and V.V. Zyablov, Codes with unequal protection of information symbols, Problemy Peredachi Informatsii 15 (3) (July–September 1979) 50–60.

[16] I.M. Boyarinov and G.L. Katsman, Linear unequal error protection codes, IEEE Trans. Inform. Theory IT-27 (2) (March 1981) 168–175.

[17] I.M. Boyarinov, Combined decoding methods for linear codes with unequal protection of information symbols, Problemy Peredachi Informatsii 19 (1) (January–March, 1983) 17–25.

[18] W.J. van Gils, Two topics on linear unequal error protection codes: bounds on their length and cyclic code classes, IEEE Trans. Inform. Theory IT-29 (6) (November, 1983).

[19] W.J. van Gils, Linear unequal error protection codes from shorter codes, IEEE Trans. Inform. Theory IT-30 (3) (May, 1984) 544–546.

[20] W.J. van Gils, On linear unequal error protection codes, Report, Research Laboratories, N.V. Philips' Gloeilampenfabriken, Eindhoven, Netherlands.

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