CYCLIC UNEQUAL ERROR PROTECTION CODES
CONSTRUCTED FROM CYCLIC CODES OF COMPOSITE LENGTH

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
**CYCLIC UNEQUAL ERROR PROTECTION CODES**

**CONSTRUCTED FROM CYCLIC CODES OF COMPOSITE LENGTH**

Mao-Chao Lin and Shu Lin

University of Hawaii

Department of Electrical Engineering

Honolulu, Hawaii 96822

**ABSTRACT**

In this paper, we first investigate the distance structure of cyclic codes of composite length. A lower bound on the minimum distance for this class of codes is derived. In many cases, the lower bound gives the true minimum distance of a code. Then, we investigate the distance structure of the direct sum of two cyclic codes of composite length. We show that, under certain conditions, the direct-sum code provides two levels of error correcting capability, and hence is a two-level unequal error protection (UEP) code. Finally, a class of two-level UEP cyclic direct-sum codes and a decoding algorithm for a subclass of these codes are presented.
I. INTRODUCTION

Unequal error protection (UEP) codes[1-11] are desirable in certain data communication situations. For example, consider a data communication system in which each message from the information source consists of several parts, and different parts have different degrees of significance. More significant parts require more protection against the channel errors, while the less significant parts require less protection against the channel errors. As a result, it is desired to use a code with unequal error protection capabilities. Another situation where UEP codes are desired is in broadcast communication systems[13-15]. 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.

UEP codes were first studied by Masnick and Wolf[1], then by many other coding theorists[2-15]. In this paper, we investigate cyclic UEP codes which are formed by taking the direct sums of cyclic codes of composite length. We first investigate the weight structure of cyclic codes of composite length. Then, we analyze the distance structure of the direct sum of two cyclic codes of composite length. We show that, under certain distance conditions, the direct-sum code provides two levels of error-correcting capability, and hence is a two-level UEP code. Finally, a class of two-level UEP cyclic direct-sum codes is presented. Also, a decoding algorithm for a subclass of two-
level UEP cyclic direct-sum codes is devised.

II. WEIGHT STRUCTURE OF BINARY CYCLIC CODES OF COMPOSITE LENGTH

Let \( n_1 \) and \( n_2 \) be two positive odd integers which are relatively prime. Let

\[ n = n_1 n_2. \]

Let \( \alpha \) be an element from some Galois field, say \( GF(2^q) \), with order \( n \). Hence \( \alpha \) is a primitive \( n \)-th root of unity. Now we consider a binary \((n,k)\) cyclic code \( C \) with generator and parity polynomials, \( g(X) \) and \( h(X) \), respectively. It is known in coding theory that the degree of \( g(X) \) is \( n-k \), the degree of \( h(X) \) is \( k \), and

\[ X^{n+1} = g(X)h(X). \]

Let

\[ Z_g = \{ \alpha^i : i=1,2,\ldots,n-k \} \]

and

\[ Z_h = \{ \alpha^j : j=1,2,\ldots,k \} \]

be the root sets of \( g(X) \) and \( h(X) \) respectively. These two sets are disjoint and their union gives all the roots of \( X^{n+1} \) in \( GF(2^q) \), i.e.,

\[ \{1, \alpha, \alpha^2, \ldots, \alpha^{n-1}\}. \]

Since every code polynomial \( c(X) \) in \( C \) has the elements in \( Z_g \) as roots, we call the elements in \( Z_g \) the zeros of \( C \). No element in \( Z_h \) can be a root of every code polynomial in \( C \). We call the elements in \( Z_h \) the nonzeros of \( C \).

A code polynomial \( c(X) \) in \( C \) is a polynomial of degree \( n-1 \) or less,
\[ c(X) = a_0 + a_1 X + a_2 X^2 + \ldots + a_{n-1} X^{n-1} \]  
with \( a_i \in \text{GF}(2) \). It is possible to arrange the coefficients of \( c(X) \) as an \( n_1 \times n_2 \) code array as shown in Figure 1.

\[
\begin{array}{cccccc}
  a_0 & a_1 & \ldots & a_{\mu} & \ldots & a_{n_2-1} \\
  a_{n_2} & a_{n_2+1} & \ldots & a_{n_2+\mu} & \ldots & a_{n_2+n_2-1} \\
  a_{2n_2} & a_{2n_2+1} & \ldots & a_{2n_2+\mu} & \ldots & a_{2n_2+n_2-1} \\
  \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\
  a_{(n_1-1)n_2} & a_{(n_1-1)n_2+1} & \ldots & a_{(n_1-1)n_2+\mu} & \ldots & a_{(n_1-1)n_2+n_2-1}
\end{array}
\]

Figure 1. The \( n_1 \times n_2 \) code array of \( c(X) \).

For \( 0 < \mu < n_2 \), the \( \mu \)-th column can be put into a polynomial of degree \( (n_1-1)n_2 \) or less as follows:

\[
A_{\mu}(X) = a_{\mu} + a_{n_2+\mu} X^{n_2} + \ldots + a_{(n_1-1)n_2+\mu} X^{(n_1-1)n_2}
= \sum_{i=0}^{n_1-1} a_{i \cdot n_2 + \mu} X^{i \cdot n_2}.
\]  

Then the code polynomial \( c(X) \) can be expressed in the following form:

\[
c(X) = A_0(X) + A_1(X)X + \ldots + A_{n_2-1}(X)X^{n_2-1}
= \sum_{\mu=0}^{n_2-1} A_{\mu}(X)X^\mu.
\]  

The expression of (3) will be used for deriving a lower bound on the weight of \( c(X) \). The main idea is to count the number of nonzero columns in the \( n_1 \times n_2 \) code array corresponding to \( c(X) \) and the number of nonzero components in every nonzero column.
Let $\beta = \alpha^{n_1}$ and $\gamma = \alpha^{n_2}$. Then $\beta$ and $\gamma$ are elements in $\text{GF}(2^q)$ with orders $n_2$ and $n_1$ respectively. Let $\rho$ be a non-negative integer less than $n$. Since $n_1$ and $n_2$ are relatively prime, there exist two unique nonnegative integers, $\ell$ and $m$, with $0 \leq \ell < n_2$ and $0 \leq m < n_1$ such that

$$\alpha^\rho = \beta^\ell \gamma^m$$

(4)

(see Appendix A). Substituting $X$ by $\alpha^\rho$ in (3) and using (4), we have

$$c(\alpha^\rho) = c(\beta^\ell \gamma^m)$$

$$= \sum_{\mu=0}^{n_2-1} A_\mu(\gamma^m) \gamma^m \beta^{\ell \mu}.$$ (5)

Let $q_1$ be the multiplicative order of 2 modulo $n_1$. Then $\text{GF}(2^{q_1})$ is a subfield of $\text{GF}(2^q)$. It can be shown that, for $0 \leq \mu < n_2$, $A_\mu(\gamma^m) \gamma^m \beta^{\ell \mu}$ is an element in $\text{GF}(2^{q_1})$. Define the following polynomial over $\text{GF}(2^{q_1})$:

$$a^{(m)}(X) = \sum_{\mu=0}^{n_2-1} A_\mu(\gamma^m) \gamma^m X^\mu.$$ (6)

It follows from (5) to (6) that

$$c(\alpha^\rho) = a^{(m)}(\beta^\ell).$$ (7)

Clearly, $\beta^\ell$ is a root of $a^{(m)}(X)$ if $\alpha^\rho$ is a root of $c(X)$.

Next we examine the weight of a code polynomial $c(X)$ in $C$. For a given $m$ with $0 \leq m < n_1$, let $V^{(m)}(c)$ be the cyclic code over $\text{GF}(2^{q_1})$ of length $n_2$ which has the following set of elements as zeros (or roots of its generator polynomial):

$$\{\beta^\ell : 0 \leq \ell < n_2 \text{ and } c(\alpha^\rho) = a^{(m)}(\beta^\ell) = 0\}.$$ (8)
Then it is clear that the polynomial $a^{(m)}(X)$ of (8) associated to $c(X)$ is a code polynomial in $V^{(m)}(c)$. Let $d^{(m)}(c)$ denote the minimum distance of $V^{(m)}(c)$. Then, if $a^{(m)}(X)$ is not a zero polynomial, the weight of $a^{(m)}(X)$ is at least $d^{(m)}(c)$.

Now we define the following set of integers associated to the code polynomialonial $c(X)$:

$$J(c) = \{m : 0 \leq m < n_1, \text{ and } c(\beta^l \gamma^m) = a^{(m)}(\beta^l) = 0 \text{ for } l = 0, 1, 2, \ldots, n_2 - 1\}. \quad (9)$$

**Lemma 1:** Consider the polynomial $a^{(m)}(X)$ of (6) associated to a code polynomial $c(X)$ in $C$. If $m$ is an integer in $J(c)$, then $a^{(m)}(X)$ is a zero polynomial and

$$A_\mu(\gamma^m) = 0$$

for $\mu = 0, 1, \ldots, n_2 - 1$.

**Proof:** If $m$ is an integer in $J(c)$, then it follows from the definition of $J(c)$ that $a^{(m)}(X)$ has $1, \beta, \beta^2, \ldots, \beta^{n_2 - 1}$ as roots. However $a^{(m)}(X)$ is a polynomial of degree $n_2 - 1$ or less. Hence if $a^{(m)}(X) \neq 0$, it has at most $n_2 - 1$ distinct roots. As a result, $a^{(m)}(X)$ must be a zero polynomial, and hence it follows from (6) that

$$A_\mu(\gamma^m) = 0$$

for $\mu = 0, 1, \ldots, n_2 - 1$.

Q.E.D.

From (8) and (9), we see that, for $m \in J(c)$, $V^{(m)}(c)$ consists of only the zero polynomial, and $d^{(m)}(c) = 0$.

Let $\overline{J}(c)$ denote the complement of $J(c)$ with respect to the set $\{0, 1, 2, \ldots, n_1 - 1\}$, i.e.,
Define
\[ D(c) = \max \{ d^{(m)}(c) : m \in \mathcal{J}(c) \}. \] (11)

Then we have Lemma 2.

Lemma 2: Let \( c(X) \) be a nonzero code polynomial in \( C \). Consider the expression of \( c(X) \) given by (3). There are at least \( D(c) \) \( A_\mu(X) \)'s in (3) which are nonzero.

Proof: First we note that \( J(c) \neq \{0,1,\ldots,n_1-1\} \), otherwise \( c(X) = 0 \). Hence \( \mathcal{J}(c) \) is not empty. Let \( m \) be an integer in \( \mathcal{J}(c) \). Then
\[ c(\beta^l \gamma^m) = a^{(m)}(\beta^l) \neq 0 \]
for some \( l \) with \( 0 \leq l < n_2 \). This says that \( a^{(m)}(X) \) given by (6) is a nonzero code polynomial in \( V^{(m)}(c) \). Since the minimum weight of \( V^{(m)}(c) \) is \( d^{(m)}(c) \), hence there are at least \( d^{(m)}(c) \) \( A_\mu(\gamma^m) \)'s in (6) are nonzero. This implies that there are at least \( d^{(m)}(c) \) \( A_\mu(X) \)'s in (3) are nonzero. Since this is true for all \( m \) in \( \mathcal{J}(c) \), hence there must be at least \( D(c) \) \( A_\mu(X) \)'s in (3) which are nonzero.

Q.E.D.

Now we define a binary cyclic code associated to a nonzero code polynomial \( c(X) \) in \( C \). Let \( W(c) \) be the binary cyclic code of length \( n_1 \) with the following set of zeros:
\[ \langle (\gamma^{n_2})^m : m \in \mathcal{J}(c) \rangle. \] (12)

Note that the order of \( \gamma^{n_2} \) is \( n_1 \) (same as the order of \( \gamma \)). Let \( d(c) \) denote the minimum distance of \( W(c) \). For \( m \in \mathcal{J}(c) \), it follows from Lemma 1 that the polynomial \( a^{(m)}(X) \) associated to \( c(X) \) is a zero polynomial and
for \( \mu=0,1,2,\ldots, n_2-1 \). Using the coefficients of \( A_\mu(X) \) of (2), we form the following polynomial:

\[
A_\mu(X) = a_\mu + a_{n_2+\mu}X + a_{2n_2+\mu}X^2 + \cdots + a_{(n_1-1)n_2+\mu}X^{n_2-1}
\]

\[
= \sum_{i=0}^{n_1-1} a_i \cdot n_2 + \mu X^i.
\]

It follows from (13) and (14) that

\[
A_\mu((\gamma^m)^{n_2}) = A_\mu(\gamma^m) = 0
\]

for \( m \in J(c) \) and \( \mu=0,1,2,\ldots, n_2-1 \). Since \( A_\mu(X) \) is binary polynomial of degree \( n_1-1 \) or less and has the elements in \( ((\gamma^{n_2})^m : m \in J(c)) \) as roots, \( A_\mu(X) \) is a code polynomial in \( W(c) \). This is to say that each column of the array shown in Figure 1 is a codeword in \( W(c) \). Hence, if \( A_\mu(X) \neq 0 \), the weight of \( A_\mu(X) \) is at least \( d(c) \). Since \( A_\mu(X) \) and \( A_\mu(X) \) have the same coefficients, the weight of \( A_\mu(X) \) is at least \( d(c) \) provided that \( A_\mu(X) \neq 0 \).

Summarizing the above results, we have Lemma 3.

**Lemma 3:** Let \( c(X) \) be a nonzero code polynomial in \( C \). The weight of any nonzero \( A_\mu(X) \) associated to \( c(X) \) is at least equal to the minimum distance \( d(c) \) of the code \( W(c) \).

\[\Delta\Delta\]

It follows from Lemmas 2 and 3 that we have Theorem 1.

**Theorem 1:** Let \( C \) be a binary cyclic code of composite length \( n=n_1 \times n_2 \) where \( n_1 \) and \( n_2 \) are relatively prime. Let \( c(X) \) be a nonzero code polynomial in \( C \). Then the weight of \( c(X) \) is at least \( D(c)d(c) \) where \( D(c) \) is given by (11) and \( d(c) \) is the minimum
weight of the binary code $W(c)$ defined by (12).

Example 1: Let $n_1=3$ and $n_2=17$. Let $\alpha$ be an element of order 51 from field $GF(2^8)$. Let $\beta=\alpha^3$ and $\gamma=\alpha^{17}$. Consider a $(51,18)$ binary cyclic code whose zeros (roots of the generator polynomial) and nonzeros (roots of the parity polynomial) are shown in Table 1. The table is a 3x17 array with 51 nonnegative integers from 0 to 50. A number $p$ in the array represents the field element $\alpha^p$. The rows of the array are numbered from 0 to 2, and the columns are numbered from 0 to 16. If $p$ is at the $m$-th row and the $l$-th column of the array, then the element $\alpha^p$ can be expressed as the product of $\gamma^m$ and $\beta^l$, i.e.,

$$\alpha^p = \beta^l \gamma^m.$$ 

For example, $\alpha^{41} = \beta^8 \gamma$. The underlined numbers in the array represent the nonzeros of the code while all the other numbers in the array represent the zeros of the code. For example, $\alpha^{29}$ is not a zero and $\alpha^{41}$ is a zero.

Table 1

<table>
<thead>
<tr>
<th>Nonzeros of a (51,18) Binary Cyclic Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 3 6 9 12 15 18 21 24 27 30 33 36 39 42 45 48</td>
</tr>
<tr>
<td>17 20 23 26 29 32 35 38 41 44 47 50 2 5 8 11 14</td>
</tr>
<tr>
<td>34 37 40 43 46 49 1 4 7 10 13 16 19 22 25 28 31</td>
</tr>
</tbody>
</table>

Let $c(X)$ be a nonzero code polynomial. From the theory of cyclic code, we know that the zeros of the code are roots of $c(X)$. 

9
From Table 1 we see that, for \( m = 0 \),
\[
c(\beta^l) = 0
\]
for \( l = 0, 1, \ldots, 16 \). For \( m = 1 \),
\[
c(\beta^l \gamma^2) \neq 0
\]
for some \( l = 0, 1, \ldots, 15 \). Therefore,
\[
J(c) = \{0\} \quad \text{and} \quad \overline{J}(c) = \{1, 2\}.
\]
Note that, for \( m = 1 \),
\[
c(\beta^l \gamma^{2}) = a^1(\beta^l) = 0
\]
for \( l = 2, 6, 7, 8, 9, 10, 11, 15 \). It follows from (8) that the code \( V(1)(c) \) has the set of zeros which includes
\[
\{\beta^2, \beta^6, \beta^7, \beta^8, \beta^9, \beta^{10}, \beta^{11}, \beta^{15}\}
\]
as a subset. Since \( V(1)(c) \) has 6 consecutive zeros (from \( \beta^6 \) to \( \beta^{11} \)), it follows from BCH bound [16] that the minimum distance \( d(1)(c) \) of \( V(1)(c) \) is at least 7. Note that \( \beta^l \) is a zero of \( V(1)(c) \) if and only if \( \beta^{2l} \) is a zero of \( V(2)(c) \). Hence \( V(2)(c) \) is equivalent to \( V(1)(c) \) and
\[
d(2)(c) = d(1)(c).
\]
Then
\[
D(c) = \max \{d(1)(c), d(2)(c)\} \geq 7.
\]
Since \( J(c) = \{0\} \), the code \( W(c) \) has \( \gamma^0=1 \) as the only zero. Hence the minimum distance \( d(c) \) of \( W(c) \) is 2. Then it follows from Theorem 1 that the weight of \( c(X) \) is at least \( D(c)d(c) \geq 14 \). Hence the minimum distance of the \((51, 18)\) code is at least 14. Note that the BCH bound of this code is 12 while the real minimum distance is 14[16].
The results derived in this section will be used to derive lower bounds on minimum distances and the multi-level error correcting capabilities of cyclic direct-sum codes of composite length in the latter sections. The result given in Theorem 1 is a slight variation of a result proved by Hartman and Tzeng[17].

III. DIRECT SUM OF TWO CYCLIC CODES

For \( i = 1 \) or \( 2 \), let \( g_i(X) \) and \( h_i(X) \) be the generator and parity polynomials of a binary \((n,k_i)\) cyclic code \( C_i \) respectively. Note that

\[
g_i(X)h_i(X) = X^{n+1}
\]

for \( i = 1,2 \). Suppose \( h_1(X) \) and \( h_2(X) \) are relatively prime. Now we want to show that the only code polynomial common to both \( C_1 \) and \( C_2 \) is the zero polynomial. Let \( c(X) \) be a code polynomial common to both \( C_1 \) and \( C_2 \). Then

\[
c(X) = a_1(X)g_1(X),
\]

\[
c(X) = a_2(X)g_2(X).
\]

It follows from (15) and (16) that

\[
c(X)h_i(X) = 0 \mod X^{n+1}
\]

for \( i = 1,2 \). Since \( h_1(X) \) and \( h_2(X) \) are relatively prime, there exists two polynomials \( b_1(X) \) and \( b_2(X) \) such that

\[
b_1(X)h_1(X) + b_2(X)h_2(X) = 1 \mod X^{n+1}.
\]

Multiplying both sides of (18) by \( c(X) \), we have

\[
c(X) = \{ b_1(X)c(X)h_1(X) + b_2(X)c(X)h_2(X) \} \mod X^{n+1}.
\]

It follows from (17) and (19) that

\[
c(X) = 0 \mod X^{n+1}.
\]
Since \( c(X) \) is a polynomial of degree less than \( n \), it follows from (20) that \( c(X) \) must be the zero polynomial. This proves that \( C_1 \) and \( C_2 \) have only the zero polynomial as the common code polynomial.

Let \( g(X) \) be the greatest common divisor of \( g_1(X) \) and \( g_2(X) \), i.e.

\[
g(X) = \text{GCD} \{g_1(X), g_2(X)\}.
\]

Since \( h_1(X) \) and \( h_1(X) \) are relatively prime, it is easy to see from (15) that

\[
g_1(X) = g(X)h_2(X),
g_2(X) = g(X)h_1(X),
x^{n+1} = g(X)h_1(X)h_2(X).
\]

The degrees of \( g(X) \) and \( h(X) = h_1(X)h_2(X) \) are \( n-k_1-k_2 \) and \( k_1+k_2 \) respectively. Let \( C \) be the direct sum of \( C_1 \) and \( C_2 \). Then \( C \) is an \( (n,k_1+k_2) \) linear code. We can readily see that every code polynomial in \( C \) is divisible by \( g(X) \). Since the degree of \( g(X) \) is \( n-k_1-k_2 \), hence \( g(X) \) generates \( C \). Therefore the direct sum \( C \) of \( C_1 \) and \( C_2 \) has \( g(X) \) and \( h(X) = h_1(X)h_2(X) \) as its generator and parity polynomials.

Let \( A_1 = \{0,1\}^{k_1} \) and \( A_2 = \{0,1\}^{k_2} \) be two message spaces. A message from \( A_i \) is denoted by \( \bar{x}_i \), where \( i=1,2 \). Let \( A \) be the Cartesian product of \( A_1 \) and \( A_2 \). Then,

\[
A = A_1 \times A_2
\]

\[
= \{(\bar{x}_1,\bar{x}_2) : \bar{x}_i \in A_i \text{ for } i = 1,2\}.
\]

We call \( A_1 \) and \( A_2 \) the first and second component message spaces of \( A \) respectively; and call \( \bar{x}_1 \) and \( \bar{x}_2 \) the first and
second component message of the message \((\overline{x}_1, \overline{x}_2)\). Let \(C_1\) and \(C_2\) be the codes for the component message spaces \(A_1\) and \(A_2\) respectively. Then the direct-sum code \(C = C_1 \oplus C_2\) is an \((n,k_1+k_2)\) code for the product space \(A\). Let \(\overline{v}(\overline{x}_1, \overline{x}_2)\) denote the codeword in \(C\) for the message \((\overline{x}_1, \overline{x}_2)\). Then \(\overline{v}(\overline{x}_1, \overline{x}_2)\) can be uniquely expressed as the sum of \(\overline{v}(\overline{x}_1)\) and \(\overline{v}(\overline{x}_2)\), where \(\overline{v}(\overline{x}_1)\) and \(\overline{v}(\overline{x}_2)\) are the codewords for component messages \(\overline{x}_1\) and \(\overline{x}_2\) in \(C_1\) and \(C_2\) respectively.

In [11,12], we have shown that, under certain distance conditions, direct sum codes have multi-level error correcting capabilities and hence are multi-level UEP codes. The main purpose of this paper is to construct UEP codes by taking direct sums of cyclic codes of composite length. For this purpose, we need to review some distance properties of direct-sum codes. These properties were proved in [11,12]. We simply state these properties here without proofs.

The error correcting capabilities of an UEP code is determined by its separation vector \(s[5,11,12]\). For an \(m\)-level UEP codes, the separation vector is a distance vector of \(m\) components. In this paper, we only consider two-level UEP codes. Consider a message \((\overline{x}_1, \overline{x}_2)\) which consists of two parts \(\overline{x}_1\) and \(\overline{x}_2\), where \(\overline{x}_1\) and \(\overline{x}_2\) are \(k_1\)-tuple and \(k_2\)-tuple over \(GF(2)\) respectively. Let \(C\) be the code for the message space \((\overline{x}_1, \overline{x}_2) : \overline{x}_1 \in (0,1)^{k_1} \text{ and } \overline{x}_2 \in (0,1)^{k_2}\). Let \(\overline{v}(\overline{x}_1, \overline{x}_2)\) be the codeword for the message \((\overline{x}_1, \overline{x}_2)\). Then, the separation vector of \(C\) is \(\overline{s} = (s_1, s_2)\) where

\[
s_1 = \min \{w(\overline{v}(\overline{x}_1, \overline{x}_2)) : \overline{x}_1 \neq \overline{0}\},
\]

13
\[ s_2 = \min \{ w(\overline{v}(\overline{x}_1, \overline{x}_2)) : \overline{x}_2 \neq \overline{0} \}, \]  
(21)

and \( w(\overline{v}) \) denote the Hamming weight of \( \overline{v} \). Clearly, the minimum distance of code \( C \) is simply \( d_{\text{min}} = \min\{s_1, s_2\} \). The component \( s_1 \) determines the level of protection for component message \( \overline{x}_1 \) against the channel errors, and the component \( s_2 \) determines the level of protection for component message \( \overline{x}_2 \) against the channel errors. For a two-level UEP code \( s_1 \neq s_2 \). Without loss of generality, we assume that \( s_1 > s_2 \). The error correcting capabilities of a two-level UEP code are stated in Theorem 2 (see [11,12] for a proof).

**Theorem 2:** Consider a two-level UEP code \( C \) for the message space \( A = \{(\overline{x}_1, \overline{x}_2) : \overline{x}_1 \in \{0,1\}^{k_1} \text{ and } \overline{x}_2 \in \{0,1\}^{k_2}\} \). Let \( \overline{s} = (s_1, s_2) \) be the separation vector of \( C \). Let \( \overline{v}(\overline{x}_1, \overline{x}_2) \) and \( \overline{r} \) be the transmitted codeword and received word respectively. Then the component message \( \overline{x}_1 \) can be decoded correctly from \( \overline{r} \) if \( \overline{r} \) contains \( t_1 = \lfloor(s_1-1)/2 \rfloor \) or fewer errors (\( \overline{x}_2 \) may not be decoded correctly). If \( \overline{r} \) contains \( t_2 = \lfloor(s_2-1)/2 \rfloor \) or fewer errors, then both \( \overline{x}_1 \) and \( \overline{x}_2 \) can be decoded correctly.

\[ \triangle \]

From Theorem 2, we see that a two-level UEP code with separation vector \( \overline{s} = (s_1, s_2) \) protects message \( \overline{x}_1 \) against \( t_1 = \lfloor(s_1-1)/2 \rfloor \) or fewer errors and protects message \( \overline{x}_2 \) against \( t_2 = \lfloor(s_2-1)/2 \rfloor \) or fewer errors.

Now we come back to direct-sum codes. Theorem 3 states the conditions under which a direct-sum code is a two-level UEP code (see [11,12] for a proof).
Theorem 3: Let $C_1$ and $C_2$ be an $(n,k_1)$ code and $(n,k_2)$ code for message spaces $A_1 = (0,1)^{k_1}$ and $A_2 = (0,1)^{k_2}$ respectively. Suppose $C_1$ and $C_2$ have only the zero vector in common. Let $C = C_1 \oplus C_2$ be the direct sum of $C_1$ and $C_2$. Suppose the following distance conditions are satisfied:

(i) The weight of any nonzero codeword in $C_2$ is at least $d_2$; and

(ii) The weight of any codeword in $C - C_2$ is at least $d_1$ with $d_1 > d_2$.

Then $C$ is a two level UEP code with a separation vector $\bar{s} = (s_1, s_2)$, where $s_1 \geq d_1$ and $s_2 \geq d_2$.

It should be noted that Theorem 2 is also valid for the case of $s_1 = s_2$ and Theorem 3 is also valid for the case of $d_1 = d_2$. However, in such a case, $C$ is not a UEP code. In the next section we will consider two-level UEP codes which are direct sums of cyclic codes of composite length.

IV. TWO-LEVEL UEP CYCLIC DIRECT-SUM CODES OF COMPOSITE LENGTH

Let $n = n_1 n_2$ where $n_1$ and $n_2$ are relatively prime. Again let $\alpha$ be an element of order $n$ from some field $GF(2^q)$. Let $\beta = \alpha^{n_1}$ and $\gamma = \alpha^{n_2}$. Then, for any $\rho$ with $0 \leq \rho < n$, there exist two integers, $m$ and $l$, with $0 \leq m < n_1$ and $0 \leq l < n_2$ such that $\alpha^\rho = \beta^m \gamma^l$.

For $i = 1, 2$, let $C_i$ be an $(n,k_i)$ binary cyclic code with generator polynomial $g_i(X)$ and parity polynomial $h_i(X)$ respectively. Note that $C_1$ and $C_2$ are two cyclic codes of composite length. Let $c_i(X)$ be a code polynomial in $C_i$ for $i =$
1,2. Define

\[ J_1 = \bigcap_{\substack{c_1(x) \neq 0 \\text{ or } \ c_1(x) \in C_1}} J(c_1) \]
\[ J_2 = \bigcap_{\substack{c_2(x) \neq 0 \\text{ or } \ c_2(x) \in C_2}} J(c_2) \]

where \( J(c_i) \) is defined by (9). It is easy to see that, for \( i=1,2, \) a number \( m \) with \( 0 \leq m < n_1 \) is in \( J_1 \) if and only if \( C_i \) contains \( \beta^l \gamma^m \) with \( l = 0,1,\ldots,n_2-1 \) as zeros. Let

\[ J_i = \{0,1,\ldots,n_1-1\} - J_i \quad (24) \]

for \( i=1,2. \) If \( \beta^l \gamma^m \) is not a zero of \( C_i \) for some \( l \) with \( 0 \leq l < n_2, \) then \( m \) is an element in \( \overline{J}_i. \)

Assume that \( \overline{J}_1 \) and \( \overline{J}_2 \) are disjoint. Apparently, \( C_1 \) and \( C_2 \) have no common nonzeros. Therefore, \( h_1(x) \) and \( h_2(x) \) are relatively prime. The direct sum of \( C_1 \) and \( C_2 \) is an \( (n_1, k_1+k_2) \) cyclic code \( C \) with generator polynomial \( g(x) = \text{GCD} \{g_1(x), g_2(x)\} \) and parity polynomial \( h(x) = h_1(x)h_2(x). \)

For \( 0 \leq m < n_1, \) let

\[ V_1^{(m)} = \bigcup_{\substack{c_1(x) \neq 0 \\text{ or } \ c_1(x) \in C_1}} V^{(m)}(c_1), \]
\[ V_2^{(m)} = \bigcup_{\substack{c_2(x) \neq 0 \\text{ or } \ c_2(x) \in C_2}} V^{(m)}(c_2). \]

where \( V^{(m)}(c_i) \) is a cyclic code associated to the code polynomial \( c_i(x) \) defined by (8). Thus \( V^{(m)}_i \) is a cyclic code of length \( n_2 \) over \( GF(2^{q_1}) \) where \( q_1 \) is the multiplicative order of 2 modulo \( n_1 \) for \( i=1,2. \) The element \( \beta^l \) is a zero of \( V^{(m)}_i \) if and only if \( \beta^l \gamma^m \) is a zero of \( C_i. \) From the results in Section II, we see that,
for \( m \in J_i \), \( V_i(m) \) consists of only the zero polynomial. Let \( d_1(m) \) for \( i=1,2 \) and \( 0 \leq m < n \). Define

\[
D_1 = \min_{m \in J_1} \{ d_1(m) \},
\]

\[
D_2 = \min_{m \in J_2} \{ d_2(m) \}.
\]

Clearly,

\[
D(c_i) \geq d_i(m)(c_i) \geq d_1(m) \geq D_i
\]

(29)

for any nonzero code polynomial \( c_i(X) \) in \( C_i \) and \( m \in J_i \) with \( i=1,2 \). Then, it follows from Lemma 2 that at least \( D_i \) of the \( n_2 \) polynomials \( A_{\mu}(X) \) associated to any nonzero code polynomial \( c_i(X) \) in \( C_i \) are nonzero for \( i=1,2 \).

Next we define two binary cyclic codes of length \( n_1 \) based on \( C_1 \) and \( C_2 \) as follows:

\[
W_1 = \bigcup_{c_1(X) \neq 0} W(c_1)
\]

(30)

\[
W_2 = \bigcup_{c_2(X) \neq 0} W(c_2)
\]

(31)

where \( W(c_i) \) is the binary cyclic code associated to a code polynomial \( c_i(X) \) defined by (12). We readily see that \( (\gamma^{n_2})^m \) is a zero of \( W_i \) if and only if \( m \in J_i \) for \( i=1,2 \). Equivalently, \( (\gamma^{n_2})^m \) is a zero of \( W_i \) if and only if \( \beta^l \gamma^m \) with \( l=0,1,\ldots,n_2-1 \) are zeros of \( C_i \). Since \( J_1 \) and \( J_2 \) are disjoint, the sets of nonzeros for \( W_1 \) and \( W_2 \) do not have any common element. Now consider the binary cyclic code \( W \) associated to the direct sum \( C = C_1 \oplus C_2 \).
\[ W = \bigcup_{c(X) \neq 0} W(c). \]  
\[ \text{Define} \]
\[ J = \bigcap_{c(X) \in C} J(c). \]  

It is easy to see that
\[ J = J_1 \cap J_2. \]  
Then \((\gamma^{n_2})^m\) is a zero of \(W\) if and only if \(m \in J\). Or, equivalently, \((\gamma^{n_2})^m\) is a zero of \(W\) if and only if \(\beta^m\gamma^l\) with \(l = 0, 1, \ldots, n_2 - 1\) are zeros of \(C\). The set of nonzeros for \(W\) is
\[ \left\{(\gamma^{n_2})^m : m \in \overline{J}\right\} \]  

where \(\overline{J} = (0, 1, \ldots, n_1 - 1) - J_1 \cap J_2\). Since \(\overline{J}_1\) and \(\overline{J}_2\) are disjoint, we can easily see that \(W\) is the direct sum of \(W_1\) and \(W_2\), i.e.,
\[ W = W_1 \oplus W_2. \]  

Let \(d_1\), \(d_2\), and \(d\) be the minimum distances of \(W_1\), \(W_2\), and \(W\) respectively. Then, \(d_1 \geq d\) and \(d_2 \geq d\).

Now we examine the distance structure of the direct sum \(C\) of \(C_1\) and \(C_2\). Any code polynomial \(c(X)\) in \(C\) can be expressed as the following sum,
\[ c(X) = c_1(X) + c_2(X) \]
where \(c_1(X) \in C_1\) and \(c_2(X) \in C_2\). Suppose \(c(X) \in C_2\) and \(c(X) \neq 0\). Then \(c_1(X) = 0\) and \(c(X) = c_2(X)\). It follows from Theorem 1 that the weight of \(c(X) = c_2(X)\) is at least \(D(c_2)d(c_2)\). Note that \(D(c_2) \geq D_2\) and \(d(c_2) \geq d_2\). Thus the weight of \(c(X)\), denoted \(w(c(X))\) is at least \(D_2d_2\), i.e.,
\[ w(c(X)) \geq D_2 d_2. \]  

Suppose \( c(X) \in C-C_2 \). Clearly \( c_1(X)=0 \). There exists an integer \( m \) in \( J_1 \) such that
\[ c_1(\beta^l \gamma^m) = 0 \]  
for some \( l \in \{0,1,\ldots,n_2-1\} \). Since \( J_1 \) and \( J_2 \) are disjoint, \( m \) must be in \( J_2 \). Consequently,
\[ c_2(\beta^l \gamma^m) = 0 \]  
for \( l = 0,1,\ldots,n_2-1 \). From (38) and (39), we have
\[
\begin{align*}
c(\beta^l \gamma^m) &= c_1(\beta^l \gamma^m) + c_2(\beta^l \gamma^m) \\
&= c_1(\beta^l \gamma^m) \\
&= 0
\end{align*}
\]
for some \( l = 0,1,\ldots,n_2-1 \). Accordingly, we have
\[
\begin{align*}
v(m)(c) &= v(m)(c_1), \\
d(m)(c) &= d(m)(c_1).
\end{align*}
\]  

It follows from Theorem 1 that the weight of \( c(X) \) is at least \( D(c)d(c) \). Note that \( D(c) \geq d(m)(c) = d(m)(c_1) \geq d_1(m) \geq D_1 \) and \( d(c) \geq d \). Thus the weight of \( c(X) \) is at least \( D_1 d \). Summarizing the above results, we have that

(1) For \( c(X) \in C-C_2 \), \( w(c) \geq D_1 d \); and

(2) For \( c(x) \in C_2 \) and \( c(x)=0 \), \( w(c) \geq D_2 d_2 \).

Suppose \( D_1 d > D_2 d_2 \). It follows from Theorem 3 that \( C \) is a two-level UEP code for the product message space \( A=A_1 \times A_2 \) with separation vector \( \bar{s}=(s_1, s_2) \) where \( A_1=(0,1)^{k_1} \), \( A_2=(0,1)^{k_2} \), \( s_1 \geq D_1 d \), and \( s_2 \geq D_2 d_2 \).

**Example 2:** Let \( n_1=3 \) and \( n_2=17 \). Let \( \alpha \) be a primitive 51-th root of unity. Let \( \beta=\alpha^3 \) and \( \gamma=\alpha^{17} \). Let \( C_1 \) be the \((51,18)\) binary
cyclic code given in Example 1. The nonzeros of $C_1$ are given in Table 1. Let $C_2$ be the (51,16) binary cyclic code with the following set of nonzeros:

$$\{\beta^l : l = 1,2,\ldots,16\}. \tag{42}$$

From Table 1 and (42), we see that the sets of nonzeros for $C_1$ and $C_2$ do not have any element in common. As a result, the direct sum $C$ of $C_1$ and $C_2$ is a (51,34) binary cyclic code. From Table 1 and (42), we find that $J_1=\{0\}$ and $J_2=\{1,2\}$. Then, $\overline{J}_1=\{1,2\}$ and $\overline{J}_2=\{0\}$. Obviously, $\overline{J}_1$ and $\overline{J}_2$ are disjoint.

From Table 1, we see that the code $V_1^{(1)}$ has $\beta^6, \beta^7, \beta^8, \beta^9, \beta^{10}$ and $\beta^{11}$ as zeros. By BCH bound, the minimum distance $d_1^{(1)}$ of $V_1^{(1)}$ is at least 7. Note that the code $V_1^{(2)}$ is equivalent to $V_1^{(1)}$ (in the sense that $\beta^l$ is a zero of $V_1^{(1)}$ if and only if $\beta^{2l}$ is a zero of $V_1^{(2)}$). Hence the minimum distance $d_1^{(2)}$ of $V_1^{(2)}$ is the same as that of $V_1^{(1)}$. As a result, $d_1^{(2)}=d_1^{(1)} \geq 7$.

From (27), we have $D_1 \geq 7$. Since $J_1=\{0\}$, the binary code $W_1$ has only one zero which is $\gamma^0=1$. The minimum distance $d_1$ of $W_1$ is at least 2. In fact $W_1$ contains the following four vectors:

$$(000),(110),(011),(101).$$

Hence $d_1=2$.

Note that $\overline{J}_2=\{0\}$. To determine $D_2$, we only need to determine the minimum distance $d_2^{(0)}$ of the code $V_2^{(0)}$. Since $\beta^0=1$ is a zero of $C_2$, $\beta^0=1$ is a zero of $V_2^{(0)}$. Hence $d_1^{(0)}$ is at least 2. From (28), we have $D_2 \geq 2$. Now consider the binary cyclic code $W_2$. Since $J_2=\{1,2\}$, the zeros of $W_2$ are $\gamma^{17}=\gamma^2$ and $(\gamma^{17})^2=\gamma$. Thus the minimum distance $d_2$ of $W_2$ is at least 3. In fact, $W_2$
consists only two codewords, (000) and (111). Hence \( d_2 = 3 \).

The binary code \( W \) is the direct sum of \( W_1 \) and \( W_2 \), and hence is the entire space \( (0,1)^3 \). Therefore, the minimum distance of \( W \) is \( d = 1 \).

From the above analysis, we have that \( D_1 d_1 \geq 7 \) and \( D_2 d_2 \geq 6 \). Therefore the direct sum \( C \) of \( C_1 \) and \( C_2 \) is a \((51,34)\) two-level UEP cyclic code with a separation vector at least \((7,6)\). The message space \( A \) for \( C \) is the product of \( A_1 = (0,1)^{18} \) and \( A_2 = (0,1)^{16} \). Thus \( C \) provides protection of the first 18 message bits against 3 or fewer random errors and protection of the next 16 message bits against 2 or fewer random errors. Note that the best single-level error correcting \((51,34)\) cyclic code has minimum distance \( d = 6 \) [16].

Some two-level UEP cyclic codes of composite length are given in Table 2. The nonzeros (roots of the parity polynomial) of each code are given. The nonzeros are represented by their exponents of \( \alpha \). The true minimum distance and BCH bound of a code are denoted by \( d \) and \( d_{\text{BCH}} \) respectively. From Table 2, we see that our algorithm gives the true minimum distances of these cyclic codes by comparing \( s_2 \) with \( d \).

Table 2

<table>
<thead>
<tr>
<th>( n )</th>
<th>( k )</th>
<th>( n_1 )</th>
<th>( n_2 )</th>
<th>( k_1 )</th>
<th>( k_2 )</th>
<th>( s_1 )</th>
<th>( s_2 )</th>
<th>( d )</th>
<th>( d_{\text{BCH}} )</th>
<th>nonzeros</th>
</tr>
</thead>
<tbody>
<tr>
<td>51</td>
<td>17</td>
<td>3</td>
<td>17</td>
<td>1</td>
<td>16</td>
<td>17</td>
<td>16</td>
<td>16</td>
<td>11</td>
<td>0, 11, 19</td>
</tr>
<tr>
<td>51</td>
<td>19</td>
<td>3</td>
<td>17</td>
<td>1</td>
<td>18</td>
<td>17</td>
<td>14</td>
<td>14</td>
<td>11</td>
<td>11, 19</td>
</tr>
<tr>
<td>51</td>
<td>35</td>
<td>3</td>
<td>17</td>
<td>18</td>
<td>17</td>
<td>7</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>0, 3, 9, 11, 17, 19</td>
</tr>
<tr>
<td>63</td>
<td>30</td>
<td>7</td>
<td>9</td>
<td>9</td>
<td>21</td>
<td>14</td>
<td>12</td>
<td>12</td>
<td>8</td>
<td>3, 9, 11, 13, 27, 31</td>
</tr>
</tbody>
</table>
V. A CLASS OF TWO-LEVEL UEP CYCLIC DIRECT-SUM CODES

There is another class of two-level UEP cyclic codes. Each code in this class is the direct sum of two cyclic codes of composite length. Let \( n = n_1 n_2 \) where \( n_1 \) and \( n_2 \) are odd positive integers and relatively prime. Again, let \( \alpha \) be an element of order \( n \) from \( \text{GF}(2^q) \). Let \( \beta = \alpha^{n_1} \) and \( \gamma = \alpha^{n_2} \). Let \( C_{11} \) be an \((n_1, k_1 + 1)\) binary cyclic code whose parity polynomial \( h_{11}(X) \) has the following set of roots:

\[
(1, \gamma^{m_1}, \gamma^{m_2}, \ldots, \gamma^{m_{k_1}}).
\]

The elements in the set of (43) are the nonzeros of \( C_{11} \). Let \( C_{22} \) be an \((n_2, k_2 + 1)\) binary cyclic code whose parity polynomial \( h_{22}(X) \) has the following set of roots:

\[
(1, \beta^{l_1}, \beta^{l_2}, \ldots, \beta^{l_{k_2}}).
\]

Then elements in the set of (44) are the nonzeros of \( C_{22} \). Let \( d_{11} \) and \( d_{22} \) be the minimum distances of \( C_{11} \) and \( C_{22} \) respectively. Let \( d_{11} \) and \( d_{22} \) be the minimum distances of the even-weight subcodes of \( C_{11} \) and \( C_{22} \) respectively.

Now we form two longer cyclic codes from \( C_{11} \) and \( C_{22} \). Let \( C_1 \) be an \((n_1 n_2, k_1)\) binary cyclic code with parity polynomial

\[
h_1(X) = h_{11}(X)/(X+1),
\]

and let \( C_2 \) be an \((n_1 n_2, k_2)\) binary cyclic code with parity polynomial

\[
h_2(X) = h_{22}(X)/(X+1).
\]

Clearly, the sets of nonzeros for \( C_1 \) and \( C_2 \) are \( \{\gamma^{m_1}, \gamma^{m_2}, \ldots, \gamma^{m_{k_1}}\} \) and \( \{\beta^{l_1}, \beta^{l_2}, \ldots, \beta^{l_{k_2}}\} \) respectively. It is easy
to show that these two sets of nonzeros are disjoint. Hence $h_1(X)$ and $h_2(X)$ are relatively prime. Note that the roots of $h_1(X)$ are zeros of $C_2$ and the roots of $h_2(X)$ are zeros of $C_1$.

Let $C$ be the direct sum of $C_1$ and $C_2$. Then $C$ is an $(n_1 n_2, k_1 + k_2)$ cyclic code with parity polynomial

$$h(X) = h_1(X) h_2(X).$$

(47)

Now we examine the distance structure of the direct-sum code $C$.

A code polynomial $c(X)$ in $C$ can be expressed as the following sum:

$$c(X) = c_1(X) + c_2(X)$$

with $c_1(X) \in C_1$ and $c_2(X) \in C_2$. First we consider the case that $c(X) \in C - C_2$

In this case, $c_1(X) \neq 0$. Hence, there exists an integer $m \in \{m_1, m_2, \ldots, m_{k_2}\}$ such that

$$c_1(\gamma^m) = 0.$$  

(48)

Since $\gamma^m$ is a zero of $C_2$, we have

$$c(\gamma^m) = c_1(\gamma^m) + c_2(\gamma^m) = c_1(\gamma^m) \neq 0.$$  

(49)

This implies that

$$m \in J(c)$$

where $J(c)$ is defined by (10). Note that $C$ has $\beta^\ell \gamma^m$ with $\ell = 1, 2, \ldots, n_2 - 1$ as zeros. Thus

$$c(\beta^\ell \gamma^m) = a^{(m)}(\beta^\ell) = 0$$

(50)

for $\ell = 1, 2, \ldots, n_2 - 1$. Then the code $V^{(m)}(c)$ associated to $c(X)$ has $\beta^\ell$ with $\ell = 1, 2, \ldots, n_2 - 1$ as zeros. It follows from the BCH bound that the minimum distance $d^{(m)}(c)$ of $V^{(m)}(c)$ is $n_2$. Hence,

$$D(c) = \max \{d^{(m)}(c) : m \in J(c)\} = n_2.$$  

(51)
It follows from Lemma 2 that all the $n_2$ polynomials, $A_\mu(X)$ with $\mu = 0, 1, \ldots, n_2 - 1$, associated to $c(X)$ are nonzero. Next, we want to determine the weight of each $A_\mu(X)$. For $0 \leq \ell < n_2$ and

$$m \in \{0, 1, \ldots, n_1 - 1\} - \{0, m_1, m_2, \ldots, m_{k_1}\},$$

$\beta^\ell \gamma^m$ is a zero of $C$. It follows from the definition of $J(c)$ given by (9) that

$$J(c) \supset \{0, 1, \ldots, n_1 - 1\} - \{0, m_1, m_2, \ldots, m_{k_1}\}.$$

This implies that the binary cyclic code $W(c)$ associated to $c(X)$ is a subcode of the code $C^*_11$ whose set of nonzeros is

$$(1, (\gamma^{n_2})^{m_1}, (\gamma^{n_2})^{m_2}, \ldots, (\gamma^{n_2})^{m_{k_1}}).$$

From (43) and (52), we see that $C^*_11$ and $C^*_11$ are equivalent. As a result, they have the same minimum distance $d_{11}$. Therefore, the minimum distance $d(c)$ of $W(c)$ is at least $d_{11}$. This implies that the weight of every nonzero $A_\mu(X)$ is at least $d_{11}$. It follows from Theorem 3 that the weight of $c(X)$ is at least

$$D(c)d(c) \geq n_2d_{11}.$$
Since the length of $C_{22}$, $n_2$, is odd and $\beta^0 = 1$ is not a zero of $C_{22}$, the weight of an even-weight code polynomial in $C_{22}$ is at most $n_2 - d_{22}$. This implies that at least $d_{22}$ of the $n_2$ coefficients, $A_0(1), A_1(1), \ldots, A_{n_2-1}(1)$ are zero. This means that at least $d_{22}$ of the $n_2$ polynomials, $A_0(X), A_1(X), \ldots, A_{n_2-1}(X)$ have even weight, which is at least $d_{11}$. As a result, the weight of $c(X)$ is at least

$$ (n_2 - d_{22})d_{11} + d_{22}d_{11} = n_2d_{11} + (d_{11}d_{11})d_{22}. \tag{54} $$

Now we consider the case for which $c(X) \in C_2$ and $c(X) \neq 0$. Then $c(X) = c_2(X) \neq 0$. It follows from the definition of $C_2$ that there exists some $l \in \{l_1, l_2, \ldots, l_{k_2}\}$ for which

$$ c(\beta^l) = c_2(\beta^l) = a^0(\beta^l) \neq 0. \tag{55} $$

For $l \in \{0, 1, 2, \ldots, n_2-1\} - \{l_1, l_2, \ldots, l_{k_2}\}$, $\beta^l$ is a zero of $C$, which implies that

$$ c(\beta^l) = a^0(\beta^l) = 0, \tag{55} $$

i.e. $\nu(0)$ contains $\beta^l$ as a zero. From (2), (6), (44), and (55), we see that $a^0(X)$ is an even weight binary polynomial in $C_{22}$. Therefore, at least $d_{22}$ of the $n_2$ coefficients of $a^0(X)$ are nonzero, or equivalently, at least $d_{22}$ of the $n_2$ polynomials,

$$ A_0(X), A_1(X), \ldots, A_{n_2-1}(X) $$

are nonzero. For $m \in \{1, 2, \ldots, n_l-1\}$ and $l \in \{0, 1, 2, \ldots, n_2-1\}$, we have

$$ c(\beta^l m) = a^m(\beta^l) = 0. $$

It follows from (2), (6) and Lemma 1 that

$$ A_\mu(\gamma^m) = 0 $$

for $\mu \in \{0, 1, 2, \ldots, n_2-1\}$ and $m \in \{1, 2, \ldots, n_l-1\}$. Thus, any
nonzero $A_{\mu}(X)$ has $n_1$ nonzero components according to BCH bound.
Since $c(X)$ contains at least $d_{22}$ nonzero $A_{\mu}(X)$'s, the weight of $c(X)$ is at least $n_1 d_{22}$.

Summarizing the above results, we have the following weight structure for the direct sum code $C$:

1. For $c(X) \in C-C_2$, $w(c) \geq n_2 d_{11} + d_{22}(d_{11} - d_{11})$;
2. For $c(X) \in C_2$ and $c(X) \neq 0$, $w(c) \geq n_1 d_{22}$.

Suppose $C_{11}$ and $C_{22}$ are chosen such that

$$n_2 d_{11} + d_{22}(d_{11} - d_{11}) > n_1 d_{22}.$$ 

Then $C$ is an $(n_1 n_2, k_1 + k_2)$ cyclic two-level UEP code with separation vector $\bar{s} = (s_1, s_2)$ where

$$s_1 \geq n_2 d_{11} + d_{22}(d_{11} - d_{11}),$$
$$s_2 \geq n_1 d_{22}.$$

The code is capable of protecting the first $k_1$ message bits against any

$$t_1 = \left\lfloor \frac{n_2 d_{11} + d_{22}(d_{11} - d_{11})}{2} \right\rfloor - 1$$

or fewer errors and protecting the next $k_2$ message bits against any

$$t_2 = \lfloor n_1 d_{22}/2 \rfloor - 1$$

or fewer errors.

Example 3: Let $n_1 = 7$ and $n_2 = 5$. Let $C_{11}$ be the $(7,4)$ Hamming code with parity polynomial $h_{11}(X) = (X+1)(X^3+X+1)$. Then the minimum distance $d_{11}$ of $C_{11}$ is 3 and the minimum distance $d_{11}$ of the even weight subcode of $C_{11}$ is 4. Let $C_{22}$ be the $(5,5)$ binary cyclic code with parity polynomial $h_{22}(X) = X^5+1$. Then the minimum
distance $d_{22}$ of $C_{22}$ is 1 and the minimum distance $d_{12}$ of the even-weight subcode of $C_{22}$ is 2. The codes $C_1$ and $C_2$ are a (35,3) and a (35,4) cyclic codes with parity polynomials $h_1(X) = X^3 + X + 1$ and $h_2(X) = X^4 + X^3 + X^2 + X + 1$ respectively. The direct sum $C$ of $C_1$ and $C_2$ is a (35,7) cyclic code with parity polynomial, $h(X) = (X^3 + X + 1)(X^4 + X^3 + X^2 + X + 1)$.

The separation vector $\bar{s}$ for $C$ has two components,

\[
s_1 \geq n_2 d_{11} + d_{22} \left( d_{11} - d_{11} \right) \\
\geq 5x3 + 1x(4-3) = 16,
\]

\[
s_2 \geq n_1 d_{22} \geq 7x2 = 14.
\]

Using this code, the first 3 message bits will be decoded correctly if there are no more than 7 errors in a received word, and the next 4 message bits will be correctly decoded if there are 6 or fewer errors in a received word. The best single-level error-correcting cyclic code of length 35 which is capable of correcting 7 or fewer errors is a (35,4) code. The best single-level error-correcting cyclic code of length 35 which is capable of correcting 6 or fewer errors is a (35,7) code.

A short list of two-level UEP codes constructed based on the above method is given in Table 3, where the nonzeros

<table>
<thead>
<tr>
<th>n</th>
<th>k</th>
<th>$n_1$</th>
<th>$n_2$</th>
<th>$k_1$</th>
<th>$k_2$</th>
<th>$s_1$</th>
<th>$s_2$</th>
<th>nonzeros</th>
</tr>
</thead>
<tbody>
<tr>
<td>35</td>
<td>7</td>
<td>7</td>
<td>5</td>
<td>3</td>
<td>4</td>
<td>16</td>
<td>14</td>
<td>5, 7</td>
</tr>
<tr>
<td>51</td>
<td>10</td>
<td>3</td>
<td>17</td>
<td>2</td>
<td>8</td>
<td>22</td>
<td>18</td>
<td>3, 17</td>
</tr>
<tr>
<td>105</td>
<td>9</td>
<td>7</td>
<td>15</td>
<td>3</td>
<td>6</td>
<td>48</td>
<td>42</td>
<td>15, 21, 35</td>
</tr>
</tbody>
</table>

Table 3

Some Two-Level UEP Cyclic Codes
of a code are given by their exponents of $\alpha$, the $n$-th primitive root of unity.

The codes constructed based on the above methods are actually direct sums of cyclic repetition codes. Van Gils has constructed some two-level majority-logic decodable UEP cyclic codes which are direct sums of majority-logic decodable repetition codes [10]. Van Gils' codes form a subclass of the codes presented in this section.

In the above construction, if we choose $C_2$ as the $(n_1n_2,k_2+1)$ code with parity polynomial

$$h_2(X) = h_{22}(X),$$

then the direct-sum $C$ of $C_1$ and $C_2$ is an $(n_1n_2,k_1+k_2+1)$ code with parity polynomial

$$h(X) = h_{11}(X)h_{22}(X)/(X+1).$$

In this case, if $n_2d_{11} > n_1d_{22}$, $C$ is a cyclic code with separation vector $\overline{s} = (n_2d_{11}, n_1d_{22})$. The proof of this result is similar to the above one.

**Example 4:** In Example 3, if we choose $C_2$ as the $(35,5)$ code with parity polynomial $h_2(X) = h_{22}(X) = X^5+1$, then the direct sum $C$ of $C_1$ and $C_2$ is a $(35,8)$ cyclic code with parity polynomial

$$h(X) = (X^3+X+1)(X^5+1).$$

The separation vector of $C$ is $\overline{s} = (15,7)$. The best single-level triple-error-correcting code of length 35 is a $(35,8)$ code with minimum distance 7.
Consider the codes of length less than 63 which we have constructed in Example 3, 4 and Table 3. By taking $s_2$ as a lower bound on the minimum distance of the corresponding cyclic code, we see from [16] that this lower bound gives the true minimum distances of these codes.

VI. DECODING

In the following, we present a procedure for decoding a subclass of cyclic direct-sum codes of composite length with two-level error correcting capabilities. The decoding is based on the algebraic structure of codes developed in section II to IV. Consider two cyclic codes, $C_1$ and $C_2$, of composite length $n=n_1n_2$, where $n_1$ and $n_2$ are relatively prime. Assume that the sets, $\mathcal{J}_1$ and $\mathcal{J}_2$, defined by (24) are disjoint. Then, the parity polynomials, $h_1(X)$ and $h_2(X)$, of $C_1$ and $C_2$ are relatively prime. The direct sum $C$ of $C_1$ and $C_2$ has a separation vector $s = (s_1,s_2)$ with $s_1 \geq D_1d$ and $s_2 \geq D_2d_2$, if $D_1d \geq D_2d_2$. Let $A_1$ and $A_2$ be the component message spaces of $C_1$ and $C_2$ respectively. The decoding to be presented can correctly decode any message $x_1$ from $A_1$ if the number of transmission errors is at most $[\frac{(D_1d_1-1)}{2}]$ with $d_1 \leq 2$. Furthermore, the decoding can correctly decode any message $x_2$ from $A_2$ if the number of transmission errors is at most $[\frac{(D_2d_2-1)}{2}]$ with $d_2 \leq 2$.

A code polynomial $c(X)$ in $C$ is the sum of a code polynomial $c_1(X)$ in $C_1$ and a code polynomial $c_2(X)$ in $C_2$, i.e.

$$c(X) = c_1(X) + c_2(X).$$

For $j=1,2$, we express $c_j(X)$ in the following form:
\[ c_j(X) = \sum_{i=0}^{n_1 n_2 - 1} a_i^{(j)} x^i \]
\[ = \sum_{\mu=0}^{n_2-1} A^{(j)}(X) x^\mu \]  \hspace{1cm} (56)

where \( A^{(j)}(X) = \sum_{i=0}^{n_1-1} a_i \cdot n_2 + i \cdot n_2. \)  \hspace{1cm} (57)

Note that (56) and (57) are simply the expressions of (2) and (3). Express \( c(X) \) in the following form:
\[ c(X) = \sum_{i=0}^{n_1 n_2 - 1} a_i x^i \]
\[ = \sum_{\mu=0}^{n_2-1} A_\mu(X) x^\mu \]  \hspace{1cm} (58)

where \( A_\mu(X) = \sum_{i=0}^{n_1-1} a_i \cdot n_2 + i \cdot n_2. \)  \hspace{1cm} (59)

Then, it follows from (56) that
\[ A_\mu(X) = A^{(1)}_\mu(X) + A^{(2)}_\mu(X) \]  \hspace{1cm} (60)

for \( \mu = 0, 1, \ldots, n_2-1. \)

Suppose that \( m \in J_1. \) Since \( J_1 \) and \( J_2 \) are disjoint, then \( m \) must be an integer in \( J_2. \) It follows from Lemma 1 and (23) that
\[ A^{(2)}_\mu(\gamma^m) = 0 \]
and
\[ A_\mu(\gamma^m) = A^{(1)}_\mu(\gamma^m) + A^{(2)}_\mu(\gamma^m) \]
\[ = A^{(1)}_\mu(\gamma^m) \]  \hspace{1cm} (61)

for \( m \in J_1 \) and \( \mu = 0, 1, \ldots, n_2-1. \) Recall that
\[ a^{(m)}_i(X) = \sum_{\mu=0}^{n_1-1} A^{(1)}_\mu(\gamma^m) \gamma^m x^\mu \]  \hspace{1cm} (62)
is a code polynomial in the code $V^{(m)}_1$ defined by (25). Suppose a code polynomial $c(X)$ is transmitted. Let $r(X)$ and $e(X)$ be the received and error polynomial respectively. Then,

$$r(X) = c(X) + e(X)$$

We express $r(X)$ and $e(X)$ in the following forms:

$$r(X) = \sum_{i=0}^{n_1 n_2 - 1} r_i X^i$$

$$= \sum_{\mu=0}^{n_2-1} R_\mu(X) X^\mu,$$  

$$e(X) = \sum_{i=0}^{n_1 n_2 - 1} e_i X^i$$

$$= \sum_{\mu=0}^{n_2-1} E_\mu(X) X^\mu,$$  

where

$$R_\mu(X) = \sum_{i=0}^{n_1-1} r_{i \cdot n_2 + \mu} X^{i \cdot n_2}$$

and

$$E_\mu(X) = \sum_{i=0}^{n_1-1} e_{i \cdot n_2 + \mu} X^{i \cdot n_2}.$$  

It follows from (63) that

$$R_\mu(X) = A_\mu(X) + E_\mu(X)$$

for $\mu = 0, 1, \ldots, n_2-1$. Clearly, for $m \in \bar{J}_1$ and $0 \leq \mu < n_2$, we have

$$R_\mu(\gamma^m) = A_\mu(\gamma^m) + E_\mu(\gamma^m)$$

$$= A^{(1)}(\gamma^m) + E_\mu(\gamma^m).$$

Suppose that $m \in \bar{J}_2$. We can easily show that

$$A^{(1)}(\gamma^m) = 0,$$
and \[ A_{\mu}(\gamma^m) = A_{\mu}^{(2)}(\gamma^m) \tag{70} \]

for \( m \in \mathcal{J}_2 \) and \( \mu = 0,1,\ldots,n_2-1 \). Let

\[
\begin{align*}
\tilde{r}'(X) &= r(X) - c_1(X) \\
&= \sum_{i=0}^{n_1n_2-1} r'_i X^i \\
&= \sum_{\mu=0}^{n_2-1} R_{\mu}(X) X^\mu, \tag{71}
\end{align*}
\]

where \( R_{\mu}(X) = \sum_{i=0}^{n_1-1} r'_i n_2 + \mu X^i n_2. \tag{72} \)

From (57), (64) and (71), we readily see that

\[ R_{\mu}(X) = R_{\mu}(X) - A_{\mu}^{(1)}(X). \tag{73} \]

It follows from (68), (70) and (73) that

\[ R_{\mu}(\gamma^m) = R_{\mu}(\gamma^m) = A_{\mu}^{(2)}(\gamma^m) + E_{\mu}(\gamma^m) \tag{74} \]

for \( m \in \mathcal{J}_2 \) and \( \mu = 0,1,\ldots,n_2-1 \). The set,

\[ \{ R_{\mu}(\gamma^m) : 0 \leq m < n_1 \text{ and } 0 \leq \mu < n_2 \} \]

is the syndrome of \( r(X) \), and will be used for decoding \( r(X) \).

For \( m \in \mathcal{J}_1 \), multiplying both sides of (69) by \( \gamma^m X^\mu \) and summing over \( \mu \), we have

\[ r(m)(X) = a_1^{(m)}(X) + e(m)(X) \tag{75} \]

where \( a_1^{(m)}(X) \) is given by (62) and

\[ r(m)(X) = \sum_{\mu=0}^{n_2-1} R_{\mu}(\gamma^m) \gamma^{m\mu} X^\mu \tag{76} \]

\[ e(m)(X) = \sum_{\mu=0}^{n_2-1} E_{\mu}(\gamma^m) \gamma^{m\mu} X^\mu. \tag{77} \]

For \( m \in \mathcal{J}_2 \), multiplying both sides of (74) by \( \gamma^m X^\mu \) and summing
over $\mu$, we have
\[
r'(m)(X) = a_2^{(m)}(X) + e^{(m)}(X) \tag{78}
\]
where
\[
r'(m)(X) = \sum_{\mu=0}^{n_2-1} R^{(m)}_\mu (\gamma^m) \gamma^{m\mu} X^\mu, \tag{79}
\]
and
\[
a_2^{(m)}(X) = \sum_{\mu=0}^{n_2-1} A^{(2)}_\mu (\gamma^m) \gamma^{m\mu} X^\mu. \tag{80}
\]

Note that, for $m \in J_1$, if $e(X)=0$, $r^{(m)}(X) = a_1^{(m)}(X)$ and is a code polynomial in $V_1^{(m)}$. Also note that, for $m \in J_2$, if $e(X)=0$, $r^{(m)}(X) = a_2^{(m)}(X)$ and is a code polynomial in $V_2^{(m)}$.

The decoding consists of two stages. First $r(X)$ is decoded into $c_1(X)$ and then $r'(X) = r(X) - c_1(X)$ is decoded into $c_2(X)$. At the first stage, we decode $r^{(m)}(X)$ into $a_1^{(m)}(X)$ which depends on $D_1$ and $d$, where $D_1$ is given by (28) and $d$ is the minimum distance of $W$ given by (32). After $a_1^{(m)}(X)$ is decoded, we can uniquely determine $A^{(1)}_\mu(X)$ from $(A^{(1)}_\mu(\gamma^m) : m \in J_1)$ for $\mu = 0, 1, \ldots, n_2-1$ (see Appendix B). Then, $c_1(X)$ is correctly recovered. At the following stage, we similarly decode $r'(m)(X)$ into $a_2^{(m)}(X)$ which depends on $D_2$ and $d_2$, where $D_2$ is given by (28) and $d_2$ is the minimum distance of $W_2$ given by (31). Then, $A^{(2)}_\mu(X)$, $\mu=0, 1, \ldots, n_2-1$, and $c_2(X)$ can be recovered.

There are two cases to be considered in decoding $r(X)$ into $c_1(X)$.

Case I

Suppose that $d = 1$. For this case, $s_1 = D_1$. The decoding of $r(X)$ into $c_1(X)$ consists of the following steps:

(1) For any $m \in J_1$, we decode the received word,
\( \overline{R}(m) = (R_0(\gamma^m), R_1(\gamma^m), \ldots, R_{n_2-1}(\gamma^m)\gamma^{m(n_2-1)}) \),

into a codeword,

\[
*\overline{A}(m) = (*A_0^{(1)}(\gamma^m), *A_1^{(1)}(\gamma^m), \ldots, *A_{n_2-1}^{(1)}(\gamma^m)\gamma^{m(n_2-1)}) , \tag{81}
\]

in \( V_1^{(m)} \) based on a certain decoding algorithm for \( V_1^{(m)} \). The codeword \(*\overline{A}(m)\) is the estimate of the real codeword,

\[
(A_0^{(1)}(\gamma^m), A_1^{(1)}(\gamma^m)\gamma^m, \ldots, A_{n_2-1}^{(1)}(\gamma^m)\gamma^{m(n_2-1)}) .
\]

(2) For any \( m \in J_1 \) and \( 0 \leq \mu < n_2 \), we set \(*A_{\mu}^{(1)}(\gamma^m) = 0\).

(3) For \( 0 \leq \mu < n_2 \) and \( 0 \leq m < n_1 \), find a codeword

\[
(*a_{\mu}^{(1)}, *a_{n_2+\mu}^{(1)}, \ldots, *a_{(n_1-1)n_2+\mu})
\]

in \( W \) such that

\[
\sum_{i=0}^{n_1-1} *a_{i,n_2+\mu}(\gamma^m)i.n_2 = *A_{\mu}^{(1)}(\gamma^m).
\]

Then the estimate for \( c_1(X) \) is

\[
*c_1(X) = \sum_{i=0}^{n_1n_2-1} *a_{i}^{(1)}X^i .
\]

Now we need to show that if the number of errors in \( e(X) \) is \([ (D_1-1)/2 ] \) or less, the above decoding results in the correct code polynomial \( c_1(X) \). Suppose \( e(X) \) contains \([ (D_1-1)/2 ] \) or fewer errors. From (65) and (67), we see that there are at most \([ (D_1-1)/2 ] \) \( E_{\mu}(X) \)'s which are nonzero. Then from (77), we see that the error polynomial \( e^{(m)}(X) \) contains at most \([ (D_1-1)/2 ] \) errors. Recall that the minimum distance of \( V_1^{(m)} \) is \( d_1^{(m)} \). From (25)
and (27), we see that $V_{1}^{(m)}$ is capable of correcting $[(D_1-1)/2]$ or fewer errors. As a result, the first step of the above decoding procedure gives the correct $a_{1}^{(m)}(X)$ for $m \in \mathcal{J}_{1}$. Once all $a_{1}^{(m)}(X)$'s for $0 \leq m < n_1$ have been determined, step 3 gives a unique solution $c_{1}(X)$ [see appendix B].

Case II

Suppose that the minimum distance $d$ of $W$ is 2. Since $W$ is a binary cyclic code, $W$ has "1" as its zero. Therefore $W$ is an even-weight code. This implies that, for $0 \leq m < n_2$, $A_{\mu}(X)$ has even weight. The procedure for decoding $r(X)$ into $c_{1}(X)$ consists of the following steps:

1. For $0 \leq \mu < n_2$, compute the modulo-2 sum of the coefficients of $R_{\mu}(X)$. If the sum is not zero, then $R_{\mu}(X)$ contains errors and $E_{\mu}(X) \neq 0$. We say that $R_{\mu}(X)$ is detected in error. In this case, we assume that

$$R_{\mu}(\gamma^{m}) = A_{\mu}^{(1)}(\gamma^{m})$$

for $m \in \mathcal{J}_{1}$. In decoding the word

$$\bar{R}(m) = (R_{0}(\gamma^{m}), R_{1}(\gamma^{m})\gamma^{m}, \ldots, R_{n_{2}-1}(\gamma^{m})\gamma^{m(n_{2}-1)})$$

(83)

if $R_{\mu}(X)$ is detected in error, the component $R_{\mu}(\gamma^{m})\gamma^{m}$ is removed to create an erasure. Hence $\bar{R}(m)$ may contain symbol errors and erasures.

2. For $m \in \mathcal{J}_{1}$, we decode $\bar{R}(m)$ into a codeword,

$$(*A_{0}^{(1)}(\gamma^{m}), *A_{1}^{(1)}(\gamma^{m})\gamma^{m}, \ldots, *A_{n_{2}-1}^{(1)}(\gamma^{m})\gamma^{m(n_{2}-1)})$$

in $V_{1}^{(m)}$ based on a certain decoding algorithm which is capable of handling both symbol errors and erasures.
(3) For $m \in J_1$ and $0 \leq \mu < n_2$, we set $A^{(1)}_{\mu}(\gamma^m) = 0$.

(4) For $0 \leq \mu < n_2$ and $0 \leq m < n_1$, find a codeword,

$(*a^{(1)}_\mu, a^{(1)}_{n_2^{-1}+\mu}, \ldots, a^{(1)}_{(n_1-1)n_2^{-1}+\mu})$,

in $W_1$ such that

$$
\sum_{i=0}^{n_1^{-1}} a^{(1)}_i \cdot n_2^{-1} + \mu(\gamma^m)^i \cdot n_2 = A^{(1)}_{\mu}(\gamma^m).
$$

Then the estimate for $c_1(x)$ is

$$
\star c_1(x) = \sum_{i=0}^{n_1^{-1}n_2^{-1}-1} a^{(1)}_i x^i.
$$

For $d = 2$, the direct sum code $C$ has a separation vector with $s_1 = 2D_1$. Now we want to show that, if there are no more than $[(2D_1-1)/2] = D_1-1$ errors in the error polynomial $e(x)$, the above decoding procedure gives the correct estimate of $c_1(x)$. Suppose there are no more than $D_1-1$ errors in $e(x)$. Let $f$ be the number of erasures in $\overline{R}(m)$. In the worst case, each of these erasure contains a single error from $e(x)$. Then there are at most

$$
t = \left\lfloor \frac{D_1-1-f}{2} \right\rfloor
$$

undetected error symbols in $\overline{R}(m)$, each contains even number of errors from $e(x)$. Since

$$
f + 2 \left\lfloor \frac{D_1-1-f}{2} \right\rfloor < D_1 \leq d_1^{(m)},
$$

the erasures and the symbol errors will be corrected at step 2. As a result, step 4 yields the correct code polynomial $c_1(x)$. 

36
Once $c_1(X)$ has been determined, we start to decode
$r'(X) = r(X) - c_1(X)$ into $c_2(X)$. As we mentioned earlier, the
decoding of $r'(X)$ into $c_2(X)$ depends on the minimum distance $d_2$ of
$W_2$. Therefore, two cases, (I) $d_2 = 1$, (II), $d_2 = 2$, need to be
considered. To decode $r'(X)$ into $c_2(X)$, we simply follow the
procedure for decoding $r(X)$ into $c_1(X)$ if we replace $r(X)$ by
$r'(X)$, $c_1(X)$ by $c_2(X)$, $J$ and $J_1$ by $J_2$, $J$ and $\bar{J}_1$ by $\bar{J}_2$, $R_\mu(X)$
by $R_\mu'(X)$, $A_\mu^{(1)}(X)$ by $A_\mu^{(2)}(X)$, $V_1^{(m)}$ by $V_2^{(m)}$, $W$ and $W_1$ by $W_2$,
$D_1$ by $D_2$, $d$ by $d_2$, and $s_1$ by $s_2$.

VII. BURST-ERROR-CORRECTION CAPABILITIES OF
CYCLIC DIRECT-SUM CODES

So far, we have studied the random error correcting
capabilities of cyclic codes through their separation vectors.
In this section, we shall see that, under some conditions, the
cyclic codes given in section IV have multi-level burst error
correcting capabilities in addition to the random error
correcting capabilities specified by their separation vectors.

Let $C$ be the direct sum of two cyclic codes, $C_1$ and $C_2$, of
composite length $n = n_1 n_2$ where $n_1$ and $n_2$ are relatively prime.
Assume that, the sets, $\bar{J}_1$ and $\bar{J}_2$, defined by (24) are disjoint.
The code $C$ has a separation vector $\bar{s}$ at least $(D_1 d, D_2 d_2)$ if
$D_1 d > D_2 d_2$. A code polynomial $c(X)$ in $C$ is the sum of a code
polynomial $c_1(X)$ in $C_1$ and a code polynomial $c_2(X)$ in $C_2$, i.e.

$$c(X) = c_1(X) + c_2(X).$$

Recall that, in section VI, the decoding of $c_j(X)$ relies on the
correct recovery of $A_\mu^{(j)}(X)$ for $\mu = 0, 1, \ldots, n_2 - 1$, where $j = 1, 2$
and \( A_{\mu}^{(j)}(X) \) is given by (57). Now we arrange the coefficients of \( c(X) \) in an \( n_1 \times n_2 \) code array as shown in Figure 1. Note that the \( \mu \)-th column of the code array for \( c(X) \) is simply the \( n_1 \)-tuple representation of \( A_{\mu}(X) \), which is given by (59). Clearly, the coefficients of \( c_{j}(X) \) can also be arranged as an \( n_1 \times n_2 \) code array for which the \( \mu \)-th column is the \( n_1 \)-tuple representation of \( A_{\mu}^{(j)}(X) \), where \( j=1, 2 \). Suppose \( c(X) \) is transmitted column by column. Then, the coefficients for the received and error polynomials, \( r(X) \) and \( e(X) \) can also be arranged as \( n_1 \times n_2 \) arrays. The \( \mu \)-th column of the \( n_1 \times n_2 \) array for \( e(X) \) is the \( n_1 \)-tuple representation of \( E_{\mu}(X) \) and the \( \mu \)-th column of the \( n_1 \times n_2 \) array for \( r(X) \) is the \( n_1 \)-tuple representation of \( R_{\mu}(X) \). It is easy to see that all the arguments in section VI are still valid.

Consider case I of decoding \( r(X) \) into \( c_1(X) \), which is given in section VI. Recall that \( d = 1 \) in this case. Suppose that the \( n_1 \times n_2 \) array associated to \( e(X) \) has no greater than \( [(D_1-1)/2] \) nonzero column. Clearly, there are at most \( [(D_1-1)/2] \) nonzero \( E_{\mu}(X) \)'s in \( e(X) \). As a result, \( a_1^{(m)}(X) \) for \( m \in \overline{J}_1 \) can be correctly decoded at step 1. Then, \( c_1(X) \) can be correctly decoded at step 3. The correctable error patterns for decoding \( r(X) \) into \( c_1(X) \) with \( d = 1 \) includes the following categories:

1. Any error pattern containing at most \( [(D_1-1)/2] \) random errors.
2. Any error burst of length up to \( [(D_1-1)/2] - 1 \) \( n_1 + 1 \).
3. Any multiple error bursts which affects no more than \( [(D_1-1)/2] \) columns in the \( n_1 \times n_2 \) array associated to...
Once $c_1(X)$ is recovered, the component message corresponding to $c_1(X)$ can be determined. Thus, we have the following result:

If $d=1$, the component message from the component message space of $C_1$ is protected against up to \([((D_1-1)/2)]\) random errors and any error burst of length up to \(((D_1-1)/2)-1\cdot n_1+1\).

Similarly, we can have the following result from decoding $r'(X)$ into $c_2(X)$:

If $d_2=1$, the component message from the component message space of $C_2$ is protected against up to \([((D_2-1)/2)]\) random errors and any error burst of length up to \(((D_2-1)/2)-1\cdot n_1+1\).

Consider case II of decoding $r(X)$ into $c_1(X)$ which is given in section VI. Note that $d=2$ in this case. Suppose the error pattern contains $D_1-1$ random errors. It has been shown in section VI that $c_1(X)$ can be recovered at step 4. Suppose the error pattern is an error burst of length at most \(((D_1-1)/2)-1\cdot n_1+2\). In the worst case, there are \(((D_1-1)/2)+1\) nonzero columns in the $n_1 \times n_2$ array associated to $e(X)$ with at least two columns containing only one nonzero component. Suppose that there are $f$ columns containing only one nonzero components in the $n_1 \times n_2$ array associated to $e(X)$ where $f \geq 2$. Thus, the $f$ corresponding $R_{\mu}(X)$'s are detected to be in error at step 1. Then, $R^{(m)}$ which is given by (83) contains $f$ erasures and at most \(((D_1-1)/2)+1-f\) undetected symbol errors. Since \(((D_1-1)/2)+1-f\cdot 2+f < D_1\) for $f \geq 2$, the erasures and the symbol errors will be corrected at step 2. Thus, $c_1(X)$ can be correctly decoded at step 4. Then, we have the following result:
If \( d=2 \), the component message from the component message space of \( C_1 \) is protected against up to \( D_1-1 \) random errors and any error burst of length up to \( \left\lfloor \frac{(D_1-1)/2}{2} \right\rfloor -1 \cdot n_1+2 \).

Similarly, we can obtain the following result from decoding \( r'(X) \) into \( c_2(X) \):

If \( d_2=2 \), the component message from the component message space of \( C_2 \) is protected against up to \( D_2-1 \) random errors and any error burst of length up to \( \left\lfloor \frac{(D_2-1)/2}{2} \right\rfloor -1 \cdot n_1+2 \).

Now we consider the \((51,34)\) code given in Example 2. We see that the first 18 message bits are protected against up to 3 random errors and any error burst of length up to 7; while the next 16 message bits are protected against up to 2 random errors. For the \((51,19)\) code given in Table 2, we see that the first bit is protected against up to 8 random errors and any error burst of length up to 22; while the next 18 bits are protected against up to 6 random errors and any error burst of length up to 8.

There exist unequal error protection codes for which all the component messages are equally protected against random errors but not equally protected against burst errors. An example is given as follows.

**Example 5:** Let \( n_1=7 \) and \( n_2=9 \). Let \( \alpha \) be a primitive element of \( \text{GF}(2^6) \). Let \( \beta=\alpha^7 \) and \( \gamma=\alpha^9 \). Table 4 is a 7x9 array with 63 nonnegative integers from 0 to 62. A number \( \rho \) in the array represents the field element \( \alpha^\rho \). If \( \rho \) is at the \( m \)-th row and the \( \ell \)-th column of the array, then the element \( \alpha^\rho \) is the product of \( \gamma^m \) and \( \beta^\ell \), i.e.
\[ \alpha^p = \beta^m \gamma^p. \]

### Table 4

<table>
<thead>
<tr>
<th>Nonzeros of a (63,24) Binary Cyclic Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>0  7  14  21  28  35  42  49  56</td>
</tr>
<tr>
<td>9  16  23  30  37  44  51  58  2</td>
</tr>
<tr>
<td>18 25  32  39  46  53  60  4  11</td>
</tr>
<tr>
<td>27 34  41* 48* 55* 62* 6* 13* 20</td>
</tr>
<tr>
<td>36 43  50  57  1  8  15  22  29</td>
</tr>
<tr>
<td>45 52* 59* 3* 10  17  24* 31* 38*</td>
</tr>
<tr>
<td>54 61* 5 12* 19* 26* 33* 40 47*</td>
</tr>
</tbody>
</table>

Let \( C_1 \) be an (63,6) binary cyclic code whose nonzeros are specified by the underlined numbers in Table 4. Let \( C_2 \) be an (63,18) binary cyclic code whose nonzeros are specified by numbers with * in Table 4. For example, \( a^{11} \) is a nonzero of \( C_1 \) and \( a^3 \) is a nonzero of \( C_2 \). Clearly, \( C_1 \) and \( C_2 \) have no nonzeros in common. Let \( C \) be the direct sum of \( C_1 \) and \( C_2 \) which is a (63,24) code. From Table 4, (22) and (23), we see that \( J_1 = \{0,3,5,6\} \) and \( J_2 = \{0,1,2,4\} \). Then \( J_1 \) and \( J_2 \) are disjoint.

From Table 4, we see that \( V_1^{(1)} \) has \( \beta^0, \beta^1, \beta^2, \beta^3, \beta^{-1}, \beta^{-2}, \beta^{-3} \) as zeros. Thus, the minimum distance \( d_1^{(1)} \) of \( V_1^{(1)} \) is at least 8.

It is easy to check that \( V_1^{(1)}, V_1^{(2)}, V_1^{(4)} \) are equivalent.
Hence, the minimum distances \( d_1^{(1)}, d_1^{(2)}, \) and \( d_1^{(4)} \) of \( V_1^{(1)}, V_1^{(2)}, \) and \( V_1^{(4)} \) are identical. From (27), we have \( D_1 \geq 8 \).

Since \( J = J_1 \cap J_2 = \{0\} \), \( W \) has only one zero which is \( \gamma^0 = 1 \).
minimum distance $d$ of $W$ is at least 2. From Table 4, we see that $v_2^{(3)}$ has $\beta^{-1}, \beta^0$ and $\beta^1$ as zeros. Thus, the minimum distance $d_2^{(3)}$ of $v_2^{(3)}$ is at least 4. We can easily check that $v_2^{(3)}$, $v_2^{(5)}$, and $v_2^{(6)}$ are equivalent. Hence, the minimum distances $d_2^{(3)}$, $d_2^{(5)}$, and $d_2^{(6)}$ of $v_2^{(3)}$, $v_2^{(5)}$, and $v_2^{(6)}$ are identical.

From (28), we have $D_2 \geq 4$. Since $J_2 = \{0, 1, 2, 4\}$, $W_2$ has $\gamma^0$, $\gamma^9 = \gamma^2$, $\gamma^{18} = \gamma^4$, and $\gamma^{36} = \gamma$ as zeros. By BCH bound, we see that the minimum distance $d_2$ of $W_2$ is at least 4. Note that $D_2 d > 16$ and $D_2 d_2 > 16$. Thus, $C$ is a $(63, 24)$ code for the product message space $A = A_1 \times A_2$ with separation vector $\bar{s} = (s_1, s_2)$, where $A_1 = \{0, 1\}^6$, $A_2 = \{0, 1\}^{18}$, $s_1 \geq 16$ and $s_2 \geq 16$. Since $d = 2$, we see that the first 6 message bits of a message are protected against up to 7 random errors and any error burst of length up to 16. However, the next 18 message bits are only protected against 7 random errors or less.

For comparison, we see that the $(63, 24)$ primitive BCH code can correct 7 random errors or less.
APPENDIX A

The Unique Expression of $\alpha^p$ as $\beta^l \gamma^m$

In this appendix, we shall prove that $\alpha^p$, for $0 \leq p < n$, can be uniquely expressed as the product of $\beta^l \gamma^m$ as given by (4), where $n = n_1 n_2$, $0 \leq l < n_2$, $0 \leq m < n_1$, and $n_1$, $n_2$ are relatively prime. Note that $\alpha$ is a primitive $n$-th root of unity, $\beta = \alpha^{n_1}$, and $\gamma = \alpha^{n_2}$.

First, we show the existence. Since $n_1$ and $n_2$ are relatively prime, there exist integers $a$ and $b$ such that

$$an_1 + bn_2 = p.$$

Clearly,

$$\alpha^p = \alpha^{an_1 + bn_2} = (\alpha^{n_1})^a (\alpha^{n_2})^b = \beta^l \gamma^m.$$

Let $l = a \mod n_2$ and $m = b \mod n_1$. Then

$$\alpha^p = \beta^l \gamma^m, \quad (A-1)$$

where $0 \leq l < n_2$ and $0 \leq m < n_1$.

Next, we show the uniqueness. Assume that

$$\alpha^p = \beta^{l'} \gamma^{m'} = \beta^{l''} \gamma^{m''}, \quad (A-2)$$

where $0 \leq l, l' < n_2$ and $0 \leq m, m' < n_1$. The condition (A-2) implies

$$\beta^{l-l'} \gamma^{m-m'} = 1,$$

or equivalently

$$\beta^{l'-l} = \gamma^{m-m'} \quad (A-3)$$

where $-n_2 < l'-l < n_2$ and $-n_1 < m-m' < n_1$.

The equation (A-3) implies $l = l'$ and $m = m'$, since

$$\{ \beta^l : l \text{ is an integer } \} \cap \{ \gamma^m : m \text{ is an integer } \} = \{1\}.$$

Thus, the expression (A-1) is unique under the condition that $0 \leq l < n_2$ and $0 \leq m < n_1$. 

43
APPENDIX B

The Recovery of $A^{(1)}_\mu(X)$

In this appendix, we shall show that $A^{(1)}_\mu(X)$ can be recovered from the set $(A^{(1)}_\mu(\gamma^m) : m \in \bar{J}_1)$ as stated in section VI, where $\mu = 0, 1, \ldots, n_2-1$.

It follows from (57) that the coefficients of $A^{(1)}_\mu(X)$ form the $n_1$-tuple

$$(a^{(1)}_\mu, a^{(1)}_{n_2+\mu}, \ldots, a^{(1)}_{(n_1-1)n_2+\mu})$$

which is a codeword of the binary cyclic code $W_1$ defined by (30).

Note that

$$A^{(1)}_\mu(\gamma^m) = \sum_{i=0}^{n_1-1} a^{(1)}_i n_2 + \mu (\gamma^m)^i n_2$$

Also note that $\gamma^{m_2^2}$, $m \in \bar{J}_1$ are nonzeros of $W_1$, where $\bar{J}_1$ is defined by (24). From the following lemma, we can easily see that $A^{(1)}_\mu(X)$ is uniquely determined by the set $(A^{(1)}_\mu(\gamma^m) : m \in \bar{J}_1)$.

**Lemma B-1:** Consider an $(n,k)$ binary cyclic code $V$ which has $a^{m_1}$, $a^{m_2}$, ..., $a^{m_k}$ as all its nonzeros, where $a$ is a primitive $n$-th root of unity. Let $v_1(X)$ and $v_2(X)$ be code polynomials of $V$. If $v_1(a^{m_i}) = v_2(a^{m_i})$ for $i=1,2,\ldots,k$, then $v_1(X) = v_2(X)$.

**Proof:** Let $v(X) = v_1(X) + v_2(X)$, which is also a code polynomial of $V$. For $i=1,2,\ldots,k$, $v_1(a^{m_i}) = v_2(a^{m_i})$ implies $v(a^{m_i}) = 0$. Combining the fact that $v(a^i) = 0$ for $i \in \{0,1,\ldots,n-1\} - (m_1, m_2, \ldots, m_k)$, we see that $v(a^i) = 0$ for $i=0,1,\ldots,n-1$. Note that $v(X)$ has degree at most $n-1$ which implies that a nonzero $v(X)$ has at most $n-1$ distinct roots. Thus, $v(X) = 0$. This implies that $v_1(X) = v_2(X)$. 

44
REFERENCES


13. L.A. Bassalygo, et.al. "Bounds for Codes with Unequal


