Order Number 9230508

Multi-level error correcting codes

Morelos-Zaragoza, Robert Henry, Ph.D.

University of Hawaii, 1992

U·M·I

300 N. Zeeb Rd.
Ann Arbor, MI 48106

MULTI-LEVEL ERROR CORRECTING CODES

A DISSERTATION SUBMITTED TO THE GRADUATE DIVISION OF THE UNIVERSITY OF HAWAII IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

IN

ELECTRICAL ENGINEERING

MAY 1992

By

Robert Henry Morelos-Zaragoza

Dissertation Committee:

Shu Lin, Chairperson

E. Bertram

J. Aslanis

N.T. Gaarder

K. Zeger

Acknowledgements

I wish to thank my advisor Professor Shu Lin for his invaluable help and guidance during the course of my thesis. His encouragement throughout my stay at the University of Hawaii is deeply appreciated. Thanks are also due to the other members of my committee, Professors Edward Bertram, James Aslanis, N. Thomas Gaarder and Kenneth Zeger. In particular I thank Professor Edward Bertram for many fruitful discussions. Thanks are also due to Professor Aslanis, whose suggestions improved the presentation of mi research work and results. Many discussions with Professors Tadao Kasami and Toru Fujiwara from Osaka University helped substantially as well. I would also like to thank my friends for their encouragement and support throughout my stay at the University of Hawaii. Lastly, I thank Naoko Tawara, my many friends in Mexico, and my family for their emotional support.

Abstract

This dissertation investigates algebraic block codes with multiple levels of error protection on the information messages, also known as unequal-error-protection (UEP) codes. New constructions of multi-level error correcting linear block codes specified by their generator matrices and by their parity-check matrices are presented and discussed. A family of nonbinary optimal LUEP codes, capable of correcting any t or less random errors affecting the most significant information symbols, while correcting any single random error in the least significant information symbols, is constructed and analyzed. This class of codes is constructed by specifying their parity check matrices, which are combinations of parity check matrices of Reed-Solomon codes and shortened nonbinary Hamming codes. Near optimal binary LUEP codes, constructed by appending (also known as time-sharing) cosets of subcodes of shorter linear Hamming codes, are proposed and analyzed. This class of LUEP codes is specified by their generator matrices. By computing the Hamming bound on binary LUEP codes with the same dimension and error correcting capabilities as the optimal binary LUEP codes, it is shown that this family of LUEP codes requires at most one additional redundant bit. Furthermore, constructions of LUEP codes by specifying their generator matrices allow the use of multi-stage decoding methods, with reduced decoding complexity, as shown in this work. Certain families of known linear block codes are analyzed to determine their UEP capabilities. We derive conditions for which some shortened Hamming codes, some nonprimitive BCH codes and some cyclic codes of composite length are LUEP codes. Bounds on the multi-level error correcting capabilities of these families of linear codes are derived. The error performance of some binary multi-level error-correcting codes

over binary symmetric channels and over additive white Gaussian noise channels with BPSK modulation is considered. Some new block coded modulation (BCM) systems using LUEP codes as component codes are presented. It is shown that these schemes offer multiple values of minimum squared Euclidean distances, one for each message part to be protected. In addition, the error performance of a BCM system using LUEP codes and QPSK modulation with Gray mapping between 2-bit symbols and signals, is analyzed. In this dissertation, we also present a new multi-stage soft-decision decoding, based on the trellis structure of LUEP codes.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

There are many practical applications in which it is required to design a code that protects messages against different levels of noise, or messages with different levels of importance over a noisy channel with the same level of noise power. Examples of such situations are: broadcast channels, multi-user channels, computer networks, pulse coded modulation (PCM) systems and source coding systems, among others. This doctoral dissertation studies the problem of designing good multi-level error correcting codes for situations, in which (1) an information source puts out messages with *different degrees of significance* (e.g., PCM and other source coding systems), or (2) messages are equally important, but are transmitted through a channel with *different levels of noise power* (e.g., Broadcast channels or multi-user channels). This work focuses on *linear block* codes that offer multiple levels of error correction.

As an illustrative example, consider the transmission of information in a computer network. Messages are transmitted in packets or blocks. In each packet, there will be a header part, followed by a block of data bits. The header part contains various control bits used to determine destination and origin of packet, priority, and other important information. This header part must be protected against a larger number of errors than the message part, since if lost, the whole packet could be lost. One solution would be to use two separate encoders, one using an error correcting code to encode the data bits, and another using a more powerful error correcting code to encode the packet header. This solution is called a *time-sharing coding*

scheme. In a time-sharing scheme, say of two codes C_1 and C_2, each codeword from C_1 is appended to (or followed by) a codeword from C_2. Time-sharing of two codes is usually denoted by $|C_1|C_2|$ or by $|\bar{\mathbf{u}}|\bar{\mathbf{v}}|$.

A similar situation occurs when a message consists of equally important message parts, but is transmitted over a channel with different levels of noise, e.g., a broadcast channel. Such a channel can be modeled as consisting of a single source transmitting information through ℓ parallel subchannels, one for each destination. In this case, more protection must be provided to the message part that is transmitted through the subchannel with the highest amount of noise level, less protection to the message part to be transmitted through the subchannel affected by the next highest noise level, and so on, until the message part that is transmitted through the subchannel with the smallest level of noise. Here again, one solution is to use ℓ different error correcting codes, each capable of correcting most of the errors induced in the particular subchannel.

Unfortunately, the time-sharing coding scheme is very inefficient, in terms of achievable rates (or, equivalently, redundancy), as has been shown by Cover [8]. An alternative solution is to consider all message parts as most important. In this case, however, more redundancy is required because the single-level error correcting code utilized will need to correct more error patterns than a multi-level error correcting code. A more efficient solution is to use a *single code* that provides the different levels of error correction required by either messages consisting of parts with different levels of significance or by channels with different levels of noise. In [8], such a coding scheme was called *information superposition.* The basic idea is to cluster codewords corresponding to the less significant message parts into *clouds*, and to associate each cloud with the most important message parts. This idea, and the basic concepts of unequal error correction, will be discussed in detail in chapter 2. For instance, going

back to the example of packet transmission in computer networks, a single two-level error correcting code can be used to encode the entire packet, protecting the header part against more errors than the data part. This code will require fewer redundant bits than its time-sharing counterpart.

A multi-level error correcting code is usually referred to as an *unequal-error-protection* (UEP) code. In this dissertation, the terms UEP code or *multi-level error correcting code* are used interchangeably. The objective of this dissertation is to present *new constructive methods* that yield families of *efficient* multi-level error-correcting codes. Issues dealing with how to measure the efficiency of a UEP code will be discussed. In addition, applications of UEP codes in conjunction with proper modulation systems, to obtain efficient communication systems for band-limited channels, are investigated.

1.1 Previous work

Linear unequal error protection (LUEP) codes were first introduced by Masnick and Wolf [34]. They discussed linear codes, specified by their parity check matrices, providing a level of error correction beyond that provided by the minimum distance of the code, for some *codeword* positions. Two construction methods were used: *matrix overlapping* and *basis method.* They also derived Hamming and Varshamov-Gilbert bounds for UEP codes based on their construction methods. Gore and Kilgus [16] introduced an example $(15, 9)$ binary cyclic code with minimum distance 4 that can correct one information bit against the occurrence of 2 errors. That is, the most significant bit can always be recovered in the presence of any two errors in a received vector. They showed an important result in the theory of LUEP codes: that systematic cyclic codes provide *equal* error protection for every information digit. The proof of this is that, if it is possible to correct a given number of errors

and decode a particular message position, then there exists a decoding procedure that will recover every position in a codeword correctly, when the same number of errors are present in a received word.

Using random coding arguments, Cover [8] has shown that there exists a coding scheme that will exceed the rate of the trivial time-sharing scheme, for the class of binary symmetric broadcast channels for two users. In this paper, the important concept of clustering codewords of a UEP code into *clouds* was introduced for the first time. It is interesting to note that if a UEP code is linear, then these clouds of codewords are *cosets* of a subcode. In this sense, it was also the first time that the idea *coset codes* [13] was introduced, though not in this terminology. More on this will be discussed in chapter 2.

Mandelbaum [33] and Kilgus and Gore [24] investigated majority logic decodable LUEP codes. Mandelbaum presented constructions based on specifying the parity-check equations of the code, resulting in LUEP codes of rate 1/2. Kilgus and Gore presented a method based on the generator matrix, to construct simple binary non-systematic cyclic LUEP codes that protect only one message position against more errors than the minimum error correcting capability of the code. These two classes of codes were later generalized by Dynkin and Togonidze [11], who constructed binary nonsystematic cyclic LUEP codes where several information bits have additional error protection.

A very important contribution to the theory of block LUEP codes was the paper by Dunning and Robbins [10]. In this paper, the authors introduced for the first time the concept of *separation vector* of a UEP code (not necessarily linear), and the corresponding *message set* which is required to protect unevenly. They did show that the separation vector is a generalization of the minimum distance of a code, by considering maximum likelihood decoding of UEP codes. The separation

vector of a UEP code has components equal to the minimum Hamming weight of all the codewords associated with message parts, and should not be confused with the distance spectrum of a code. In addition, the paper presented a procedure to construct generator matrices of optimal LUEP codes, where by optimal LUEP code is meant the shortest LUEP code for a given message set and separation vector. This construction method was used by Van Gils [48], to perform an exhaustive computer search of all binary optimal LUEP codes of length up to 15.

Other contributions to constructions of LUEP codes were [47] and [48], where several optimal LUEP codes of short length, mainly based on generator matrices, are analyzed. A table of binary cyclic UEP codes of odd length up to 39 was compiled in [48]. This table was later extended in [29] to include all binary cyclic UEP codes of odd length up to 65. For the particular case of cyclic LUEP codes of composite length up to 85, offering two levels of error correction, a computer search was performed in this research work. These results helped to determine sufficient conditions on the UEP properties of some nonprimitive BCH codes (see chapter 4). In [48], the author also examined majority-logic decodable LUEP codes, very much in the same line of argument as the paper by Dynkin and Togonidze [11].

Using simple forms of generator matrices, Dirssen [9] presented a class of LUEP codes, based on unions of cosets of rate-1/2 codes, to obtain low rate LUEP codes of moderate minimum distances. Van Gils [49], considered methods of combining shorter LUEP codes using well known methods, such as the $|\bar{u}|\bar{u}+\bar{v}|$ construction, the product of two UEP codes, and the concatenation of a UEP code with a single-level error correcting code. In [7], an analysis of binary images of Reed-Solomon (RS) codes was presented. Under some conditions, when shortened, these binary image codes are LUEP codes. The authors showed that these codes are multi-level concatenated codes [50] and presented a simple example to illustrate the decoding. A

relatively unknown paper was [45], wherein binary UEP codes of rate 1/2, based on specifying their parity check matrices and using adjacency matrices of rank-3 graphs, were discussed. These codes have minimum distance 3 or 4 and provide additional error correcting capability on half of the message symbols. Ma and Wolf [31] present binary UEP codes constructed from binary convolutional codes, by the so called *generalized tail biting* construction. These multi-level error correcting codes have the advantage of yielding a range of rates and error correcting levels using the same encoder structure.

In his doctoral dissertation [26], Lin introduced several classes of LUEP codes. Constructions of two-level LUEP codes specified by their generator and parity-check matrices were analyzed. A class of optimal binary LUEP codes of minimum distance 3, protecting the most important message part against 2 random errors, was described. In addition, the author presented two-level LUEP direct-sum codes of composite length, and showed their random and burst error correcting capabilities. Direct-sums of product and concatenated codes and their decodings , for random and burst errors, were also presented.

Another contribution to the theory of LUEP codes was that of Boyarinov and Katsman [5]. In their paper, they presented necessary and sufficient conditions, on the weight of an arbitrary codeword of a linear code, for the code to be an LUEP code. This result extends to necessary and sufficient conditions on the linear dependency between columns of the parity check matrix of an LUEP code. On the other hand, the authors proved that if the set of minimum weight vectors of a linear code C does not span C, then C has UEP capabilities. In addition, a class of optimal binary 2-level LUEP codes, based on parity-check matrices of binary BCH codes, was discussed. Constructions based on direct-sums of concatenated and product codes were also introduced. Finally, the authors considered a majority

logic decoding algorithm for LUEP codes.

In a milestone paper [3], Blokh and Zyablov generalized the class of concatenated codes and introduced multi-level concatenated codes. In a subsequent paper [50], multi-level concatenated codes were shown to have UEP capabilities, not only against random errors, but also against burst errors, and against combinations of both. In paper [50], a decoding algorithm for m-level concatenated codes - called algorithm $\phi(m)$ - was formulated and shown to be capable of correcting both random and burst errors. This is a class of very powerful UEP codes (both linear and non linear) that has found many applications recently in coded modulation schemes.

Decoding algorithms for some classes of linear multi-level error correcting codes were presented in a paper by Boyarinov [4]. He discussed a decoding procedure for the class of optimal 2-level LUEP codes constructed in [5]. A decoding procedure for compound LUEP codes, specified by their generator matrices, was also introduced. The author also considered a majority logic decoding procedure with generalized metrics, and illustrated it for the case of direct-sums of product codes. In [26], [42], [27], and [43], other decoding procedures were considered. Note, however, that *soft-decision* decoding for UEP codes has not been explicitly addressed in the literature. The exception is in the case of combined coding and modulation schemes using multi-level concatenated codes [30], where in practical applications sub-optimal multi-stage soft-decision decoding is used.

The paper by Bassalygo et al. [2] presented asymptotic bounds on the rates of 2-level error correcting codes. The bounds were generalizations of the Varshamov-Gilbert lower bound and the Elias upper bound. In [23], asymptotic bounds for *linear* UEP codes were considered. It was shown that linear UEP codes are asymptotically inferior to nonlinear UEP codes. This observation was recently validated in [12]. Van Gils, [47], [48], introduced bounds on the length of LUEP codes and

some classes of simple LUEP codes based on generator matrices. Kasami et al. [19], presented a class of direct-sum UEP codes to derive, using a graph-theoretical approach, bounds on the achievable rates of a two-level UEP code for the binary symmetric broadcast channel with two receivers.

In summary, research on multi-level error correcting codes started some 25 years ago. Since then, several construction schemes have been devised. Most known constructions concentrate in binary linear multi-level error correcting codes or LUEP codes. Not much work has been done in designing efficient nonbinary LUEP codes. Constructions based on specifying the parity check matrices of LUEP codes have resulted in binary optimal LUEP codes of minimum distance 3. Specifying the generator matrices of LUEP codes resulted in optimal LUEP codes that provide additional protection only to a few information bits, or in optimal LUEP codes of short lengths. In the area of analysis of linear codes, only shortened binary images of Reed-Solomon codes have been shown to have multi-level error correcting capabilities. The $|\bar{u}|\bar{u}+\bar{v}|$ construction is an efficient way of combining linear codes to obtain longer linear codes, possibly with error correcting capabilities beyond those provided by their minimum distances. Another construction method that yields efficient LUEP codes is multi-level concatenation. Several decoding methods of LUEP codes have been devised, but no *soft-decision* decoding methods for LUEP codes have been considered so far. Finally, research effort has been invested in deriving bounds on the rates and length of linear and nonlinear UEP codes.

1.2 Overview of this research work

In chapter 2, basic ideas and concepts necessary to understand the new constructions and error performance of UEP codes are presented. In particular, the concepts of a cloud and cloud partition of a UEP code, used intensively in this dissertation.

Not much is known about *nonbinary* codes with multiple levels of error correction. The only published work is on multi-level concatenated codes [3], [50], although no result has been published on their particular application to construct nonbinary UEP codes. Much less is known about *optimal* nonbinary LUEP codes. In chapter 3, a construction of a family of optimal nonbinary two-level error-correcting codes is proposed, based on a combination of Reed-Solomon codes and Hamming codes over a Galois field. These codes are specified by their parity-check matrices and can be transformed into systematic form - that is, with information symbols appearing explicitly within a codeword - with the same levels of error correction. It is shown that these codes are asymptotically optimal in the sense of achieving the Hamming bound on the number of redundant symbols, for two-level error-correcting codes with the same dimension and error correcting capabilities, for large values of their length.

In chapter 3, *new constructions* of LUEP codes by combining shorter linear codes specified by their generator matrices are also presented. The results indicate that it is possible to obtain optimal or nearly optimal multi-level error correcting codes in this fashion. In addition, these constructions result in codes which are very easy to decode, with multi-stage suboptimal decoding using either bounded distance decoding or majority logic decoding for the component codes.

The *analysis* of families of conventional single level error correcting codes, to determine their possible multi-level error protection capabilities, is an interesting research problem that has not been reported before in the literature. In chapter 4 of this dissertation, nonprimitive BCH codes are analyzed and, using both their concatenated structure and number theoretic arguments, sufficient conditions for these codes to be LUEP codes are given. Shortened binary Hamming codes were analyzed in [14] to determine their weight distribution and probability of unde-

tected error. In chapter 4, based on the fact that the number of minimum weight codewords decreases as the shortening length increases, we determine the lengths for which a shortened Hamming code is a UEP code. The resulting two-level codes are not good compared to other families of UEP codes. However, this result is of interest, because of the extensive use of shortened Hamming codes in computer network communications. In addition, chapter 4 introduces an example $(63, 24, 15)$ primitive BCH code, containing a $(63, 22, 15)$ shortened Reed-Muller (RM) code, that has UEP capabilities. This example breaks new ground on the analysis of binary primitive BCH codes to determine their UEP capabilities.

In chapter 5, an analysis of block coded modulation schemes using ideas and concepts from UEP codes is introduced. Many of the known block coded modulation (BCM) schemes are particular cases of UEP codes based on the so-called *generalized concatenation* scheme [3], with inner codes used to encode labels of signal constellations, defined in Euclidean space. Constructions of new families of BCM schemes using other types of UEP codes are proposed. We present and discuss a new multi-stage soft-decision decoding of LUEP codes (when combined with signal constellations in Euclidean space). This multi-stage soft-decision decoding uses the trellis structure of LUEP codes and its subcodes, and offers a good trade-off between decoding complexity and error performance.

One of the results of this research is that criteria to determine the efficiency of a multi-level error-correcting code have been found. In previously published results, all the known bounds on the volume and length of single-level error correcting codes have been generalized to cover codes with multiple levels of error protection. More realistic measures of the efficiency of a code are presented. Specifically, we compare of the redundancy required by single-level error-correcting codes, one for each level of error protection, as opposed to a multi-level error-correcting code C. If the

latter requires redundancy between that of a single-level error-correcting code for the highest level of error protection and that of a single-level error-correcting code for the lowest level of error protection, then code C is said to be *good*. In chapter 5, an analysis of the error performance of multi-level error-correcting codes, for the cases of (1) the binary symmetric channel (BSC) and (2) the additive white Gaussian noise (AWGN) channel with BPSK modulation is made and compared with conventional single-level error-correcting codes.

The dissertation finishes with some conclusions and a number of recommendations on future research problems dealing with this increasingly practical area of channel coding.

Chapter 2

Basic concepts

In this chapter, basic concepts of the theory of block codes with multiple levels of error correction are presented. These concepts are necessary to understand the proposed new constructions and in the analysis of UEP codes.

2.1 Separation vector

When a code is used to provide multiple levels of error protection, the conventional definition of minimum distance must be generalized. Since different levels of error protection are possible with a UEP code, a vector of minimum distances, one for each level of error protection, needs to be defined, as shown below.

Let C be an (n, k) block code (not necessarily linear) over a finite alphabet A, $n \geq k$. That is, C is a one-to-one mapping from A^k to A^n, i.e.,

$$\bar{\mathbf{m}} \in A^k \qquad \overset{C}{\mapsto} \qquad \bar{\mathbf{c}}(\bar{\mathbf{m}}) \in A^n,$$

where

$$A^k = \underbrace{A \times A \times \cdots \times A}_{k \text{ times}}.$$

As usual, an element $\bar{\mathbf{m}}$ from A^k is called a *message*, and an element $\bar{\mathbf{c}}(\bar{\mathbf{m}})$ from C is called a *codeword*. In other words, a message is a k-tuple with elements from A, while a codeword is an n-tuple with elements from A. A^k is known as the *message set*.

Let A^k be decomposed into the direct product of ℓ disjoint *message subsets*, A^{k_i}, $1 \leq i \leq \ell$, such that

$$A^k = A^{k_1} \times A^{k_2} \times \cdots \times A^{k_\ell},$$

and a message $\bar{\mathbf{m}} \in A^k$ can be expressed as

$$\bar{\mathbf{m}} = (\bar{\mathbf{m}}_1, \bar{\mathbf{m}}_2, \cdots, \bar{\mathbf{m}}_\ell), \quad \bar{\mathbf{m}}_i \in A^{k_i}, \quad 1 \leq i \leq \ell,$$

where each $\bar{\mathbf{m}}_i$ is called the *i-th message part*, $1 \leq i \leq \ell$.

The *separation vector* of C is defined as the ℓ-tuple $\bar{\mathbf{s}} = (\mathbf{s}_1, \mathbf{s}_2, \cdots, \mathbf{s}_\ell)$, where

$$\mathbf{s}_i \triangleq \min\{d(\bar{\mathbf{c}}(\bar{\mathbf{m}}), \bar{\mathbf{c}}'(\bar{\mathbf{m}}')) : \bar{\mathbf{m}}_i \neq \bar{\mathbf{m}}'_i, \quad \bar{\mathbf{m}}_i, \bar{\mathbf{m}}'_i \in A^{k_i}\}, \quad 1 \leq i \leq \ell,$$

where $d(\bar{\mathbf{x}}, \bar{\mathbf{x}}')$ denotes the Hamming distance between $\bar{\mathbf{x}}$ and $\bar{\mathbf{x}}'$ in A^n. Note that there are no constraints on the j-th message parts, $\bar{\mathbf{m}}_j$, for $j \neq i$, in the above definition of separation vector. Assume that C has all the components of its separation vector distinct and arranged in decreasing order, i.e.,

$$\mathbf{s}_1 > \mathbf{s}_2 > \cdots > \mathbf{s}_\ell,$$

so that C is an (n, k) block code of minimum distance $\mathbf{s}_\ell$. Code C is said to be an (n, k) *ℓ-level UEP code* with separation vector

$$\bar{\mathbf{s}} = (\mathbf{s}_1, \mathbf{s}_2, \cdots, \mathbf{s}_\ell),$$

for the message set $A^{k_1} \times A^{k_2} \times \cdots \times A^{k_\ell}$.

This work concentrates on *linear* block multi-level error correcting codes. That is, A is taken to be the Galois field $GF(q)$ of q elements, where q is a power of a prime. For a *linear* multi-level error correcting code, or LUEP code, each element of the separation vector is given by

$$\mathbf{s}_i \triangleq \min\{\mathrm{wt}(\bar{\mathbf{c}}(\bar{\mathbf{m}})) : \bar{\mathbf{m}}_i \neq \bar{0}, \quad \bar{\mathbf{m}}_i \in GF(q)^{k_i}\}, \quad 1 \leq i \leq \ell. \tag{2.1}$$

A linear code C is said to be an (n, k) ℓ-level LUEP code with separation vector $\bar{\mathbf{s}} = (\mathbf{s}_1, \mathbf{s}_2, \cdots, \mathbf{s}_\ell)$, for the *message space* $GF(q)^{k_1} \times GF(q)^{k_2} \times \cdots \times GF(q)^{k_\ell}$.

In general, it is very hard to determine the separation vector of a code C. This task involves considering all possible decompositions of the message set. Furthermore, for a given decomposition of its message set, a code may have several separation vector values. This is to say that, for a given message set, there might be several codes, of the *same length*, that yield different separation vector values. In [48], a table of all LUEP codes of length up to 15 is presented. In some cases, there exist several LUEP codes for a given value of its length n and dimension k. For instance, there are two *different* $(8, 3)$ binary LUEP codes [48]:

(1) An $(8, 3)$ 2-level LUEP code with separation vector $\bar{\mathbf{s}} = (6, 2)$ for the message space $\{0, 1\} \times \{0, 1\}^2$; and

(2) an $(8, 3)$ 2-level LUEP code with separation vector $\bar{\mathbf{s}} = (5, 4)$ for the message space $\{0, 1\} \times \{0, 1\}^2$.

In this work, the following more practical problem is considered:

> *Given* ℓ and a message set to be protected with ℓ error correcting levels, find the shortest possible *linear* UEP code C with a separation vector whose elements are at least as large as the minimum distances corresponding to desired error correcting capabilities.

In other words, given the dimension k and the decomposition of the message set $GF(q)^k$ into ℓ message subspaces $GF(q)^{k_i}$, to be protected against any t_i or less random errors, $1 \leq i \leq \ell$, the problem is to find an (n, k) LUEP code over the same message space (and subspaces) such that the separation vector of C, $(\mathbf{s}_1, \cdots, \mathbf{s}_\ell)$, has components $\mathbf{s}_i \geq 2t_i + 1$, $1 \leq i \leq \ell$, with n as small as possible. An (n, k) ℓ-level

UEP code with separation vector

$$\bar{\mathbf{s}} = (\mathbf{s}_1, \mathbf{s}_2, \cdots, \mathbf{s}_\ell), \quad \mathbf{s}_i \geq 2t_i + 1, \quad 1 \leq i \leq \ell$$

will sometimes be called an (n,k) $(t_1, t_2, \cdots, t_\ell)$-*error correcting code.* An example of a binary LUEP code is presented in Example 1 below.

Example 1: Let C be a linear code defined by the mapping:

$$\begin{aligned} 00 &\mapsto 0000 \\ 10 &\mapsto 1011 \\ 11 &\mapsto 1101 \\ 01 &\mapsto 0110 \end{aligned}$$

Then,

$$\begin{aligned} s_1 &= \min\{wt(1011), wt(1101)\} = \min\{3,3\} = 3, \\ s_2 &= \min\{wt(1101), wt(0110)\} = \min\{3,2\} = 2, \end{aligned}$$

and C is a $(4,2)$ two-level LUEP code with separation vector $\bar{\mathbf{s}} = (3,2)$ for the message set $\{0,1\} \times \{0,1\}$. Note that the redundancy of C is between the redundancy of a $(3,2,2)$ parity-check code and the redundancy of a $(5,2,3)$ shortened binary Hamming code. $\triangle\ \triangle$

Note that for LUEP codes, definition (2.1) of the separation vector involves the *generator matrix.* As a result, linear multi-level error correcting codes can be constructed by specifying their generator matrices. The generator matrix is selected such that all the elements of the separation vector are distinct, so that, for example, $\bar{\mathbf{s}} = (6,2,2)$ is the same as $\bar{\mathbf{s}} = (6,2)$. Several constructions of LUEP codes using this approach are presented in chapter 3.

Another way of defining a linear code is by specifying its parity-check matrix, H. That is, we need to specify the different levels of error protection in terms of the linear dependence between the columns of H. Let C be a conventional linear single-level error-correcting code. The *minimum distance* of C is given by the following theorem [37],

Theorem 1 *Let C be an (n, k, d) linear code with parity-check matrix H. The minimum distance of the code is at least d if and only if every combination of $d-1$ or fewer columns of H is linearly independent.*

Proof: See [37], p. 44. $\triangle\ \triangle$

For multi-level error correcting codes, the minimum distance is generalized to the separation vector. Accordingly, the separation vector of an ℓ-level error correcting code C, $\bar{\mathbf{s}} = (\mathbf{s}_1, \mathbf{s}_2, \cdots, \mathbf{s}_\ell)$, defines the linear dependence relation among columns of its parity-check matrix. Let H be the parity-check matrix of C. Assume that H can be transformed into *systematic* form, say H', and that the equivalent linear systematic code, whose parity-check matrix is H', has the same error-correcting capabilities and message space as the original code. Then

$$H' = (I|P) \tag{2.2}$$

where I is the $(n-k)$-by-$(n-k)$ identity matrix, and P is an $(n-k)$-by-k matrix that defines the parity-check equations of the code. If the parity-check matrix can be written as in (2.2), then we have the following result, which is a generalization of Theorem 1 for LUEP codes (see [34]):

Theorem 2 *Let C' be an (n, k) ℓ-level LUEP code with separation vector*

$$\bar{\mathbf{s}} = (\mathbf{s}_1, \mathbf{s}_2, \cdots, \mathbf{s}_\ell),$$

for the message space $M = GF(q)^{k_1} \times GF(q)^{k_2} \times \cdots \times GF(q)^{k_\ell}$. *Then every combination of* $\mathbf{s}_i - 1$ *or fewer columns from* H', *with at least one column from the first* $k_1 + k_2 + \cdots + k_i$ *columns of* H', *is linearly independent, for* $1 \leq i \leq \ell$.

Proof: See [34], Theorem 1, p.601. $\triangle\,\triangle$

It should be noted [10] that it is *not* always possible to transform a nonsystematic LUEP code into an equivalent systematic LUEP code, with the same separation vector and message space. This means that there might be codes with multi-level error-correcting capabilities but protecting *code positions* instead of *message positions*. This is to say that it is not always possible to have an LUEP code protecting message bits explicitly, as can be seen in the case of some LUEP codes specified by their generator matrix (see section 3.2).

2.2 Partition into clouds

Multi-level error correcting codes have the following useful set theoretic interpretation. Let C be an ℓ-level UEP code with separation vector $\bar{\mathbf{s}} = (\mathbf{s}_1, \mathbf{s}_2, \cdots, \mathbf{s}_\ell)$ for the message set $A^{k_1} \times A^{k_2} \times \cdots \times A^{k_\ell}$. For $1 \leq i \leq \ell$, consider the set of codewords such that the first message part, $\bar{\mathbf{m}}_1$, is fixed, i.e.,

$$C\ell(\bar{\mathbf{x}}_{i_1}) \stackrel{\Delta}{=} \{\bar{\mathbf{c}}(\bar{\mathbf{m}}) \in C : \bar{\mathbf{m}}_1 = \bar{\mathbf{x}}_{i_1}\}.$$

The set of codewords $C\ell(\bar{\mathbf{x}}_{i_1})$ is called a *first-level cloud.* It is clear that the *volume* (number of codewords) of a first-level cloud equals $|A|^{k_2+k_3+\cdots+k_\ell}$. The number of first-level clouds equals the number of choices for $\bar{\mathbf{x}}_1$, i.e., $|A|^{k_1}$. These first-level clouds are disjoint and partition code C, in such a way that C can be expressed as

a union of first-level clouds,

$$C = \bigcup_{\bar{\mathbf{x}}_{i_1} \in A^{k_1}} C\ell(\bar{\mathbf{x}}_{i_1})$$

where $\bar{\mathbf{x}}_{i_1}$ runs trough all distinct elements in A^{k_1}. This is called the *first-level partition* of C, and is denoted $C/C\ell(\bar{\mathbf{m}}_1)$.

Similarly, define a *second-level cloud* as the set of codewords of C such that the first and second message parts are fixed,

$$C\ell(\bar{\mathbf{x}}_{i_1}, \bar{\mathbf{x}}_{i_2}) \triangleq \{\bar{\mathbf{c}}(\bar{\mathbf{m}}) \in C \ : \ \bar{\mathbf{m}}_1 = \bar{\mathbf{x}}_{i_1} \text{ and } \bar{\mathbf{m}}_2 = \bar{\mathbf{x}}_{i_2}, \ \ \bar{\mathbf{a}}_{i_1} \in A^{k_1}, \bar{\mathbf{x}}_{i_2} \in A^{k_2}\}.$$

Each second-level cloud consists of $|A|^{k_3+\cdots+k_\ell}$ codewords of C, and the number of second-level clouds is equal to the number of choices of values of the first and second message parts, $\bar{\mathbf{x}}_{i_1}$ and $\bar{\mathbf{x}}_{i_2}$, which equals $|A|^{k_1+k_2}$. Note that each first-level cloud can be written as a union of second-level clouds,

$$C\ell(\bar{\mathbf{m}}_1) = \bigcup_{\bar{\mathbf{x}}_{i_2} \in A^{k_2}} C\ell(\bar{\mathbf{m}}_1, \bar{\mathbf{x}}_{i_2}),$$

where $\bar{\mathbf{x}}_{i_2}$ runs through all distinct elements in A^{k_2}. After the *second-level partition* of C, denoted $C/C\ell(\bar{\mathbf{m}}_1)/C\ell(\bar{\mathbf{m}}_1, \bar{\mathbf{m}}_2)$, C can be expressed as

$$C = \bigcup_{\bar{\mathbf{x}}_{i_1} \in A^{k_1}} \bigcup_{\bar{\mathbf{x}}_{i_2} \in A^{k_2}} C\ell(\bar{\mathbf{x}}_{i_1}, \bar{\mathbf{x}}_{i_2})$$

where $\bar{\mathbf{x}}_{i_1}$ runs through all distinct elements in A^{k_1}, and $\bar{\mathbf{x}}_{i_2}$ runs through all distinct elements in A^{k_2}.

This process of partitioning a UEP code into clouds can be continued until the ℓ-th level partition of C is reached, using *ℓ-level clouds*, which are defined as

$$C\ell(\bar{\mathbf{x}}_{i_1}, \bar{\mathbf{x}}_{i_2}, \cdots, \bar{\mathbf{x}}_{i_\ell}) \triangleq \{\bar{\mathbf{c}}(\bar{\mathbf{m}}) \in C \ : \ \bar{\mathbf{m}}_1 = \bar{\mathbf{x}}_{i_1}, \bar{\mathbf{m}}_2 = \bar{\mathbf{x}}_{i_2}, \cdots, \bar{\mathbf{m}}_\ell = \bar{\mathbf{x}}_{i_\ell}\}.$$

Each of these ℓ-level clouds consists of a single codeword, and the number of ℓ-level clouds is equal to the volume of the code, $|A|^k$. After the *ℓ-level partition* of code

C, denoted $C/C\ell(\bar{\mathbf{m}}_1)/C\ell(\bar{\mathbf{m}}_1, \bar{\mathbf{m}}_2)/\cdots/C\ell(\bar{\mathbf{m}}_1, \cdots, \bar{\mathbf{m}}_\ell)$,

$$C = \bigcup_{\bar{\mathbf{x}}_{i_1} \in A^{k_1}} \bigcup_{\bar{\mathbf{x}}_{i_2} \in A^{k_2}} \cdots \bigcup_{\bar{\mathbf{x}}_{i_\ell} \in A^{k_\ell}} C\ell(\bar{\mathbf{x}}_{i_1}, \cdots, \bar{\mathbf{x}}_{i_\ell})$$

Recently, Forney [13] introduced the concept of coset codes, based on partitioning codes into cosets. If a UEP code is linear, then the cloud partition process just presented is equivalent to the coset decomposition process used by Forney. Furthermore, the cloud concept for codes for the binary symmetric broadcast channel for two users was first introduced by Cover [8].

2.3 Encoder structure

An encoder for a multi-level error-correcting code, based on its cloud partition, consists of interconnected blocks, one for each partition level. A block diagram of an encoder for a two-level error correcting code is shown in Figure 2.1.

The first block is a *first-level cloud selector*, using the first message part $\bar{\mathbf{m}}_1$ as input. Message part $\bar{\mathbf{m}}_1$ serves as *index* for first-level clouds. Once a first-level cloud is selected, it is used as input for the *second-level cloud selector*, which in addition uses the second message part $\bar{\mathbf{m}}_2$ as input, so that a second-level cloud is indexed by both $\bar{\mathbf{m}}_1$ and $\bar{\mathbf{m}}_2$. Intermediate cloud selectors work in a similar fashion. For the *i-th cloud selector*, one input comes from the previous $(i-1)$-th level cloud selector while the another input is from the corresponding i-th message part. Thus i-level clouds are indexed by $\bar{\mathbf{m}}_1$, $\bar{\mathbf{m}}_2$, $\cdots$, $\bar{\mathbf{m}}_i$. Finally, the *ℓ-level cloud selector* will output codewords from C, indexed by $\bar{\mathbf{m}} = (\bar{\mathbf{m}}_1, \bar{\mathbf{m}}_2, \cdots, \bar{\mathbf{m}}_\ell)$ Note that the partition process gives a nice interpretation of the encoding process, where each message part, of a given message to be encoded by a UEP code C, selects clouds at the corresponding level of the partition of C.

As an illustrative example, consider an (n, k) 2-level UEP code C over A^n with

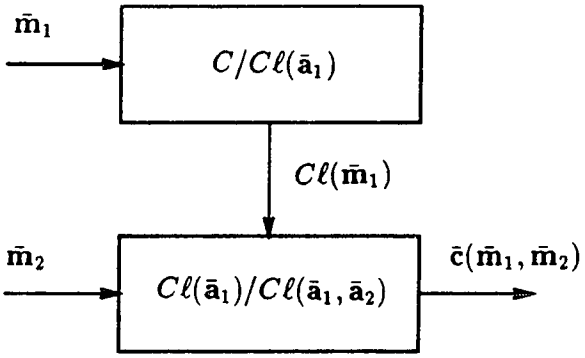

Figure 2.1: Encoder of a two-level error-correcting code

separation vector $\bar{\mathbf{s}} = (\mathbf{s}_1, \mathbf{s}_2)$, where $\mathbf{s}_1 > \mathbf{s}_2$, for the message set $A^{k_1} \times A^{k_2}$. Then C admits a two-level partition into clouds. The encoding of C can be viewed as follows:

(1) The first message part $\bar{\mathbf{m}}_1 \in A^{k_1}$ *selects a 1-level cloud* $C\ell(\bar{\mathbf{m}}_1)$, which consists of $|A|^{k_2}$ 2-level clouds (codewords).

(2) The second message part $\bar{\mathbf{m}}_2 \in A^{k_2}$ *selects a 2-level cloud* $C\ell(\bar{\mathbf{m}}_1, \bar{\mathbf{m}}_2)$ (simply a codeword) from the $|A|^{k_2}$ 2-level clouds (codewords) in the previously selected (indexed by $\bar{\mathbf{m}}_1$) 1-level cloud.

This particular case of a 2-level UEP code will be used to give a set theoretic interpretation of the separation vector. Note that the minimum distance between a codeword $\bar{\mathbf{c}}$ in a 1-level cloud $C\ell(\bar{\mathbf{x}}_{i_1})$ and a codeword $\bar{\mathbf{c}}'$ in a distinct 1-level cloud $C\ell(\bar{\mathbf{x}}'_{i_1})$, $\bar{\mathbf{x}}_{i_1} \neq \bar{\mathbf{x}}'_{i_1}$, is given by

$$\mathbf{s}_1 = \min\{d(\bar{\mathbf{c}}(\bar{\mathbf{m}}), \bar{\mathbf{c}}'(\bar{\mathbf{m}}')) \ : \ \bar{\mathbf{x}}_{i_1} \neq \bar{\mathbf{x}}'_{i_1}\}$$

while the minimum distance between a codeword $\bar{\mathbf{c}}$ from a 2-level cloud $C\ell(\bar{\mathbf{m}}_1, \bar{\mathbf{m}}_2)$ and a codeword $\bar{\mathbf{c}}'$ from another 2-level cloud $C\ell(\bar{\mathbf{m}}_1, \bar{\mathbf{m}}_2)$ is equal to the minimum distance of the code, i.e., $\bar{\mathbf{s}}_2$, because in the case of 2-level UEP codes, 2-level clouds are simply codewords.

2.4 Relationship between separation vector and clouds

Define the *inter-set* distance between sets U and V, $U \subseteq A^n$, $V \subseteq A^n$, as follows,

$$d[U, V] \triangleq \min\{d(\bar{\mathbf{u}}, \bar{\mathbf{v}}) : \bar{\mathbf{u}} \in U \text{ and } \bar{\mathbf{v}} \in V\}$$

where $d(\bar{\mathbf{y}}, \bar{\mathbf{z}})$ denotes the Hamming distance between $\bar{\mathbf{y}}, \bar{\mathbf{z}} \in A^n$. The minimum inter-set distance between any two distinct 1-level clouds, $C\ell(\bar{\mathbf{x}}_{i_1}) \neq C\ell(\bar{\mathbf{x}}'_{i_1})$, is

$$\mathbf{s}_i = \min\{d[C\ell(\bar{\mathbf{x}}_{i_1}), C\ell(\bar{\mathbf{x}}'_{i_1})]\}$$

Therefore, the first component of the separation vector of a UEP code equals the minimum inter-set distance, or *separation*, between 1-level clouds. This set theoretic interpretation of the separation vector holds at every level of the partition, so that the j-th component of the separation vector is equal to the minimum inter-set distance or separation between j-level clouds, i.e., for $C\ell(\bar{\mathbf{x}}_{i_1}, \cdots, \bar{\mathbf{x}}_{i_j}) \neq C\ell(\bar{\mathbf{x}}'_{i_1}, \cdots, \bar{\mathbf{x}}'_{i_j})$,

$$\mathbf{s}_j = \min\{d[C\ell(\bar{\mathbf{x}}_{i_1}, \cdots, \bar{\mathbf{x}}_{i_j}), C\ell(\bar{\mathbf{x}}'_{i_1}, \cdots, \bar{\mathbf{x}}'_{i_j})]\}$$

for $1 \leq j \leq \ell$, and under the assumption that the separation vector is arranged in strictly decreasing order, i.e., $\mathbf{s}_1 > \mathbf{s}_2 > \cdots > \mathbf{s}_\ell$.

Viewing an ℓ-level error correcting code as partitioned into clouds in ℓ stages is useful in: (1) analyzing the structure of this class of codes; (2) designing multi-stage decoding methods for multi-level error-correcting codes; and (3) analyzing block coded modulation (BCM) schemes. For example, in BCM each partition level is associated with a corresponding *label* partition stage. Tanner [44] was the first to point out the connection between BCM and UEP codes. In this dissertation, concepts from the theory of multi-level error correcting codes are used in constructing block modulation codes (see chapter 5).

2.5 Summary

In this chapter, we have presented basic concepts necessary to understand the theory of multi-level error correcting codes. In this dissertation, only *linear* UEP codes

(LUEP codes) are considered. We have defined the separation vector of an LUEP code, which constitutes a generalization of the minimum Hamming distance of a linear code. This separation vector has one component for each message subspace, and provides information about the multi-level error correcting capabilities of the code, just as the minimum distance gives information about the error correction capability of a (single-level) linear code. The problem of finding efficient LUEP codes is difficult, because sometimes there is no unique mapping for a given message space that will provide some predefined multi-level error correcting capabilities. This dissertation concentrates on the problem of designing and analyzing multi-level error-correcting codes of the shortest length for a given message space and separation vector. An LUEP code can be specified by the linear dependency among the columns of its parity-check matrix. However, this method of construction can only be useful when the resulting LUEP code can be transformed into a systematic LUEP code with the same message space and separation vector, or when it is of interest to protect codeword positions instead of message positions. The partition into clouds of an LUEP code has been presented. It was shown that the components of the separation vector of an LUEP code correspond to the minimum Hamming distance between codewords in different clouds at the corresponding partition level. The number of levels of the partition of an LUEP code into clouds is thus equal to the number of component of its separation vector. The cloud structure of an LUEP code served to give an interpretation of the encoding process, as one of selecting clouds at different levels of the partition. It was also mentioned that the cloud structure also serves to design multi-stage decoding methods for multi-level error correcting codes, and in analyzing some block coded modulation schemes.

Chapter 3

Constructions of LUEP codes

In this chapter, new families of optimal or near optimal linear multi-level error correcting codes are introduced. Optimality of an LUEP code C is shown using the Hamming bound on the number of redundant symbols for an LUEP code with the same separation vector and message space.

3.1 Specifying the parity check matrix

In this section, a new family of optimal nonbinary LUEP codes, whose parity-check matrix is a combination of the parity-check matrices of Reed-Solomon codes and Hamming codes, is introduced. An asymptotic form of the Hamming bound for LUEP codes over the field $GF(2^s)$ is derived. It is shown that these codes are optimal in the sense of achieving the Hamming lower bound on the number of redundant symbols for an LUEP code with the same separation vector and message space.

3.1.1 Optimal binary (2,1)-error correcting codes

Boyarinov and Katsman [5] have constructed binary LUEP codes with separation vector $\bar{\mathbf{s}} \geq (2t+1, 3)$, $t \geq 2$, based on specifying a parity check matrix constructed as a combination of the parity check matrices of binary BCH codes and Hamming codes. These codes are optimal in the sense described in the previous paragraph. In [28], for the case $t = 2$, these optimal binary LUEP codes are generalized by using,

instead of the parity check matrix of a Hamming code, the parity check matrix of a *shortened* Hamming code, as follows.

Let α be a primitive element of $GF(2^m)$. Let C be the binary linear code with parity check matrix,

$$H = \left(\begin{array}{cccc|ccc} 1 & \alpha & \cdots & \alpha^{2^m-2} & 0_m & \cdots & 0_m \\ 1 & \alpha^3 & \cdots & \alpha^{3(2^m-2)} & & & \\ 0_\ell & 0_\ell & \cdots & 0_\ell & \beta_1 & \cdots & \beta_{2^{m+\ell}-2^m} \end{array} \right), \tag{3.1}$$

where each power of α is represented by a binary column vector of length m, 0_m represents the all zero column vector of length m, and, for $\ell \geq 0$, $\beta_1, \cdots, \beta_{2^{m+\ell}-2^m}$ represent all binary column vectors of length $m+\ell$ having the last ℓ entries not all zero.

Theorem 3 *C is a $(2^{m+\ell}-1, 2^{m+\ell}-2m-\ell-1)$ LUEP code, with separation vector $\bar{\mathbf{s}} = (\mathbf{s}_1, \mathbf{s}_2)$, $\mathbf{s}_1 \geq 5$, and $\mathbf{s}_2 \geq 3$, for the message space $M = M_1 \times M_2$, where $M_1 = \{0,1\}^{2^m-m-1}$ and $M_2 = \{0,1\}^{2^{m+\ell}-2^m-m-\ell}$.*

Proof: See [28]. $\triangle\triangle$

3.1.2 Optimal nonbinary (2,1)-error correcting codes

A construction of a class of LUEP codes over the field $GF(2^s)$, obtained from generalizing the above class of binary LUEP codes is proposed. Instead of parity check matrices of BCH codes *over* $GF(2)$, parity check matrices of BCH codes *over* $GF(2^s)$ are used. The idea is to obtain *optimal* LUEP codes with separation vector $\bar{\mathbf{s}} = (\mathbf{s}_1, \mathbf{s}_2)$, where $\mathbf{s}_1 \geq 5$ and $\mathbf{s}_2 \geq 3$, this time over the field $GF(2^s)$, for $s \geq 2$. The parity check matrix of the proposed class of nonbinary LUEP codes is specified as follows. Let γ be a primitive element of $GF(2^{ms})$ over $GF(2^s)$. Let $C(2^s)$ be the linear code over $GF(2^s)$ with parity check matrix,

$$H = \left(\begin{array}{cccc|ccc} 1 & \gamma & \cdots & \gamma^{2^{sm}-2} & 0_m & \cdots & 0_m \\ 1 & \gamma^2 & \cdots & \gamma^{2(2^{sm}-2)} & 0_m & \cdots & 0_m \\ 1 & \gamma^3 & \cdots & \gamma^{3(2^{sm}-2)} & & & \\ 1 & \gamma^4 & \cdots & \gamma^{4(2^{sm}-2)} & \phi_1 & \cdots & \phi_{n_b} \\ 0_\ell & 0_\ell & \cdots & 0_\ell & & & \end{array} \right), \tag{3.2}$$

where each power of γ is represented as a column vector of length m over $GF(2^s)$, $s > 1$, 0_i represents a (2^s-ary) column vector of i zeros, and $\phi_1, \cdots, \phi_{n_b}$ represent column vectors, not multiples of each other, of length $2m + \ell$ over $GF(2^s)$ for which the last ℓ entries are not all zeros, where

$$n_b = \frac{2^{s(2m)}(2^{s\ell} - 1)}{2^s - 1.}$$

Note that H can be written as,

$$H = \begin{pmatrix} H_{aa} & 0_{ab} \\ H_{ab} & H_{ba} \\ 0_{ba} & H_{bb} \end{pmatrix} = (H_1 | H_2), \tag{3.3}$$

where,

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

is the parity check matrix of a BCH code C_a over $GF(2^s)$ of length $n_a = 2^{sm} - 1$, dimension $k_a \geq 2^{sm} - 4m - 1$, and minimum distance $d_a \geq 5$; H_{aa} is the parity check matrix of a BCH code C_{aa} over $GF(2^s)$, which contains C_a, of length n_a, dimension $k_{aa} \geq 2^{sm} - 2m - 1$, and minimum distance $d_{aa} \geq 3$;

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

is the parity check matrix of a shortened Hamming code C_b of length $n_b = 2^{2sm}(2^{s\ell} - 1)/(2^s - 1) = 2^{2sm}(2^{s(\ell-1)} + \cdots + 2^s + 1)$, dimension $k_b = n_b - 2m - \ell$ and minimum distance $d_b \geq 3$; and H_{bb} is the parity-check matrix of a linear code C_{bb} over $GF(2^s)$, containing C_b, of length n_b, dimension $k_{bb} \geq n_b - \ell$ and minimum distance $d_{bb} = 2$.

Theorem 4 *C is an (n,k) LUEP code over $GF(2^s)$, $s>2$, with parameters,*

$$\begin{aligned} n &= 2^{2sm}(2^{s(\ell-1)}+2^{s(\ell-2)}+\cdots+2^s+1)+2^{sm}-1, \\ k &\geq n-4m-\ell, \\ \bar{\mathbf{s}} &= (\mathbf{s}_1,\mathbf{s}_2), \quad \mathbf{s}_1 \geq 5, \quad \mathbf{s}_2 \geq 3, \end{aligned}$$

for the message space $M = GF(2^s)^{k_1} \times GF(2^s)^{k_2}$, where

$$\begin{aligned} k_1 &\geq 2^{sm}-2m-1, \quad \textit{and} \\ k_2 &\geq 2^{2sm}(2^{s(\ell-1)}+\cdots+2^s+1)-2m-\ell. \end{aligned}$$

In other words, C protects the first $2^{sm}-2m-1$ information symbols against any combination of 2 or less symbol errors, and the remaining information symbols against any single symbol error.

Proof: Note first that H_b is of the form

$$H_b = (\, M_0 \;\mid\; M_1 \;\mid\; \cdots \;\mid\; M_{2^{s\ell}-2}\,),$$

where

$$M_i = \begin{pmatrix} 0 & 1 & \xi & \cdots & \xi^{2^{2sm}-2} \\ \mu^i & \mu^i & \mu^i & \cdots & \mu^i \end{pmatrix}, \quad 0 \leq i \leq 2^{s\ell}-2,$$

ξ is a primitive element of $GF(2^{2sm})$, and μ is a primitive element of $GF(2^{s\ell})$. That C has minimum distance $\mathbf{s}_2 = 3$ follows from the fact that all columns of matrix H in equation (3.3) are different, and we can find 3 columns from submatrix H_2 in equation (3.3) that add to the all-zero vector [37]. By Theorem 1 of Chapter 2, it follows that $\mathbf{s}_2 = 3$. It remains to show that $\mathbf{s}_1 \geq 5$. Let $\bar{h}_i^{(j)}$ denote the i-th column of submatrix H_j in equation (3.3), for $j = 1,2$. We will prove that any column $\bar{h}_i^{(1)}$ is linearly dependent on no less than four other columns of H. We do this by considering linear combinations of columns of H, dividing into two cases: (1) linear combinations of three columns, and (2) linear combinations of four columns.

(1) Three columns:

(i) $\bar{h}_{i_1}^{(1)} + \bar{h}_{i_2}^{(1)} + \bar{h}_{i_3}^{(1)} \neq \bar{0}$, by definition of H_a, the parity check matrix of a BCH code of minimum distance 5.

(ii) $\bar{h}_{i_1}^{(1)} + \bar{h}_{i_2}^{(1)} + \bar{h}_{i_3}^{(2)} \neq \bar{0}$, since $\bar{h}_{i_3}^{(2)} \neq \bar{0}$.

(iii) $\bar{h}_{i_1}^{(1)} + \bar{h}_{i_2}^{(2)} + \bar{h}_{i_3}^{(2)} \neq \bar{0}$, since $\bar{h}_{i_1}^{(1)} \neq \bar{0}$.

(2) Four columns:

(i) $\sum_{j=1}^{4} \bar{h}_{i_j}^{(1)} \neq \bar{0}$, by definition of H_a, the parity check matrix of a BCH code of minimum distance 5.

(ii) $\sum_{j=1}^{3} \bar{h}_{i_j}^{(1)} + \bar{h}_{i_4}^{(2)} \neq \bar{0}$, since $\bar{h}_{i_4}^{(2)} \neq \bar{0}$.

(iii) $\bar{h}_{i_1}^{(1)} + \bar{h}_{i_2}^{(1)} + \bar{h}_{i_3}^{(2)} + \bar{h}_{i_4}^{(2)} \neq \bar{0}$, since all columns are different.

(iv) $\bar{h}_{i_1}^{(1)} + \sum_{j=2}^{4} \bar{h}_{i_j}^{(2)} \neq \bar{0}$, since $\bar{h}_{i_1}^{(1)} \neq \bar{0}$. $\triangle\triangle$

As in the binary case (previous section), code C can be transformed into a systematic code with the same parameters and separation vector [5, 28]. Note that for $m = 1$, C_a and C_{aa} are *Reed-Solomon* (RS) codes over $GF(2^s)$. It should be noted that the proof of Theorem 4 is much simpler than that used in [28] for the binary case. Example 2 illustrates a code in this class.

Example 2: Let $\ell = m = 1$ and $s = 3$. C is then a $(71, 66)$ two-level LUEP code over $GF(2^3)$, with 5 information symbols protected against any two or less random errors, and 61 information symbols protected against any single random error. In the next subsection we will show that this code achieves the Hamming lower bound on the number of redundant symbols from $GF(2^3)$ with equality. Code C is an example of an optimal two-level (2,1)-error correcting code over $GF(2^3)$. $\triangle\triangle$

Hamming bound

For a binary linear (t_1, t_2)-error correcting (n, k) code, the following Hamming bound was derived by Boyarinov and Katsman [5],

$$2^{n-k} \geq \sum_{i=0}^{t_2} \binom{n}{i} + \sum_{j=t_2+1}^{t_1} \sum_{i=0}^{t_2} \binom{n-k_1}{i} \binom{k_1}{j-i} \tag{3.4}$$

For the codes of Theorem 4, we obtain a lower bound on the number of redundant symbols as follows: (i) the number of cosets is now $2^{s(n-k)}$; (ii) the number of vectors in $(GF(2^s))^n$ of weight less than or equal to t_2 is,

$$\sum_{i=0}^{t_2} \binom{n}{i} (2^s - 1)^i$$

and (iii) the number of vectors over $GF(2^s)$ of Hamming weight w, such that $t_2 < w \leq t_1$, with at least one nonzero component in the k_1 most significant positions is,

$$\sum_{j=t_2+1}^{t_1} \sum_{i=0}^{t_2} \binom{n-k_1}{i} \binom{k_1}{j-i} (2^s - 1)^j$$

The Hamming bound for a linear two-level (t_1, t_2)-error correcting code over $GF(2^s)$ is obtained as follows: The number of cosets of a linear code in $GF(2^s)^n$ must be at least equal to the number of correctable error patterns, i.e.,

$$2^{s(n-k)} \geq \sum_{i=0}^{t_2} \binom{n}{i} (2^s - 1)^i + \sum_{j=t_2+1}^{t_1} \sum_{i=0}^{t_2} \binom{n-k_1}{i} \binom{k_1}{j-i} (2^s - 1)^j \tag{3.5}$$

For the class of codes of Theorem 4, we let $t_1 = 2$ and $t_2 = 1$ in (3.5), obtaining

$$\begin{aligned} 2^{s(n-k)} \geq\ & 1 + (2^{sm} - 1)(2^s - 1) + (2^{sm} - 2m - 1)(2^{sm-1} + m - 1)(2^s - 1)^2 \\ & + 2^{2sm}(2^{s\ell} - 1)[1 + (2^{sm} - 2m - 1)(2^s - 1)] \end{aligned} \tag{3.6}$$

Evaluating (3.6) for different values of ℓ, s and m helps us to determine that Theorem 4 gives optimal codes for $m = 1$, $s > 2$ and $\ell > 0$. From (3.6) with $m = 1$, we have that

$$2^{s(n-k)} \geq 2^{s(\ell+4)-1}(1 + \Delta),$$

where

$$\Delta = 1 + [1 + (2^s - 1)^2 (1 + 2^{s-1}(2^s - 3))] 2^{1-s(\ell+4)}$$

$$-2^{1-s\ell} - 2^{3-s} + 2^{3-s\ell-s} + 2^{3-2s} - 2^{3-s\ell-2s}$$

and $0 < \Delta < 1$, for $m = 1$, $s \geq 3$ and $\ell > 0$. Therefore, $n - k \geq \ell + 4$. Note that for $m = 1$, codes from Theorem 4 have redundancy $n - k = \ell + 4$.

Theorem 4 gives optimal codes for $m = 1$, $s > 2$ and $\ell > 0$. It follows that (2,1)-error correcting codes over $GF(2^s)$, with parity check matrix of the form (3.2) and $m = 1$ (i.e., the upper left submatrix is the parity check matrix of an RS code), are *optimal codes* in the sense of achieving the Hamming lower bound on the number of redundant symbols with equality.

3.1.3 Optimal $(t, 1)$-error correcting codes over $GF(2^s)$

In this section, we generalize the results of the previous section, to construct a family of $(t, 1)$-error correcting codes over $GF(2^s)$, with $t \geq 2$. Instead of using the parity check matrix of a 2-error correcting nonbinary BCH code, the parity check matrix of a *t-error correcting* nonbinary BCH code is used to obtain nonbinary LUEP codes with separation vector $\bar{\mathbf{s}} = (\mathbf{s}_1, \mathbf{s}_2)$, $\mathbf{s}_1 \geq 2t + 1$ and $\mathbf{s}_2 \geq 3$. In addition, we show that if the parity check matrices of RS codes and shortened nonbinary Hamming codes are combined, then *asymptotically optimal* nonbinary $(t, 1)$-error correcting codes are obtained. Let C be the linear code over $GF(2^s)$ with parity check matrix as in equation (3.3), where H_a is now the parity check matrix of a t-error correcting BCH code C_a over $GF(2^s)$ of length $n_a = 2^{sm} - 1$ and dimension $k_a \geq 2^{sm} - 2mt - 1$, and H_{aa} is the parity check matrix of a $(t-1)$-error correcting BCH code C_{aa} over $GF(2^s)$ of length $n_{aa} = 2^{sm} - 1$ and dimension $k_{aa} \geq 2^{sm} - 2m(t-1) - 1$.

Theorem 5 *C is an (n,k) LUEP code over $GF(2^s)$, $s>2$, with parameters*

$$\begin{aligned} n &= 2^{2sm}(2^{s(\ell-1)}+2^{s(\ell-2)}+\cdots+2^s+1)+2^{sm}-1 \\ k &\geq n-2mt-\ell \\ \bar{\mathbf{s}} &= (\mathbf{s}_1,\mathbf{s}_2),\quad \mathbf{s}_1\geq 2t+1,\ \mathbf{s}_2\geq 3 \end{aligned} \tag{3.7}$$

for the message space $M=GF(2^s)^{k_1}\times GF(2^s)^{k_2}$, where

$$\begin{aligned} k_1 &\geq 2^{sm}-2m(t-1)-1 \quad \textit{and} \\ k_2 &\geq 2^{2sm}(2^{s(\ell-1)}+\cdots+2^s+1)-2m-\ell \end{aligned}$$

Proof: (Similar to the proof of Theorem 4) The minimum distance of C is $\mathbf{s}_2=3$. That any column $\bar{h}_i^{(1)}$ from submatrix H_1 in (3.3) is linearly dependent on no less than $2t$ other columns of H is shown by considering the linear dependency among columns of (3.3) in the following way. Let $\bar{h}_i^{(j)}$ denote the i-th column of submatrix H_j in (3.3).

(1) Up to $2t-2$ columns: $\sum_{j=1}^{m}\bar{h}_{i_j}^{(1)}+\sum_{j'=1}^{2t-2-m}\bar{h}_{i_{j'}}^{(2)}=\bar{0}$, contradicts the definition of H_{aa}, the parity check matrix of a $(t-1)$-error correcting BCH code, for $1\leq m\leq 2t-2$.

(2) $2t-1$ columns: $\sum_{j=1}^{m}\bar{h}_{i_j}^{(1)}+\sum_{j'=1}^{2t-1-m}\bar{h}_{i_{j'}}^{(2)}=\bar{0}$. Divide into the following cases:

  (i) $m=2t-1$, contradicts the definition of H_a, the parity check matrix of a t-error correcting BCH code.

  (ii) $1\leq m\leq 2t-2$, contradicts definition of H_{aa}, the parity check matrix of a $(t-1)$-error correcting BCH code.

(3) $2t$ columns: $\sum_{j=1}^{m}\bar{h}_{i_j}^{(1)}+\sum_{j'=1}^{2t-m}\bar{h}_{i_{j'}}^{(2)}=\bar{0}$. Divide in three cases:

  (i) $m=2t$, contradicts the definition of H_a, the parity check matrix of a t-error correcting BCH code.

(ii) $m = 2t - 1$, impossible because $\bar{h}^{(2)}_{i_{j'}} \neq \bar{0}$.

(iii) $1 \leq m \leq 2t - 2$, contradicts definition of H_{aa}, the parity check matrix of a $(t-1)$-error correcting BCH code. $\Delta\Delta$

As in Theorem 4, code C can be transformed into a systematic code with the same parameters and separation vector [5, 28]. Note that for $m = 1$, C_a and C_{aa} are *Reed-Solomon* (RS) codes over $GF(2^s)$.

Hamming bound for $\bar{\mathbf{s}} = (7, 3)$

Let $t_1 = 3$ and $t_2 = 1$ in inequality (3.5). Then

$$2^{s(n-k)} \geq \sum_{i=0}^{1} \binom{n}{i} (2^s - 1)^i + \sum_{j=2}^{3} \sum_{i=0}^{1} \binom{n-k_1}{i} \binom{k_1}{j-i} (2^s - 1)^j.$$

With $m = 1$, we have for codes from Theorem 5, using the parameters in (3.7),

$$\begin{aligned} 2^{s(n-k)} \geq\ & 1 + 2^{2s}(2^{s\ell} - 1) + (2^s - 1)(2^s - 1) \\ & + (2^s - 5)(2^{s-1} - 3)(2^s - 1)^2 \\ & + 2^{2s}(2^{s\ell} - 1)(2^s - 1) + 4(2^s - 5)(2^s - 1)^2 \\ & + \tfrac{1}{6}(2^s - 5)(2^s - 6)(2^s - 7)(2^s - 1)^3 \\ & + \{2^{2s}(2^{s\ell} - 1)(2^s - 1)^2 + 4(2^s - 1)^3\}(2^s - 5)(2^s - 3) \end{aligned}$$

After some manipulation, we have that

$$2^{s(n-k)} \geq 2^{s(\ell+6)-1}(1 + \Delta)$$

where

$$\begin{aligned} \Delta\ =\ & 1 + \Big[2^{2s}(2^{s\ell} + 1) + (2^s - 1)^2 + (2^s - 5)(2^{s-1} - 3)(2^s - 1)^2 \\ & + 2^{2s}(2^{s\ell} - 1)(2^s - 1) + 4(2^s - 5)(2^s - 1)^2 \\ & + \tfrac{1}{6}(2^s - 5)(2^s - 6)(2^s - 7)(2^s - 1)^3\Big] \times 2^{1-s(\ell+6)} \\ & - 2^{1-s\ell} - (2^{s+3} - 15)(2^{s(\ell+4)} - 2^{4s}) \times 2^{1-s(\ell+6)} \\ & - (2^{s(\ell+2)} - 2^{2s})(2^{s+1} - 1)(2^{2s} - 2^{s+3} + 15) \times 2^{1-s(\ell+6)} \\ & + 4(2^{3s} - 3 \times 2^{2s} + 3 \times 2^s + 1)(2^{2s} - 2^{s+3} + 15) \times 2^{1-s(\ell+6)} \end{aligned}$$

and $-0.075 < \Delta < 1$, for $s \geq 4$ and $\ell > 0$. Therefore, a linear nonbinary two-level (3,1)-error correcting code with parameters as in (3.7) requires at least $\ell + 6$ redundant symbols. For $t = 3$ and $m = 1$, the codes given by Theorem 5 have $2t + \ell = \ell + 6$ redundant symbols and therefore are optimal.

Asymptotic Hamming bound

For $t_1 = t$, $t > 3$ and $t_2 = 1$, the Hamming bound (3.5) becomes practically impossible to evaluate. In this subsection, we derive an asymptotic equivalent for the class of nonbinary LUEP codes given by Theorem 5. From (3.5), we obtain

$$2^{s(n-k)} \geq n(2^s - 1) + 1 + \sum_{j=2}^{t} \binom{k_1}{j} (2^s - 1)^j + (n - k_1) \sum_{j=2}^{t} \binom{k_1}{j-i} (2^s - 1)^j \quad (3.8)$$

We are going to derive an asymptotic equivalent of (3.8), for fixed t and large s. A good lower bound on (3.8) is obtained by taking only the most dominant term,

$$2^{s(n-k)} > (n - k_1) \sum_{j=2}^{t} \binom{k_1}{j-1} (2^s - 1)^j \quad (3.9)$$

A lower bound on the sum of binomial coefficients is [37]

$$\sum_{j=2}^{t} \binom{k_1}{j-1} \geq 2^{k_1 \left\{ H\left(\frac{t-1}{k_1}\right) - \frac{1}{2k_1} \log_2 \left[8k_1 \left(\frac{t-1}{k_1}\right) \left(\frac{k_1 - t + 1}{k_1}\right) \right] \right\}} \quad (3.10)$$

By double application of L'Hopital rule, we can show that

$$\lim_{n \to \infty} \frac{k_1 \left\{ H\left(\frac{t-1}{k_1}\right) - \frac{1}{2k_1} \log_2 \left[8k_1 \left(\frac{t-1}{k_1}\right) \left(\frac{k_1 - t + 1}{k_1}\right) \right] \right\}}{k_1 H\left(\frac{t-1}{k_1}\right)} = 1 - \frac{1}{2(t-1)} \quad (3.11)$$

$$\lim_{k_1 \to \infty} \frac{k_1 H\left(\frac{t-1}{k_1}\right)}{(t-1)\log_2 k_1} = 1 \quad (3.12)$$

In addition,

$$k_1 H\left(\frac{t-1}{k_1}\right) > (t-1)\log_2 k_1 - (t-1)\log_2(t-1) \tag{3.13}$$

and

$$\lim_{k_1\to\infty} \frac{(t-1)\log_2 k_1}{(t-1)\log_2 k_1 + (t-1)\log_2(t-1)} = 1 \tag{3.14}$$

Using (3.11)-(3.14) in (3.10), and the inequality $(2^s-1)^t > 2^{(s-1)t}$, we obtain

$$\begin{aligned} s(n-k) \;&\gtrsim \log_2(n-k_1) \\ &+ \left(1 - \frac{1}{2(t-1)}\right)\left[(t-1)\log_2 k_1 - (t-1)\log_2(t-1)\right] + (s-1)t \end{aligned} \tag{3.15}$$

where $a(n) \sim b(n)$ (read $a(n)$ is asymptotic to $b(n)$) means that

$$\lim_{n\to\infty} \frac{a(n)}{b(n)} = 1.$$

In other words, the expression on the right hand side (RHS) of (3.9) is asymptotic (after taking logarithm base 2) to the RHS of (3.15), and at the same time, the RHS of (3.9) is greater than the RHS of (3.15).

Inequality (3.15) can be rewritten as follows:

$$\begin{aligned} s(n-k) \;&\gtrsim \log_2 n + (t-3/2)\log_2 k_1 + st \\ &+ \log_2(1-k_1/n) - \left[(t-3/2)\log_2(t-1) + t\right]. \end{aligned} \tag{3.16}$$

Now, let $c(t) = (t-3/2)\log_2(t-1) + t$, a constant that depends on t but not on s, n or k_1. For large s, we have that

$$\frac{c(t)}{s} \approx 0.$$

In addition, we assume that $k_1 < cn$, where $0 \le c << 1$. It follows from (3.16) that

$$(n-k) \gtrsim \left\lceil \frac{1}{s}\left[\log_2 n + (t-3/2)\log_2 k_1\right] + t \right\rceil, \tag{3.17}$$

which is the desired asymptotic Hamming lower bound, on the number of redundant symbols of a $(t,1)$-error correcting code, over $GF(2^s)$. For codes with parameters as those in (3.7) we have, for large s,

$$n \gtrsim 2^{s(2m+\ell-1)} \qquad \text{and} \qquad k_1 \gtrsim 2^{sm-1}. \tag{3.18}$$

Let $m = 1$. It follows from (3.17) and (3.18) that the number of redundant symbols has the following asymptotic Hamming lower bound

$$n - k \gtrsim \left\lceil 2t + \ell - \left(\frac{1}{2} + \frac{(t-3/2)}{s} \right) \right\rceil = 2t + \ell,$$

because $(t-3/2)/s \approx 0$. Note that, for $m = 1$, LUEP codes of Theorem 5 have exactly $2t + \ell$ redundant symbols, and thus achieve the Hamming bound. We have shown that the LUEP codes obtained from Theorem 5 are optimal when their parity-check matrices are combinations of parity-check matrices of t-error correcting *Reed-Solomon* codes and parity-check matrices shortened Hamming codes, both over the field $GF(2^s)$, for large s. In Table 3.1, a list of some optimal nonbinary LUEP codes is presented.

3.2 Specifying the generator matrix

In this section, constructions of LUEP codes by appending (or time sharing) cosets of subcodes in linear codes using *very simple* constituent linear codes are proposed. Two specific families of *very good* LUEP codes are presented. These codes are good in the sense of requiring only one to three redundant bits more than LUEP codes achieving the Hamming lower bound, on the number of redundant bits with equality. A simple two-stage decoding is proposed and it is shown that decoding up to the unequal error protection capacity of the LUEP code is possible using only the decodings of the constituent codes. This is an important feature that makes this

Table 3.1: Some optimal $(t, 1)$-error correcting codes over $GF(2^s)$

s	ℓ	n	k	k_1	k_2	t
3	1	71	66	5	61	2
3	2	583	577	5	572	2
4	1	271	266	13	253	2
4	1	271	264	11	253	3
4	2	4367	4361	13	4348	2
4	2	4367	4359	11	4348	3
5	1	1055	1050	29	1021	2
5	1	1055	1048	27	1021	3
5	1	1055	1046	25	1021	4
6	1	4159	4152	59	4093	3
6	1	4159	4150	57	4093	4
6	1	4159	4148	55	4093	5

type of LUEP codes attractive for practical applications. When compared to known constructions of LUEP codes, the proposed new codes are better or very close in terms of number of redundant bits. One important advantage of the new LUEP codes presented in this section, is that they have a two-stage (suboptimal) decoding which uses only the decoding of very simple constituent codes. Finally, the use of LUEP codes as component codes themselves in these constructions is considered. In particular, we analyze the case when an LUEP code is combined, using construction X, with a linear code partitioned into cosets of its linear subcode. Optimal LUEP codes may be obtained in this way. This adds to the attractiveness of designing LUEP codes using constructions based on combining cosets of subcodes in linear codes.

3.2.1 Near-optimal LUEP codes based on construction X

For $i = 1, 2, 3$, let C_i denote a linear (n_i, k_i, d_i) binary code. Assume $C_3 \subseteq C_2$, so that $k_3 \leq k_2$ and $d_3 \geq d_2$. Let C_X be the linear code whose generator matrix is

$$G_X = \begin{pmatrix} G_1 & G_2 \\ 0 & G_3 \end{pmatrix}$$

where G_1, $[G_2^T G_3^T]^T$ and G_3 are the generator matrices of C_1, C_2 and C_3 respectively (Note that it is required that $k_1 = k_2 - k_3$). Then C_X is an $(n_1 + n_3, k_1 + k_3)$ linear code with minimum distance $d_X = \min\{d_3, d_1 + d_2\}$, [32]. This method of combining shorter linear codes to obtain a linear code with increased length and minimum distance is known as *Construction X*. It is interesting to note that C_X can be viewed as dividing C_2 into cosets of its subcode C_3 and appending (or time sharing) a codeword of C_1 to each coset of C_3 in C_2. In this sense, construction X constitutes a generalization of the so called $|\bar{u}|\bar{u} + \bar{v}|$ construction. The UEP capabilities of linear codes, obtained from combining shorter linear codes, using construction X, were analyzed previously in [26, 4] in a different context. The separation vector of C_X is

$$\bar{s} = (s_1, s_2), \quad s_1 \geq d_1 + d_2, \quad s_2 \geq \min\{d_3, d_1 + d_2\}$$

for the message space $M = \{0,1\}^{k_1} \times \{0,1\}^{k_3}$.

In this section, we present two specific families of *near optimal* binary LUEP codes using construction X.

Example 3: Let C_1 be the $(7, 3, 4)$ code, C_2 be the trivial (i.e., no coding) $(7, 7, 1)$ code and C_3 be the $(7, 4, 3)$ Hamming code. Then C_X is a $(14, 7)$ LUEP code with separation vector $\bar{s} = (5, 3)$, for the message space $\{0,1\}^3 \times \{0,1\}^4$. In other words, C_X protects 3 message bits against any 2 errors, while the remaining 4 message bits are protected against any single error. Note that the dimension of C_X is between the

dimensions of a $(14, 8, 3)$ shortened Hamming code and a $(14, 6, 5)$ shortened BCH code. Time sharing an $(11, 3, 5)$ shortened BCH code and a $(7, 4, 3)$ Hamming code, gives an $(18, 7)$ LUEP code with the same message space and separation vector, requiring 4 more redundant bits than code C_X. Note in addition that there exists a $(15, 7)$ shortened generalized concatenated (GC) code [50], with the same separation vector and message space, that requires only one more redundant bit than C_X. $\triangle\triangle$

A family of very good (2,1)-error correcting codes is obtained using construction X and generalizing the previous example as follows. Let code C_2 be the trivial $(2^m - 1, 2^m - 1, 1)$ code (no coding), let code C_3 be a $(2^m - 1, 2^m - 1 - m, 3)$ Hamming code, and let code C_1 be the shortest length linear code obtained from an extended $(2^l, 2^l - l - 1, 4)$ Hamming code, of dimension $k_1 = k_2 - k_3 = m$. In other words, C_1 is a $(2^l - s, 2^l - s - l - 1, 4)$ linear code, where l is such that $2^l - s - l - 1 = m$.

Applying construction X to these three codes, a $(2^m + 2^l - s - 1, 2^m - 1)$ LUEP code C_X with separation vector $\bar{\mathbf{s}} = (5, 3)$ for the message space $\{0,1\}^{k_1} \times \{0,1\}^{k_2}$, where $k_1 = m$ and $k_2 = 2^m - 1 - m$ is obtained.

To evaluate the effectiveness of this family of codes, we use the Hamming lower bound on the number of redundant bits of an LUEP code with the same length, same separation vector $\bar{\mathbf{s}} = (2t_1 + 1, 2t_2 + 1)$, $t_1 \geq t_2$, and same subdimension k_1. Upon substituting the parameters of the family of codes in question in (3.4), we

have that

$$\begin{aligned} 2^{n-k} &\geq (2^m - 1 + 2^l - s) + \tfrac{1}{2}(m^2 - m) + (2^m - 1 + 2^l - s - m)m \\ &= (2^m - 1 + 2^l - s) + \tfrac{1}{2}(m^2 - m) + (2^m - 1 + 2^l - s - m)(2^l - l - s - 1) \\ &= 2^{m+l-1}\{2 + 2^{2l-m-l+1} - 2^{-m-l+1}(2^m + 2^l)(l + s) - (s+1)2^{-m-l+1} \\ &\qquad +(m + s + 1)(l + s + 1)2^{-m-l+1} + \tfrac{1}{2}(m^2 - m)2^{-m-l+1}\} \\ &= 2^{m+l-1}(1 + \Delta), \end{aligned} \tag{3.19}$$

where

$$\begin{aligned} \Delta &= 1 + 2^{l-m+1} - 2^{-m-l+1}(2^m - 2^l)(l - s) - (s+1)2^{-m-l+1} \\ &\qquad +(m + s + 1)(l + s + 1)2^{-m-l+1} + \tfrac{1}{2}(m^2 - m)2^{-m-l+1} \end{aligned}$$

and $0 < \Delta < 1$, given that the shortening length s is such that $0 \leq s \leq 2^{l-1} - l$. Since $m = 2^l - l - s - 1$, it follows from (3.19) that

$$n - k \geq 2^l - s - 1.$$

On the other hand, code C_X requires $(n - k) = 2^l - s$ redundant bits. It follows that code C_X requires only one more redundant bit than an optimal LUEP code (i.e., a code achieving the Hamming bound (3.4) with equality), with the same separation vector $\bar{\mathbf{s}}$ and same subdimension k_1. Table 3.2 lists the first codes from this family of LUEP codes. In Table 3.2, Δ_H is the difference between the number of redundant bits of the LUEP code, and the Hamming bound on the number of redundant bits of an LUEP code with the same parameters. The next Example 4 shows how to obtain a binary (3,2)-error correcting code using construction X.

Example 4: Let C_1 be a $(8, 4, 4)$ extended Hamming code, C_2 be a $(15, 11, 3)$ Hamming code and C_3 be a $(15, 7, 5)$ BCH code. Then C_X is a $(23, 11)$ LUEP code with separation vector $\bar{\mathbf{s}} = (7, 5)$ for the message space $\{0, 1\}^4 \times \{0, 1\}^7$. In

Table 3.2: LUEP codes with $\bar{s} = (5,3)$ based on construction X

m	l	s	C_2	C_3	C_1	C_X	Δ_H
3	3	1	(7,7,1)	(7,4,3)	(7,3,4)	(14,7)	+1
4	3	0	(15,15,1)	(15,11,3)	(8,4,4)	(23,15)	+1
5	4	6	(31,31,1)	(31,26,3)	(10,5,4)	(41,31)	+2
6	4	5	(63,63,1)	(63,57,3)	(11,6,4)	(74,63)	+2
7	4	4	(127,127,1)	(127,120,3)	(12,7,4)	(139,127)	+1
8	4	3	(255,255,1)	(255,247,3)	(13,8,4)	(268,255)	+1
9	4	2	(511,511,1)	(511,502,3)	(14,9,4)	(525,511)	+1

other words, C_X protects 4 message bits against any three errors, while 7 bits are protected against any two errors. Note that the dimension of C_X is between the dimensions of a $(23, 13, 5)$ shortened BCH code and a $(23, 8, 7)$ shortened BCH code. Time sharing a $(14, 4, 7)$ shortened BCH code and a $(15, 7, 5)$ BCH code, results in a $(30, 11)$ LUEP code with the same message space and separation vector, requiring 7 more redundant bits than C_X. It is also interesting to note that there exists a $(35, 11)$ binary cyclic UEP code [29], with the same separation vector and message space, that requires 12 more redundant bits than C_X. $\triangle\triangle$

From construction X, and generalizing Example 4 above, a family of good two-level error correcting codes, with separation vector $\bar{s} = (7,5)$, is constructed as follows: Let C_1 be the shortest length linear code obtained from an extended $(2^l, 2^l - l - 1, 4)$ Hamming code, of dimension $k_1 = k_2 - k_3$, i.e., a $(2^l - s, , m, 4)$ linear code, where l is such that $2^l - l - 1 - s = m$; let C_2 be a $(2^m - 1, 2^m - 1 - m, 3)$ Hamming code; and let C_3 be a $(2^m - 1, 2^m - 1 - 2m, 5)$ double-error correcting BCH code, contained in C_2. Applying construction X to the these three codes, a $(2^m + 2^l - s - 1, 2^m - 1 - m)$ LUEP code C_X with separation vector $\bar{s} = (7,5)$ for the message space $\{0,1\}^m \times \{0,1\}^{2^m - 1 - 2m}$ is obtained.

Table 3.3: LUEP codes with $\bar{s} = (7, 5)$ based on construction X

m	l	s	C_2	C_3	C_1	C_X	Δ_H
4	3	0	(15,11,3)	(15,7,5)	(8,4,4)	(23,11)	0
5	4	6	(31,26,3)	(31,21,5)	(10,5,4)	(41,26)	+1
6	4	5	(63,57,3)	(63,51,5)	(11,6,4)	(74,57)	+2
7	4	4	(127,120)	(127,113,5)	(12,7,4)	(139,120)	+2
8	4	3	(255,247,3)	(255,239,5)	(13,8,4)	(268,255)	+2
9	4	2	(511,502,3)	(511,493,5)	(14,9,4)	(525,502)	+2

For this family of LUEP codes, the Hamming bound (3.4) contains more terms and is difficult to evaluate. For short lengths, these codes are very good, in the sense of requiring only 1 or 2 redundant bits more than optimal codes, as shown in Table 3.3, where a list of (3,2)-error correcting codes based on construction X is presented, and where Δ_H is the same as in Table 3.2.

3.2.2 LUEP codes based on construction X4

The key idea in construction X4 is to do a *double-sided* construction X, using four linear codes, as follows. A linear code C_{X4} is constructed by dividing a linear code C_2 into cosets of its proper subcode C_1, and dividing a second linear code C_4 into cosets of its proper subcode C_3, and appending (or time sharing) a coset of C_1 to a coset of C_3 in all possible ways [41]. For $1 \leq i \leq 4$, let C_i denote a linear (n_i, k_i, d_i) binary code. Let $d_1 \geq d_3$. Assume $C_1 \subseteq C_2$, so that $k_1 \leq k_2$ and $d_1 \geq d_2$, and $C_3 \subseteq C_4$, with $k_3 \leq k_4$ and $d_3 \geq d_4$. Then C_{X4} is the linear code whose generator matrix is

$$G_{X4} = \begin{pmatrix} G_1 & 0 \\ G_2 & G_4 \\ 0 & G_3 \end{pmatrix}$$

where G_1, $[G_1^T G_2^T]^T$, G_3 and $[G_4^T G_3^T]^T$ are the generator matrices of C_1, C_2, C_3 and C_4 respectively Note that $k_2 - k_1 = k_4 - k_3 = \mu$. Code C_{X4} is a $(n_1 + n_3, k_2 + k_3)$

linear code with minimum distance $d_{X4} = \min\{d_1, d_2 + d_4\}$ [41].

Theorem 6 *Code C_{X4} is an $(n_1 + n_3, k_1 + k_3 + \mu)$ 3-level LUEP code with separation vector $\bar{\mathbf{s}} = (\mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3)$, where*

$$\begin{aligned} \mathbf{s}_1 &\geq \min\{d_1, d_2 + d_4\}, \\ \mathbf{s}_2 &\geq d_2 + d_4, \quad \textit{and} \\ \mathbf{s}_3 &\geq \min\{d_3, d_2 + d_4\}, \end{aligned} \tag{3.20}$$

for the message space $\{0,1\}^{k_1} \times \{0,1\}^{\mu} \times \{0,1\}^{k_3}$.

Proof: Let $\bar{\mathbf{m}}$ be a message from $\{0,1\}^{k_1+\mu+k_3}$ to be encoded using code C_{X4}. Divide $\bar{\mathbf{m}}$ into three message parts, $\bar{\mathbf{m}}_1 \in \{0,1\}^{k_1}$, $\bar{\mathbf{m}}_2 \in \{0,1\}^{\mu}$ and $\bar{\mathbf{m}}_3 \in \{0,1\}^{k_3}$. Let $\bar{\mathbf{c}}(\bar{\mathbf{m}})$ be a codeword of C_{X4}. Then

$$\begin{aligned} \bar{\mathbf{c}}(\bar{\mathbf{m}}) = \bar{\mathbf{m}} G_{X4} &= (\bar{\mathbf{m}}_1, \bar{\mathbf{m}}_2, \bar{\mathbf{m}}_3) \begin{pmatrix} G_1 & 0 \\ G_2 & G_4 \\ 0 & G_3 \end{pmatrix} \\ &= (\bar{\mathbf{m}}_1 G_1 + \bar{\mathbf{m}}_2 G_2, \bar{\mathbf{m}}_2 G_4 + \bar{\mathbf{m}}_3 G_3) \end{aligned}$$

and the Hamming weight of $\bar{\mathbf{c}}(\bar{\mathbf{m}})$ is thus given by

$$\mathrm{wt}(\bar{\mathbf{c}}(\bar{\mathbf{m}})) = \mathrm{wt}(\bar{\mathbf{m}}_1 G_1 + \bar{\mathbf{m}}_2 G_2) + \mathrm{wt}(\bar{\mathbf{m}}_2 G_4 + \bar{\mathbf{m}}_3 G_3)$$

By considering all cases where different message parts are nonzero, it is possible to obtain the separation vector of code C_{X4}. We arrange such cases in a table, where $\bar{\mathbf{m}}_i = 1$ indicates that the i-th message part is nonzero, as shown in Table 3.4, where, for $i = 2, 4$, δ_i denotes the minimum distance of a linear (n_i, μ) code D_i whose generator matrix is G_i. Note that $D_i \subset C_i$, so that $\delta_i \geq d_i$.

It follows that the separation vector of C_{X4} is $\bar{\mathbf{s}} = (\mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3)$, where

$$\begin{aligned} \mathbf{s}_1 &\geq \min\{d_1, d_2 + \delta_4, d_1 + d_3, d_2 + d_4\} = \min\{d_1, d_2 + d_4\} \\ \mathbf{s}_2 &\geq \min\{d_2 + \delta_4, d_2 + d_4, \delta_2 + \delta_4, \delta_2 + d_4\} = d_2 + d_4 \\ \mathbf{s}_3 &\geq \min\{d_1 + d_3, d_2 + d_4, d_3, \delta_2 + d_4\} = \min\{d_3, d_2 + d_4\} \end{aligned}$$

Table 3.4: Bounds on the separation vector for construction X4

$\bar{\mathrm{m}}_1$	$\bar{\mathrm{m}}_2$	$\bar{\mathrm{m}}_3$	$\mathrm{wt}(\bar{\mathrm{x}})$	Bound
1	0	0	$\mathrm{wt}(\bar{\mathrm{m}}_1 G_1)$	d_1
1	1	0	$\mathrm{wt}(\bar{\mathrm{m}}_1 G_1 + \bar{\mathrm{m}}_2 G_2) + \mathrm{wt}(\bar{\mathrm{m}}_2 G_4)$	$d_2 + \delta_4$
1	0	1	$\mathrm{wt}(\bar{\mathrm{m}}_1 G_1) + \mathrm{wt}(\bar{\mathrm{m}}_3 G_3)$	$d_1 + d_3$
1	1	1	$\mathrm{wt}(\bar{\mathrm{m}}_1 G_1 + \bar{\mathrm{m}}_2 G_2) + \mathrm{wt}(\bar{\mathrm{m}}_2 G_4 + \bar{\mathrm{m}}_3 G_3)$	$d_2 + d_4$
0	1	0	$\mathrm{wt}(\bar{\mathrm{m}}_2 G_2) + \mathrm{wt}(\bar{\mathrm{m}}_2 G_4)$	$\delta_2 + \delta_4$
0	0	1	$\mathrm{wt}(\bar{\mathrm{m}}_3 G_3)$	d_3
0	1	1	$\mathrm{wt}(\bar{\mathrm{m}}_2 G_2) + \mathrm{wt}(\bar{\mathrm{m}}_2 G_4 + \bar{\mathrm{m}}_3 G_3)$	$\delta_2 + d_4$

$\triangle\triangle$

Code C_{X4} may or may not be a three-level LUEP code. This will depend on the relationship between the minimum distances of the component codes. In other words, in order for C_{X4} to have unequal error protection capabilities, we require that $d_1 < d_2 + d_4$, or $d_3 < d_2 + d_4$, Suppose $d_1 \geq d_2 + d_4$ and $d_3 < d_2 + d_4$. Then C_{X4} will be a *two-level* LUEP code with separation vector $\bar{\mathrm{s}} = (\mathrm{s}_1, \mathrm{s}_2)$, where $\mathrm{s}_1 \geq d_2 + d_4$ and $\mathrm{s}_2 \geq d_3$, for the message space $\{0,1\}^{k_1+\mu} \times \{0,1\}^{k_3}$. Example 5 below illustrates a code belonging to this case. On the other hand, if both $d_1 < d_2 + d_4$ and $d_3 < d_2 + d_4$, then C_{X4} will be a *three-level* LUEP codes with separation vector $\bar{\mathrm{s}} = (\mathrm{s}_1, \mathrm{s}_2, \mathrm{s}_3)$, where $\mathrm{s}_1 \geq d_2 + d_4$, $\mathrm{s}_2 \geq d_1$ and $\mathrm{s}_3 \geq d_3$, for the message space $\{0,1\}^{\mu} \times \{0,1\}^{k_1} \times \{0,1\}^{k_3}$.

Example 5: Let C_2 be a $(31, 16, 7)$ BCH code and C_1 be a $(31, 11, 11)$ BCH code. Then $\mu = 5$. So let C_4 be a $(31, 30, 2)$ even-weight code and C_3 be a $(31, 25, 4)$ shortened even-weight Hamming code. Then C_{X4} is a $(62, 41)$ LUEP code with separation vector $\bar{\mathrm{s}} = (9, 4)$ for the message space $M = \{0,1\}^{16} \times \{0,1\}^{25}$. Note that the dimension of C_{X4} is between the dimensions of a $(62, 55, 4)$ shortened even-weight Hamming code and a $(62, 38, 9)$ shortened BCH code. By time sharing a

$(36, 16, 9)$ linear code (obtained from shortening a $(37, 17, 9)$ quadratic residue code, from the table of best linear codes in [3]) and a $(31, 25, 4)$ shortened even-weight Hamming code, we obtain a $(67, 41)$ LUEP code with the same separation vector and same message space that requires 5 more redundant bits than C_{X4}. Note also that there exists a $(63, 43)$ binary cyclic UEP code [10], which when shortened results in a $(61, 41)$ LUEP code with the same separation vector and message space, that requires one redundant bit less than C_{X4}. $\triangle\,\triangle$

Although the code from Example 5 above has one more redundant bit than an LUEP code obtained from shortening a binary cyclic UEP code, it has the advantage of very simple component codes, and can be decoded using a two-stage decoding (see section 3.2.4 below).

3.2.3 Dual codes of optimum binary LUEP codes

The class of linear codes obtained by construction X4, contains the duals of the optimum binary LUEP codes discussed in section 3.1.1. Let C be a binary LUEP code with separation vector $\bar{\mathrm{s}} = (5, 3)$ and parity check matrix (3.1),

$$H(2) = \left(\begin{array}{cccc|ccc} 1 & \alpha & \cdots & \alpha^{2^m-2} & 0_m & \cdots & 0_m \\ 1 & \alpha^3 & \cdots & \alpha^{3(2^m-2)} & & & \\ 0_\ell & 0_\ell & \cdots & 0_\ell & \beta_1 & \cdots & \beta_{2^{m+\ell}-2^m} \end{array} \right),$$

where each power of α is represented by a binary column vector of length m, 0_m represents the all zero column vector of length m, and, for $\ell \geq 0$, $\beta_1, \cdots, \beta_{2^{m+\ell}-2^m}$ represent all binary column vectors of length $m + \ell$ having the last ℓ entries not all zero.

Comparing the above matrix with the generator matrix of a linear code based on construction X4,

$$G_{X4} = \begin{pmatrix} G_1 & 0 \\ G_2 & G_4 \\ 0 & G_3 \end{pmatrix},$$

it is clear that the dual code of code C has generator matrix of this form, where

$$G_1 = (\,1 \quad \alpha \quad \alpha^2 \quad \cdots \quad \alpha^{2^m-2}\,)$$

generates a simplex (or maximal-length) $(2^m-1, m, 2^{m-1})$ code,

$$\begin{pmatrix} G_1 \\ G_2 \end{pmatrix} = \begin{pmatrix} 1 & \alpha & \alpha^2 & \cdots & \alpha^{2^m-2} \\ 1 & \alpha^3 & \alpha^6 & \cdots & \alpha^{3(2^m-2)} \end{pmatrix}$$

generates the dual of a double-error correcting BCH code, i.e., a $(2^m-1, 2m, d_2)$ linear code, with minimum distance

$$d_2 = \begin{cases} 2^{m-1} - 2^{(m+1)/2-1}, & m \geq 2 \quad \text{odd;} \\ 2^{m-1} - 2^{(m+2)/2-1}, & m \geq 4 \quad \text{even.} \end{cases}$$

The matrix

$$G_3 = (\,1 \quad \cdots \quad 1 \quad | \quad \alpha \quad \alpha \quad \cdots \quad \alpha \quad | \quad \cdots \quad | \quad \alpha^{2^m-2} \quad \cdots \quad \alpha^{2^m-2}\,)$$

generates a linear code whose codewords consist of 2^m copies of a codeword in a $(2^\ell - 1, \ell, 2^{\ell-1})$ simplex code. Therefore,

$$d_3 = 2^m(2^{\ell-1}) = 2^{m+\ell-1}.$$

Finally, matrix

$$\begin{pmatrix} G_4 \\ G_3 \end{pmatrix} = \begin{pmatrix} 0 & 1 & \alpha & \cdots & \alpha^{2^m-2} & | & \cdots & | & 0 & 1 & \alpha & \cdots & \alpha^{2^m-2} \\ 1 & 1 & 1 & \cdots & 1 & | & \cdots & | & \beta^{2^\ell-2} & \beta^{2^\ell-2} & \beta^{2^\ell-2} & \cdots & \beta^{2^\ell-2} \end{pmatrix}$$

generates a $(2^{m+\ell} - 2^m, m+\ell, d_4)$ linear code which consists of time-sharing $2^\ell - 1$ translates of a $(2^m, m, 2^{m-1})$ (extended) simplex code. As a result, the minimum distance is given by

$$d_4 = (2^\ell - 1)2^{m-1} = 2^{m+\ell-1} - 2^{m-1}.$$

On the other hand, from matrix G_4 it follows that $\delta_4 = d_4$, while

$$G_2 = (\,1 \quad \alpha^3 \quad \alpha^6 \quad \cdots \quad \alpha^{3(2^m-2)}\,)$$

generates a $(2^m - 1, m, 2^{m-1})$ simplex code, and $\delta_2 = (2^\ell - 1)2^{m-1} = \delta_4$.

From the results of section 3.2.2, we have that the components of the separation vector of the dual code of an optimal binary LUEP code with $\bar{s} = (5, 3)$ are,

$$s_1 \geq \begin{cases} \min\{2^{m-1}, 2^{m-1} - 2^{(m+1)/2-1} + 2^{\ell+m-1} - 2^{m-1}\}, & m \geq 3 \quad \text{odd}; \\ \min\{2^{m-1}, 2^{m-1} - 2^{(m+2)/2-1} + 2^{\ell+m-1} - 2^{m-1}\}, & m \geq 4 \quad \text{even}. \end{cases}$$

$$= \begin{cases} \min\{2^{m-1}, 2^{\ell+m-1} - 2^{(m+1)/2-1}\}, & m \geq 3 \quad \text{odd}; \\ \min\{2^{m-1}.2^{\ell+m-1} - 2^{(m+2)/2-1}\}, & m \geq 4 \quad \text{even}. \end{cases}$$

$$s_2 \geq \begin{cases} \min\{2^{m+\ell-1} - 2^{m-1}, 2^{m-1} - 2^{(m+1)/2-1} + 2^{m+\ell-1} - 2^{m-1}, \\ \qquad 2^{m+\ell-1} - 2^{m-1} + 2^{m+\ell-1} - 2^{m-1}\}, & m \geq 3, \text{ odd}; \\ \min\{2^{m+\ell-1} - 2^{m-1}, 2^{m-1} - 2^{(m+2)/2-1} + 2^{m+\ell-1} - 2^{m-1}, \\ \qquad 2^{m+\ell-1} - 2^{m-1} + 2^{m+\ell-1} - 2^{m-1}\}, & m \geq 4, \text{ even}. \end{cases}$$

$$= 2^{m+\ell-1} - 2^{m-1}, \quad \text{and}$$

$$s_3 \geq \min\{2^{m+\ell-1}, 2^{m+\ell-1} - 2^{m-1}\} = 2^{m+\ell-1} - 2^{m-1}.$$

It follows that C_{X4} is a $(2^{m+\ell} - 2^m, 2m + \ell)$ two-level LUEP code with separation vector $\bar{s} = (s_1, s_2)$, where

$$s_1 \geq \begin{cases} \min\{2^{m-1}, 2^{m+\ell-1} - 2^{(m+1)/2-1}\}, & m \geq 3 \quad \text{odd}; \\ \min\{2^{m-1}, 2^{m+\ell-1} - 2^{(m+2)/2-1}\}, & m \geq 4 \quad \text{even}. \end{cases}$$

$$s_2 \geq 2^{m+\ell-1} - 2^{m-1}$$

for the message space $M = \{0,1\}^m \times \{0,1\}^{m+\ell}$.

Tables 3.5 and 3.6 show specific examples of duals of optimal binary LUEP codes with separation vector $\bar{s} = (5, 3)$. The variables in the tables are those from Theorem 3 and the following,

$k_d = k_{d_1} + k_{d_2}$: dimension of the dual code,

s_{d_1}, s_{d_2} : components of separation vector of the dual code, for the message space $M = \{0,1\}^{k_{d_1}} \times \{0,1\}^{k_{d_2}}$.

Table 3.5: Length 31 codes

m	ℓ	k	k_1	k_2	k_d	k_{d_1}	k_{d_2}	s_{d_1}	s_{d_2}
0	5	26	0	26	5	0	5	-	16
2	3	24	1	23	7	2	5	2	14
3	2	23	4	19	8	3	5	4	12
4	1	22	11	11	9	4	5	8	8
5	0	21	21	0	10	5	5	12	-

Table 3.6: Length 63 codes

m	ℓ	k	k_1	k_2	k_d	k_{d_1}	k_{d_2}	s_{d_1}	s_{d_2}
0	6	57	0	57	6	0	6	-	32
2	4	55	1	54	8	2	6	2	30
3	3	54	4	50	9	6	3	4	28
4	2	53	11	42	10	4	6	8	21
5	1	52	26	26	11	5	6	16	16
6	0	51	51	0	12	6	6	24	-

For example, the dual code of the (31,24) binary optimal LUEP code, protecting 1 bit against any two or less random errors and 23 bits against any single random error, is a (31,7) binary LUEP code that protects 5 bits against any six or less random errors and detects an error in the remaining 2 information bits.

3.2.4 Multi-stage decoding of constructions X and X4

Constructions X and X4 have been provided with decoding procedures in the original paper [41]. When these constructions yield LUEP codes, the decoding will be carried out in two stages, with the first decoding stage used to decode the most important message part [4]. For example, in construction X the separation vector is $\bar{\mathbf{s}} = (\mathbf{s}_1, \mathbf{s}_2)$, with $\mathbf{s}_1 \geq d_1 + d_2$ and $\mathbf{s}_2 \geq \min\{d_3, d_1 + d_2\}$. If $d_3 < d_1 + d_2$, then we have an LUEP code which can be decoded first by using decoders for codes C_1 and C_2 in the first decoding stage, to correct up to $\lfloor (d_1 + d_2 - 1)/2 \rfloor$ errors, and then use the decoded first message part and a decoder for subcode C_3 to decode the least significant message part, which is protected against up to $\lfloor (d_3 - 1)/2 \rfloor$ errors. In the case of construction X4, decoding will also be carried out in two stages. In the first decoding stage, decoders for codes C_2 and C_4 will be used to recover the most significant message part, which is protected against $\lfloor (d_2 + d_4 - 1)/2 \rfloor$ errors. The second decoding stage uses the decoded first message part and a decoder for subcode C_1, or a decoder for subcode C_3, or both, to estimate the least significant message part.

Recently, a coset decoding [17] has been proposed for codes based on the so called $|\bar{\mathbf{u}}|\bar{\mathbf{u}} + \bar{\mathbf{v}}|$ construction. This method can be applied for constructions X and X4, which are also based on partitioning a linear code into cosets of its subcode, and either appending codewords of another code (as in construction X), or appending cosets of a subcode in another linear code (as in construction X4). If the component

codes in constructions X and X4 are majority logic decodable, then the resulting LUEP code will also be majority logic decodable. This can be shown as follows. Consider construction X. For $i = 1, 2$, let code C_i be majority-logic decodable in L_i steps. Suppose $\bar{\mathbf{x}} = \bar{\mathbf{m}}G$ is a codeword of LUEP code C_X. Then $\bar{\mathbf{x}} = (\bar{\mathbf{x}}_1, \bar{\mathbf{x}}_2)$, where $\bar{\mathbf{x}}_1 \in C_1$ and $\bar{\mathbf{x}}_2 \in C_2$. Let $\bar{\mathbf{r}} = (\bar{\mathbf{r}}_1, \bar{\mathbf{r}}_2)$ be the received vector. Let

$$\bar{\mathbf{m}}_1 = \phi_k^{(1)}(\bar{\mathbf{r}}_1), \qquad 1 \leq k \leq d_1 - 1$$

$$\bar{\mathbf{m}}_1 = \phi_k^{(2)}(\bar{\mathbf{r}}_2), \qquad 1 \leq k \leq d_2 - 1$$

be a set of $d_1 + d_2 - 2$ equations orthogonal on the first message part $\bar{\mathbf{m}}_1$, based on codes C_1 and C_2, respectively. To obtain the $d_1 + d_2$ equations required for majority logic decoding, we add the trivial equations $\bar{\mathbf{m}}_1 = \bar{\mathbf{m}}_1^{(i)}$, where $\bar{\mathbf{m}}_1^{(i)}$ is the estimated first message part based on code C_i, for $i = 1, 2$. Once the first message part $\bar{\mathbf{m}}_1$ has been decoded, we form the message $(\bar{\mathbf{m}}_1, \bar{\mathbf{0}})$ to obtain a codeword $\bar{\mathbf{c}}_2 \in C_2$. We then proceed as indicated in [3], subtracting $\bar{\mathbf{c}}_2$ from the second part of received vector, $\bar{\mathbf{r}}_2$, and decoding the resulting vector $\bar{\mathbf{r}}_2 - \bar{\mathbf{c}}_2$ using the decoding (which may or may not be majority logic decoding) for code C_3 to obtain an estimate of the least significant message part $\bar{\mathbf{m}}_2$.

Consider now decoding of construction X4. Assume, as in Example 5, that $d_1 \geq d_2 + d_4$ and $d_3 < d_2 + d_4$. For $i = 2, 4$, let code C_i be majority-logic decodable in L_i steps. Let $\bar{\mathbf{x}} = (\bar{\mathbf{x}}_1, \bar{\mathbf{x}}_2)$, $\bar{\mathbf{x}}_1 \in C_2$, $\bar{\mathbf{x}}_2 \in C_4$, be the transmitted codeword, and let $\bar{\mathbf{r}} = (\bar{\mathbf{r}}_1, \bar{\mathbf{r}}_2)$ be the received vector. Let

$$(\bar{\mathbf{m}}_1, \bar{\mathbf{m}}_2) = \phi_k^{(2)}(\bar{\mathbf{r}}_1), \qquad 1 \leq k \leq d_2 - 1$$

$$(\bar{\mathbf{m}}_1, \bar{\mathbf{m}}_2) = \phi_k^{(4)}(\bar{\mathbf{r}}_2), \qquad 1 \leq k \leq d_4 - 1$$

be a set of $d_2 + d_4 - 2$ equations orthogonal to message part $(\bar{\mathbf{m}}_1, \bar{\mathbf{m}}_2) \in \{0, 1\}^{k_1 + \mu}$, based on codes C_2 and C_4 respectively. To obtain the required $d_2 + d_4$ orthogonal equations for majority-logic decoding, we add the trivial equations

$$(\bar{\mathbf{m}}_1, \bar{\mathbf{m}}_2) = (\bar{\mathbf{m}}_1, \bar{\mathbf{m}}_2)^{(i)},$$

where $(\bar{\mathbf{m}}_1, \bar{\mathbf{m}}_2)^{(i)}$ is the estimated message part based on code C_i, for $i = 2, 4$. Once message part $(\bar{\mathbf{m}}_1, \bar{\mathbf{m}}_2)$ is estimated at the first stage, we form message $(\bar{\mathbf{m}}_2, \bar{\mathbf{0}})$ to obtain a codeword $\bar{\mathbf{c}}_4 \in C_4$, and decode vector $\bar{\mathbf{r}}_2 - \bar{\mathbf{c}}_4$ using the decoder for subcode C_3 to obtain an estimate of the least significant part, $\bar{\mathbf{m}}_3$. Finally, if $d_1 < d_2 + d_4$ and $d_3 < d_2 + d_4$, then at the first stage of decoding we use the following set of $d_2 + d_4$ equations orthogonal to message part $\bar{\mathbf{m}}_2$:

$$\begin{aligned}
\bar{\mathbf{m}}_2 &= \phi_k^{(2)}(\bar{\mathbf{r}}_1), \qquad 1 \leq k \leq d_2 - 1 \\
\bar{\mathbf{m}}_2 &= \phi_k^{(4)}(\bar{\mathbf{r}}_2), \qquad 1 \leq k \leq d_4 - 1 \\
\bar{\mathbf{m}}_2 &= \bar{\mathbf{m}}_2^{(2)} \\
\bar{\mathbf{m}}_2 &= \bar{\mathbf{m}}_2^{(4)}
\end{aligned}$$

where as before, for $i = 2, 4$, $\bar{\mathbf{m}}_2^{(i)}$ indicates the estimated message part $\bar{\mathbf{m}}_2$ using code C_i. At the second decoding stage, we form message $(\bar{\mathbf{0}}, \bar{\mathbf{m}}_2)$ to obtain a codeword $\bar{\mathbf{c}}_2 \in C_2$, and form message $(\bar{\mathbf{m}}_2, \bar{\mathbf{0}})$ to obtain a codeword $\bar{\mathbf{c}}_4 \in C_4$. We then decode vector $\bar{\mathbf{r}}_1 - \bar{\mathbf{c}}_2$ using the decoding for subcode C_1, and decode vector $\bar{\mathbf{r}}_2 - \bar{\mathbf{c}}_4$ using the decoding for subcode C_3, to obtain an estimate for message parts $\bar{\mathbf{m}}_1$ and $\bar{\mathbf{m}}_3$, respectively.

It is interesting to point out that in all of the above two-stage decodings, the first decoding stage is equivalent to finding to which coset the received vector is closest, while the second decoding stage finds a codeword, within that coset, which is as close to the received vector as any other codeword.

3.2.5 Construction X using an LUEP code as component code

Given optimal (or good) LUEP codes, it may possible to obtain longer LUEP codes, with increased separation vector, by combining these shorter LUEP codes with linear

codes. This was done in [49] for the $|\bar{\mathbf{u}}|\bar{\mathbf{u}}+\bar{\mathbf{v}}|$ construction. In this section, we consider using construction X to obtain good LUEP codes based on shorter LUEP codes.

Let C_1 be an $(n_1, k_1 = k_{11} + k_{12})$ LUEP code with separation vector $\bar{\mathbf{s}} = (\mathbf{s}_1, \mathbf{s}_2)$ for the message space $M = \{0,1\}^{k_{11}} \times \{0,1\}^{k_{12}}$. Then C_1 has generator matrix of the form

$$G_1 = \begin{pmatrix} G_{11} \\ G_{12} \end{pmatrix}$$

where G_{11} and G_{12} are k_{11}-by-n_1 and k_{12}-by-n_1 matrices respectively. Let C_2 and C_3 be (n_2, k_2, d_2) and (n_2, k_3, d_3) linear codes such that $C_3 \subset C_2$. Combining the three codes above using construction X, we obtain an $(n_1 + n_2, k_2 = k_{11} + k_{12} + k_3)$ LUEP code C_X.

Separation vector

Again, we need to consider cases with message parts $\bar{\mathbf{m}}_1 \in \{0,1\}^{k_{11}}$, $\bar{\mathbf{m}}_2 \in \{0,1\}^{k_{12}}$ and $\bar{\mathbf{m}}_3 \in \{0,1\}^{k_3}$ nonzero. We start by writing the generator matrix of C_X as

$$G_X = \begin{pmatrix} G_{11} & G_{21} \\ G_{12} & G_{22} \\ 0 & G_3 \end{pmatrix}$$

for some G_{21} and G_{22} such that $G_2 = [G_{21}^{\mathsf{T}} G_{22}^{\mathsf{T}}]^{\mathsf{T}}$, where $[G_{21}^{\mathsf{T}} G_{22}^{\mathsf{T}} G_3^{\mathsf{T}}]^{\mathsf{T}}$ and G_3 are the generator matrices of codes C_2 and C_3 respectively. Then we form a table listing all cases where the message parts are nonzero, just as we did for construction X4. Let $\bar{\mathbf{x}}$ be a codeword of C_X. Again, let $\bar{\mathbf{m}}_i = 1$ indicate the i-th message part being nonzero. Table 3.7 shows the resulting cases.

As a result, the separation vector of C_X is $\bar{\mathbf{s}'} = (\mathbf{s}'_1, \mathbf{s}'_2, \mathbf{s}'_3)$, where

$$\mathbf{s}'_1 \geq s_1 + d_2 \quad \mathbf{s}'_2 \geq s_2 + d_2 \quad \mathbf{s}'_3 \geq \min\{d_3, s_2 + d_2\}$$

for the message space $M = \{0,1\}^{k_{11}} \times \{0,1\}^{k_{12}} \times \{0,1\}^{k_3}$.

Table 3.7: Bounds on the separation vector for construction X

$\bar{m}_1$	$\bar{m}_2$	$\bar{m}_3$	$\mathrm{wt}(\bar{x})$	Bound
1	0	0	$\mathrm{wt}(\bar{m}_1 G_{11}) + \mathrm{wt}(\bar{m}_1 G_{21})$	$s_1 + d_2$
1	1	0	$\mathrm{wt}((\bar{m}_1, \bar{m}_2) G_1) + \mathrm{wt}((\bar{m}_1, \bar{m}_2) G_2)$	$s_1 + d_2$
1	0	1	$\mathrm{wt}(\bar{m}_1 G_{11}) + \mathrm{wt}(\bar{m}_1 G_{21} + \bar{m}_3 G_3)$	$s_1 + d_2$
1	1	1	$\mathrm{wt}((\bar{m}_1, \bar{m}_2) G_1) + \mathrm{wt}((\bar{m}_1, \bar{m}_2) G_2 + \bar{m}_3 G_3)$	$s_1 + d_2$
0	1	0	$\mathrm{wt}(\bar{m}_2 G_{12}) + \mathrm{wt}(\bar{m}_2 G_{22})$	$s_2 + d_2$
0	0	1	$\mathrm{wt}(\bar{m}_3 G_3)$	d_3
0	1	1	$\mathrm{wt}(\bar{m}_2 G_{12}) + \mathrm{wt}(\bar{m}_2 G_{22} + \bar{m}_3 G_3)$	$s_2 + d_2$

Example 6: Let C_1 be an $(5, 3)$ optimal LUEP code with separation vector $\bar{s} = (3, 2)$ for the message space $\{0,1\}^1 \times \{0,1\}^2$. Let C_2 be a $(15, 14, 2)$ even parity code and C_3 be a $(15, 11, 3)$ Hamming code. Then C_X is a $(20, 14)$ *optimal* LUEP code with separation vector $\bar{s} = (5, 4, 3)$ for the message space $\{0,1\}^1 \times \{0,1\}^2 \times \{0,1\}^{11}$. Note that C_X has dimension between the dimensions of a $(20, 10, 5)$ shortened BCH code and a $(20, 15, 3)$ shortened Hamming code. Time sharing a $(5, 1, 5)$ repetition code, a $(6, 2, 4)$ shortened even-weight Hamming code and a $(15, 11, 3)$ Hamming code, we obtain a $(26, 14)$ LUEP code with the same separation vector and same message space that requires 6 more redundant bits than C_X. $\triangle\ \triangle$

3.3 Summary

In this chapter, we presented construction methods of linear multi-level error correcting codes, by specifying their parity-check matrices or their generator matrices. A new family of optimal nonbinary LUEP codes was introduced. These optimal nonbinary LUEP codes are specified by their parity-check matrices, which are a combination of parity-check matrices of RS codes and parity-check matrices of shortened nonbinary Hamming codes. It was shown that these codes are optimal in the

sense of achieving the Hamming lower bound on the number of redundant symbols. These nonbinary LUEP codes have minimum distance 3 and in addition provide an error correcting capability against t or less random errors for part of the message symbols. We also presented constructions of LUEP codes by appending (or time sharing) cosets of linear subcodes in linear codes. These codes are constructed by specifying their generator matrices. Two new families of efficient binary two-level error correcting codes were introduced. These new families of LUEP codes were obtained by combining very simple constituent codes, such as Hamming codes, extended Hamming codes, and 2-error correcting BCH codes. We obtained binary LUEP codes of minimum distance 3, protecting some information bits against 2 or less random errors, whose number of redundant bits is only one more than the number of redundant bits required by optimal LUEP codes of the same separation vector and message space. We also introduced short to moderate length binary LUEP codes of minimum distance 5, protecting some information bits against 3 or less random errors. These LUEP codes require only one to two more redundant bits than optimal LUEP codes. The duals of some optimal binary LUEP codes were characterized, and their multi-level error correcting capabilities determined. In addition, these generator matrix based constructions lend themselves quite naturally to multi-stage decoding methods. In this section we have presented a multi-stage decoding method, using majority logic decoding of constituent codes, for LUEP codes based on specifying their generator matrices.

Chapter 4

Analysis of known linear codes

The analysis of well known families of linear codes, such as BCH and Hamming codes, is presented in this chapter to determine conditions for these linear codes to be unequal error protection codes.

4.1 Binary nonprimitive BCH codes

One way of constructing UEP codes is by *direct sum.* Let C be a linear block code and $\bar{\mathbf{c}}(\bar{\mathbf{m}})$ a codeword in C, corresponding to a message $\bar{\mathbf{m}} \in M$. Let

$$C_i = \{\bar{\mathbf{c}}(\bar{\mathbf{m}}_i) : \bar{\mathbf{m}}_i \in M_i\}$$

be a linear block code for the message space M_i, for $1 \leq i \leq \ell$. Suppose $C_1, C_2, \cdots, C_\ell$ satisfy the following conditions:

(1) $C_i \bigcap C_j = \{\bar{0}\}, \quad i \neq j,$

(2) $\bar{\mathbf{c}}(\bar{\mathbf{m}}_1) + \bar{\mathbf{c}}(\bar{\mathbf{m}}_2) + \cdots + \bar{\mathbf{c}}(\bar{\mathbf{m}}_\ell) = \bar{\mathbf{c}}(\bar{\mathbf{m}}'_1) + \bar{\mathbf{c}}(\bar{\mathbf{m}}'_2) + \cdots + \bar{\mathbf{c}}(\bar{\mathbf{m}}'_\ell) \iff \bar{\mathbf{m}}_i = \bar{\mathbf{m}}'_i.$

Then

$$C = \{\bar{\mathbf{c}}(\bar{\mathbf{m}}_1) + \bar{\mathbf{c}}(\bar{\mathbf{m}}_2) + \cdots + \bar{\mathbf{c}}(\bar{\mathbf{m}}_\ell) : \bar{\mathbf{c}}(\bar{\mathbf{m}}_i) \in C_i, \quad 1 \leq i \leq \ell\}$$

is the *direct sum* of $C_1, C_2, \cdots, C_\ell$, denoted

$$C = C_1 \oplus C_2 \oplus \cdots \oplus C_\ell.$$

In [26], it was shown that if the minimum weight of codewords in

$$C - \{C_{i+1} \oplus C_{i+2} \oplus \cdots \oplus C_\ell\} \tag{4.1}$$

is *at least* d_i, and $d_1 \geq d_2 \geq \cdots \geq d_\ell$, then C is an ℓ-level LUEP code with separation vector $\bar{\mathbf{s}} = (\mathbf{s}_1, \mathbf{s}_2, \cdots, \mathbf{s}_\ell)$, where $\mathbf{s}_i \geq d_i$, for $1 \leq i \leq \ell$. Using the terminology of chapter 2, the set given by (4.1) is an i-th level cloud.

4.1.1 Binary cyclic codes of composite length

Let $n = n_1 n_2$, where n_1 and n_2 are two odd positive and relatively prime integers, and let α be an n-th root of unity from the field $GF(2^q)$. Then $\beta = \alpha^{n_1}$ and $\gamma = \alpha^{n_2}$ are elements of orders n_2 and n_1 respectively. For $i = 1, 2$, let C_i be an (n, k_i) binary cyclic codes with generator polynomial $g_i(x)$ and parity-check polynomial $h_i(x) = (x^n + 1)/g_i(x)$.

Suppose $h_1(x)$ and $h_2(x)$ are relatively prime in $GF(2)[x]$ modulo $(x^n + 1)$. Then the only polynomial common to C_1 and C_2 is $c(x) = 0$, and the direct sum $C = C_1 \oplus C_2$ is an $(n, k_1 + k_2)$ binary cyclic code with generator polynomial

$$g(x) = \gcd\{g_1(x), g_2(x)\}$$

and parity-check polynomial $h(x) = h_1(x)h_2(x)$.

Since $\gcd(n_1, n_2) = 1$, there exist two unique integers ℓ and m such that any element in $GF(2^q)$ can be uniquely represented as $\alpha^\rho = \beta^\ell \gamma^m$, where $0 \leq \rho < n$, $0 \leq \ell \leq n_2$ and $0 \leq m < n_1$. For $i = 1, 2$, let $c_i(x)$ be a code polynomial of C_i. Then

$$c_i = \sum_{j=0}^{n_2 - 1} A_j(x) X^j,$$

where

$$A_j(x) = \sum_{i=0}^{n_1 - 1} a_{n_2 i + j} x^{n_2 i}, \quad a_{n_2 i + j} \in GF(2).$$

It can be verified that

$$c_i(\alpha^\rho) = a_i^{(m)}(\beta^\ell),$$

where

$$a_i^{(m)}(x) = \sum_{j=0}^{n_2} A_j(\gamma^m)\gamma^{mj}x^j.$$

Let $i = 1, 2$ and $0 \leq m < n_1$. Let $V^{(m)}(C_i)$ be a cyclic code over $GF(2^{q_1})$ of length n_2 and zeros $\{\beta^\ell : 0 \leq \ell < n_2, c_i(\alpha^\rho) = a_i^{(m)}(\beta^\ell) = 0\}$, where q_1 is the multiplicative order of 2 modulo n_1 and $c_i(x) \in C_i$. Let $d^{(m)}(C_i)$ denote the minimum distance of $V^{(m)}(C_i)$. Associated with a code polynomial $c_i(x)$, define the following sets,

$$J(C_i) = \{m : 0 \leq m < n_1, \ \ c_i(\alpha^\rho) = a_i^{(m)}(\beta^\ell) = 0, \ \ 0 \leq \ell < n_2\} \qquad (4.2)$$

$$\bar{J}(C_i) = \{0, 1, \cdots, n_1\} - J(C_i)$$

$$J_i = \bigcap_{c_i(x) \neq 0 \in C_i} J(C_i)$$

$$\bar{J}_i = \{0, 1, \cdots, n_1 - 1\} - J_i$$

Let code

$$V^{(m)} = \bigcup_{c_i(x) \neq 0 \in C_i} V(m)(C_i)$$

have minimum distance $d_i^{(m)}$. Let

$$D_i = \min_{m \in \bar{J}_i} \{d_i^{(m)}\}.$$

Then $d^{(m)}(C_i) \geq d_i^{(m)} \geq D_i$, and there are at least D_i nonzero polynomials $A_j(x)$, $0 \leq j < n_2$, associated to an arbitrary nonzero polynomial $c_i(x) \in C_i$. Based on

$$W_i = \bigcup_{c_i(x) \neq 0 \in C_i} W(C_i),$$

where $W(C_i)$ is a binary cyclic code of length n_1 with zeros $\{(\gamma^{n_2})^m : m \in J(C_1)\}$, and $J(C_i)$ is given by (4.2). Then $W = W_1 \oplus W_2$ is a binary code associated with $C = C_1 \oplus C_2$. Let d, d_1 and d_2 be the minimum distances of codes W, W_1 and W_2 respectively. Then $d_1 \geq d$, $d_2 \geq d$ and it can be shown that if $c(x) \neq 0$ and

(1) if $c(x) \in C - C_2$, then $\mathrm{wt}(c(x)) \geq D_1 d$, and

(2) if $c(x) \in C_2$, then $\mathrm{wt}(c(x)) \geq D_2 d$,

where $\mathrm{wt}(c(x))$ denotes the Hamming weight of $c(x) \in C$. If $D_1 d \geq D_2 d_2$, then C is a two-level binary cyclic UEP code in which k_i information bits are protected against any combination of $\lfloor (\mathrm{s}_i - 1)/r \rfloor$ or less errors, where $\mathrm{s}_1 \geq D_1 d$ and $\mathrm{s}_2 \geq D_2 d_2$.

With the aid of a computer, a systematic search for good two-level cyclic UEP codes of composite length n, $n \leq 75$, was performed. The BCH lower bound was used to estimate the minimum distance of the associated code. In this search, a code is good if its minimum distance $d_{\min}$ is greater than or equal to the largest minimum distance d^* among all binary cyclic codes of the same length and dimension. The true minimum distance of codes was determined from [37]. Some good cyclic UEP codes of composite length found in our search are presented in table 4.1.

4.1.2 A class of two-level UEP BCH codes

Let C_1 be a $(3p, 2p+1)$ modified Fire code with generator polynomial

$$g_1(x) = (x^p + 1)\phi_1(x),$$

where $\phi_1(x)$ is the minimal polynomial of α, a $3p$-th root of unity in the field $GF(2^s)$. Let C_2 be a $(3p, p)$ repetition code with parity check polynomial $h_2(x) = x^p + 1$. Note that $h_1(x) = (x^{3p} + 1)/g_1(x)$ and that $h_1(x)$ and $h_2(x)$ are relatively prime. The direct sum $C = C_1 \oplus C_2$ is a $(3p, 2p+1)$ binary nonprimitive BCH code with generator polynomial $g(x) = \phi_1(x)$. The minimum distance of C is exactly 3 [37].

To determine the separation vector of code C, we use the analysis outlined in the previous section, and an array of exponents of α, shown in Table 4.2. A number $\rho \pmod{3p}$ appearing at the m-th row and μ-th column of the array of Table 4.2,

Table 4.1: Some good cyclic UEP codes of composite length

n	k	n_1	n_2	k_1	k_2	s_1	s_2	d^*	d_{bch}	Nonzeros
15	9	3	5	1	8	5	4	4	3	0,1,7
21	13	3	7	1	12	7	4	4	3	0,1,5
33	23	3	11	12	11	5	3	3	3	5, 11,0,3
35	11	5	7	4	7	7	5	5	5	7,0,5,15
39	27	3	13	14	13	6	3	3	3	7, 13,0,3
39	15	3	13	1	14	13	10	10	7	0,7,13
45	33	5	9	1	32	9	4	4	3	0,1,3,7,21
51	34	3	17	18	16	7	6	6	4	11, 17, 19,3,9
51	19	3	17	1	18	17	14	14	11	0,11,17,19
55	15	5	11	4	11	11	5	5	5	11,0,5
57	39	3	19	20	19	5	3	3	3	5, 19,0,3
63	13	7	9	1	12	27	24	24	15	0,3,13
69	56	3	23	12	44	5	4	4	3	0, 3,1,5
75	53	3	25	5	48	5	4	4	3	0, 15,1,5,7,35
75	34	3	25	4	30	10	6	6	6	15,5,7,25,35
75	29	3	25	1	28	25	8	8	8	0,1,5,35

Table 4.2: Array of exponents of α

	0	1	2	$\cdots$	$p-1$
0	0	3	6	$\cdots$	$3(p-1)$
1	p	$p+3$	$p+6$	$\cdots$	$p+3(p-1)$
2	$2p$	$2p+3$	$2p+6$	$\cdots$	$2p+3(p-1)$

indicates that $\alpha^\rho = \beta^\mu \gamma^m$, where $\beta = \alpha^3$ and $\gamma = \alpha^p$, for $0 \leq \rho < 3p$, $0 \leq m < 2$ and $0 \leq \mu < p$. Equivalently,

$$3\mu + pm \equiv \rho \pmod{3p}.$$

Following the analysis of the previous section, we find that $\bar{J}_1 = \{1, 2\}$, $\bar{J}_2 = \emptyset$, $D_2 = 1$, $d_2 = 3$, and $d = 1$. Hence $s_1 \geq D_1$, and $s_2 = 3$, where D_1 will be found using the BCH lower bound, by counting the number of consecutive roots of $\phi_1(x)$ in the second and third rows of the array in Table 4.2, i.e., the number of consecutive zeros of code $V_1^{(1)}$ or $V_1^{(2)}$.

Let

$$C_o^\star \triangleq \{\rho : \phi_1(\alpha^\rho) = 0, \alpha^\rho = \gamma\beta^\mu, 0 < \mu < p\}, \quad \text{and}$$

$$C_e^\star \triangleq \{\rho : \phi_1(\alpha^\rho) = 0, \alpha^\rho = \gamma^2\beta^\mu, 0 < \mu < p\}.$$

Then $C_o^\star$ and $C_e^\star$ are the sets of zeros (also known as *defining sets*) of codes $V_1^{(1)}$ and $V_1^{(2)}$, respectively. The exponents of the roots of $\phi_1(x)$ are grouped in a *cyclotomic coset* [32]:

$$C^\star = \{1, 2, 4, 8, \cdots, 2^{s-1}\},$$

where s is the multiplicative order of 2 modulo $3p$. We have that $C^\star = C_o^\star \bigcup C_e^\star$, and if p is a prime number and 2 is primitive modulo p, then $|C^\star| = p - 1$ and thus $\alpha \in GF(2^{p-1})$. Let L_o and L_e be the sets obtained from taking the elements in $C_o^\star$ and $C_e^\star$ modulo p, respectively. Let Q and N be the sets of quadratic residues and nonresidues modulo p, respectively. Suppose 2 is a primitive root modulo p. Then

$C^\star$ is a complete set of residues modulo p and we have either $Q = L_o^\star$ and $N = L_e^\star$, or $Q = L_e^\star$ and $N = L_o^\star$. Number theory permits us to find sufficient conditions on a prime number p, such that a given element in the set $\{1, 2, 3, \cdots, p\}$ belongs to Q or N [36].

Example 7: For $p = 11$, we have the following 3-by-11 array of exponents of α, a 33-th root of unity in $GF(2^{10})$. The rows are indexed by m while the columns are indexed by μ,

	0	1	2	3	4	5	6	7	8	9	10
0	0	3	6	9	12	15	18	21	24	27	30
1	11	14	17	20	23	26	29	32	2	5	8
2	22	25	28	21	1	4	7	10	13	16	19

e.g., $\alpha = \beta^4\gamma^2$. We have,

$$C^\star = \{1, 2, 4, 8, 16, 32, 31, 29, 25, 17\},$$

$$C_o^\star = \{2, 8, 32, 29, 17\}, \quad C_e^\star = \{1, 4, 16, 31, 25\},$$

$$L_o^\star = N = \{2, 8, 10, 7, 6\}, \quad \text{and} \quad L_e^\star = Q = \{1, 4, 5, 9, 3\}. \; \triangle\,\triangle$$

The next lemmas give sufficient conditions on p such that some 3 consecutive integers in Q or N exist.

Lemma 1 *If* $p \equiv \pm 19, \pm 29 \; (mod \; 120)$ *then* $4, 5, 6 \in Q$.

Proof: From elementary number theory [36], it is known that

(i) if $p \equiv \pm 3 \pmod 8$ then $2 \in N$,

(ii) if $p \equiv \pm 1 \pmod 5$ then $5 \in Q$,

(iii) if $p \equiv \pm 5 \pmod{12}$ then $3 \in N$.

The conclusion of the lemma follows from the fact that if $a, b \in N$ then $ab \in Q$, and therefore $4 \in Q$ and $6 \in Q$. Solving the set of congruences (i) to (iii) above, using the Chinese remainder theorem, we obtain $p \equiv \pm 19, \pm 29 \pmod{120}$. $\triangle\ \triangle$

All the proofs of the lemmas presented in this section follow the same line of argument as for Lemma 1 and thus their proofs are omited.

Lemma 2 *If* $p \equiv \pm 1, \pm 11 \ (mod\, 60)$ *then* $3, 4, 5 \in Q$.

Lemma 3 *If* $p \equiv \pm 11, \pm 13, \pm 61 \ (mod\, 168)$ *then* $6, 7, 8 \in N$.

Combining Lemmas 1-3, we obtain sufficient conditions for codes $V_1^{(1)}$ or $V_1^{(2)}$ to have some *three consecutive zeros.* It follows from the BCH lower bound that $\mathbf{s}_1 \geq 4$ and we have proved the following result.

Theorem 7 *Let p be a prime number such that 2 is primitive modulo p and at least one of the following congruences is satisfied:*

(i) $p \equiv \pm 11, \pm 19, \pm 29, \pm 59 \ (mod\, 120)$,

(ii) $p \equiv \pm 11, \pm 13, \pm 61 \ (mod\, 168)$.

Then there exists a $(3p, 2p + 1)$ *two-level BCH UEP code which provides single-error correction and double-error detection for* $k_1 = p + 1$ *information bits, and single-error correction for the remaining* $k_2 = p$ *information bits. In other words, the separation vector is* $\bar{\mathbf{s}} \geq (4, 3)$, *for the message space* $\{0,1\}^{p+1} \times \{0,1\}^p$.

Stronger conditions for the existence of 4 consecutive integer in Q or N are given by the next lemmas.

Lemma 4 *If p is a prime number such that*

$$p \equiv \pm 13, \pm 107, \pm 157, \pm 323, \pm 347, \pm 397 \ (mod\, 840),$$

then $5, 6, 7, 8 \in N$.

Lemma 5 *If p is a prime number such that*

$$p \equiv \pm 19, \pm 29, \pm 139, \pm 149, \pm 221, \pm 389 \ (mod\, 840),$$

then $4, 5, 6, 7 \in Q$.

Lemma 6 *If p is a prime number such that*

$$p \equiv \pm 37, \pm 83, \pm 107, \pm 133, \pm 157, \pm 227, \pm 347, \pm 397 \ (mod\, 1320),$$

then $9, 10, 11, 12 \in Q$.

Lemmas 4-6 prove the following theorem.

Theorem 8 *Let p be a prime number such that 2 is a primitive root modulo p and at least one of the following congruences holds*

$$p \equiv \pm 13, \pm 19, \pm 29, \pm 107, \pm 139, \pm 149, \pm 157, \pm 221, \pm 323, \pm 347, \pm 389, \pm 397 \pmod{840},$$

$$p \equiv \pm 37, \pm 83, \pm 107, \pm 133, \pm 157, \pm 227, \pm 347, \pm 397, \pm 493, \pm 563 \pmod{1320}.$$

Then there exists a $(3p, 2p+1)$ two-level BCH UEP code with separation vector $\bar{\mathbf{s}} \geq (5, 3)$ for the message space $\{0,1\}^{p+1} \times \{0,1\}^{p}$.

Even when 2 is not a primitive root modulo p, but $(p-1)/2$ is a prime number, there still exist two-level BCH UEP codes, as the following Theorem 9 shows.

Theorem 9 *Let p be a prime number such that 2 is not a primitive root modulo p and $p = 2q + 1$, where q is a prime number $q > 3$. If $p \equiv \pm 1 \ (mod\, 24)$, then there exists a $(3p, 2p+1)$ two-level BCH UEP code with separation vector $\bar{\mathbf{s}} \geq (5, 3)$ for the message space $\{0,1\}^{p+1} \times \{0,1\}^{p}$.*

Proof: The powers of 2 form a complete set of residues modulo q. Moreover, if $p \equiv \pm 1 \pmod 8$ and $p \equiv \pm 1 \pmod{12}$, then $1, 2, 3, 4 \in Q$. Using the Chinese remainder theorem we obtain the congruence shown in the theorem. $\triangle\triangle$

Some two-level BCH UEP codes with a higher lower bound on $\mathbf{s}_1$ are given by the following Theorem 10.

Theorem 10 *Let p be a prime number such that 2 is not a primitive root modulo p and the following congruences are simultaneously satisfied:*

(i) $p \equiv \pm 53, \pm 187, \pm 197, \pm 283, \pm 307, \pm 317 \ (mod\, 840)$,

(ii) $p \equiv \pm 3, \pm 5, \pm 17, \pm 21, \pm 31, \pm 33, \pm 35, \pm 37, \pm 39 \ (mod\, 92)$,

(iii) $p \equiv \pm 7, \pm 11, \pm 13, \pm 21, \pm 23, \pm 29, \pm 33, \pm 35, \pm 37 \ (mod\, 76)$,

(iv) $p \equiv \pm 1, \pm 5, \pm 7, \pm 9, \pm 19 \ (mod\, 44)$.

Then there exists a $(3p, 2p+1)$ two-level BCH UEP code with separation vector $\bar{\mathbf{s}} \geq (7, 3)$ for the message space $\{0,1\}^{p+1} \times \{0,1\}^p$.

Proof: The set of congruences given in the theorem corresponds to sufficient conditions for which $(18 + i) \in N$, for $0 \leq i \leq 5$. $\triangle\triangle$

In Table 4.3, a list of two-level BCH UEP codes is presented. In the table, the theorem corresponding to each code, when applicable, is indicated.

4.2 Binary shortened Hamming codes

An analysis of multi-level error-correcting capabilities of binary shortened Hamming codes is presented in this section. First it is shown that shortening a Hamming code by 0, 1 or 2 bits does not result in an LUEP codes. The proof of this result is very simple and involves examining a generator matrix of a binary Hamming

Table 4.3: Some two-level BCH UEP codes

n	k	p	k_1	k_2	s_1	s_2	Nonzeros	Theorem
33	23	11	12	11	4	3	5, 11,0,3	1
39	27	13	14	13	5	3	7, 13,0,3	1,2
57	39	19	20	19	5	3	5, 19,0,3	1,2
69	47	23	24	23	5	3	5, 23,0,3,15	3
87	59	29	30	29	5	3	5, 29,0,3	1,2
111	75	37	38	37	5	3	11, 37,0,3	2
141	95	47	48	47	5	3	5, 47,0,3,15	3
159	107	53	54	53	7	3	11, 53,0,3	4
177	119	59	60	59	6	3	5, 59,0,3	1
183	123	61	62	61	7	3	5, 61,0,3	1
201	135	67	68	67	7	3	7, 67,0,3	
213	143	71	72	71	7	3	7, 71,0,3,21	
237	159	79	80	79	7	3	7, 79,0,3,9	
249	167	83	84	83	8	3	11, 83,0,3,	2
303	203	101	102	101	8	3	5, 101,0,3	1
309	207	103	104	103	8	3	5, 103,0,3,9	
321	215	107	108	107	7	3	7, 107,0,3	1,2
393	263	131	132	131	9	3	5, 131,0,3	1
417	279	139	140	139	6	3	5, 139,0,3	1,2
447	299	149	150	149	7	3	5, 149,0,3	1,2

code. In addition, a procedure to obtain a lower bound on - and sometimes the exact value of - the number of linearly independent minimum weight vectors of a shortened Hamming code is proposed. The unequal error protection capabilities of some shortened Hamming codes will be determined using this approach, for code lengths from 7 to 1023.

In order to analyze the multi-level error correcting capabilities of shortened Hamming codes, we need the following definition: Let C be an (n, k, d) linear code and denote by M the *set of minimum weight codewords*,

$$M \triangleq \{\bar{c} \in C : \mathrm{wt}(\bar{c}) \leq 2\delta,\ \delta > t\}, \tag{4.3}$$

where $\mathrm{wt}(\bar{c})$ is the Hamming weight of vector $\bar{c}$, and $t = \lfloor (d-1)/2 \rfloor$. Denote by r_M the number of vectors of the maximum linearly independent subsystem of vectors in M, i.e., r_M is the rank of M. The following result is known [5]:

Lemma 7 *To provide the protection level δ for at least k^* information digits of a (n, k, d) linear code C, it is necessary and sufficient that the rank r_M of the set of minimum weight codewords M be no greater than $k - k^*$.*

Proof: See [5]. $\triangle\triangle$

In other words, if the minimum weight codewords in a linear code C do not span it, then C is an LUEP code. In analyzing Hamming codes, we use the following special case of (4.3),

$$M \triangleq \{\bar{c} \in C : \mathrm{wt}(\bar{c}) = 3\}. \tag{4.4}$$

Let C_H be a $(2^m - 1, 2^m - m - 1, 3)$ binary Hamming code. Let M denote the set of minimum weight vectors in C_H, as defined in (4.4). Then there exists a linear subspace of C_H, denoted S, such that

$$M \subseteq S \qquad \text{and} \qquad \dim(S) \leq k \tag{4.5}$$

i.e., S is the *span* of M (sometimes denoted $< M >$). Without loss of generality, the generator matrix of C_H is

$$G = \left(\begin{array}{ccccc|c} 1 & 0 & 0 & \cdots & 0 & \bar{\mathbf{v}}_1 \\ 0 & 1 & 0 & \cdots & 0 & \bar{\mathbf{v}}_2 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & \bar{\mathbf{v}}_k \end{array}\right) = \left(\begin{array}{c} \bar{\mathbf{u}}_1 \\ \bar{\mathbf{u}}_2 \\ \vdots \\ \bar{\mathbf{u}}_k \end{array}\right) \tag{4.6}$$

It is clear that

$$\{\bar{\mathbf{v}}_1, \bar{\mathbf{v}}_2, \cdots, \bar{\mathbf{v}}_k\} = \{\bar{\mathbf{v}} : \mathrm{wt}(\bar{\mathbf{v}}) \geq 2 \text{ and } \bar{\mathbf{v}} \in \{0,1\}^m\}$$

Lemma 8 *C_H is spanned by its minimum weight vectors.*

Proof: Suppose that $\dim(S) < k$. Then there exists an integer i, with $1 \leq i \leq k$, such that $\bar{\mathbf{u}}_i \notin S$ and $\mathrm{wt}(\bar{\mathbf{v}}_i) \geq 3$.
Let i_o be an integer such that $\bar{\mathbf{u}}_{i_o} \notin S$ and

$$\mathrm{wt}(\bar{\mathbf{u}}_{i_o}) = \min\{\mathrm{wt}(\bar{\mathbf{u}} : \bar{\mathbf{u}} \notin S \text{ and } \bar{\mathbf{u}} \in C_H\}$$

Since $\mathrm{wt}(\bar{\mathbf{v}}_{i_o}) \geq 3$, there exists a $\bar{\mathbf{u}}_j$ such that

$$\mathrm{wt}(\bar{\mathbf{u}}_j) = \mathrm{wt}(\bar{\mathbf{u}}_{i_o}) - 1 \tag{4.7}$$

and

$$\mathrm{wt}(\bar{\mathbf{u}}_{i_o} + \bar{\mathbf{u}}_j) = 3. \tag{4.8}$$

From equations (4.7) and (4.8), we conclude that $\bar{\mathbf{u}}_j \in S$. On the other hand, from (4.8) it follows that $\bar{\mathbf{u}}_{i_o} + \bar{\mathbf{u}}_j \in S$. Hence $\bar{\mathbf{u}}_{i_o} \in S$, a contradiction. We have shown that $S = C_H$. $\triangle\triangle$

Combining Lemmas 7 and 8, we conclude that a Hamming code C_H has no multi-level error correcting capabilities. The proof of Lemma 8 can be extended to the case of $(2^m - 1 - \ell, 2^m - m - 1 - \ell, 3)$ shortened Hamming codes, for shortening lengths

$\ell = 0, 1, 2$. The argument is that, for any $\bar{\mathbf{u}}_i$ with $\mathrm{wt}(\bar{\mathbf{u}}_i) \geq 3$, there are at least three possible vectors $\bar{\mathbf{u}}_j \in C_H$ such that $\mathrm{wt}(\bar{\mathbf{u}}_i + \bar{\mathbf{u}}_j) = 3$. This assertion can be easily verified by direct examination of the generator matrix (4.6) of the Hamming code C_H.

When the shortening length $\ell > 2$, we treat C_H as a binary cyclic code with generator polynomial $g(x)$ (a primitive polynomial) of degree m. Let C_S denote the shortened cyclic code obtained from C_H by deleting the ℓ leading high order coefficients in code polynomials. Then C_S is an $(2^m - 1 - \ell, 2^m - 1 - m - \ell, d \geq 3)$ linear code and is no longer cyclic. To find the minimum weight polynomials of C_S, we start by defining the following set of integers associated with the generator polynomial of C_H, $g(x)$ [14]

$$I_3 \triangleq \{i : x^{i-1} + x^j + 1 \bmod g(x) = 0 \text{ and } 0 < j < i \leq 2^m - 1\}$$

Let a_0, a_1 be the smallest integers in I_3 such that $a_1 > a_0$. Consider the case $n_s \geq a_0$. Then the following polynomials are linearly independent:

$$\begin{aligned} v(x) &= x^{a_0-1} + x^j + 1 \\ xv(x) &= x^{a_0} + x^{j+1} + x \\ &\vdots \qquad\qquad \vdots \\ x^{n_s-a_0}v(x) &= x^{n_s-1} + x^{n_s-a_0+j} + x^{n_s-a_0} \end{aligned}$$

where $0 < j < a_0 - 1$. This set of linearly independent polynomials belongs to a basis of S, the linear subspace of C_S containing the set of minimum weight polynomials, M'. Therefore, the dimension of S is given by

$$\begin{aligned} \dim(S) &\geq n_s - a_0 + 1, \text{ for } n_s \geq a_1; \\ \dim(S) &= n_s - a_0 + 1, \text{ for } a_1 > n_s \geq a_0. \end{aligned}$$

These conditions, with $a_0, a_1 \in I_3$, are equivalent to the following:

- If $a_0 = m + 1$ then no linear UEP code exists because

$$\begin{aligned}\dim(S) &\geq (2^m - 1 - \ell) - (m+1) + 1 \\ &= 2^m - 1 - m - \ell = k_s\end{aligned}$$

- If $a_0 > m + 1$ then we have three cases:

  (i) For $2 \leq \ell \leq 2^m - 1 - a_1$ ($n_s \geq a_1$), we need to examine S directly because

$$\dim(S) \geq 2^m - a_0 - \ell \qquad \text{and} \qquad 2^m - a_0 - \ell < k_s$$

  (ii) For $2^m - a_1 \leq \ell \leq 2^m - 1 - a_0$ ($a_1 > n_s \geq a_0$), C_S is a linear UEP code with separation vector $\bar{s} \geq (4, 3)$ for the message spaces $M_1 = \{0,1\}^{a_0 - 1 - m}$ and $M_2 = \{0,1\}^{2^m - a_0 - \ell}$. This is true since

$$\dim(S) = 2^m - a_0 - \ell < 2^m - 1 - m - \ell = k_s$$

  (iii) For $2^m - a_0 \leq \ell \leq 2^m - m - 3$ ($n_s < a_0$), no codewords of weight 3 exist, and we must check again conditions but this time for some other integers $a_0, a_1 \in I_\omega$, $\omega \geq 4$, where

$$\begin{aligned}I_\omega \triangleq \{i : x^{i-1} + x^{j_1} + \quad \cdots + x^{j_{\omega-2}} + 1 \mod g(x) = 0, \\ 0 < j_1 < \cdots < j_{\omega-2} < i - 1 < 2^m - 1\}\end{aligned}$$

With the aid of a computer, the values of $a_0, a_1 \in I_3$, for all primitive polynomials of degrees $3 \leq m \leq 10$ were determined. The results are shown in Tables 4.4-4.6. Note that given a polynomial $g(x)$ of degree m with fixed a_0 and a_1, the reciprocal polynomial of $g(x)$, $g^*(x) = x^m g(x^{-1})$ has the same values a_0 and a_1. Therefore, only one polynomial from $\{g(x), g^*(x)\}$ is listed in Tables 4.4-4.6. From these Tables, we find values of $g(x)$ that result in linear UEP codes with minimum distance 3. Tables 4.7-4.9 show LUEP codes C_S with separation vector $\bar{s} = (s_1, s_2)$, $s_1 \geq 4$, $s_2 = 3$ for

the message spaces $M_1 = \{0,1\}^{k_1}$ and $M_2 = \{0,1\}^{k_2}$. The range of the length and subdimension k_2 of each code are shown, i.e.,

$$n_{inf} \leq n \leq n_{sup} \quad \text{and} \quad k_{2,inf} \leq k_2 \leq k_{2,sup}.$$

Although LUEP codes obtained from shortening Hamming codes are not optimal, the analysis presented in this work is of interest because of the many applications of shortened Hamming codes for error control in data communications. For example, shortened Hamming codes are used in local area networks, under IEEE Standard 802.3.

4.3 Binary primitive BCH codes

A recent result [1] suggests that binary primitive BCH codes, those containing second-order Reed-Muller codes as proper subcodes, are not spanned by their set of minimum weight codewords. In view of Lemma 7, these codes are LUEP codes. The purpose of this section is to introduce an example of this class of LUEP codes, and to present a table of some binary BCH LUEP codes.

Example 8: Let C be a $(63, 24, 15)$ BCH code. Then C contains a $(63, 22, 15)$ second-order Reed-Muller code, C_{RM}, as proper subcode. It is known [21], [22] that a boolean polynomial representation of two codewords which form a basis of the coset leaders of C/C_{RM} is

$$f^{(3)}_{21,1} = x_1x_3x_6 \oplus x_1x_4x_5 \oplus x_1x_4x_6 \oplus x_2x_3x_5 \oplus x_2x_3x_6 \oplus x_2x_4x_5$$

$$f^{(3)}_{21,2} = x_1x_3x_5 \oplus x_1x_4x_6 \oplus x_2x_3x_6 \oplus x_2x_4x_5 \oplus x_2x_4x_6$$

All three nonzero coset leaders have Hamming weight 18. By explicitly evaluating the weight distribution of all the cosets of C_{RM} in C, we have verified that all

Table 4.4: Values of $a_0, a_1 \in I_3$ for $3 \leq m \leq 8$

m	$g(x)$	a_0	a_1
3	1101	4	6
4	11001	5	9
5	101001	6	11
	111011	8	12
	111101	9	10
6	1100001	7	13
	1101101	9	14
	1110011	12	16
7	11000001	8	15
	10010001	8	15
	10111001	11	21
	11010011	22	28
	11010101	20	22
	11100101	15	22
	11110001	20	22
	11110111	20	21
	11111101	14	20
8	101100101	24	27
	101101001	17	33
	101110001	22	26
	110001101	21	24
	110101001	14	27
	111000011	28	34
	111100111	28	32
	111110101	14	27

Table 4.5: Values of $a_0, a_1 \in I_3$ for $m = 9$

m	$g(x)$	a_0	a_1
9	1000010001	10	19
	1001011001	37	43
	1100110001	62	63
	1000101101	21	34
	1001110111	42	60
	1101100001	30	44
	1011011011	12	23
	1110000101	20	39
	1111101001	30	46
	1111100011	30	38
	1110001111	40	48
	1101101011	12	23
	1001101111	14	27
	1111001101	58	64
	1101110011	16	31
	1111001011	30	50
	1001111101	20	34
	1111010101	46	47
	1010010101	40	58
	1010111101	27	35
	1111111011	56	60
	1100010101	37	62
	1010110111	28	52

Table 4.6: Values of $a_0, a_1 \in I_3$ for $m = 10$

m	$g(x)$	a_0	a_1
10	10000001001	11	21
	10100001101	50	66
	11111111001	66	82
	10001101111	24	47
	11101001101	14	27
	10111111011	27	53
	10000011011	40	43
	10100100011	38	75
	10100110001	66	82
	11000010011	61	86
	11101100011	33	65
	10111100101	18	35
	10100011001	54	80
	11001111111	52	64
	11101010101	58	86
	10110001111	18	35
	11100111001	36	71
	11011010011	24	45
	11101000111	56	58
	11110010011	84	99
	10111000111	33	47
	10011010111	15	29
	11010110101	58	71
	10000101101	56	66
	11101111101	25	49
	11001111001	62	65
	10000100111	50	55
	11000010101	50	71
	11011011111	80	81
	11010001001	41	51

Table 4.7: LUEP Shortened Hamming Codes with $\bar{s} \geq (4,3)$

m	g(x)	n_{inf}	n_{sup}	k_1	$k_{2,inf}$	$k_{2,sup}$
5	111011	8	11	2	1	4
	111101	9	9	3	1	1
6	1110011	12	15	5	1	4
	1101101	9	13	2	1	5
7	10111001	11	20	3	1	10
	11010011	22	27	14	1	6
	11010101	20	21	12	1	2
	11100101	15	21	7	1	7
	11110001	20	21	12	1	2
	11110111	20	20	12	1	1
	11111101	14	19	6	1	6
8	101100101	24	26	15	1	3
	101101001	17	32	8	1	16
	101110001	22	25	13	1	4
	110001101	21	23	12	1	3
	110101001	14	26	5	1	13
	111000011	28	33	19	1	6
	111100111	28	31	19	1	4
	111110101	14	26	5	1	13

Table 4.8: LUEP Shortened Hamming Codes with $\bar{s} \geq (4,3)$

m	g(x)	n_{inf}	n_{sup}	k_1	$k_{2,inf}$	$k_{2,sup}$
9	1001011001	37	42	27	1	6
	1100110001	62	62	52	1	1
	1000101101	21	33	11	1	13
	1001110111	42	59	32	1	18
	1101100001	30	43	20	1	14
	1011011011	12	22	2	1	11
	1110000101	20	38	10	1	19
	1111101001	30	45	20	1	16
	1111100011	30	37	20	1	8
	1110001111	40	47	30	1	8
	1101101011	12	22	2	1	11
	1001101111	14	26	4	1	13
	1111001101	58	63	48	1	6
	1101110011	16	30	6	1	15
	1111001011	30	49	20	1	20
	1001111101	20	33	10	1	14
	1111010101	46	46	36	1	1
	1010010101	40	57	30	1	18
	1010111101	27	34	17	1	8
	1111111011	56	59	46	1	4
	1100010101	37	61	27	1	25
	1010110111	28	51	18	1	24

Table 4.9: LUEP Shortened Hamming Codes with $\bar{\mathbf{s}} \geq (4,3)$

m	g(x)	n_{inf}	n_{sup}	k_1	$k_{2,inf}$	$k_{2,sup}$
10	10100001101	50	66	39	1	17
	11111111001	66	82	55	1	17
	10001101111	24	47	13	1	24
	11101001101	14	27	3	1	14
	10111111011	27	53	16	1	27
	10000011011	40	43	29	1	4
	10100100011	38	75	27	1	38
	10100110001	66	82	55	1	17
	11000010011	61	86	50	1	26
	11101100011	33	65	22	1	33
	10111100101	18	35	7	1	18
	10100011001	54	80	43	1	27
	11001111111	52	64	41	1	13
	11101010101	58	86	47	1	29
	10110001111	18	35	7	1	18
	11100111001	36	71	25	1	36
	11011010011	24	45	13	1	22
	11101000111	56	58	45	1	3
	11110010011	84	99	73	1	16
	10111000111	33	47	22	1	15
	10011010111	15	29	4	1	15
	11010110101	58	71	47	1	14
	10000101101	56	66	45	1	11
	11101111101	25	49	14	1	25
	11001111001	62	65	51	1	4
	10000100111	50	55	39	1	6
	11000010101	50	71	39	1	22
	11011011111	80	81	69	1	2
	11010001001	41	51	30	1	11

Table 4.10: Some binary primitive BCH LUEP codes

m	n	δ	k_{BCH}	k_{RM}	k_1	k_2
6	63	15	24	22	2	22
7	127	31	36	29	7	29
8	255	63	55	37	18	37
9	511	127	85	46	39	46
10	1023	255	133	56	77	56

the minimum weight vectors of C are contained in C_{RM}. Therefore, C is a two-level error correcting code with separation vector $\bar{\mathbf{s}} = (18, 15)$ for the message space $\{0,1\}^2 \times \{0,1\}^{22}$. $\Delta\ \Delta$

In Table 4.10, we present some $(2^m - 1, k_{BCH}, \delta)$ binary primitive BCH codes C with LUEP capabilities, where δ is the designed minimum distance, $n = 2^m - 1$, k_{RM} denotes the dimension of a second-order Reed-Muller code of minimum distance δ contained in C, and k_1 and k_2 are the dimensions of the message subspaces, with corresponding separation vector $\bar{\mathbf{s}} = (s_1, \delta)$. Only for the first code in this table we have shown that $s_1 = 18$. Values of s_1 for the other codes in the table remain to be solved.

4.4 Summary

In this chapter, an analysis of the multi-level error correcting capabilities of some families of well known linear codes, such as binary Hamming codes and binary BCH codes, was presented. A table of some good binary cyclic UEP codes of composite length, obtained from a computer search for two-level error correcting codes among all binary cyclic codes of composite length up to 85, was introduced. It was shown that some nonprimitive binary BCH codes of composite length $3p$, where p is a prime number, are actually multi-level error correcting codes. We derived sufficient

conditions on a prime number p, such that $(3p, 2p+1)$ nonprimitive binary BCH codes of minimum distance 3, providing additional error correcting capabilties for $p+1$ information bits, exist. We also presented an analysis of the multi-level error correcting capabilities of shortened binary Hamming codes. By examining the set of minimum weight codewords, we obtained ranges of shortening lengths such that the resulting codes have at least two levels of error correction. We analyzed all primitive polynomials over $GF(2)$ of degree m, where $3 \leq m \leq 10$. Finally, we introduced an example $(63, 24, 15)$ binary primitive BCH code, C, that is not spanned by its set of minimum weight codewords. We determined the multi-level error correcting capabilities of this example code, by analyzing the weight structure of four cosets of a $(63, 22, 15)$ cyclic Reed-Muller, a subcode of C. An open research problem is to find other cases of binary primitive BCH codes having multiple levels of error correction.

Chapter 5

Applications in block coded modulation

In 1974, Massey [35] introduced the idea of considering channel coding and modulation as a single combined operation, to improve the performance of digital communication systems. Active research on the design of bandwidth efficient communication systems, combining error-correcting coding and digital modulation, started in 1982 with the classic work of Ungerboeck [46]. Since then, extensive research has been done in this area. When *block* error-correcting codes are used, then the resulting systems are known as *block coded modulation* (BCM) systems. Combined coding/modulation schemes can be constructed based on *partitions of a set of elementary signals* into non-intersecting subsets, each of which has a minimum squared Euclidean distance greater than that of the set of elementary signals [15]. Most of the known BCM systems are in fact an adaptation of the ideas of generalized concatenated codes [3] to signaling systems with arbitrary signal distance. In this chapter, we present several techniques of combining multi-level error correcting codes and M-PSK modulation systems to achieve coded modulation schemes that offer *distinct values of squared Euclidean distance*, one for each message part to be protected. In this way, most important message parts will have a larger squared Euclidean distance between code sequences than that corresponding to less important message parts. If data transmission is performed over an additive white Gaussian noise (AWGN) channel, and the channel code is selected properly (a soft-decision decoding and

a small number of nearest neighbors), then at high signal-to-noise ratios, we will have a *smaller* probability of bit error for the most important message parts than for the rest of the message. To fully achieve the performance promised by a given minimum squared Euclidean distance, in this chapter we present a new *multistage soft-decision decoding* of block linear multi-level error correcting codes, that uses their trellis structure.

5.1 Introduction

The building blocks for most coded modulation systems are phase shift-keying (PSK) or quadrature amplitude modulation (QAM) signal sets. In this work, we consider bandwidth efficient data communication over a power-limited channel, such as the satellite channel, for which M-PSK signals are used. In a 2^{μ}-PSK signal constellation, 2^{μ} signal points are equally spaced on a circle of radius ρ, centered at the coordinate origin. The average signal power is $P_{av} = \rho^2$ and the *minimum squared Euclidean distance*, normalized with respect to the average signal power is

$$\delta_{\mu}^2 = \frac{d_e^2}{P_{av}} = 4\sin^2\left(\frac{\pi}{2^{\mu}}\right) \tag{5.1}$$

Only *block* LUEP codes will be considered, although convolutional LUEP codes, and even nonlinear UEP codes, may be used as well. Advantages of using block codes as component codes in coded modulation systems include no error propagation and low complexity decoding. Among the disadvantages of using block coded modulation systems the most important is the high number of nearest neighbors (code sequences separated by the minimum squared Euclidean distance), which causes a drop in the coding gain. In this research, we focus our attention on the following techniques of combining error correcting codes and modulation:

(1) One technique used here to construct a *block modulation code* is the so-called

mapping by set partitioning, in which the signal constellation is partitioned into subsets with increasing intraset minimum squared Euclidean distances. For a 2^μ-PSK signal constellation, a μ-stage partition is made, and μ conventional error correcting codes are used, one for each partition stage [18],[40],[15],[39]. Modulation codes obtained this way are usually known as *multi-level modulation codes.*

(2) A second way of designing BCM systems using 2^μ-PSK signal constellations is as follows. A $(\mu n, \mu k, d)$ binary error correcting code, C, is used. Each codeword in C is decomposed into n μ-bit symbols and each symbol is placed in correspondence, via an appropriate label-to-signal mapping, with a signal point in the 2^μ-PSK signal constellation [38].

(3) Another way of obtaining a block modulation code is to use an (n, k, d) block code C over $GF(2^\mu)$. Then each code symbol selects, again by an appropriate symbol-to-signal mapping, a particular signal point in the 2^μ-PSK signal constellation. For the case $\mu = 1$, we have a trivial BCM system in which code bits of a codeword in a binary error correcting code select one of two phases in a BPSK (2-PSK) signal constellation.

5.1.1 Performance over binary symmetric channels

Before analyzing combined UEP coding and modulation schemes, we examine the error performance of LUEP codes over binary symmetric channels. It is shown that the additional error correcting capabilities of UEP codes result in a reduction of the probability of a block error, P_e, compared to that of single-level error correcting codes.

We start by considering an (n, k, d) conventional linear error correcting code C, of length n, dimension k and minimum distance d. Suppose we use a *bounded-distance decoder*. The probability of correct decoding, denoted by P_{cd}, is given

by the probability of occurrence of error patterns of Hamming weight up to the error correcting capability of C, $t = \lfloor (d-1)/2 \rfloor$. If we suppose that errors are independent over a BSC with crossover probability p, then the probability of a specific error pattern of weight i over a span of n transmitted bits is given by $p^i(1-p)^{n-i}$. Correct decoding will take place whenever $0 \leq i \leq t$, and there are $\binom{n}{i}$ such patterns, so that the probability of correct decoding is

$$P_{cd} = \sum_{i=0}^{t} \binom{n}{i} p^i (1-p)^{n-i}$$

Now let C' be an (n,k) two-level UEP code with separation vector $\bar{\mathbf{s}} = (\mathbf{s}_1, \mathbf{s}_2)$ for the message space $M = \{0,1\}^{k_1} \times \{0,1\}^{k_2}$, with $k = k_1 + k_2$, and $t_1 \geq t_2$, where $t_i = \lfloor (\mathbf{s}_i - 1)/2 \rfloor$, $i = 1,2$. That is, C' protects k_1 information symbols against up to t_1 errors and k_2 information symbols against up to t_2 errors. The error patterns that C' is capable of correcting (using a *bounded-distance decoder*) are those whose Hamming weight is:

1. Up to t_2 (The minimum error correcting capability of the code).

2. Greater than t_2 but less than or equal to t_1, affecting k_1 information bits.

Then the probability of correct decoding P_{cd} is given by the following expression,

$$P_{cd} = \sum_{i=0}^{t_2} \binom{n}{i} p^i (1-p)^{n-i} + \sum_{j=t_2+1}^{t_1} \sum_{i=0}^{t_2} \binom{n-k_1}{i} \binom{k_1}{j-i} p^j (1-p)^{n-j} \qquad (5.2)$$

Note that we are assuming a linear *systematic* UEP code, so that the k_1 most significant information bits can be identified explicitly in some k_1 bit positions within a codeword.

Block error probability.

For a code (not necessarily linear) of length n and number of codewords M, the *block error probability*, P_e is

$$P_e \triangleq \frac{1}{M}\sum_{i=1}^{M} \Pr\{\hat{\bar{\mathbf{x}}} \neq \bar{\mathbf{x}}(\bar{\mathbf{m}}_i)|\bar{\mathbf{x}}(\bar{\mathbf{m}}_i) \text{ sent}\}$$

where $\hat{\bar{\mathbf{x}}}$ is the received word, and $\bar{\mathbf{x}}(\bar{\mathbf{m}}_i)$ is the transmitted codeword corresponding to message $\bar{\mathbf{m}}_i$. For linear binary codes, if $\bar{\mathbf{e}}$ denotes an error word induced by the channel, and we use standard array decoding (i.e., *maximum likelihood decoding*), then

$$P_e = \Pr\{\bar{\mathbf{e}} \text{ is not a coset leader}\}$$

Let α_i denote the number of coset leaders of weight i. Then the probability of a block error is

$$P_e = 1 - \sum_{i=0}^{n} \alpha_i p^i (1-p)^{n-i}$$

Note that $\sum_{i=0}^{n} \alpha_i = 2^{n-k}$, the total number of coset leaders, or the volume (number of words) in a decoding sphere centered around each codeword. It is also interesting to point out that the probability of correct decoding - using *maximum likelihood decoding* - is

$$P'_{cd} = \sum_{i=0}^{n} \alpha_i p^i (1-p)^{n-i}$$

and therefore

$$P_e = 1 - P'_{cd}$$

An upper bound on the block error probability of a code is obtained when instead of P'_{cd} we use the probability of correct decoding *using bounded-distance decoding*, P_{cd}, so that

$$P_e \leq 1 - \sum_{i=0}^{t} \binom{n}{i} p^i (1-p)^{n-i} \tag{5.3}$$

Table 5.1: Optimal 2-level LUEP codes of length 31

n	k	k_1	k_2	s_1	s_1
31	23	4	19	5	3
31	22	11	11	5	3

Table 5.2: Optimal 2-level LUEP codes of length 63

n	k	k_1	k_2	s_1	s_1
63	54	4	50	5	3
63	53	11	42	5	3
63	52	26	26	5	3

It follows from (5.2) that the block error probability for a binary linear UEP code has an upper bound as follows:

$$P_e \leq 1 - \sum_{i=0}^{t_2} \binom{n}{i} p^i (1-p)^{n-i} - \sum_{j=t_2+1}^{t_1} \sum_{i=0}^{t_2} \binom{n-k_1}{i} \binom{k_1}{j-i} p^j (1-p)^{n-j} \qquad (5.4)$$

If a code is optimal in the sense of achieving the Hamming bound with equality (i.e., packing the space), then this bound on the probability of block error is achieved with equality. Figures 5.1 and 5.2 present the performance of several optimal linear UEP codes (from [26] and [48]), comparing their block error probabilities (since the codes are optimal, the bounds are achieved) with those of conventional single-level error correcting codes. In Figure 5.1 the optimal two-level LUEP codes shown in Table 5.1 are compared with a $(31, 26, 3)$ Hamming code and a $(31, 21, 5)$ BCH code. In Figure 5.2 we compare optimal two-level LUEP codes in Table 5.2 with a $(63, 57, 3)$ Hamming code and a $(63, 51, 5)$ BCH code [26].

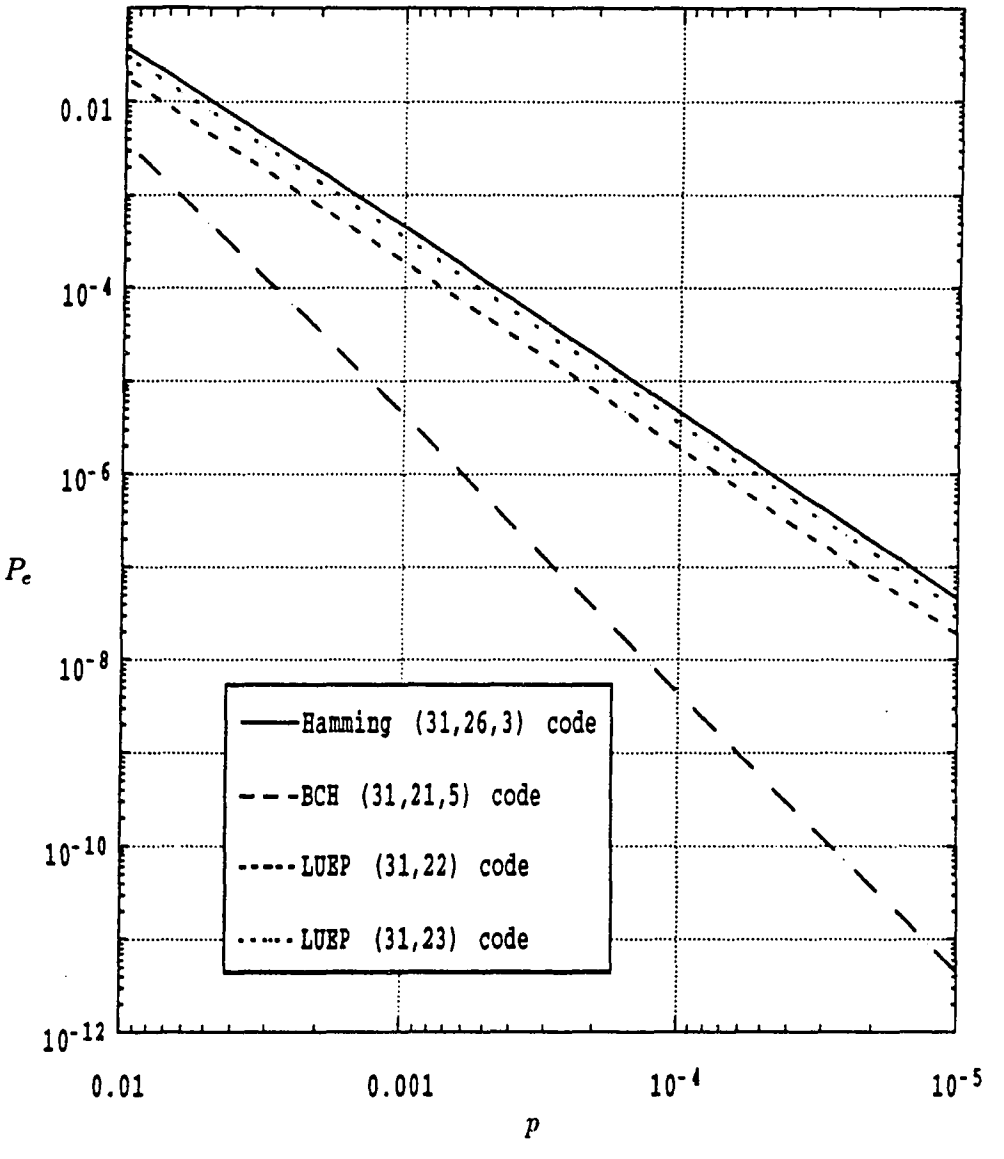

Figure 5.1: Block error probability for optimal 2-level LUEP codes of length 31

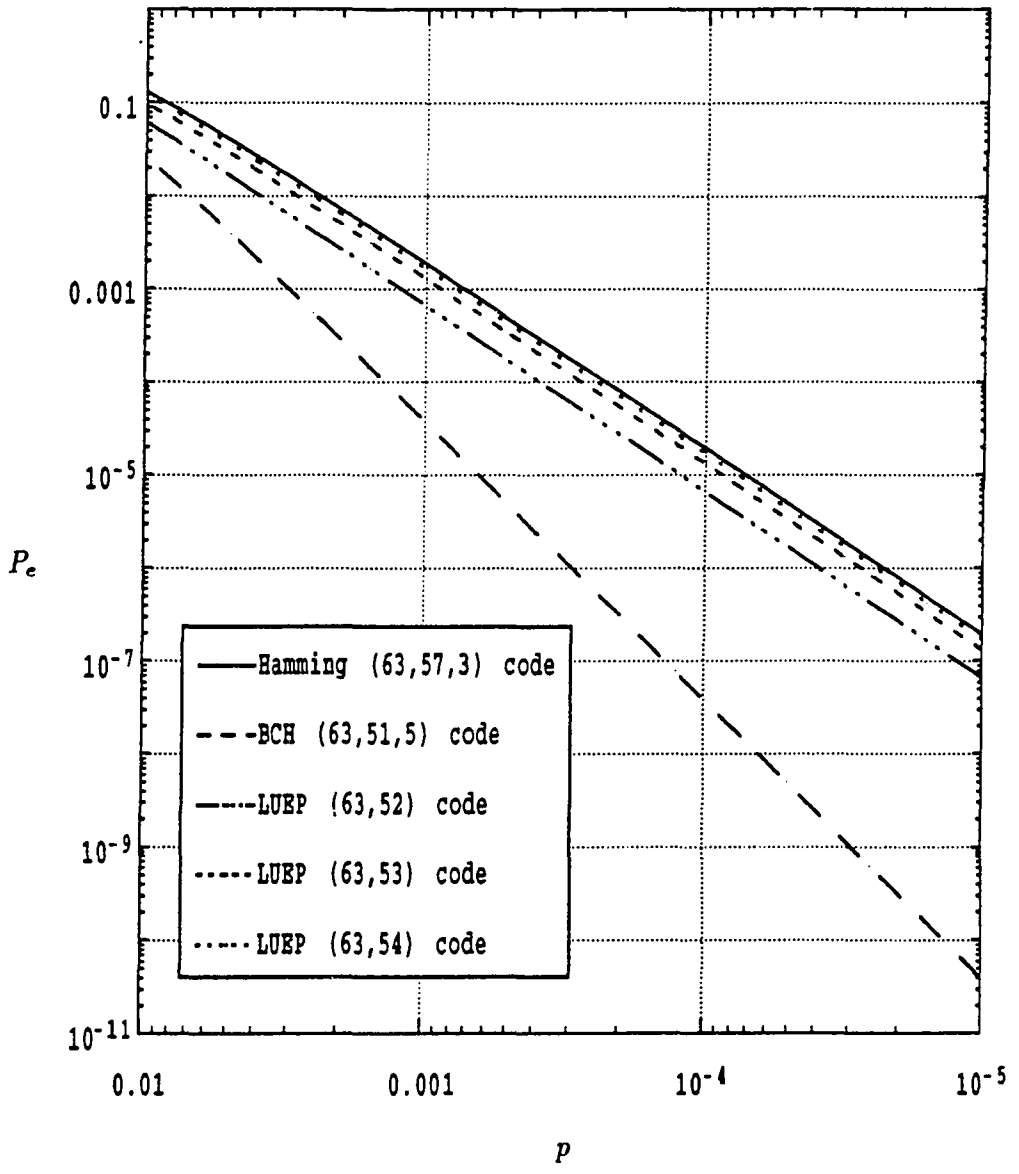

Figure 5.2: Block error probability for optimal 2-level LUEP codes of length 63

5.2 LUEP BPSK block modulation codes

In this section, we analyze the error performance of some *optimal* LUEP codes and compare it with the error performance of conventional linear error correcting codes used in an additive white Gaussian noise (AWGN) channel with BPSK modulation. We show that for short lengths, LUEP codes combined with BPSK modulation perform no better than conventional linear codes of the same minimum distance, due to their lower rates, which are not compensated by increased minimum squared Euclidean distances.

Block error probability

The probability of bit error in data communication over an AWGN with BPSK modulation is given by

$$p = \frac{1}{2}\text{erfc}\left(\sqrt{\frac{E_b}{N_o}}\right) \tag{5.5}$$

Let C be an (n, k, d) error correcting code. We consider using C to transmit data over an AWGN simply by mapping bit symbols into the phases of a BPSK modulated signal. To compare the performance of C over the uncoded case, we need to derive the probability of block error (or block error rate, BER) expressions for both the uncoded and the coded cases. For the uncoded case, we assume a block of k information bits to be transmitted using BPSK modulation. In this case, the BER is the probability of at least one bit error among the k bits within a block. Assuming that symbols in the BPSK signal are independent, we have

$$\text{BER} = 1 - (1-p)^k \tag{5.6}$$

where p is given by equation (5.5).

In the coded case, we assume that the system works at the *same data rate*, that is, over a block time period, the same number of bits are sent to the user. Under

this assumption, the amount of energy per coded bit E_c is smaller than the original energy per bit E_b by a factor of k/n (the rate of code C). The expressions for the block error rate in this case are the same as equations (5.3) and (5.4), i.e.,

$$P_e \leq 1 - \sum_{i=0}^{t} \binom{n}{i} p^i (1-p)^{n-i}, \text{ and} \tag{5.7}$$

$$P_e \leq 1 - \sum_{i=0}^{t_2} \binom{n}{i} p^i (1-p)^{n-i} - \sum_{j=t_2+1}^{t_1} \sum_{i=0}^{t_2} \binom{n-k_1}{i} \binom{k_1}{j-i} p^j (1-p)^{n-j}. \tag{5.8}$$

but where now the probability of bit error is given by

$$p = \frac{1}{2}\text{erfc}\left(\sqrt{\frac{E_c}{N_o}}\right) = \frac{1}{2}\text{erfc}\left(\sqrt{\frac{k}{n}\frac{E_b}{N_o}}\right) \tag{5.9}$$

In Figures 5.3 and 5.4, the error performance of some optimal LUEP codes of lengths 63 and 255 [26] is presented and compared to that of Hamming and two-error-correcting BCH codes of the same length, and with uncoded BPSK of appropriate block lengths. It is clear that UEP codes in this case are worse than Hamming codes of the same length, mainly because of their lower rates. It is also clear from these results that, as the length increases, the performance of optimal LUEP codes improves.

These LUEP BPSK block modulation codes have very low rates and do not offer significant coding gains (savings in signal-to-noise ratio between coded and uncoded systems to achieve the same bit error rate) over uncoded BPSK modulation. To improve both the rate and the coding gain we have to use an expanded signal constellation and more powerful codes.

5.3 LUEP QPSK block modulation codes

In a QPSK signal constellation with *gray mapping* between labels and signal points, depicted in Figure 5.5, the squared Euclidean distance between signal points is *twice*

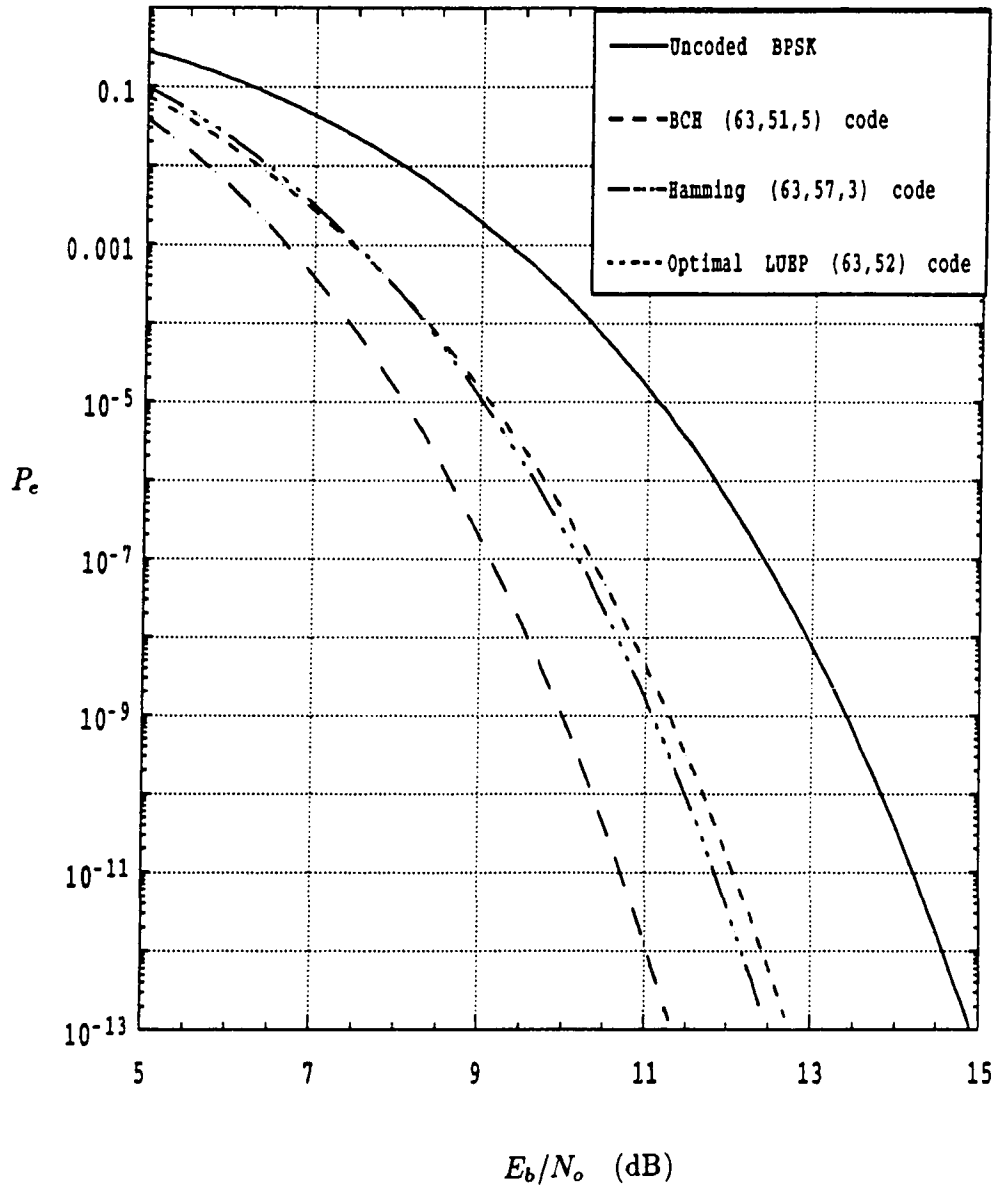

Figure 5.3: Block error probability for optimal LUEP BPSK codes of length 63

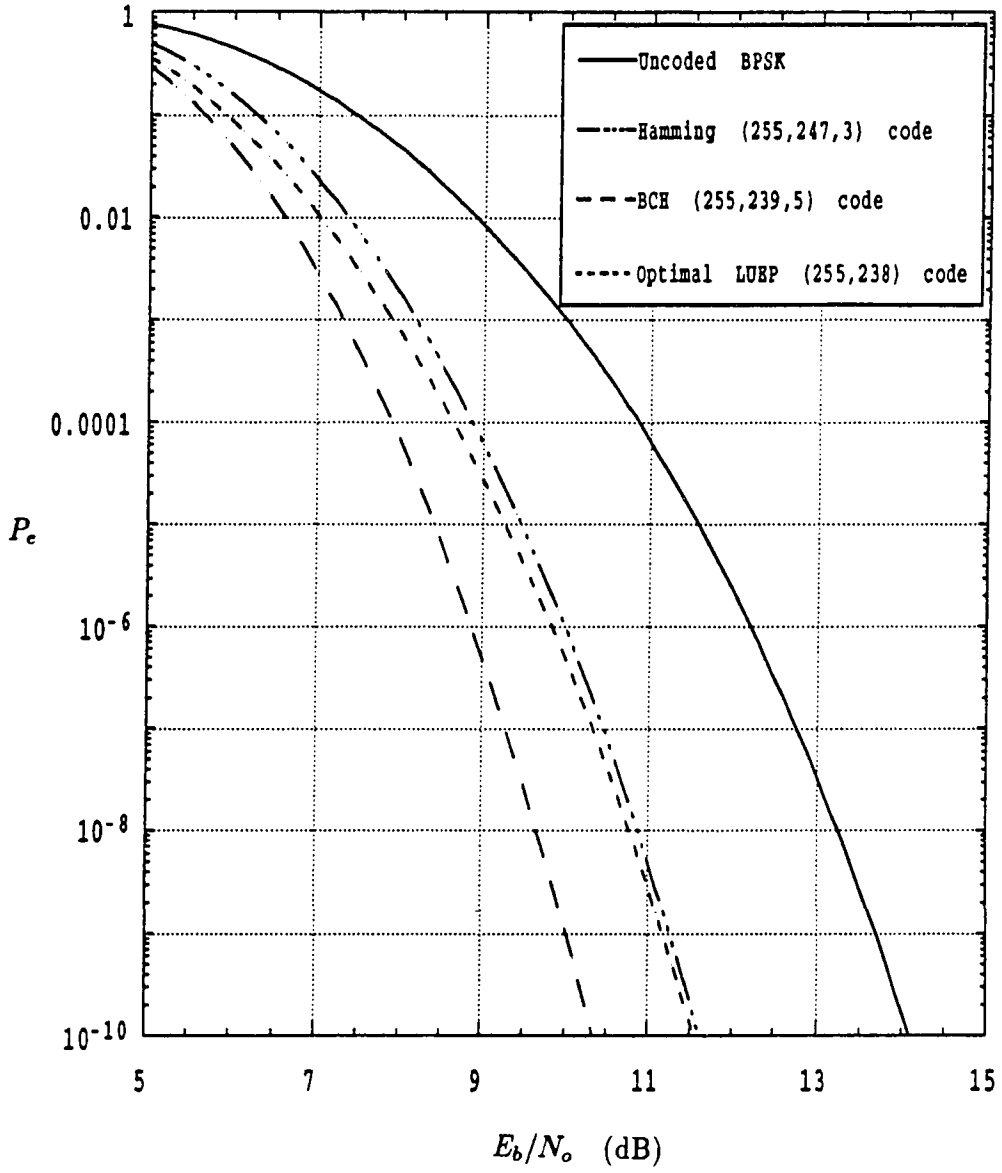

Figure 5.4: Block error probability for optimal LUEP BPSK codes of length 255

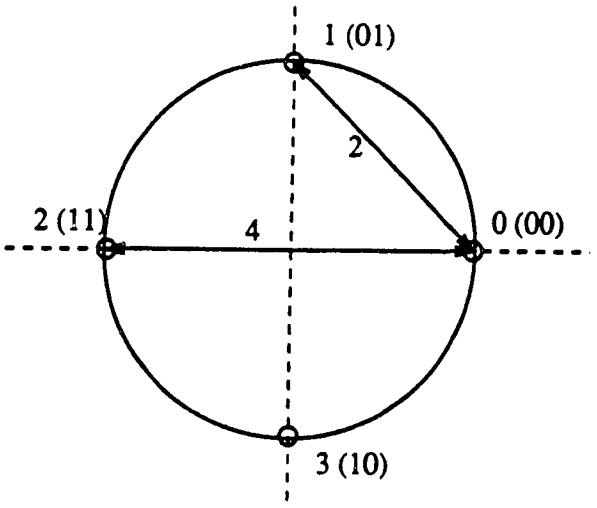

Figure 5.5: A QPSK signal constellation with Gray mapping

the Hamming distance between the corresponding labels. We say that a QPSK signal constellation forms a *second-order Hamming space* [25].

By mapping 2-bit symbols into signal points in a QPSK signal set, we can combine $(2n, k_1 + k_2)$ 2-level LUEP codes and QPSK modulation to achieve a block coded modulation system that offers two values of minimum squared Euclidean distances, one for each message part. The resulting LUEP QPSK block modulation code has the same minimum squared Euclidean distance as the best QPSK block modulation code with the same rate and length, while offering an additional squared Euclidean distance between code sequences associated with most important message parts. The proposed construction is as follows: Let C_b be a $(2n, k_1 + k_2)$ binary LUEP code with separation vector $\bar{\mathbf{s}} = (\mathbf{s}_1, \mathbf{s}_2)$ for the message space $\{0,1\}^{k_1} \times \{0,1\}^{k_2}$. Let S denote the QPSK signal set and define the following

(Gray) mapping $\mathcal{M}$ between 2-bit symbols and S,

$$\begin{aligned} 00 &\mapsto 0 \\ 01 &\mapsto 1 \\ 11 &\mapsto 2 \\ 10 &\mapsto 3 \end{aligned}$$

Then $C = \mathcal{M}(C_b)$ is a 2-level LUEP QPSK block modulation code of length n, dimension k and rate

$$R = k/n \quad \text{(bits/T)},$$

where T indicates the time duration of a signal. Then the *squared Euclidean separation vector* of C is

$$\bar{\mathbf{S}}_{SED} = (2\mathbf{s}_1, 2\mathbf{s}_2).$$

As in the case of single-level block modulation codes, given the minimum squared Euclidean distance (MSED) and rate of a modulation code, an *asymptotic coding gain* G is defined for an AWGN channel [13]. For high signal-to-noise ratios, G equals the ratio of the MSED of the coded system to the MSED of an uncoded system transmitting at the same rate (number of bits per signal). Accordingly, for each component of $\bar{\mathbf{S}}_{SED}$ we have an asymptotic coding gain value. In this work we define an *asymptotic coding gain vector* as

$$\bar{G} = (G_1, G_2),$$

where, for $i = 1, 2$,

$$G_i = 10 \log_{10} \left[\frac{2\mathbf{s}_i}{4 \sin^2 (\pi/2^R)} \right] \quad \text{(dB)}$$

Note that, as in the case of conventional block coded modulation systems, these asymptotic coding gains can only be reached if a *soft-decision* decoding is available. In section 5.6, we address the problem of soft decision decoding of LUEP codes, using their trellis structure. To illustrate this construction method, in Table 5.3 some

Table 5.3: Some LUEP QPSK block modulation codes

$2n$	k	k_1	k_2	s_1	s_2	R (bits/T)	G_1 (dB)	G_2 (dB)
4	2	1	1	3	2	1	1.76	0.00
8	5	1	4	4	2	5/4	3.28	0.27
8	5	4	1	3	2	5/4	2.03	0.27
8	6	1	5	3	2	3/2	2.71	0.95
10	5	1	4	5	4	1	3.98	3.01
10	7	1	6	4	2	7/5	3.65	0.64
10	7	4	3	3	2	7/5	2.40	0.64
10	8	1	7	3	2	8/5	3.06	1.30
12	6	1	5	6	4	1	4.77	3.01
12	6	2	4	5	4	1	3.98	3.01
12	9	1	8	4	2	3/2	3.96	0.95
12	9	4	5	3	2	3/2	2.71	0.95
12	10	1	9	3	2	5/3	3.32	1.56
14	7	1	6	7	4	1	5.44	3.01
14	7	4	3	5	4	1	3.98	3.01
14	8	1	7	5	4	8/7	4.07	3.10
14	11	1	10	4	2	11/7	4.21	1.20
14	11	4	7	3	2	11/7	2.96	1.20
14	12	1	11	3	2	12/7	3.51	1.75

LUEP BPSK block modulation codes are listed. Codes in Table 5.3 have the same minimum squared Euclidean distance as that of the *best* QPSK block modulation code with the same values of rate and length [40], and provide additional coding gain, or equivalently smaller probability of bit error, for the k_1 most important message bits. To increase the rate of a block coded modulation system with LUEP codes, we must use an expanded signal constellation. In section 5.5, we present a construction method of LUEP 8-PSK block modulation codes.

5.4 Nonbinary LUEP M-PSK block modulation codes

Another construction method of BCM schemes is to use a linear code over $GF(2^\mu)$ and a mapping $\mathcal{M} : GF(2^\mu) \mapsto S$, where S is a 2^μ-PSK signal constellation. In chapter 3 we introduced a new family of *optimal* 2-level LUEP codes over $GF(2^\mu)$. In this section, we propose to use this codes to obtain 2^μ-PSK block modulation codes as follows. Let C be a 2-level (n, k) LUEP code over $GF(2^\mu)$ with separation vector $\bar{\mathbf{s}} = (\mathbf{s}_1, \mathbf{s}_2)$ for the message space

$$M = GF(2^\mu)^{k_1} \times GF(2^\mu)^{k_2}.$$

Let δ_μ^2 denote minimum Euclidean distance of a 2^μ-PSK signal constellation. After mapping symbols into signal points, we obtain an (n, k) 2^μ-PSK block modulation code C_M, with squared Euclidean separation vector

$$\bar{\mathbf{S}}_{SED} = (\delta_\mu^2 \mathbf{s}_1, \delta_\mu^2 \mathbf{s}_2), \tag{5.10}$$

and rate

$$R = \frac{\mu k}{n} \quad \text{(bits/T)}.$$

Note that any mapping between $GF(2^\mu)$ and S will give the squared Euclidean separation vector (5.10).

Example 9: Let C be a $(71, 66)$ LUEP code over $GF(2^3)$ with separation vector $\bar{\mathbf{s}} = (5, 3)$ that protects 5 information symbols against the occurrence of any two random errors and the remaining 61 information symbols against any single random error. This is the code from Example 2, presented in Chapter 3. Then C_M is a $(71, 66)$ 8-PSK block modulation code with rate

$$R = \frac{3 \times 66}{71} = 2.788 \quad \text{(bits/T)},$$

and squared Euclidean separation vector $\bar{\mathbf{S}}_{SED} = (2.93, 1.76)$. A hypothetical reference system working at the same rate would have a minimum squared Euclidean distance $d_u^2 = 4\sin^2(\pi/2^{2.788}) = 0.77$ and coding gain vector $\bar{\mathbf{G}} = (5.80, 3.68)$, for the message space $M = \{0,1\}^{15} \times \{0,1\}^{183}$. $\triangle\triangle$

Although codes constructed using this technique have relatively good squared Euclidean distances, they require a *soft decision decoding* of nonbinary codes, which requires a large number of computations. In addition, as the size of the signal constellation grows, LUEP codes with increased values of minimum distance (or separation vector) are required. This is not the case for the family of optimal nonbinary 2-level LUEP codes introduced in chapter 3, which always have the same minimum Hamming distance. Finally, as with most block modulation codes, there is a large number of nearest neighbors, resulting in error performance reduction. For the nonbinary optimal LUEP code of example 8 above, the number of codewords of weight 3 is 28244, while the number of codewords of weight 5 is 144207.

5.5 LUEP 8-PSK block modulation codes

To increase the rate of block modulation codes we need to encode the labels of an expanded signal constellation set, as mentioned in section 5.3. The idea is to use the same construction of LUEP QPSK codes for the second stage of a partition of an 8-PSK block modulation code where 8-PSK signals are decomposed into QPSK signals and an additional error correcting code selects which coset of QPSK in 8-PSK is to be used. This is illustrated in Figure 5.6. The original 8-PSK constellation is partitioned into 2 QPSK signal sets, and each code bit from a codeword of a (n, k_1, d_1) binary code C_1 selects which QPSK signal set is to be used. Once a QPSK signal set is selected, each code bit in a codeword of a $(2n, k_2)$ binary LUEP

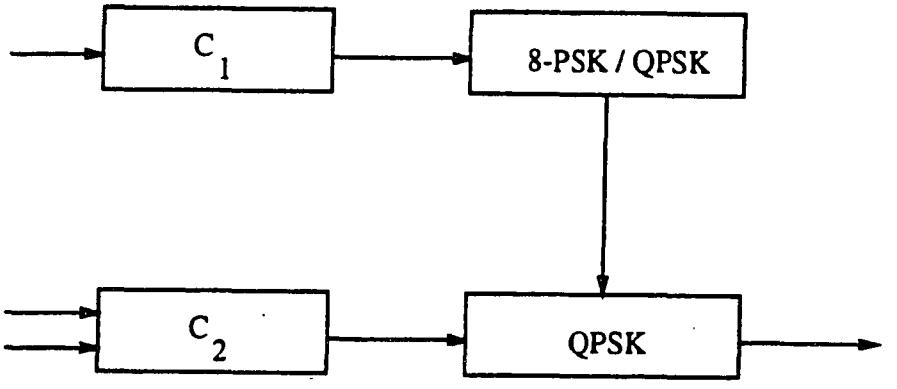

Figure 5.6: Block diagram of an encoder for an LUEP 8-PSK block modulation code

code C_2 is used to select, via Gray mapping, a QPSK signal from within the selected set to be transmitted.

Let C_2 be a $(2n, k_2)$ binary 2-level LUEP code with separation vector $\bar{\mathbf{s}} = (\mathbf{s}_1, \mathbf{s}_2)$ for the message space

$$\{0,1\}^{k_{21}} \times \{0,1\}^{k_{22}},$$

where $k_{21} + k_{22} = k_2$. The resulting code is an $(n, k_1 + k_2)$ LUEP 8-PSK block modulation code of rate

$$R = \frac{k_1 + k_{21} + k_{22}}{n} \quad \text{(bits/T)},$$

and squared Euclidean separation vector

$$\bar{\mathbf{S}}_{SED} = (\mathbf{S}_1, \mathbf{S}_2),$$

where $\mathbf{S}_1 = \min\{0.586 \times d_1, 2 \times \mathbf{s}_1\}$ and $\mathbf{S}_2 = \min\{0.586 \times d_1, 2 \times \mathbf{s}_2\}$, for the message space

$$M = \{0,1\}^{k_1+k_{21}} \times \{0,1\}^{k_{22}}.$$

In table 5.4 some LUEP 8-PSK block modulation codes are listed.

Table 5.4: Some LUEP 8-PSK block modulation codes

n	k_1	d_1	k_{21}	k_{22}	K_1	K_2	s_1	s_2	S_1	S_2	R
7	1	7	1	11	2	11	3	2	4.10	4.00	13/7
8	1	8	1	13	2	13	3	2	4.87	4.00	15/8
9	1	9	1	15	2	15	3	2	5.27	4.00	17/9
10	1	10	1	17	2	17	3	2	5.86	4.00	19/10
11	1	11	1	19	2	19	3	2	6.00	4.00	21/11

5.6 Multi-stage soft-decision decoding of LUEP codes

In previous sections of this chapter, we have presented several alternatives of combining linear block multi-level error correcting codes and 2^μ-PSK modulation systems. Other combinations are possible and may lead to coded modulation schemes with increased minimum squared Euclidean distance or increased rate. However, as mentioned before, to achieve the coding gain promised by a given minimum squared Euclidean distance between code sequences, a *soft-decision* decoding method is required. The penalty in coding gain caused by performing hard-decision decoding is so much (2 to 3 dB), that in some cases it will result in a system that performs worse than the uncoded modulation system. In this section, a new multi-stage soft-decision decoding of multi-level error correcting codes is introduced. It is shown that using the trellises of component codes of a multi-level error correcting code, its trellis structure can be derived. The trellis of an LUEP code can then be used to perform soft-decision decoding using the Viterbi algorithm, with squared Euclidean distance as metric. Multi-stage decoding of BCM schemes was introduced by Imai and Hirakawa [18], and widely used ever since. Multi-level error correcting codes lend themselves naturally to a multi-stage decoding, as mentioned in chapter 3.

Let C be an (n, k) two-level LUEP code with separation vector $\bar{\mathbf{s}} = (\mathbf{s}_1, \mathbf{s}_2)$ for the message space $\{0,1\}^{k_1} \times \{0,1\}^{k_2}$. Then C can be represented as the direct sum of subcodes C_1 and C_2, $C = C_1 \oplus C_2$, i.e,

$$C = \{\bar{\mathbf{c}} = \bar{\mathbf{c}}_1 + \bar{\mathbf{c}}_2 : \bar{\mathbf{c}}_1 \in C_1 \quad \text{and} \quad \bar{\mathbf{c}}_2 \in C_2\},$$

where C_2 is an $(n, k_2, \mathbf{s}_2)$ subcode which contains all codewords of minimum weight of C, and C_1 is an $(n, k_1, \mathbf{s}_1)$ linear code spanned by a system of coset representatives of C_2 in C. Let T_i be a trellis for subcode C_i of C, $i = 1, 2$. Then a trellis of C will be equal to the direct product of T_1 and T_2, $T_1 \otimes T_2$, where the direct product of two trellises is defined as replacing each branch of T_1 at the i-th stage by the i-th stage of T_2. Viterbi maximum likelihood decoding algorithm can then be applied to T to estimate the most likely codeword of C using soft decisions. To reduce the number of computations in soft-decision decoding of a block code, a technique called *multi-stage decoding* is commonly used [18],[39],[20]. A two-stage soft-decision decoding of LUEP codes is as follows:

1. Using soft-decisions (squared Euclidean distance) and the Viterbi algorithm, determine the closest path $\hat{c}_1$ in T_1 to the received sequence. At this decoding step, the most significant message part is estimated.

2. Using soft-decisions and the Viterbi algorithm, determine the closest path $\hat{c}_2$ in $\hat{c}_1 + T_2$ to the received sequence, to estimate the less significant message part. Here $\hat{c}_1 + T_2$ indicates that the metrics of $\hat{c}_1$, obtained in the first decoding stage, are used at each decoding stage of the Viterbi algorithm operating on trellis T_2.

This two-stage soft-decision decoding can be generalized to ℓ-level error correcting codes as follows: If C is an (n, k) ℓ-level error correcting code, then C can be

represented as

$$C = C_1 \oplus C_2 \oplus \cdots \oplus C_\ell,$$

and can be decoded in ℓ stages using T_i of C_i, $i = 1, 2, \cdots, \ell$, and maximum-likelihood decoding using the Viterbi algorithm. Although at each stage the decoding is maximum-likelihood, the multi-stage soft-decision decoding method described above is suboptimal. At each decoding stage, the most likely path is estimated using only part (T_i) of the trellis T of C. This suboptimal multi-stage soft-decision decoding also increases the effective number of nearest neighbors. However, in many cases there is only a fraction of coding gain lost using this multi-stage soft-decision decoding method [25], [6].

Example 10: In this example we construct an LUEP QPSK block modulation code of length 7, and decode it using a two-stage soft-decision decoding algorithm. We use the so-called $|\bar{\mathbf{u}}|\bar{\mathbf{u}} + \bar{\mathbf{v}}|$ construction, which can be viewed as a particular case of *construction X* from chapter 3, where $G_1 = G_2$ is a generator matrix of a linear code C_1 containing codewords $\bar{\mathbf{u}}$, and G_3 is a generator for a linear code C_2 containing codewords $\bar{\mathbf{v}}$. Let C_1 be a $(7, 6, 2)$ parity-check code and C_2 be a $(7, 1, 7)$ repetition code. Then applying construction X we obtain a $(14, 7)$ LUEP code with separation vector $\bar{\mathbf{s}} = (7, 4)$, for the message space $\{0, 1\}^1 \times \{0, 1\}^6$. Now we combine this LUEP code with QPSK modulation just as described in section 5.2. With Gray mapping between 2-bit symbols and QPSK signals, we obtain an LUEP QPSK code C of length 7, rate $R = 1$ (bits/T) and squared Euclidean separation separation vector $\bar{\mathbf{S}}_{SED} = (14, 8)$. The reference uncoded system is BPSK, which has a MSED of 4. It follows that the asymptotic coding gain vector for this block modulation code is $\bar{G} = (5.44, 3.01)$. The trellises of codes C_1 and C_2 are illustrated in Figure 5.7.

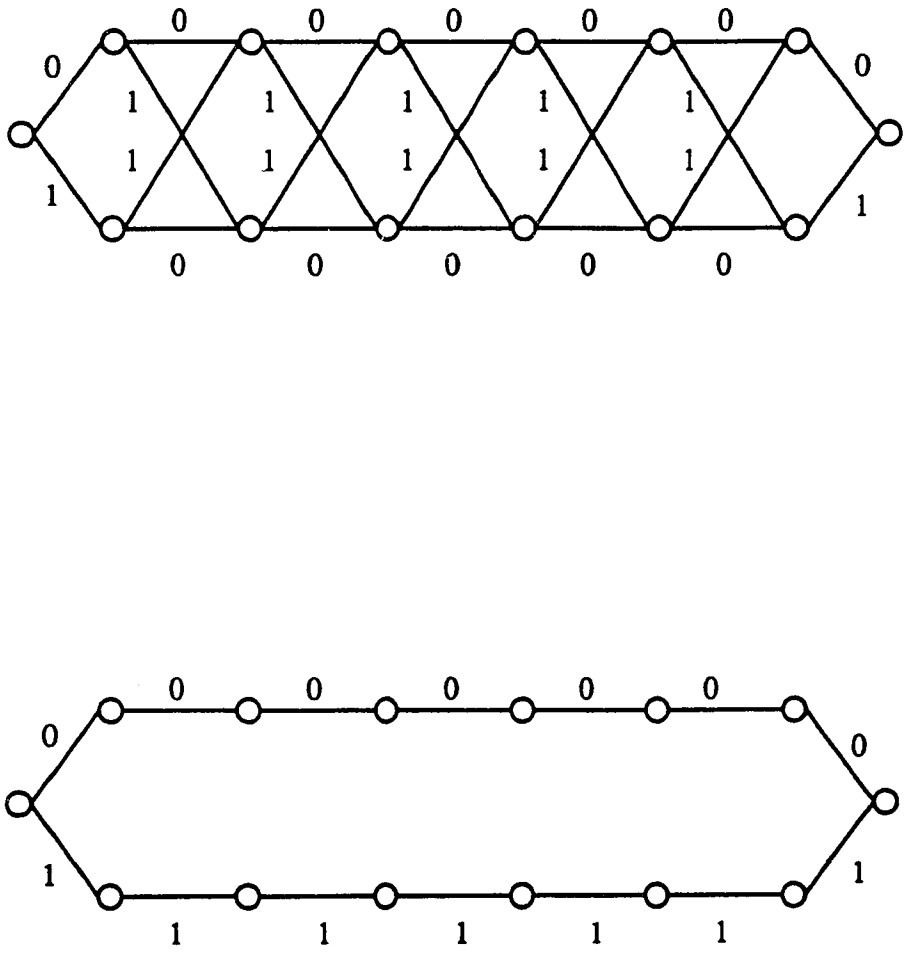

Figure 5.7: Trellises of $(7, 6, 2)$ and $(7, 1, 7)$ codes

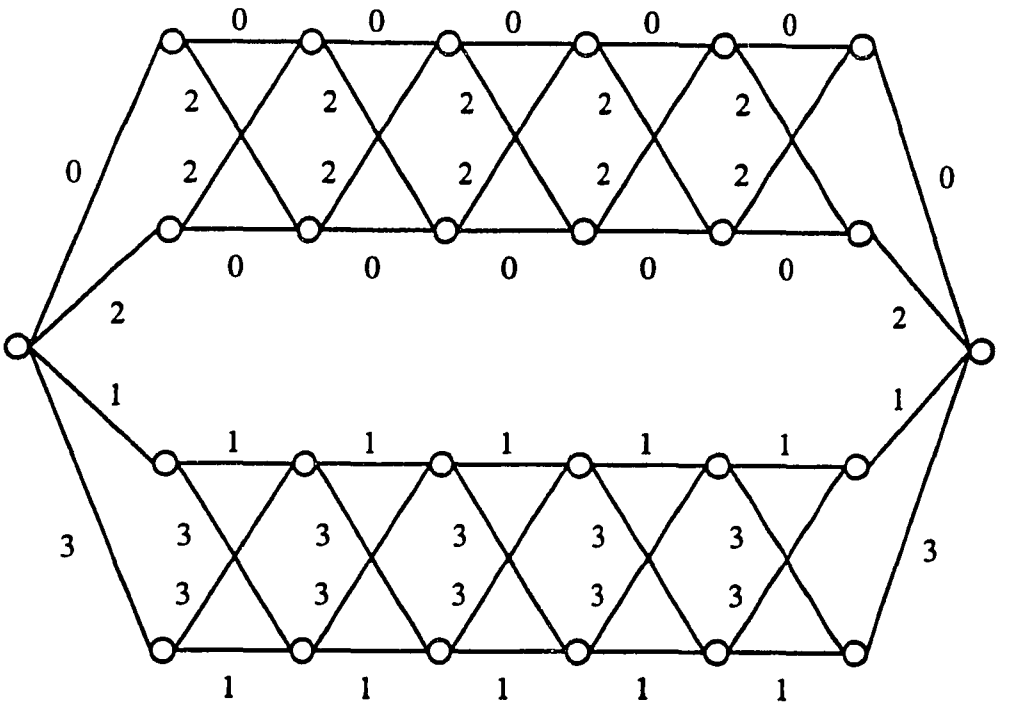

Figure 5.8: Trellis diagram for an LUEP QPSK code of length 7

To obtain the trellis for code C, we repeat each branch of T_1, the trellis of code C_1 twice. This is the $|\bar{\mathbf{u}}|\bar{\mathbf{u}}|$ part of the construction. This is equivalent to substituting 0 by 00 and 1 by 11. We then modify trellis T_2 of code C_2 by interleaving a 0 before each branch, thus constructing the $|0|\bar{\mathbf{v}}|$ part of the code. In this case, this is equivalent of replacing 0 by 00 and 1 by 01. The trellis of the LUEP code is then the direct product of T_1 and T_2, $T_1 \otimes T_2$, corresponding to $|\bar{\mathbf{u}}|\bar{\mathbf{u}}| + |0|\bar{\mathbf{v}}|$. After this is done, we apply Gray mapping and replace each 2-bit symbol by a number in $\{0, 1, 2, 3\}$, the label set for the QPSK signal points. The resulting trellis is shown in Figure 5.8.

Note that the minimum squared Euclidean distance between the subtrellis on the top and the subtrellis on the bottom is $2 \times 7 = 14$, while the minimum squared Euclidean distance between paths within a subtrellis is $2 \times 4 = 8$. Also, the signals

points used in a subtrellis belong to the same BPSK signal subconstellation, i.e., $\{0,2\}$ for the top subtrellis and $\{1,3\}$ for the bottom subtrellis. A two-stage soft-decision decoding can now be performed using the Viterbi algorithm and squared Euclidean distance as a metric. At high signal-to-noise ratio on an AWGN channel, the probability of block error P_e is dominated by the probability of taking a path in the trellis at minimum squared Euclidean distance, which can be approximated by

$$P_e \approx N(d_{min})Q\left(\frac{a\sqrt{d_{min}}}{2\sigma}\right),$$

a^2 is the average signal power. For this LUEP QPSK block modulation code, the probability of block error depends on what message part is being considered. For the least significant bits, we have

$$P_{e_2} = 21Q\left(a\sqrt{2}\right) + 35Q\left(2a\right) + 7Q\left(a\sqrt{8}\right),$$

while for the most significant part (one bit in this case),

$$P_{e_1} = Q\left(a\sqrt{3.5}\right)$$

In both of the above expressions, we assume zero-mean unit-variance additive white Gaussian noise. In Figure 5.9, we plot the probability of bit error for uncoded BPSK and compare it with the bit error probability of the least significant message part, which is approximately $P_{b_2} \approx P_{e_2}/k_2$, and that of the most significant message part, $P_{b_1} = P_{e_1}$. We see form the plot that at $P_b = 10^{-10}$, the real coding gain of this LUEP QPSK block modulation code is about 2.8 dB, while in addition for the most important bit there approximately a 5.4 dB gain. $\triangle\triangle$

Example 11: To illustrate the trellis structure of LUEP codes, in this example we use more powerful codes as component codes in the $|\bar{\mathbf{u}}|\bar{\mathbf{u}} + \bar{\mathbf{v}}|$ construction. Let C_1 be a $(16, 15, 2)$ parity-check code, and C_2 a $(16, 5, 8)$ first-order Reed-Muller code. Then

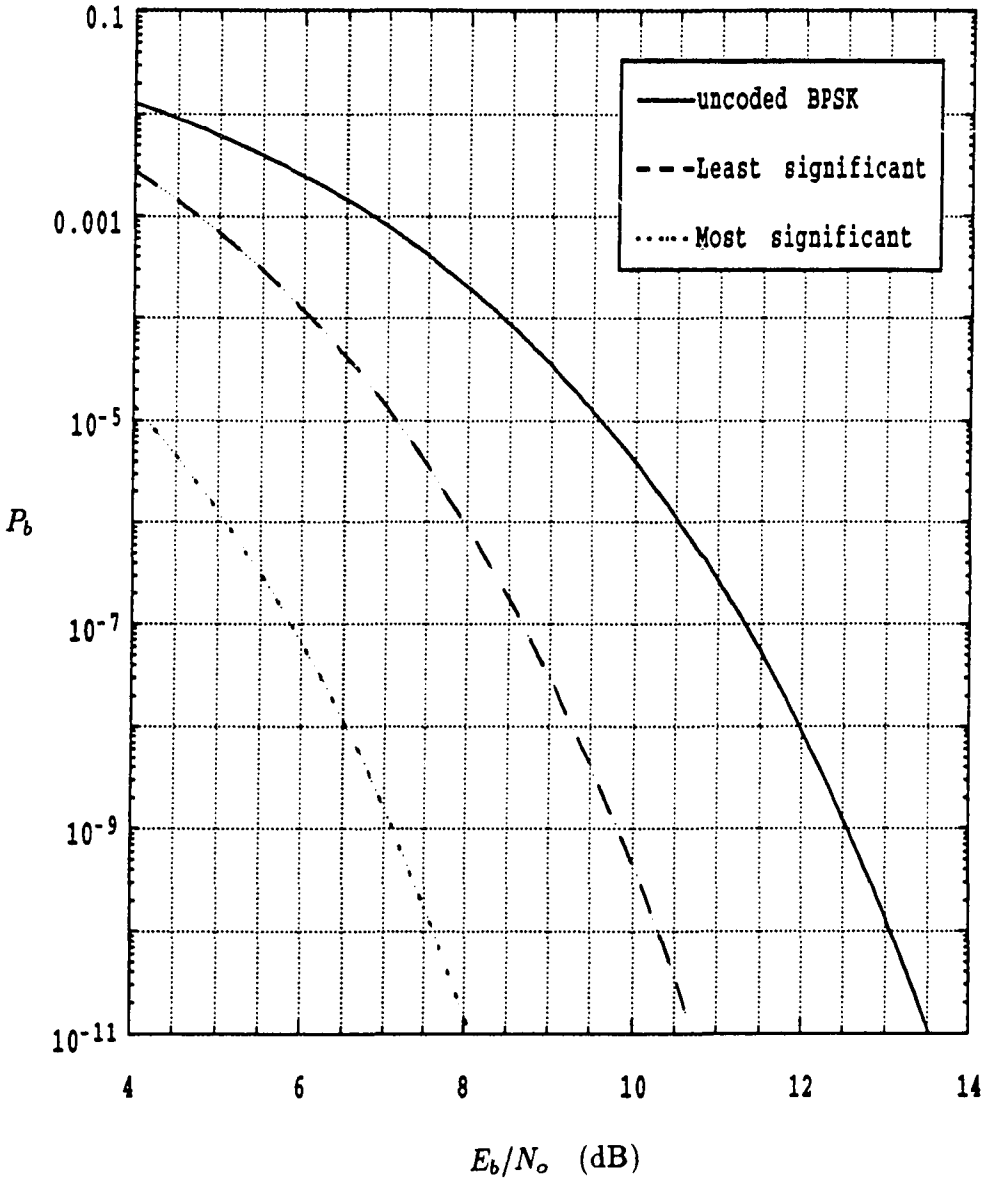

Figure 5.9: Error performance of the LUEP QPSK modulation code of example 10

we obtain a $(32, 20)$ LUEP code with separation vector $\bar{\mathbf{s}} = (8, 4)$ for the message space $\{0,1\}^5 \times \{0,1\}^{15}$. By mapping 2-bit symbols into the set $\{0, 1, 2, 3\}$ of labels of the QPSK signal constellation, we obtain an LUEP QPSK block modulation code of length 16 and rate $R = 5/4$ (bits/T), with squared Euclidean separation vector $\bar{\mathbf{S}} = (16, 8)$. The asymptotic coding gain vector, with respect to a hypothetical PSK system transmitting at the same rate, is $\bar{G} = (6.3, 3.28)$ (dB). A two-stage soft-decision decoding of this code uses the Viterbi algorithm on two trellises, as mentioned above, one for each decoding stage. The trellises are shown in Figure 5.10. $\triangle\triangle$

For LUEP codes based on construction X of section 3.2.1, a two-stage soft-decision may be performed as well, although it is more difficult to determine the trellis structure of component codes.

Example 12: In this example we construct the trellis of a LUEP QPSK block modulation code based on construction X. Let C_1 be a $(8, 4, 4)$ first-order RM code, C_2 be the $(8, 8, 1)$ trivial code and C_3 be a $(8, 4, 4)$ first-order RM code contained in C_2. Then applying construction X to these codes we obtain a $(16, 8)$ LUEP code C_X with separation vector $\bar{\mathbf{s}} = (5, 4)$ for the message space $\{0,1\}^4 \times \{0,1\}^4$. Using the method of section 5.2, we obtain an LUEP QPSK block modulation code of length 8, rate $R = 1$ (bits/T), that has a squared Euclidean separation vector $\bar{\mathbf{S}} = (10, 8)$. The asymptotic coding gain vector with respect to uncoded BPSK is $\bar{G} = (3.98, 3.01)$ (dB). The trellises for each decoding stage are presented in Figure 5.11. $\triangle\triangle$

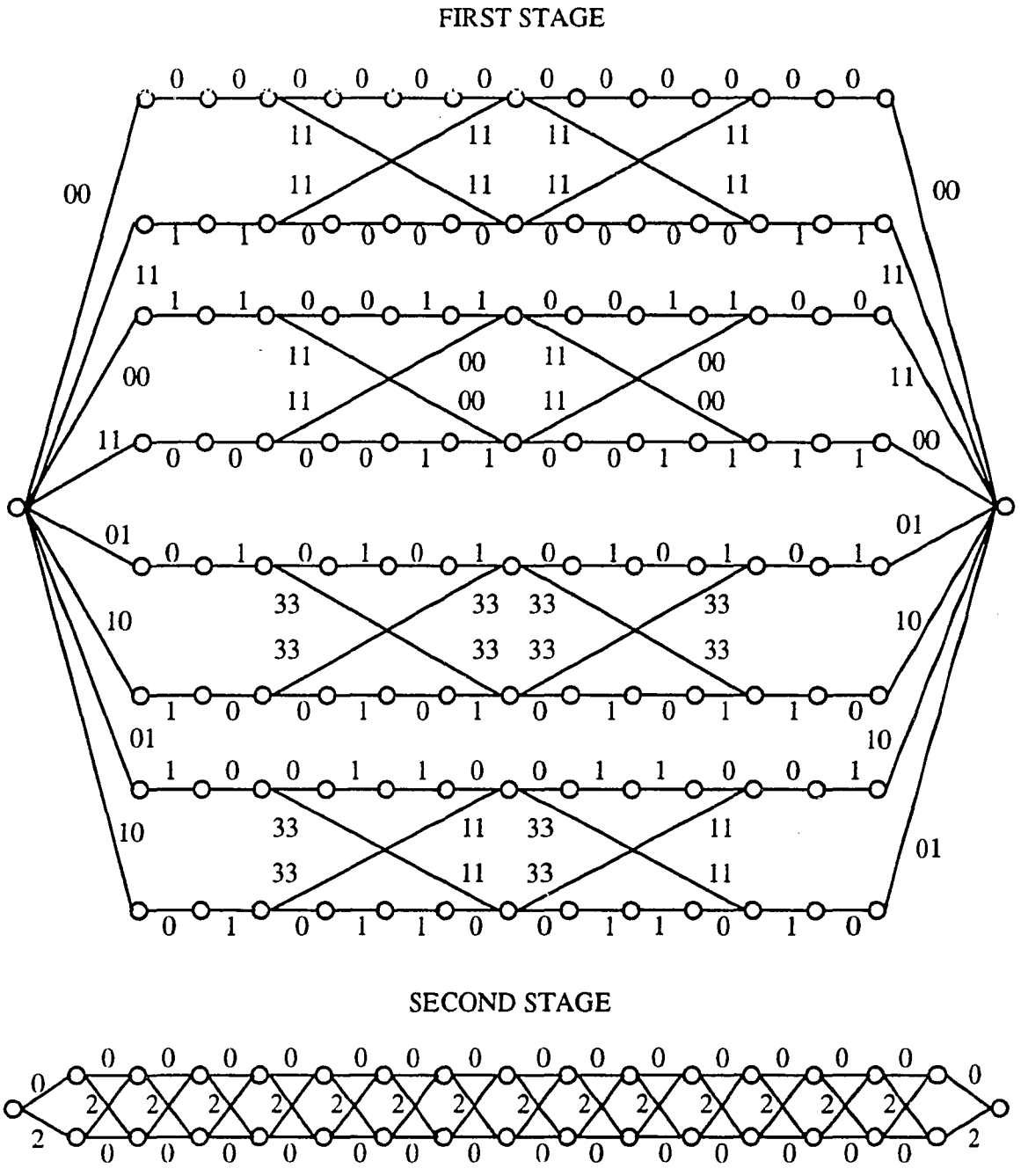

Figure 5.10: Trellises for the LUEP QPSK block modulation code of example 11

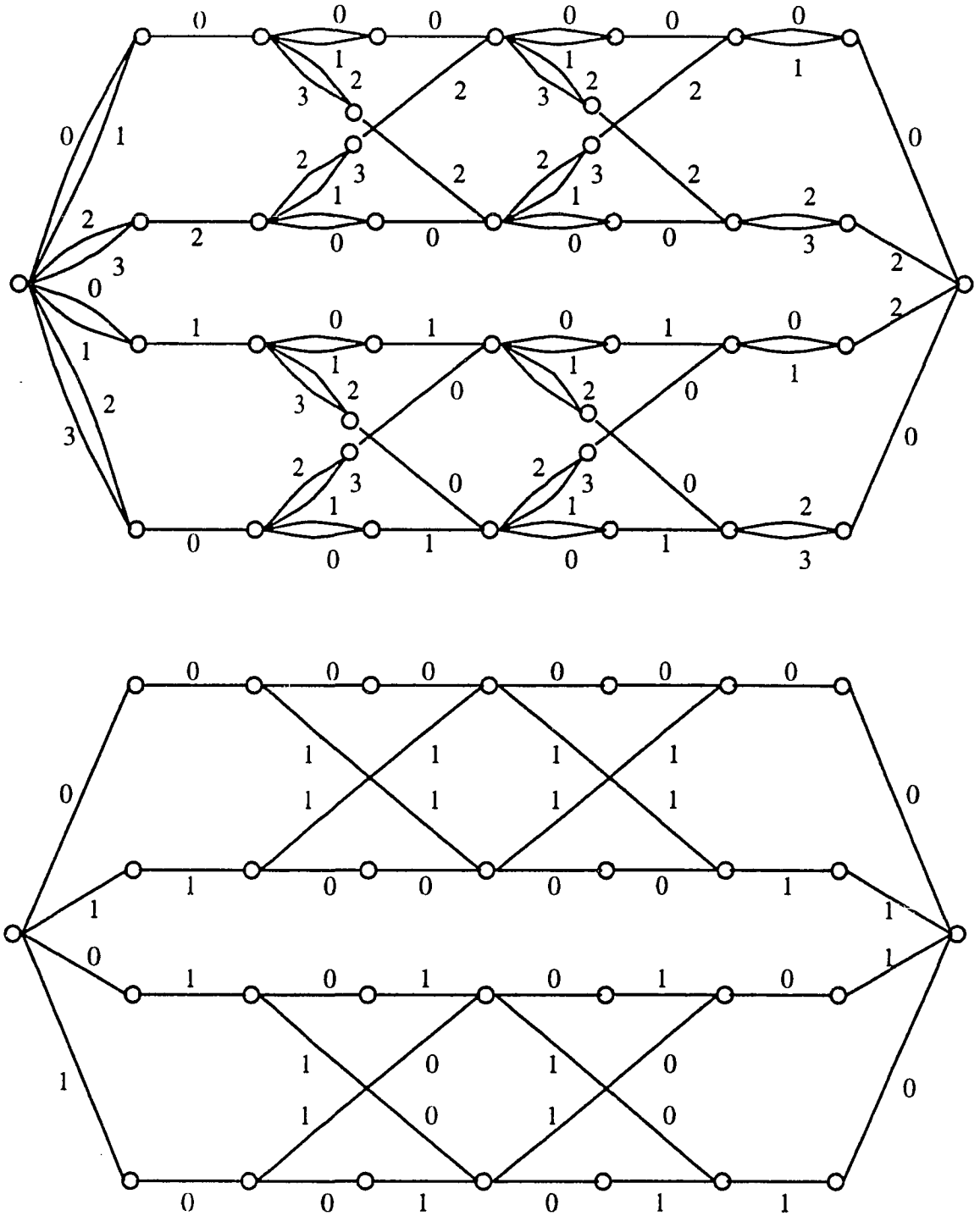

Figure 5.11: Trellises for the LUEP QPSK block modulation code of example 12

5.7 Summary

In this chapter, we presented some applications of multi-level error correcting codes in channels such as the binary symmetric channel (BSC) and the additive white Gaussian noise (AWGN) channel. Bounds on the probability of a block error were derived for the BSC, and shown that optimal binary LUEP codes outperform single-level linear codes of the same minimum distance, because of their additional error correcting capabilities. Error performance graphs were presented, comparing LUEP codes of minimum distance 3, and error correcting capability against 2 or less random error for some message bits, against single-error correcting Hamming codes and double-error correcting BCH codes of the same length. We also examined the error performance of LUEP codes over the AWGN channel with binary phase-shift-keying (BPSK). Bounds on the probability of a block error were derived. It turned out that, in this application, optimal LUEP codes perform worse than single-level error correcting codes of the same minimum distance, unless the length is large enough. This degradation in performance seems to be caused by the lower code rate of LUEP codes, as compared to single-level error correcting codes, which is not compensated by their additional error correcting capabilities. It was also shown that, using quaternary phase-shift-keying (QPSK) and Gray mapping between two-bit symbols and signals, efficient block modulation codes may be obtained. These LUEP QPSK block modulation codes have rate between those of uncoded BPSK and uncoded QPSK, and offer two levels of error correction. Error correction in this application is measured by the minimum squared Euclidean distance among signal sequences associated with a particular message part to be protected. Consequently, we defined a squared Euclidean separation vector (SESV) and a asymptotic coding gain vector. A table of good LUEP QPSK block modulation code was presented. In

addition, the possibility of mapping symbols from $GF(2^s)$ into labels of a 2^s-PSK signal set was considered. Again, a SESV and an asymptotic coding gain vector were defined. An example showed that reasonable asymptotic coding gains can be achieved, using an optimal nonbinary LUEP code over $GF(2^3)$, if a soft-decision decoding is available. We introduced LUEP 8-PSK block modulation codes. These codes are constructed using a two-level partition of an 8-PSK signal set. At the first level, a partition of an 8-PSK signal set into two QPSK signal sets is used, in conjunction with a single-level error correcting code. At the second level, once a QPSK signal set sequence is selected, the QPSK signal sequence to be transmitted is chosen by an LUEP QPSK block modulation code. The corresponding SESV and asymptotic coding gain vector were derived, and a table of efficient LUEP 8-PSK block modulation codes was presented. Finally, to achieve the potential coding gains promised by a given minimum squared Euclidean distance, multi-stage soft-decision decodings were considered. In particular, a two-level soft-decision decoding was presented and illustrated through several examples, showing the trellis structure of the component subcodes decoded at each stage. A plot of the average probability of a bit error against signal-to-noise ratio was presented for a particular LUEP QPSK block modulation code, to show that it is possible to obtain very good performance using LUEP QPSK block modulation codes, while at the same providing a smaller probability of bit error for the most important message part.

Chapter 6

Conclusions and future work

In this thesis, three aspects of multi-level error correcting codes have been considered. First, we have presented several methods of synthesis of LUEP codes, by specifying their parity-check matrix or their generator matrix. Optimal and near optimal LUEP codes were constructed and analyzed. A family of nonbinary optimal LUEP codes, capable of correcting any t or less random errors affecting the most significant information symbols, while correcting any single random error in the least significant information symbols, was introduced. These codes are specified by their parity check matrix, which is a combination of the parity check matrices of Reed-Solomon codes and shortened nonbinary Hamming codes. Near optimal binary LUEP codes, constructed by time-sharing cosets of subcodes in shorter linear Hamming codes, were proposed and analyzed. This class of LUEP codes is specified by their generator matrix. An important advantage of LUEP codes specified by their generator matrix is that they allow the use of multi-stage decoding methods, with reduced decoding complexity. Multi-stage decoding was presented for the case of hard-decision by majority logic and also for soft-decision using trellises of subcodes in an LUEP code. Future research in the area of synthesis of LUEP codes includes the construction of repeated-root cyclic codes, which are based on the $|\bar{\mathbf{u}}|\bar{\mathbf{u}}+\bar{\mathbf{v}}|$ construction. A general construction of nonbinary multi-level error correcting codes based on Reed-Solomon codes, which would have a variety of values of separation vectors, is worth future consideration and research effort.

Second, the analysis of some classes of linear binary block codes was presented. We performed a computer search for good binary cyclic codes of composite length. Using number theoretic arguments, bounds on the multi-level error correcting capabilities of some nonprimitive binary BCH codes were derived. Conditions on the shortening length of shortened binary Hamming codes, for which we obtain two-level error correcting codes, were derived. In future research, the analysis of primitive BCH codes, those whose minimum weight codewords do not span them, must be performed. In particular, primitive BCH codes containing second-order Reed-Muller codes have been shown to have multi-level error-correcting capabilities. Determining the separation vector of these codes remains to be done. Not only primitive BCH codes containing second-order RM codes may be multi-level error correcting codes. Primitive BCH codes containing r-th order RM codes in general may also have multi-level error correcting capabilities. This is a good research problem certainly worth future research effort.

The third aspect covered in this work was the application of multi-level error-correcting codes in block coded modulation schemes. In this direction, we presented the error performance of some binary multi-level error-correcting codes over binary symmetric channels and over additive white Gaussian noise channels with BPSK modulation. To improve the rate and minimum squared Euclidean distance, two new block coded modulation systems using LUEP codes and QPSK or 8-PSK signal constellations were introduced and analyzed. It was shown that with these BCM schemes it is possible to offer multiple values of minimum squared Euclidean distances, each associated with a message part to be protected. The error performance of a specific BCM system using a two-level LUEP code and QPSK modulation with Gray mapping between 2-bit symbols and signal points, was analyzed. Future research in other methods of combining multi-level error correcting codes and M-PSK

or M-QAM signal sets is promising. As in the case of conventional BCM, new constructions should offer higher rates and higher real coding gains, while at the same time trying to keep the decoding complexity as small as possible.

An important finding of this thesis is a new multi-stage soft-decision decoding, based on the trellis structure of LUEP codes. Although suboptimal, this type of decoding permits to achieve most of the coding gain promised by the squared Euclidean separation vector, with reduced decoding complexity (measured by the number of paths and states in the trellis). The analysis of the error performance of multi-stage soft-decision decoding of LUEP codes is a good research problem which deserves further attention.

Bibliography

[1] D. Augot, P. Charpin and N. Sendrier, "Studying the Locator Polynomials of Minimum Weight Codewords of BCH Codes," Abstracts of papers, *1991 IEEE International Symposium of Information Theory*, Budapest, Hungary, June 23-28, 1991.

[2] L.A. Bassalygo, V.A. Zinovev, V.V. Zyablov, M.S. Pinsker and G.S. Poltyrev, "Bounds for Codes with Unequal Protection of Two Sets of Messages," *Probl. Pered. Inform.*, Vol. 15, No. 3, pp. 40-49, July-September 1979.

[3] E.L. Blokh and V.V. Zyablov, "Coding of Generalized Concatenated Codes," *Problem. Pered. Inform.*, Vol. 10, No. 3, pp.45-50, July-September 1974.

[4] I.M. Boyarinov, "Combined Decoding Methods for Linear Codes with Unequal Error Protection of Information Symbols," *Problem. Pered. Inform.*, Vol. 19, No. 1, pp. 17-25, January-March 1983.

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

[6] A.R. Calderbank, "Multilevel Codes and Multistage Decoding," *IEEE Transactions on Information Theory*, Vol. 37, No. 3, pp. 222-229, March 1989.

[7] C. Couvreur and P. Piret, "Codes Between BCH and RS codes," *Philips J. Res.*, Vol. 39, pp. 195-205, 1984.

[8] T.M. Cover, "Broadcast Channels," *IEEE Transactions on Information Theory*, Vol. IT-18, No. 1, pp.2-14, January 1972.

[9] L. Driessen, "On an Infinite Series of (4n,2n) Binary Codes," *IEEE Transactions on Information Theory*, Vol. IT-30, No. 2, pp. 392-395, March 1984.

[10] L.A. Dunning and W.E. Robbins, "Optimal Encodings of Linear Block Codes for Unequal Error Protection," *Information and Control*, Vol. 37, pp. 150-177, May 1978.

[11] V.N. Dynkin and V.A. Togonidze, "Cyclic Codes with Unequal Symbol Protection," *Probl. Pered. Inform.*, Vol.12, No. 1, pp.24-28, January-March 1976.

[12] E.K. Englund, "Nonlinear Unequal Error-Protection Codes are Sometimes Better than Linear Ones," *IEEE Transactions on Information Theory*, Vol. 37, No.5, pp. 1418-1420, September 1991.

[13] G.D. Forney, Jr., "Coset Codes I: Introduction and Geometrical Classification," *IEEE Transactions on Information Theory*, Vol. IT-34, pp. 1123-1151, September 1988, Part II.

[14] T. Fujiwara, T. Kasami and S. Lin, "Error Detecting Capabilities of the Shortened Hamming Codes Adapted for Error Detection in the IEEE Standard 802.3," *IEEE Transactions on Communications*, Vol. 37, No. 9, pp. 986-989, September 1989.

[15] V.V. Ginzburg, "Multidimensional Signals for a Continuous Channel," *Problemy Peredachi Informatsii*, Vol. 20, No. 1, pp. 28-46, January-March, 1984.

[16] W.C. Gore and C.C. Kilgus, "Cyclic Codes with Unequal Error Protection," *IEEE Transactions on Information Theory*, Vol. IT-17, No. 2, pp. 214-215, March 1971.

[17] F. Hemmati, "Closest Coset Decoding of $|u|u+v|$ Codes," *IEEE Journal on Selected Areas in Communications*, Vol. SAC-7, pp. 982-988, August 1989.

[18] H. Imai and S. Hirakawa, "A New Multilevel Coding Method Using Error-Correcting Codes," *IEEE Transactions on Information Theory*, Vol. IT-23, No. 3, pp. 371-377, May, 1977.

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

[20] T. Kasami, T. Takata, T. Fujiwara and S. Lin, "Linear Structure and Error Performance Analysis of Block PSK Modulation Codes," *Transactions of IEICE of Japan*, Vol. J73-A, No. 2, pp. 314-321, February 1990.

[21] T. Kasami, T. Takata, T. Fujiwara and S. Lin, "Representation of Codewords of a Cyclic Code by Boolean Polynomials and Its Application to Trellis Diagram Construction," *Preceedings of the 12th Symposium on Information Theory and Its Applications*, Inuyama, Japan, December 6-9, 1989.

[22] T. Kasami, T.Takata, T. Fujiwara and S. Lin, "On Complexity of Trellis Structure of Linear Block Codes," submitted to *IEEE Transactions on Information Theory*, 1990(in revision for publication).

[23] G.L. Katsman, "Bounds on Volume of Linear Codes with Unequal Information-Symbol Protection," *Probl. Pered. Inform.*, Vol. 16, No. 2, pp. 25-32, April-June 1980.

[24] C.C. Kilgus and Gore, W.C., "A Class of Cyclic Unequal-Error-Protection Codes," *IEEE Transactions on Information Theory*, Vol. IT-18, No. 5, pp. 687-690, September 1972.

[25] F.R. Kischischang, P.G. De Buda and S. Pasupathy, "Block Coset Codes for M-ary Phase Shift Keying," *IEEE Journal on Selected Areas in Communications*, Vol. 7, No. 6, pp. 900-913, August, 1989.

[26] M.C. Lin, *Coding for Unequal Error Protection*, Ph.D. Dissertation, University of Hawaii, December 1986.

[27] M.C. Lin and S. Lin, "Cyclic Unequal Error Protection Codes Constructed from Cyclic Codes of Composite Length," *IEEE Transactions on Information Theory*, Vol. IT-34, No. 4, pp. 867-870, July 1988.

[28] M.C. Lin and S. Lin, "Codes with Multi-Level Error Correcting Capabilities," *Discrete Mathematics*, Vol. 83, pp. 301-314, 1990.

[29] M.C. Lin, C.C. Lin and S. Lin, "Computer Search for Binary Cyclic UEP Codes of Odd Length up to 65," *IEEE Transactions on Information Theory*, Vol. 36, No. 4, pp. 924-935, July 1990.

[30] S. Lin, T. Kasami, T. Takata and T. Fujiwara, "On Multi-Level Block Modulation Codes," *IEEE Transactions on Information Theory*, Vol. 38, No. 4, July 1991.

[31] H.H. Ma and J.K. Wolf, "Binary Unequal Error-Protection Codes from Convolutional Codes by Generalized Tail-Biting," *IEEE Transactions on Information Theory*, Vol. IT-32, No. 6, pp. 776-786, November 1986.

[32] J.F. MacWilliams and N.J.S. Sloane, *The Theory of Error-Control Codes*, Amsterdam: North Holland, 1978.

[33] D. Mandelbaum, "Unequal-Error-Protection Codes Derived from Difference Sets," *IEEE Transactions on Information Theory*, Vol. IT-18, No. 5, pp. 686-687, September 1972.

[34] B. Masnick and J. Wolf, "On Linear Unequal Error Protection Codes," *IEEE Transactions on Information Theory*, Vol. IT-13, No. 4, pp. 600-607, July 1967.

[35] J.L. Massey, "Coding and Modulation in Digital Communications," *Proceedings of the 1974 International Zurich Seminar on Digital Communications*, Zurich, Switzerland, pp. E2(1)-(4), March 1974.

[36] I. Niven and H.S. Zuckerman, *An Introduction to the Theory of Numbers*, John Wiley and Sons, Inc, New York, 1966.

[37] W.W. Peterson and E.J. Weldon, Jr., *Error Correcting Codes*, MIT Press, 2nd ed., 1972.

[38] S.L. Portnoi, "Characteristics of Coding and Modulation Systems from the Standpoint of Concatenated Codes," *Problem. Pered. Inform.*, Vol. 21, No. 3, pp. 14-27, July-September 1985.

[39] G.J. Pottie and D.P. Taylor, "Multilevel Codes Based on Partitioning," *IEEE Transactions on Information Theory*, Vol. 35, No. 1, pp. 87-98, January, 1989.

[40] S.L. Sayegh, "A Class of Optimum Block Codes in Signal Space," *IEEE Transactions on Communications*, Vol. COM-34, no. 10, pp. 1043-1045, October 1986.

[41] N.J.A. Sloane, S.M. Reddy and C.L. Chen, "New Binary Codes," *IEEE Transactions on Information Theory*, Vol. IT-18, No. 4, pp. 503-510, July 1972.

[42] P. Stevens, *About the Transformation of Information Word into Codeword, and Viceversa, for Binary Cyclic LUEP-Codes*, Faculty of Applied Sciences, Vrije Universiteit Brussel, December 1986.

[43] P. Stevens, "On Decoding Unequal Error Protection Product Codes", *IEEE Transactions on Information Theory*, Vol. 36, No. 4, pp. 890-895, July 1990.

[44] R.M. Tanner, "Algebraic Construction of Large Euclidean Distance Combined Coding Modulation Systems," Abstract of Papers, *1986 IEEE International Symposium on Information Theory*, Ann Harbor, October 6-9, 1986.

[45] V.D. Tonchev, "Rank-3 Graphs, Block Designs, and Codes with Unequal Symbol Protection," *Probl. Pered. Inform.*, Vol. 17, No. 2, pp. 19-25, April-June 1981.

[46] G. Ungerboeck, "Channel Coding With Multilevel/Phase Signals," *IEEE Transactions on Information Theory*, Vol. IT-28, No. 1, January 1982, pp. 55-67.

[47] W.J. Van Gils, *On Linear Unequal Error Protection Codes*, Master's Thesis, Eindhoven University of Technology, July 1982.

[48] W.J. Van Gils, "Two Topics on Linear Unequal Error Protection Codes: Bounds on Their Length and Cyclic Code Classes," *IEEE Transactions on Information Theory*, Vol. IT-29, No. 6, pp. 866-876, November 1983.

[49] W.J. Van Gils, "Linear Unequal Error Protection Codes from Shorter Codes," *IEEE Transactions on Information Theory*, Vol. IT-30, No. 3, pp. 544-546, May 1984.

[50] V.A. Zinovev and V.V. Zyablov, "Codes with Unequal Protection of Information Symbols," *Problem. Pered. Inform.*, Vol. 15, No. 3, pp. 50-60, July-September 1979.

[51] V.V. Zyablov and S.L. Portnoi, "Modulation/Coding System for a Gaussian Channel," *Problemy Peredachi Informatsii*, Vol. 23, No. 1, pp. 18-26, July-September, 1987.