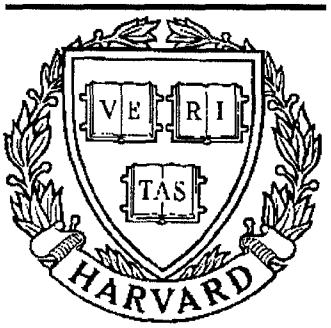
THESIS REPORT
Ph.D.

SYSTEMS
RESEARCH
CENTER



Supported by the National Science Foundation Engineering Research Center Program (NSFD CD 8803012), the University of Maryland, Harvard University, and Industry

New Codes and New Approaches for Code Division Multiple Access on Optical Fiber Networks

by G-C. Yang
Advisors: T.E. Fuja

Ph.D. 92-12

Abstract

Title of Dissertation: NEW CODES AND NEW APPROACHES FOR
CODE DIVISION MULTIPLE ACCESS
ON OPTICAL FIBER NETWORKS

Guu-Chang Yang, Doctor of Philosophy, 1992

Dissertation directed by: Thomas E. Fuja, Assistant Professor
Electrical Engineering Department

This thesis concerns the use of optical fiber in a multi-user communication network. Specifically, it considers the use of code-division multiple access (CDMA) techniques that permit many users to share a single optical channel through the assignment of unique "signature sequences".

In most optical systems – where incoherent processing means that only signal intensity is measured – there are no negative signal components, and the effect on code design is profound. This was noted by Salehi and others in their design of *optical orthogonal codes* (OOC's).

The results in this thesis extend previous work on OOC's in several areas. First, we consider codes for which the auto- and cross-correlation constraints are not equal. We observe that the effects of the two constraints on system performance are not identical, and so considering only codes for which they *are* identical may lead to a sub-optimal code. Bounds on such OOC's are derived and techniques for constructing them are described.

OOC's with unequal auto- and cross-correlation constraints may be viewed as constant-weight unequal error protection (UEP) codes; therefore we interpret the bounds and constructions in that context and compare them with previous work on UEP codes.

We develop bounds on the size of "variable weight" OOC's and demonstrate techniques for building them.

By considering the original CDMA system as an inner code and adding a channel encoder for each user as an outer code, we demonstrate a concatenated coding scheme that can improve system performance without increasing the weight of the OOC sequences.

A new approach to CDMA on optical fiber networks employing pulse position modulation (PPM) is developed. We propose assigning to each user in the network two signature sequences – a synchronization sequence and a data sequence. We stipulate the required properties of such sequences, and we demonstrate that in some cases more users can be accommodated with this approach than with the assignment of single sequences that "do it all".

NEW CODES AND NEW APPROACHES FOR CODE DIVISION MULTIPLE ACCESS ON OPTICAL FIBER NETWORKS

by

Guu-Chang Yang

Dissertation submitted to the Faculty of the Graduate School
of The University of Maryland in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
1992

Advisory Committee:

Assistant Professor Thomas E. Fuja, Chairman/Advisor
Professor Evaggelos Geraniotis
Professor Raymond Miller
Assistant Professor Adrianos Papamarcou
Assistant Professor KuoJuey Ray Liu

Acknowledgments

I would like to thank Professor Fuja for his guidance and encouragement during the preparation of this thesis. This work could not have been completed without his advice and support. I also like to thank my parents and my fiancée, especially for their endless love and encouragement. They are always around me when I need them.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Background

A multiple access communication system is a communication system where a number of users share a common transmission medium to transmit messages to a number of destinations. One of the key issues that must be resolved in moving from a single user communication system to a multi-user communication system is how we can efficiently divide the available transmission medium among all users. Time division multiple access (TDMA) and frequency division multiple access (FDMA) are two common multiplexing techniques. In TDMA (FDMA), each user is allocated to a different time slot (frequency) for transmission. TDMA and FDMA schemes are extremely effective when the traffic is heavy. In terms of bursty or sporadic traffic, TDMA and FDMA, however, are inefficient. This behavior is typical in local area networks (LANs) or data communications. Illustrations of TDMA and FDMA schemes are in Figures 1.1 and 1.2.

The solutions which are widely used in LANs are random multiple access techniques. One of random multiple access techniques which is used in the popular Ethernet LAN is called carrier sense multiple access/collision detection

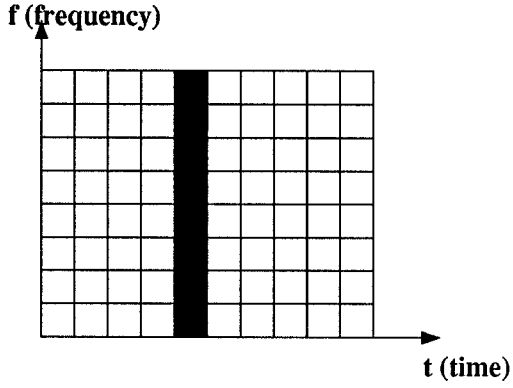


Figure 1.1: A time division multiple access scheme.

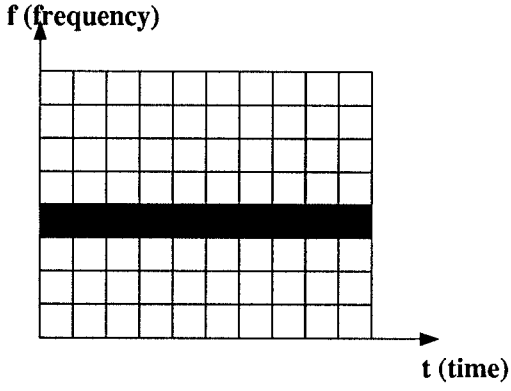


Figure 1.2: A frequency division multiple access scheme.

(CSMA/CD). The "collision" problem that occurs when several active users send messages simultaneously is handled by CSMA/CD. Each active user first listens to the medium to determine if a transmission is in progress before proceeding to transmit, so that it does not interfere with the ongoing transmission. Furthermore, if two active users start to transmit almost simultaneously, they will shortly detect a collision in process and both cease transmitting. After a random delay, each user will retransmit his message. CSMA/CD can increase throughput and decrease average delay significantly[1].

An alternative for random multiple access techniques is code division multiple access (CDMA), which provides multiple access capability with a fixed delay (the delay is proportional to the length of the signature codeword). This thesis investigates problems for CDMA on optical fiber networks. We begin with reviewing some basic concepts of CDMA.

1.2 Review of CDMA

CDMA—also known as spread spectrum multiple access (SSMA)—was developed from spread spectrum techniques established for antijam and multipath rejection applications as well as accurate ranging and tracking[2]. CDMA techniques permit many users to share a single transmission medium through the assignment of unique "signature sequences." This approach has a long history as applied to communication channels, such as radio waves, coaxial cable, and satellite, where the modulated signals can have both positive and negative components – e.g. binary phase shift keying. However, in optical systems – where the incoherent processing means that only signal intensity is measured – there are no negative components. The lack of "negative components" in current optical transmission

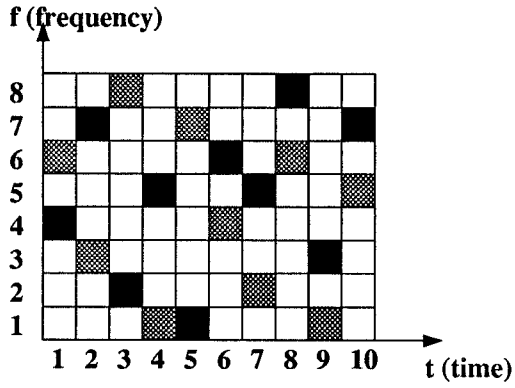


Figure 1.3: A FH CDMA scheme with FH sequences $(4, 7, 2, 5, 1, 6, 5, 8, 3, 7)$ and $(6, 3, 8, 1, 7, 4, 2, 6, 1, 5)$.

technology dictates the use of the $(0, 1)$ notation, which is different from related techniques where the $(+1, -1)$ notation reflects the fact that binary phase shift keying (BPSK) is explicitly or implicitly assumed[3][4].

Depending on the different spreading techniques, there are different forms of CDMA, such as direct sequence (or pseudo-noise) CDMA, frequency-hopping CDMA, time-hopping CDMA, and hybrid forms. Among these spreading techniques, direct sequence (DS) and frequency-hopping (FH) are the most commonly used. The difference between DS CDMA and FH CDMA is that the bandwidth is spread by direct modulation of a data-modulated carrier with a wideband spreading code in DS CDMA; on the other hand, the spreading code in FH CDMA does not directly modulate the data-modulated carrier but is instead used to control the sequence of carrier frequencies. Thus the transmitted signal appears as a data-modulated carrier which is hopping from one frequency to the next[5]. Figure 1.3 illustrates two FH sequences in the time-frequency plane.

DS CDMA optical fiber systems take advantage of excess bandwidth in single-mode fibers to map low information rate electronical or optical signals

into high rate optical a sequences to achieve random, asynchronous communication access, free of network control among the users[6]. The use of a DS CDMA fiber-optic system is described in [6][7]. In an optical network we assume that time is broken into discrete slots; within each slot each user can either transmit an optical pulse (i.e. - a physical "1") or not transmit an optical pulse (i.e. - a physical "0"). If we assume incoherent processing – i.e., the phase of the signal is not available – then multiple pulses transmitted simultaneously by different users add. Specifically, if J users transmit a pulse during the same slot then the receiver observes a pulse of intensity JE, where E is the intensity received when a single user transmits a pulse.

Each user in the network is assigned a codeword containing w ones from a CDMA code. The codeword assigned to a user is that user's "signature sequence", and when the user wishes to convey a logical "1" he transmits the corresponding sequence of pulses and pauses; when the user wishes to convey a logical "0" he transmits nothing for n slots. At the receiver each user computes the correlation of the received sequence with that user's signature sequence; because of the low auto- and cross-correlation properties the correlation typically stays low until a logical "1" is indicated by a correlation of w. In this way each user can recover his own logical sequence.

The block diagram in Figure 1.4 depicts a DS CDMA scheme.

The i^{th} user's binary signal $s_i(t)$ at the output of i^{th} channel encoder is given by

$$s_i(t) = b_i(t)c_i(t),$$

where $b_i(t)$ and $c_i(t)$ are the i^{th} user's binary modulated data signal and binary modulated signature sequence, respectively.

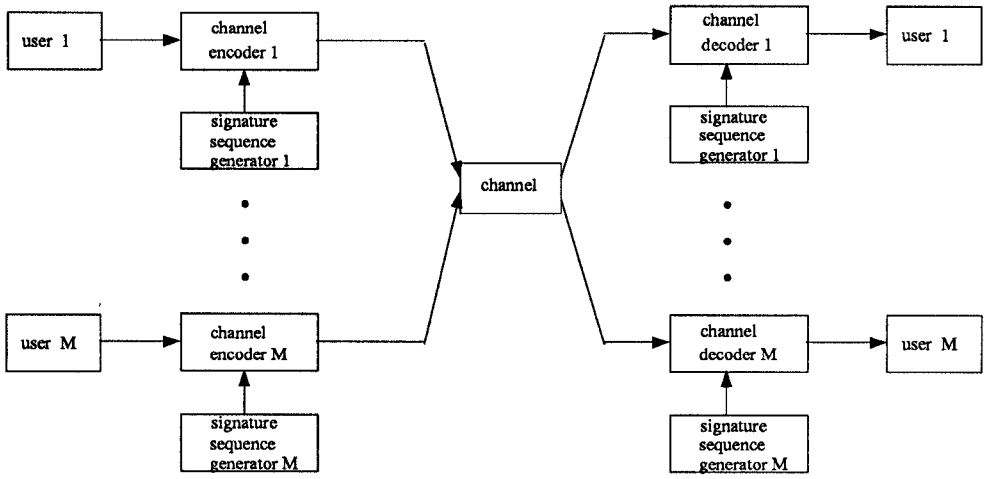


Figure 1.4: A DS CDMA scheme.

The i^{th} user's binary modulated data signal $b_i(t)$ is given by

$$b_i(t) = \sum_{k=0}^{\infty} b_k^{(i)} P_{T_d}(t - kT_d),$$

where $b_k^{(i)}$ is the i^{th} user's k^{th} data signal that takes on 0 or 1 (we use on-of keying in optical fiber CDMA system.) and $P_{T_d}(t)$ is a rectangular pulse of duration T_d which starts at $t = 0$.

The i^{th} user's binary modulated signature sequence is given by

$$c_i(t) = \sum_{l=0}^{\infty} c_l^{(i)} P_{T_c}(t - lT_c),$$

where $c_l^{(i)}$ (0 or 1) is the l^{th} bit of i^{th} user's signature sequence repeated cyclically with period (= the length of the sequence) $n = \frac{T_d}{T_c}$, i.e., $c_l^{(i)} = c_{n+l}^{(i)}$, and P_{T_c} is a rectangular pulse of duration T_c which is start at $t = 0$.

The encoding process is illustrated in Figure 1.5.

The decoding process is as follows. The received signal at the front end of each decoder is given by

$$r(t) = \sum_{i=1}^{M} s_i(t - \tau_i) = \sum_{i=1}^{M} b_i(t - \tau_i) c_i(t - \tau_i),$$

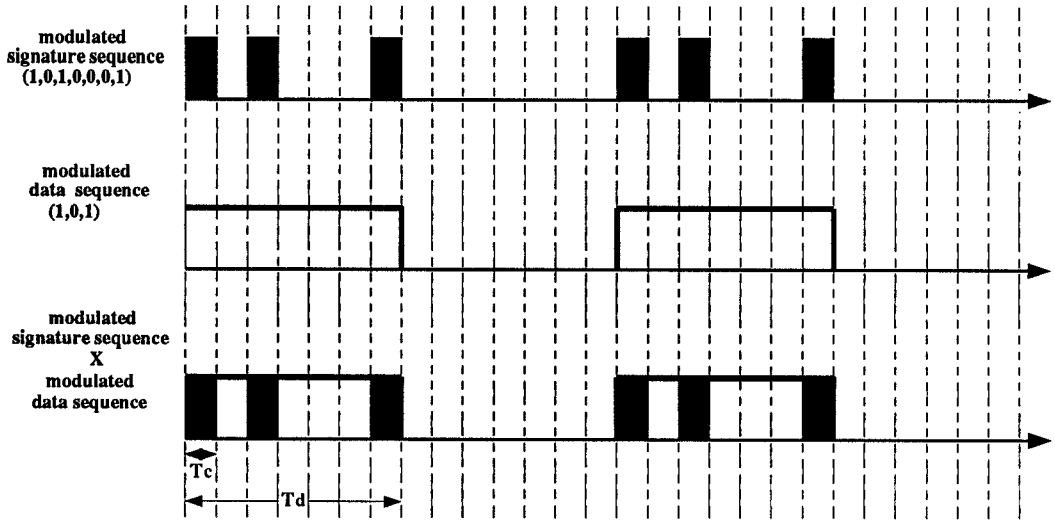


Figure 1.5: An encooding process of a DS CDMA scheme.

where τ_i is the time delay associated with the i^{th} signal.

The i^{th} user's signature sequence generator in the decoder end will store a replica of i^{th} user's signature sequence in the encoder end. Hence, the output of i^{th} user correlation decoder Z_i is equal to

$$Z_i = b_i^{(i)} + I_i,$$

where I_i is the undesired signal comes from the other users.

The output Z_i would then be compared to a threshold level at the comparator for the data recovery.

1.3 Motivation and Outline of Thesis

In order to extract its signature sequence in the presence of other users' signature sequences for a desired user, a set of signature sequences need to satisfy the following two properties:

- **Auto-correlation property:** Each sequence can easily be distinguished from a shifted version of itself.
- **Cross-correlation property:** Each sequence can be easily distinguished from every other sequence in the set.

An optical orthogonal code (OOC) is a collection of binary sequences with good auto- and cross-correlation properties; they were defined by Salehi and others as a means of obtaining code division multiple access on optical networks.

What follows is the definition of an OOC given by Salehi *et. al.* [8].

Definition: An $(n, w, \lambda_a, \lambda_c)$ optical orthogonal code $\mathcal{C}$ is a collection of binary n-tuples, each of Hamming weight w, such that the following two properties hold:

- (Auto-correlation) For any $\mathbf{x} = [x_0, \ldots, x_{n-1}] \in \mathcal{C}$ and any integer τ, $0 < \tau < n$,
$$\sum_{t=0}^{n-1} x_t x_{t\oplus\tau} \leq \lambda_a.$$
- (Cross-correlation) For any $\mathbf{x} = [x_0, \ldots, x_{n-1}] \in \mathcal{C}$ and any $\mathbf{y} = [y_0, \ldots, y_{n-1}] \in \mathcal{C}$ such that $\mathbf{x} \neq \mathbf{y}$ and any integer τ,
$$\sum_{t=0}^{n-1} x_t y_{t\oplus\tau} \leq \lambda_c.$$

Note: OOC's were defined in terms of periodic correlation; thus the addition in the subscripts above – denoted "$\oplus$" – is all modulo-n.

The two correlation constraints serve two purposes:

- The auto-correlation constraint guarantees that each signature sequence is unlike cyclic shifts of itself. This property is used to enable the receiver to obtain *synchronization* – that is, to find the beginning of its message and subsequently locate the codeword boundaries.

- The cross-correlation constraint guarantees that each signature sequence is unlike cyclic shifts of the *other* signature sequences. This property enables the receiver to estimate its message in the presence of interference from other users. Thus, the cross-correlation constraint aids both synchronization in the presence of multiple users and permits each receiver to "track" its message after synchronization is achieved.

Therefore, the auto-correlation constraint contributes only to synchronization, while the cross-correlation constraint affects both synchronization *and* operation.

A reasonable "figure of merit" for a code is the number of interfering users necessary to cause the code to fail. For instance, assume synchronization has been achieved; then the only errors the i^{th} user can make are $0 \rightarrow 1$ errors, and they can only occur when enough *other* users interfere to make the correlation at the i^{th} receiver exceed w. Since each of those other users can contribute at most λ_c to the correlation, the performance "figure of merit" is w/λ_c. In a similar vein, the synchronization "figure of merit" is $(w - \lambda_a)/\lambda_c$ for multiple-access synchronization and $w - \lambda_a$ for single-user synchronization.

Up to now all work on OOC's have assumed that the constraint placed on the auto-correlation and that placed on the cross-correlation are the same [6][7][8][9]. In Chapter 2 we consider codes for which the two constraints are *not* equal. Specifically, we develop bounds on the size of such OOC's and demonstrate construction techniques for building them. The results demonstrate that a significant increase in the code size is possible by letting the auto-correlation constraint exceed the cross-correlation constraint. These results suggest that for a given performance requirement the optimal OOC may be one with unequal constraints[10][11][12].

In terms of multi-performance requirements for CDMA optical networks, we propose two approaches to achieve the requirements. The first approach is to use variable weight OOCs in a CDMA optical network. Up to now all work on OOC's have assumed that the weight of each codeword is the same. In Chapter 3 we develop bounds on the size of OOC's when this assumption is removed. In addition, we demonstrate construction techniques for building such "variable weight" OOC's. The results demonstrate that it is possible to assign codewords with different weights among the users. Since the weight of a user's signature sequence directly affects the user's performance, this is useful for CDMA on optical networks with different performance requirements among the users.

Another approach is to use a so-called concatenated coding scheme. To obtain a given level of performance for CDMA on an optical system with a given number of users, we often choose a bigger weight OOC. In Chapter 4 we demonstrate a concatenated coding scheme that achieves a required performance without increasing the weight of the OOC sequences. By considering the original CDMA system as the inner code and adding a channel encoder for each user as an outer code, we have demonstrated that concatenated coding scheme can increase the information rate when compared with a "pure" CDMA system. Furthermore, by using a different outer code for each user, we can meet the performance requirements for users with different priorities.

The increasing use of optical communications has fostered a renewed interest in pulse position modulation (PPM) [13][14]. In Chapter 5 we considers a new approach to CDMA on optical fiber networks employing PPM. We propose assigning to each user in the network two signature sequences – a synchronization sequence that the user will use to establish frame synchronization and a data sequence that will be used to convey information once frame synchronization is

established.

Previous work on CDMA for optical channels has assumed the assignment of a single signature sequence to each user. In some designs – e.g., prime sequences – the auto-correlation of each signature sequences is high, creating possible problems with respect to synchronization[15]. In other designs – e.g., optical orthogonal codes – the auto-correlation of sequences is kept low; however, this property is useless once frame synchronization is maintained, and it limits the number of signature sequences that can be constructed.

Our approach requires a set of synchronization sequences with good auto-correlation properties as well as a set of data sequences with good cross-correlation properties; in addition, good "global" correlation properties – e.g. constraints on the cross-correlation between data and synchronization sequences – is necessary. We demonstrate how this can be attained using extended quadratic congruence (QC) sequences [16] for synchronization and extended prime sequences as data sequences.

Finally, conclusions and further researches are presented in Chapter 6.

1.4 Other Applications

There are several other applications for $(0, 1)$ sequences with good correlation properties other than in CDMA optical fiber system.

The definition of an OOC can be recast in terms of Hamming distance; doing so makes clearer the parallels between OOC's and constant weight error correcting codes.

Notation: Given a binary n-tuple $\mathbf{x} = [x_0, x_1, \ldots, x_{n-1}]$, let $D^i\mathbf{x}$ $(0 \leq i \leq n-1)$ denote the binary n-tuple obtained by performing i right-cyclic shifts on

$\mathbf{x}$ – i.e., $D^i\mathbf{x} = [x_{n-i}, x_{n-i+1}, \ldots, x_{n-1}, x_0, x_1, \ldots, x_{n-i-1}]$.

Alternate Definition: An $(n, w, \lambda_a, \lambda_c)$ optical orthogonal code $\mathcal{C}$ is a collection of binary n-tuples, each of Hamming weight w, such that the following two properties hold:

- (Auto-correlation) For any $\mathbf{x} = [x_0, \ldots, x_{n-1}] \in \mathcal{C}$ we have $d^a_{\min}(\mathbf{x}) \geq 2w - 2\lambda_a$, where $d^a_{\min}(\mathbf{x}) \triangleq \min\{d_H(\mathbf{x}, D^\tau\mathbf{x}) : \tau = 1, 2, \ldots, n-1\}$.
- (Cross-correlation) For any $\mathbf{x} = [x_0, \ldots, x_{n-1}] \in \mathcal{C}$ and any $\mathbf{y} = [y_0, \ldots, y_{n-1}] \in \mathcal{C}$, we have $d^c_{\min}(\mathbf{x}, \mathbf{y}) \geq 2w - 2\lambda_c$, where $d^c_{\min}(\mathbf{x}, \mathbf{y}) \triangleq \min\{d_H(\mathbf{x}, D^\tau\mathbf{y}) : \tau = 0, 1, \ldots, n-1\}$.

Consider the partition of binary n-tuples into "clouds", where every cloud consists of cyclic shifts of the same n-tuple. Then constructing an OOC consists of picking at most one n-tuple from every cloud under two constraints; the first constraint specifies the minimum Hamming distance *within* a cloud, while the second specifies the minimum Hamming distance *between* clouds.

An unequal error protection (UEP) code is a code that protects some positions in a message word against more errors than other ones. Then an OOC with unequal auto- and cross-correlation constraints can be considered as a two-level UEP code[17][18][19]. Suppose you have such an OOC with cardinality M. Now consider the two-level UEP code consisting of all the n-tuples of the OOC and all their cyclic shifts, the resulting code has nM codewords. Some information bits of the UEP code pick a cloud and others pick codeword from cloud.

Throughout the thesis, OOC's and PPM codes are defined in terms of periodic correlation. The increased interest in spread spectrum communication has led a corresponding increased interest in codes with aperiodic correlation constraints. Since periodic correlation is a more restricted constraint than ape-

riodic correlation, therefore, codes which satisfy periodic correlation constraints will satisfy aperiodic correlation correlation constraints. Codes with aperiodic correlation constraints can be obtained by shifting and truncating codes with periodic correlation constraints[20]. Various bounds and constructions for codes with aperiodic correlation constraints are given in [21][22][23].

In radar, sonar, Costas arrays, and FH CDMA codes, the auto- and cross-correlation are treated in a two dimensional time-frequency array [24][25]. However, the design methods such as balanced incomplete block designs and finite projective planes are used in both one and two dimensional cases. The idea to consider the unequal auto- and cross-correlation constraints can be used in these two dimensional arrays.

In 1964, Kautz and Singleton defined two classes of codes that permit many users to share a common OR-channel[26]. The first class of codes is uniquely decodable codes. A T-user code $(C_1, C_2, \cdots, C_T)$ is said to be uniquely decodable if all sums ("Or" operation) consisting of one codeword from each constituent code are distinct. The other class of codes is disjunctive codes. Disjunctive codes of order m can be used for identifying m out of T ($m << T$) users sharing a multiple access OR-channel. The fact that asynchronous communication is assumed reduces the size of the OOC's since no cyclic shift of any codeword can be used. If both block and bit synchronism are maintained in the system, a disjunctive code guarantees unique identification of active users as long as the number of active users does not exceed m out of T ($m << T$)[27][28]. We can relate OOC's to disjunctive codes by using all cyclic shifts of the codewords as the codewords of disjunctive codes. Let $N(m, T)$ denote the minimum possible length of a disjunctive code of order m and size T. Then an (n, w, λ, λ) OOC with t codewords is an $N(\lfloor \frac{w-1}{\lambda} \rfloor, nt)$ disjunctive code.

Codes which are designed for CDMA on optical fiber networks can be used for CDMA on radio or satellite network where binary phase shift keying (BPSK) is explicitly or implicitly assumed. An $(n, w, \lambda_a, \lambda_c)$ OOC is an $(n, w, n - 4w + 4\lambda_a, n - 4w + 4\lambda_c)$ code with BPSK modulation technique.

There are other applications such as neuromorphic CDMA systems and coherent ultra-short light pulse CDMA systems. The details are in [29].

Chapter 2

Optical Orthogonal Codes with Unequal Auto- and Cross-correlation Constraints

An optical orthogonal code (OOC) is a collection of binary sequences with good auto- and cross-correlation properties; they were defined by Salehi and others as a means of obtaining code division multiple access on optical networks. Up to now all work on OOC's have assumed that the constraint placed on the auto-correlation and that placed on the cross-correlation are the same. In this chapter we consider codes for which the two constraints are *not* equal. Specifically, we develop bounds on the size of such OOC's and demonstrate construction techniques for building them. The results demonstrate that a significant increase in the code size is possible by letting the auto-correlation constraint exceed the cross-correlation constraint. These results suggest that for a given performance requirement the optimal OOC may be one with unequal constraints.

This chapter also views OOC's with unequal auto- and cross-correlation constraints as constant-weight unequal error protection (UEP) codes with two levels of protection. The bounds derived are interpreted from this viewpoint and are compared with previous work on UEP codes.

2.1 Introduction

This chapter concerns the use of optical fiber in a multi-user communication network. Specifically, it considers the use of code-division multiple access techniques that permit many users to share a single optical channel through the assignment of unique "signature sequences".

This approach has a long history as applied to communication channels where the modulated signals can have both positive and negative components – e.g. binary phase shift keying. However, in optical systems – where incoherent processing means that only signal intensity is measured – there are no negative components, and the effect on code design is profound. This was noted by Salehi and others in their design of *optical orthogonal codes* (OOC's) [6][7][8][9].

The results in this chapter extend previous work on optical orthogonal codes in that they consider codes for which the auto- and cross-correlation constraints are equal. We demonstrate that the effects of the two constraints on system performance are not identical, and so considering only codes for which they are identical may lead to a sub-optimal code. Bounds on such OOC's are derived and techniques for constructing them are described.

Finally, OOC's with unequal auto- and cross-correlation constraints may be viewed as constant-weight unequal error protection (UEP) codes; therefore we interpret the bounds and constructions in that context and compare them with previous work on UEP codes.

2.2 Background and Motivation

In this section we briefly review previous work on optical orthogonal codes and indicate why the problem considered in this chapter is important.

2.2.1 Definitions and Past Work

What follows is the definition of an OOC given by Salehi *et. al.* [8].

Definition: An $(n, w, \lambda_a, \lambda_c)$ optical orthogonal code $\mathcal{C}$ is a collection of binary n-tuples, each of Hamming weight w, such that the following two properties hold:

- (Auto-correlation) For any $\mathbf{x} = [x_0, \ldots, x_{n-1}] \in \mathcal{C}$ and any integer τ, $0 < \tau < n$,

$$\sum_{t=0}^{n-1} x_t x_{t\oplus\tau} \leq \lambda_a.$$

- (Cross-correlation) For any $\mathbf{x} = [x_0, \ldots, x_{n-1}] \in \mathcal{C}$ and any $\mathbf{y} = [y_0, \ldots, y_{n-1}] \in \mathcal{C}$ such that $\mathbf{x} \neq \mathbf{y}$ and any integer τ,

$$\sum_{t=0}^{n-1} x_t y_{t\oplus\tau} \leq \lambda_c.$$

Note: OOC's were defined in terms of periodic correlation; thus the addition in the subscripts above – denoted "$\oplus$" – is all modulo-n.

The definition of an OOC can be recast in terms of Hamming distance; doing so makes clearer the parallels between OOC's and constant weight error correcting codes.

Notation: Given a binary n-tuple $\mathbf{x} = [x_0, x_1, \ldots, x_{n-1}]$, let $D^i\mathbf{x}$ $(0 \leq i \leq n-1)$ denote the binary n-tuple obtained by performing i right-cyclic shifts on $\mathbf{x}$.

Alternate Definition: An $(n, w, \lambda_a, \lambda_c)$ optical orthogonal code $\mathcal{C}$ is a collection of binary n-tuples, each of Hamming weight w, such that the following two properties hold:

- (Auto-correlation) For any $\mathbf{x} = [x_0, \ldots, x_{n-1}] \in \mathcal{C}$ we have $d^a_{\min}(\mathbf{x}) \geq 2w - 2\lambda_a$, where $d^a_{\min}(\mathbf{x}) \triangleq \min\{d_H(\mathbf{x}, D^\tau\mathbf{x}) : \tau = 1, 2, \ldots, n-1\}$.
- (Cross-correlation) For any $\mathbf{x} = [x_0, \ldots, x_{n-1}] \in \mathcal{C}$ and any $\mathbf{y} = [y_0, \ldots, y_{n-1}] \in \mathcal{C}$, we have $d^c_{\min}(\mathbf{x}, \mathbf{y}) \geq 2w - 2\lambda_c$, where $d^c_{\min}(\mathbf{x}, \mathbf{y}) \triangleq \min\{d_H(\mathbf{x}, D^\tau\mathbf{y}) : \tau = 0, 1, \ldots, n-1\}$.

Consider the partition of binary n-tuples into "clouds", where every cloud consists of cyclic shifts of the same n-tuple. Then constructing an OOC consists of picking at most one n-tuple from every cloud under two constraints; the first constraint specifies the minimum Hamming distance *within* a cloud, while the second specifies the minimum Hamming distance *between* clouds. Thus if the two constraints are equal – i.e., if $\lambda_a = \lambda_c = \lambda$ – then an OOC, taken together with all of the cyclic shifts of each OOC codeword, represents a constant-weight cyclic error correcting code with minimum distance $2w - 2\lambda$.

The use of OOC's for multiple access is described in [6][7]. In an optical network we assume that time is broken into discrete slots; within each slot each user can either transmit an optical pulse (i.e. - a physical "1") or not transmit an optical pulse (i.e. - a physical "0"). If we assume incoherent processing – i.e., the phase of the signal is not available – then multiple pulses transmitted simultaneously by different users add.

Each user in the network is assigned a codeword from an optical orthogonal code. The codeword assigned to a user is that user's "signature sequence", and when the user wishes to convey a logical "1" he transmits the corresponding sequence of pulses and pauses; when the user wishes to convey a logical "0" he transmits nothing for n slots. At the receiver each user computes the correlation of the received sequence with that user's signature sequence; because of the low auto- and cross-correlation properties the correlation typically stays low until a

logical "1" is indicated by a correlation of w. In this way each user can recover his own logical sequence.

In [6][7] Salehi introduced optical orthogonal codes and computed the error probability assuming a channel where the only "noise" is interference from other users. In [8] Chung, Salehi, and Wei described constructions of OOC's for the case $\lambda_a = \lambda_c = 1$ and derived bounds on the cardinality on an (n, w, λ, λ) code. In [9] Chung and Kumar described a construction technique for the case $\lambda_a = \lambda_c = 2$ and derived new upper bounds on the cardinality of an optimal OOC – again for the case $\lambda_a = \lambda_c = \lambda$.

2.2.2 Why Consider $\lambda_a \neq \lambda_c$?

The two correlation constraints serve two purposes.

- The auto-correlation constraint guarantees that each signature sequence is unlike cyclic shifts of itself. This property is used to enable the receiver to obtain *synchronization* – that is, to find the beginning of its message and subsequently locate the codeword boundaries.
- The cross-correlation constraint guarantees that each signature sequence is unlike cyclic shifts of the *other* signature sequences. This property is used to enable the receiver to estimate its message in the presence of interference from other users. Thus the cross-correlation constraint aids both synchronization in the presence of multiple users and permits each receiver to "track" its message after synchronization is achieved.

Thus the auto-correlation constraint contributes only to synchronization, while the cross-correlation constraint affects both synchronization and operation.

A reasonable "figure of merit" for a code is the number of interfering users necessary to cause the code to fail. For instance, assume synchronization has

been achieved; then the only errors the i^{th} receiver can make in estimating its logical sequence are $0 \rightarrow 1$ errors, and they can only occur when enough *other* users interfere to make the correlation at the i^{th} receiver exceed w. Since each of those other users can contribute at most λ_c to the correlation, the performance "figure of merit" is w/λ_c. In a similar vein, the synchronization "figure of merit" is $(w - \lambda_a)/\lambda_c$ for multiple-access synchronization and $w - \lambda_a$ for single-user synchronization.

Taking these as our performance criteria, we see why "asymmetric" OOC's – i.e., codes with $\lambda_a \neq \lambda_c$ – might be preferable to "symmetric" codes. If we compare (for instance) an $(n, w+m, \lambda+m, \lambda)$ OOC with either an (n, w, λ, λ) code or an $(n, w+m, \lambda+m, \lambda+m)$ code, we see the asymmetric code is more robust.

So the performance of an $(n, w+m, \lambda+m, \lambda)$ OOC will be at least as good as comparable "symmetric" OOC's. However, we will see in this chapter that the cardinality of the $(n, w+m, \lambda+m, \lambda)$ code can actually exceed that of the less powerful codes – thus more users can be provided even better performance. Clearly, this motivates the study of such OOC's.

2.3 Some New Bounds on Optical Orthogonal Codes

Define $\Phi(n, w, \lambda_a, \lambda_c)$ to be the cardinality of an optimal optical orthogonal code with the given parameters – i.e.,

$$\Phi(n, w, \lambda_a, \lambda_c) \triangleq \max\{|\, \mathcal{C} \,|: \mathcal{C} \text{ is an } (n, w, \lambda_a, \lambda_c) \text{ OOC}\}.$$

In this section we derive some new bounds on $\Phi(n, w, \lambda_a, \lambda_c)$. Before this can

be done, however, we need to set up the notation and derive some preliminary results.

Definition: Let $\mathbf{x} = [x_0, x_1, \cdots, x_{n-1}]$ be a binary n-tuple of weight w; assume $x_{j_0} = x_{j_1} = \cdots = x_{j_{w-1}} = 1$. The *adjacent relative delay vector* associated with $\mathbf{x}$ is denoted $\mathbf{t_x} = [t_0, t_1, \cdots, t_{w-1}]$ and is defined by

$$t_i = \begin{cases} j_{i+1} - j_i, & \text{for } i = 0, 1, \cdots, w-2. \\ n + j_0 - j_{w-1}, & \text{for } i = w-1. \end{cases}$$

More generally, the *relative delay* between two 1's in a binary n-tuple is the number (modulo n) of cyclic shifts required to "line up" the two 1's; $\mathbf{t_x}$ consists of the relative delay between all adjacent 1's in $\mathbf{x}$.

Notation: Let $\mathbf{x}$ be a binary n-tuple of weight w and let $\mathbf{t_x} = [t_0, t_1, \ldots, t_{w-1}]$ be its adjacent relative delay vector. Let $R_{\mathbf{x}} = [r_{\mathbf{x}}(i,j)]$ denote the $(w-1) \times w$ array of integers whose $(i,j)^{th}$ element is given by

$$r_{\mathbf{x}}(i,j) = \sum_{k=0}^{i} t_{j \uplus k}.$$

(Note: The subscript addition above and in the definition of $M_{\mathbf{x},\lambda}$ below is all modulo w – denoted "$\uplus$".)

More Notation: For any $\mathbf{x} \in \{0,1\}^n$ and any integer λ $(1 \leq \lambda \leq w-1)$ let $M_{\mathbf{x},\lambda}$ be the set of integer λ-tuples given by

$$M_{\mathbf{x},\lambda} \triangleq \{[\sum_{k_0=0}^{i_0} t_{j \uplus k_0}, \sum_{k_1=i_0+1}^{i_1} t_{j \uplus k_1}, \sum_{k_2=i_1+1}^{i_2} t_{j \uplus k_2}, \cdots, \sum_{k_{\lambda-1}=i_{\lambda-2}+1}^{i_{\lambda-1}} t_{j \uplus k_{\lambda-1}}] : \\ 0 \leq i_0 < i_1 < \cdots < i_{\lambda-1} \leq w-2, j = 0, 1, \cdots, w-1\},$$

where $\mathbf{t_x} = [t_0, t_1, \ldots, t_{x-1}]$

There are at most $w\binom{w-1}{\lambda}$ vectors in $M_{\mathbf{x},\lambda}$; this is because there are $\binom{w-1}{\lambda}$ ways to pick the i_ℓ's and w ways to pick the j's. If every such selection yields a different vector then $|M_{\mathbf{x},\lambda}| = w\binom{w-1}{\lambda}$; otherwise $|M_{\mathbf{x},\lambda}| < w\binom{w-1}{\lambda}$.

Example: Let $\mathbf{x} = [1001100010000]$. Then

$$t_{\mathbf{x}} = [t_0, t_1, t_2, t_3] = [3, 1, 4, 5].$$

Furthermore,

$$\begin{aligned} R_{\mathbf{x}} &= \begin{pmatrix} t_0 & t_1 & t_2 & t_3 \\ t_0+t_1 & t_1+t_2 & t_2+t_3 & t_3+t_0 \\ t_0+t_1+t_2 & t_1+t_2+t_3 & t_2+t_3+t_0 & t_3+t_0+t_1 \end{pmatrix} \\ &= \begin{pmatrix} 3 & 1 & 4 & 5 \\ 4 & 5 & 9 & 8 \\ 8 & 10 & 12 & 9 \end{pmatrix} \end{aligned}$$

and

$$\begin{aligned} M_{\mathbf{x},2} &= \{[t_0,t_1],[t_1,t_2],[t_2,t_3],[t_3,t_0], \\ &\quad [t_0+t_1,t_2],[t_1+t_2,t_3],[t_2+t_3,t_0],[t_3+t_0,t_1], \\ &\quad [t_0,t_1+t_2],[t_1,t_2+t_3],[t_2,t_3+t_0],[t_3,t_0+t_1]\}. \\ &= \{[3,1],[1,4],[4,5],[5,3], \\ &\quad [4,4],[5,5],[9,3],[8,1], \\ &\quad [3,5],[1,9],[4,8],[5,4]\}. \end{aligned}$$

The significance of $R_{\mathbf{x}}$ and $M_{\mathbf{x},\lambda}$ to OOC's is given in the following three lemmas.

Lemma 2.1: Let $\mathbf{x} = [x_0, x_1, \cdots, x_{n-1}]$ be a binary n-tuple. Then the inequality

$$\sum_{t=0}^{n-1} x_t x_{t\oplus\tau} \leq \lambda$$

holds for all $1 \leq \tau \leq n-1$ if and only if no component in $R_{\mathbf{x}}$ is repeated more than λ times.

Proof: The elements of $R_{\mathbf{x}}$ indicate the relative delay between every pair of 1's in $\mathbf{x}$. Therefore, $R_{\mathbf{x}}$ contains $\lambda+1$ repeated elements if and only there exist two sequences $\{i_0, i_1, \ldots, i_\lambda\}$ and $\{i'_0, i'_1, \ldots, i'_\lambda\}$ such that for all $j = 0, 1, \ldots, \lambda$

$$x_{i_j} = x_{i'_j} = 1$$

and

$$i_j - i'_j = \tau^* \neq 0.$$

But this is true if and only if $\sum_{t=0}^{n-1} x_t x_{t+\tau^*} \geq \lambda + 1$. QED.

Lemma 2.2: Let $\mathbf{x} = [x_0, x_1, \ldots, x_{n-1}]$ and $\mathbf{y} = [y_0, y_1, \ldots, y_{n-1}]$ be binary n-tuples. Then the inequality

$$\sum_{t=0}^{n-1} x_t y_{t\oplus\tau} \leq \lambda$$

holds for all $0 \leq \tau \leq n-1$ if and only if $M_{\mathbf{x},\lambda}$ and $M_{\mathbf{y},\lambda}$ are disjoint.

Proof: $M_{\mathbf{x},\lambda}$ is a collection of integer λ-tuples. A vector $\mathbf{m} = [a_0, a_1, \ldots, a_{\lambda-1}]$ is in $M_{\mathbf{x},\lambda}$ if and only if there exists a sequence of $\lambda+1$ distinct integers – call them $i_0, i_1, \ldots, i_\lambda$ – such that

$$x_{i_j} = 1 \quad \text{for } j = 0, 1, \ldots, \lambda$$

and

$$i_{j+1} - i_j = a_j \quad \text{for } j = 0, 1, \ldots, \lambda - 1.$$

Therefore, $M_{\mathbf{x},\lambda} \cap M_{\mathbf{y},\lambda} = \emptyset$ if and only if it's impossible to "line up" $\lambda+1$ binary 1's in $\mathbf{x}$ with $\lambda+1$ binary 1's in $\mathbf{y}$ with cyclic shifts – i.e., if and only if $\sum_{t=0}^{n-1} x_t y_{t\oplus\tau} \leq \lambda$. QED.

Lemma 2.3: Let $\mathbf{x} = [x_0, x_1, \ldots, x_{n-1}]$ be a binary n-tuple. Then the inequality

$$\sum_{t=0}^{n-1} x_t x_{t\oplus\tau} \leq \lambda$$

holds for all $\tau = 1, 2 \cdots, n-1$ if and only if $|M_{\mathbf{x},\lambda}| = w\binom{w-1}{\lambda}$ – i.e., if and only if the vectors defining $M_{\mathbf{x},\lambda}$ are all distinct.

Proof: Similar to the proof of Lemma 2.2. The presence of a λ-tuple in $M_{\mathbf{x},\lambda}$ corresponds to $\lambda + 1$ non-zero components of $\mathbf{x}$ such that the relative delays between the non-zero components are given by the λ-tuple. If $|M_{\mathbf{x},\lambda}| < w\binom{w-1}{\lambda}$ then there are two different sets of $\lambda + 1$ non-zero components with the same relative delays between them; thus we can "line up" the $\lambda + 1$ binary ones and obtain an auto-correlation of at least $\lambda + 1$. Conversely, if $|M_{\mathbf{x},\lambda}| = w\binom{w-1}{\lambda}$ then every set of $\lambda + 1$ non-zero components of $\mathbf{x}$ have a different relative delay structure, so it's impossible to obtain an auto-correlation of $\lambda+1$ or more. QED.

2.3.1 An Upper Bound

In this section we will use the characterizations developed above to provide an upper bound on $\Phi(n, w, \lambda_a, \lambda_c)$.

First consider the case $\lambda_a = \lambda_c = \lambda$; the bound we derive is identical to one in [8] derived from the Johnson bound for constant weight error correcting codes; it is re-derived here to illustrate the approach that will be taken in the proof of the new bounds.

Theorem 2.1: [Johnson Bound] The following inequality holds:

$$\Phi(n, w, \lambda, \lambda) \leq \frac{(n-1)(n-2)\ldots(n-\lambda)}{w(w-1)\ldots(w-\lambda)}.$$

Proof: Let $\mathcal{C}$ be an optimal (n, w, λ, λ) OOC – i.e., $|\mathcal{C}| = \Phi(n, w, \lambda, \lambda)$. From

Lemma 2.3 we know that for every $\mathbf{x} \in \mathcal{C}$ the set $M_{\mathbf{x},\lambda}$ consists of $w\binom{w-1}{\lambda}$ distinct integer λ-tuples. Furthermore, from Lemma 2.2 we know that for $\mathbf{x}, \mathbf{y} \in \mathcal{C}$, $\mathbf{x} \neq \mathbf{y}$, the sets $M_{\mathbf{x},\lambda}$ and $M_{\mathbf{y},\lambda}$ are disjoint. Therefore the union of $M_{\mathbf{x},\lambda}$ as $\mathbf{x}$ varies over all $\mathbf{x} \in \mathcal{C}$ consists of $\Phi(n, w, \lambda, \lambda) \cdot w\binom{w-1}{\lambda}$ distinct integer λ-tuples. However, by definition if $[a_0, a_1, \ldots, a_{\lambda-1}] \in M_{\mathbf{x},\lambda}$ then $a_0 + a_1 + \ldots + a_{\lambda-1} \leq n-1$. But the number of ways to select λ positive a_i's that sum to no more than $n-1$ is just the number of compositions of n with $\lambda + 1$ positive parts – and that is equal to $\binom{n-1}{\lambda}$. We have thus shown that

$$\Phi(n, w, \lambda, \lambda) \cdot w\binom{w-1}{\lambda} \leq \binom{n-1}{\lambda},$$

which was to be proven. QED.

Our next goal is to bound $\Phi(n, w, \lambda_a, \lambda_c)$ for $\lambda_a > \lambda_c$. To do so we first need a preliminary lemma.

Lemma 2.4: Let $\mathbf{x} \in \mathcal{C}$, where $\mathcal{C}$ is an $(n, w, \lambda + m, \lambda)$ optical orthogonal code. (Assume $m \geq 0$ is an integer.) Then if $w > 2\lambda - 2$,

$$|M_{\mathbf{x},\lambda}| \geq \frac{w\binom{w-1}{\lambda}}{m+1}.$$

If $w \leq 2\lambda - 2$,

$$|M_{\mathbf{x},\lambda}| \geq \frac{w\binom{w-1}{\lambda}}{\lambda+m}.$$

Proof: See Appendix A.

Theorem 2.2: Let m be a non-negative integer. Then if $w > 2\lambda - 2$,

$$\Phi(n, w, \lambda + m, \lambda) \leq \frac{(n-1)(n-2)\ldots(n-\lambda)(m+1)}{w(w-1)(w-2)\ldots(w-\lambda)}.$$

If $w \leq 2\lambda - 2$,

$$\Phi(n, w, \lambda + m, \lambda) \leq \frac{(n-1)(n-2)\ldots(n-\lambda)(\lambda+m)}{w(w-1)(w-2)\ldots(w-\lambda)}.$$

Proof: Let $\mathcal{C}$ be an $(n, w, \lambda + m, \lambda)$ OOC such that $|\mathcal{C}| = \Phi(n, w, \lambda + m, \lambda)$. By Lemma 2.4, for any $\mathbf{x} \in \mathcal{C}$ and if $w > 2\lambda - 2$, $|M_{\mathbf{x},\lambda}| \geq w\binom{w-1}{\lambda}/(m+1)$. Furthermore, by Lemma 2.2 $M_{\mathbf{x},\lambda}$ and $M_{\mathbf{y},\lambda}$ are disjoint for $\mathbf{x}, \mathbf{y} \in \mathcal{C}$ and $\mathbf{x} \neq \mathbf{y}$; therefore $|M_{\mathbf{x},\lambda}|$ summed up over all $\mathbf{x} \in \mathcal{C}$ cannot exceed the total number of integer λ-tuples that are "allowable" as elements of $M_{\mathbf{x},\lambda}$ – a number shown in the proof of Theorem 2.1 to be $\binom{n-1}{\lambda}$. So:

$$\begin{aligned}
\binom{n-1}{\lambda} &\geq \sum_{\mathbf{x}\in\mathcal{C}} |M_{\mathbf{x},\lambda}| \\
&\geq \Phi(n, w, \lambda + m, \lambda) \cdot \min\{|M_{\mathbf{x},\lambda}| : \mathbf{x} \in \mathcal{C}\} \\
&\geq \Phi(n, w, \lambda + m, \lambda) \cdot \frac{w\binom{w-1}{\lambda}}{m+1}
\end{aligned}$$

which was to be proven. The proof for $w \leq 2\lambda - 2$ immediately follows the proof above. QED.

Examining Theorem 2.2, we find (for instance) that the upper bound on $\Phi(n, w, \lambda, 1)$ is λ times greater than the analogous bound on $\Phi(n, w, 1, 1)$. It should also be noted that a trivial upper bound on $\Phi(n, w, \lambda + m, \lambda)$ is given by any upper bound on $\Phi(n, w, \lambda + m, \lambda + m)$. For "typical" OOC values – i.e., $n \gg w$ – an upper bound derived this way will be much looser than the bound in Theorem 2.2. For instance, considering $(n, w, 2, 1)$ OOC's, the bound in Theorem 2.2 is tighter than the Johnson bound for $(n, w, 2, 2)$ codes provided $n - 2 > 2(w - 2)$.

2.3.2 Lower Bounds

In [8][30] a lower bound on $\Phi(n, w, \lambda, \lambda)$ was derived for odd prime n. Subsequently, Victor Wei [31] derived an alternate lower bound – again for odd prime n. In what follows we use the general approach of Wei [31] to bound $\Phi(n, w, \lambda_a, \lambda_c)$ for $\lambda_a \neq \lambda_c$ and any n.

Theorem 2.3:

$$\Phi(n, w, \lambda_a, \lambda_c) \geq \frac{\binom{n}{w} - A}{B},$$

where

$$A = \sum_{\substack{\delta=1 \\ \delta | n}}^{\lfloor n/2 \rfloor} \sum_{N=0}^{\lfloor w\delta/n \rfloor} \left[\triangle(N - w\delta/n) \binom{\delta}{N} + \sum_{c=\lceil w\delta/n \rceil}^{\substack{w-\lambda_a \\ +N-1}} \binom{c}{N} \binom{w - Nn/\delta - 1}{c - N - 1} \right.$$

$$\left. \binom{\delta}{N} \binom{n - Nn/\delta}{c - N} \right] + |\{\delta : 1 \leq \delta \leq \lfloor n/2 \rfloor, \delta \not| n\}| \sum_{c=1}^{w-\lambda_a-1} \binom{w-1}{c-1} \binom{n}{c},$$

$$\triangle(x) = \begin{cases} 1, & \text{if } x = 0; \\ 0, & \text{otherwise,} \end{cases}$$

and

$$B = n \sum_{i > \lambda_c} \binom{n-w}{w-i} \binom{w}{i}.$$

Proof: As in [8][30], the proof consists of demonstrating that A is an upper bound on the number of binary n-tuples that violate the auto-correlation constraint and B is an upper bound on the number of binary n-tuples that violate the cross-correlation constraint for a given binary n-tuple $\mathbf{x}$. The result follows from an application of the greedy algorithm. The validity of B as an upper bound was demonstrated in [8][30]. Thus the proof consists of demonstrating that there are at most A binary n-tuples that violate the auto-correlation constraint. A proof of this is given in Appendix B.

2.3.3 Asymptotic Bounds

In this section we examine how the cardinality of an optimal $(n, w, \lambda_a, \lambda_c)$ optical orthogonal code behaves for large blocklength. The goal is to see how quickly the parameters w, λ_a, and λ_c should grow with blocklength n in an $(n, w, \lambda_a, \lambda_c)$ OOC.

Lemma 2.5: Let λ_c be a positive integer, and let p and q be a non-negative constants such that $p > (\lambda_c + q)/(\lambda_c + 1)$. Then

$$\lim_{n\to\infty} \Phi(n, \lceil \alpha n^p \rceil, \lceil \beta n^q \rceil, \lambda_c) = 0,$$

for any positive real α and β.

Proof: From Theorem 2.2,

$$\begin{aligned}\Phi(n, \lceil \alpha n^p \rceil, \lceil \beta n^q \rceil, \lambda_c) &\leq \frac{(n-1)\ldots(n-\lambda_c)(\beta n^q + 1 - \lambda_c + 1)}{\alpha n^p(\alpha n^p - 1)\ldots(\alpha n^p - \lambda_c)} \\ &= n^{\lambda_c + q - p(\lambda_c+1)}(\beta - (\lambda_c - 2)/n^q)\alpha^{-(\lambda_c+1)} \prod_{i=1}^{\lambda_c} \frac{1-(i/n)}{1-(i/\alpha n^p)}.\end{aligned}$$

Since by assumption $\lambda_c + q - p(\lambda_c + 1) < 0$ we have the desired result. QED

Lemma 2.6: Let λ_a and λ_c be positive integers, and let p be a constant such that $p < \min\{\lambda_a/(2\lambda_a + 3), \lambda_c/(2\lambda_c + 3)\}$. Then

$$\lim_{n\to\infty} \Phi(n, \lfloor \alpha n^p \rfloor, \lambda_a, \lambda_c) = \infty,$$

for any positive real α.

Proof: See Appendix C.

Considering the case $\lambda_a = \lambda_c = 1$, Lemma 2.5 tell us that the weight should grow no faster than $\sqrt{n}$, and Lemma 2.6 suggests it should grow no faster than $n^{1/5}$.

Lemma 2.7: Let p, q, and r be positive constants between zero and one such that $p < (1/2)$, then

$$\lim_{n\to\infty} \Phi(n, \lfloor \alpha n^p \rfloor, \lfloor \beta n^q \rfloor, \lfloor \gamma n^r \rfloor) = \infty,$$

for any positive real α,β and γ.

Proof: See Appendix D.

2.4 OOC's as Constant-Weight Unequal Error Protection Codes

In this section we briefly describe the connection between $(n, w, \lambda_a, \lambda_c)$ optical orthogonal codes and unequal error protection (UEP) code with two levels of protection.

An unequal error protection code is an error control code with a "twist"; the code is designed so that different digits in a message have varying levels of reliability. This may be convenient in applications where the position of a digit in a message determines its importance. The archetypical example of this is a message containing a bank balance; if the balance is \$1376.62 it's much more important that the "1" be uncorrupted than that the "2" be error-free.

An encoder for an (n, k) binary error control code is a mapping $f : \{0,1\}^k \to \{0,1\}^n$. The message $\mathbf{x} \in \{0,1\}^n$ is represented by the codeword $f(\mathbf{x}) \in \{0,1\}^n$. If $\min\{d(f(\mathbf{x}), f(\mathbf{y})) : \mathbf{x}, \mathbf{y} \in \{0,1\}^k, \mathbf{x} \neq \mathbf{y}\} \geq 2t+1$ then we say the code is t-error correcting. (Here, $d(\mathbf{c}_1, \mathbf{c}_2)$ is the Hamming distance between the n-tuples $\mathbf{c}_1$ and $\mathbf{c}_2$.)

Definition: Given an encoder $f : \{0,1\}^k \to \{0,1\}^n$, the *separation vector* associated with the encoder is an integer k-tuple $\mathbf{s} = [s_0, s_1, \ldots, s_{k-1}]$ defined by

$$s_i = \min\{d(f(\mathbf{x}), f(\mathbf{y})) : \mathbf{x}, \mathbf{y} \in \{0,1\}^k \text{ and } x_i \neq y_i\}.$$

Note that the separation vector is associated with the *encoder* rather than the code – i.e., the image of the encoder. It's possible that a code may have multiple separation vectors associated with it – corresponding to different encoders for the same code.

Let $f(\cdot)$ be an encoder with separation vector $\mathbf{s} = [s_0, s_1, \ldots, s_{k-1}]$. Suppose a message k-tuple $\mathbf{x}$ is used to select a codeword $f(\mathbf{x}) \in \mathcal{C}$ which is then transmitted over a noisy channel. It's obvious that minimum-distance decoding will correctly recover the i^{th} bit of the message provided no more than $t_i = \lfloor (s_i - 1)/2 \rfloor$ errors occur during transmission. A code with an encoder whose separation vector has the property that $t_i \neq t_j$ for some $i \neq j$ is called an unequal error protection (UEP) code.

There is a substantial body of literature on UEP codes. (See [17][18][19][32] for references.) However, there has been no investigation of *constant weight* UEP codes.

Throughout this section, we assume $\lambda_a \geq \lambda_c$ – the inter-cloud Hamming distance $d^c_{\min}$ is no less than the intra-cloud Hamming distance $d^a_{\min}$ for an $(n, w, \lambda_a, \lambda_c)$ OOC.

Then an $(n, w, \lambda_a, \lambda_c)$ OOC with $\lambda_a > \lambda_c$ can be used to construct a constant weight UEP code. Suppose you have such an OOC with cardinality M. Now consider the error control code consisting of all the n-tuples of the OOC and all their cyclic shifts. The resulting code has nM codewords, all of weight w. Furthermore, such a code consists of M "clouds" of codewords, where two codewords belong to the same cloud if and only if they're cyclic shifts of one another. The distance between any two codewords within the same cloud is at least $2(w - \lambda_a)$; the distance between any two codewords from two different clouds is at least $2(w - \lambda_c)$

So, consider the following encoder. Take $k_2 = \lfloor \log_2 M \rfloor$ message bits and use them to pick a cloud; then take $k_1 = \lfloor \log_2 n \rfloor$ message bits and use them to pick an n-tuple from within the chosen cloud. Any two messages that differ in the first k_2 bits will have codewords that differ in at least $2(w - \lambda_c)$ positions; any two messages that differ in the last k_1 bits will have codewords that differ in at least $2(w - \lambda_a)$ positions. Therefore we have described an encoder for a $(k_1 + k_2, n)$ code with separation vector

$$\mathbf{s} = (\underbrace{2(w-\lambda_c), \ldots, 2(w-\lambda_c)}_{k_2}, \underbrace{2(w-\lambda_a), \ldots, 2(w-\lambda_a)}_{k_1})$$

This observation means that our lower bound for OOC's may be interpreted as an existence result for constant-weight UEP codes with two levels of protection.

Notation: Let $M(n, w, \lambda_a, \lambda_c)$ denote the lower bound on $\Phi(n, w, \lambda_a, \lambda_c)$ derived in Theorem 2.3 – i.e., $M(n, w, \lambda_a, \lambda_c) = (\binom{n}{w} - A)/B$, where A and B are given in Theorem 2.3.

Theorem 2.4: Let α and β be positive, even integers. Then there exists a weight-w $(n, k_1 + k_2)$ error control code with separation vector

$$\mathbf{s} = [\underbrace{\alpha, \alpha, \ldots, \alpha}_{k_2}, \underbrace{\beta, \beta, \ldots, \beta}_{k_1}],$$

where

$$k_2 = \lfloor \log_2 M(n, w, w - (\beta/2), w - (\alpha/2)) \rfloor \quad \text{and} \quad k_1 = \lfloor \log_2 n \rfloor.$$

It is interesting to compare the bound in Theorem 2.4 with existing bounds for non-constant weight UEP codes. Bassalgyo *et.al.* [33] used Gilbert-Varshamov style reasoning to derive the following result: There exists a UEP code

with blocklength n and rate $R = R_1 + R_2$ with a separation vector consisting of $k_1 = R_1 n$ entries of $d_1 = 2t_1 + 1$ and $k_2 = R_2 n$ entries of $d_2 = 2t_2 + 1$ $(t_1 \leq t_2)$ provided the following inequality hold:

$$R_2 \geq 1 - R_1 - \frac{t_1 \log_2 n}{n} - \frac{3t_2 \log_2(1/R_1) + t_1 + 2}{n}.$$

If we let $k_1 = \lfloor \log_2 n \rfloor$, this bound may be directly compared to the bound in Theorem 2.4. Figure 2.1 shows two different lower bounds on the number of information bits that can be protected against two errors while simultaneously protecting $k_1 = \lfloor \log_2(n) \rfloor$ information bits against single errors. One bound is from Bassalgyo *et.al* and the other comes from Theorem 2.4 when we set $w = \lfloor n/2 \rfloor$. The lower bound in Theorem 2.4 is a clear improvement over the bound in [33].

This may at first seem surprising; the bound from Theorem 2.4 is one on *constant-weight* UEP codes, while the Bassalgyo bound has no such constraint. However, there are similar results with regard to "regular" t-error correcting codes. For instance, the traditional Gilbert-Varshamov bound for codes with blocklength $n = 8$ and minimum distance four is given by $\lceil 2^8/(1+\binom{8}{2}+\binom{8}{3}) \rceil = 4$. By comparison, if we use a Gilbert-Varshamov-type technique to bound the cardinality of the optimal weight-four code with blocklength $n = 8$ and minimum distance four we obtain $\lceil \binom{8}{4}/(1+4^2) \rceil = 5$.

The phenomenon can be explained by the proof of the Gilbert-Varshamov bound. The bound states that, when drawing codewords from a set $\mathcal{X}$ with the property that every sphere of radius $d-1$ contains V_{d-1} elements, it is always possible to find a code with minimum distance d and cardinality at least $|\mathcal{X}|/V_{d-1}$. In going to a constant weight code we are taking a subset of $\mathcal{X}$ in such a way that the ratio of the cardinality of the set to the sphere "volume" increases.

2.5 Constructions of OOC's

2.5.1 An (n, w, 2, 1) OOC (Construction 1)

As indicated in Section 2.2, we now demonstrate a technique for constructing an $(n, w, 2, 1)$ OOC. The method is a variation on the techniques proposed by Wilson and Hanani[34][35] to construct $(n, w, 1, 1)$ codes. We begin by considering specific values of w ($w = 5$ and $w = 6$) and then generalize the technique.

2.5.1.1 Some Specific Examples

An $(n, 5, 2, 1)$ OOC: Let n be a prime number such that $n = 12t + 1$ for an integer t. Let α be a primitive element of the field GF(n) such that $\alpha^q = \alpha^{3t} - 1$ and $\alpha^r = 2$, where q and r are integers satisfying one of the following two conditions:

- $q \equiv 1 \pmod 3$ and $r \equiv 2 \pmod 3$;
- $q \equiv 2 \pmod 3$ and $r \equiv 1 \pmod 3$.

Then we can construct an $(n, 5, 2, 1)$ OOC $\mathcal{C}$ with cardinality $|\mathcal{C}| = t$ as follows. The i^{th} codeword $\mathbf{x}_i$ contains a "1" in positions 0, α^{3i}, α^{3t+3i}, α^{6t+3i}, and α^{9t+3i} and a "0" everywhere else. This holds for $i = 0, 1, \ldots, t-1$. (Note: We say that the code consists of the "blocks" $\{[0, \alpha^{3i}, \alpha^{3t+3i}, \alpha^{6t+3i}, \alpha^{9t+3i}] : i = 0, 1, \ldots, t-1\}$.)

To see that this construction yields an $(n, 5, 2, 1)$ code let $R_{\mathbf{x}_0}$ denote the array consisting of all the relative delays between pairs of 1's in $\mathbf{x}_0$. Keeping in mind that $x^{12t} = 1$ and $x^{6t} = -1$, simple algebra reveals that

$$R_{\mathbf{x}_0} = \begin{pmatrix} 1 & \alpha^{3t}-1 & \alpha^{3t}(\alpha^{3t}-1) & \alpha^{6t}(\alpha^{3t}-1) & \alpha^{3t} \\ \alpha^{3t} & 2\alpha^{6t} & 2\alpha^{9t} & 1 & \alpha^{9t}(\alpha^{3t}-1) \\ \alpha^{6t} & \alpha^{3t}(\alpha^{3t}-1) & \alpha^{9t} & 2 & 2\alpha^{3t} \\ \alpha^{9t} & \alpha^{6t} & \alpha^{6t}(\alpha^{3t}-1) & \alpha^{9t}(\alpha^{3t}-1) & \alpha^{3t}-1 \end{pmatrix}.$$

Every component of $R_{\mathbf{x}}$ is of the form $\beta\alpha^{3jt}$ where $\beta \in \{1, 2, \alpha^{3t} - 1\}$ and $j \in \{0, 1, 2, 3\}$. Therefore as long as the base-α logarithms of 2 and $\alpha^{3t} - 1$ are not equivalent to each other mod 3 and are not equivalent to zero mod 3,

$$\beta_1 \neq \beta_2 \text{ or } j_1 \neq j_2 \Rightarrow \beta_1\alpha^{3j_1t} \neq \beta_2\alpha^{3j_2t}.$$

And so no element of $R_{\mathbf{x}_0}$ is repeated more than twice; this implies the auto-correlation constraint for $\mathbf{x}_0$ is met. Furthermore, the matrix $R_{\mathbf{x}_i}$ is obtained by multiplying $R_{\mathbf{x}_0}$ by α^{3i}, so the auto-correlation constraint is met for all $\mathbf{x}_i$.

To check the cross-correlation constraint, we note as before that $M_{\mathbf{x},1}$ is a set whose "vectors" are just the 1-tuples from $R_{\mathbf{x}}$. Therefore as long as the components of $R_{\mathbf{x}_i}$ and the components of $R_{\mathbf{x}_j}$ form disjoint sets for $i \neq j$, we will have proven the cross-correlation constraint is met. But (as mentioned above) $R_{\mathbf{x}_i}$ is obtained by multiplying $R_{\mathbf{x}_0}$ by α^{3i}; thus the components of $R_{\mathbf{x}_i}$ are of the form $\beta\alpha^{3(jt+i)}$; as long as $0 \leq i \leq t-1$ the components form disjoint sets and so the cross-correlation constraint is met.

Example: Let $n = 37$ and $t = 3$. Choose $\alpha = 2$ as the primitive element of GF(37) and so $2^{3t} - 1 = 30 = 2^{14}$ while $2 = 2^1$ - i.e., $q = 14$ and $r = 1$. Then the code consists of the blocks $\{[0, 1, 6, 31, 36], [0, 8, 11, 26, 29], [0, 10, 14, 23, 27]\}$ and so the three codewords are

$$\begin{aligned}
\mathbf{x}_0 &= [1100001000000000000000000000100001] \\
\mathbf{x}_1 &= [1000000010010000000000000010010000000] \\
\mathbf{x}_2 &= [1000000000100010000000010001000000000].
\end{aligned}$$

An $(n, 6, 2, 1)$ OOC: Let n be a prime number such that $n = 18t + 1$ for an integer t. Let α be a primitive element of the field GF(n) such that $\alpha^q = \alpha^{6t} - 1$

and $\alpha^r = 2$, where q and r are integers satisfying one of the following two conditions:

- $q \equiv 1 \pmod 3$ and $r \equiv 2 \pmod 3$;
- $q \equiv 2 \pmod 3$ and $r \equiv 1 \pmod 3$.

Then we can construct a $(n, 6, 2, 1)$ OOC $\mathcal{C}$ with cardinality $|\mathcal{C}| = t$ with the blocks $\{[\alpha^{3i}, \alpha^{3t+3i}, \alpha^{6t+3i}, \alpha^{9t+3i}, \alpha^{12t+3i}, \alpha^{15t+3i}] : i = 0, 1, \ldots, t-1\}$.

The proof that this construction yields an $(n, 6, 2, 1)$ code is analogous to the result for the $w = 5$ code and is omitted.

2.5.1.2 Generalizing the Examples

The techniques described in Section 2.5.1.1 are easily generalized to different values of w; the details are given below.

An (n, w, 2, 1) OOC for Even w: Let $w = 2m$ and choose n to be a prime number such that $n = w^2t/2 + 1$. Let α be a primitive element of GF(n) such that $\{\log_\alpha[\alpha^{kmt} - 1] : 1 \leq k \leq m\}$ are all distinct modulo m. Then the code consisting of the blocks

$$\{[\alpha^{mi}, \alpha^{m(i+t)}, \alpha^{m(i+2t)}, \ldots, \alpha^{m(i+(2m-1)t)}] : i = 0, 1, \ldots, t-1\}$$

is an $(n, w, 2, 1)$ OOC.

An (n, w, 2, 1) OOC for Odd w: Let $w = 2m + 1$ and choose n to be a prime number such that $n = (w^2 - 1)t/2 + 1$. Let α be a primitive element of GF(n) such that $\{\log_\alpha[\alpha^{k(m+1)t} - 1] : 1 \leq k \leq m\}$ are all distinct modulo $m + 1$ and non-zero modulo $m + 1$. Then the code consisting of the blocks

$$\{[0, \alpha^{(m+1)i}, \alpha^{(m+1)(i+t)}, \alpha^{(m+1)(i+2t)}, \ldots, \alpha^{(m+1)(i+(2m-1)t)}] : i = 0, 1, \ldots, t-1\}$$

is an $(n, w, 2, 1)$ OOC.

2.5.2 Construction of an (n, w, 1, 1) OOC

Bose used a balanced incomplete block design (BIBD) to design the first $(n, w, 1, 1)$ OOC for $w = 3, 4, 5$ in 1939[36]. Wilson generalized the results to arbitrary w in 1972[34]. Hanani also used BIBD but with different parameters to construct $(n, 6, 1, 1)$ OOC in 1961[35]. We have generalized Hanani's result to any $w = 4m + 2$ or $4m + 3$ (m is an integer). All four constructions for $(n, w, 1, 1)$ OOC's are optimal, since the cardinality t reaches the Johnson bound in Theorem 2.2.

First we restate the Wilson's construction on an $(n, w, 1, 1)$ OOC for completeness.

An (n, w, 1, 1) OOC for Odd w:[34] Let $w = 2m + 1$ and choose n to be a prime number such that $n = w(w-1)t + 1$. Let α be a primitive element of GF(n) such that $\{\log_\alpha[\alpha^{2mkt} - 1] : 1 \leq k \leq m\}$ are all distinct modulo m. Then the code consisting of the blocks

$$\{[\alpha^{mi}, \alpha^{mi+2mt}, \alpha^{mi+4mt}, \ldots, \alpha^{mi+4m^2t}] : i = 0, 1, \ldots, t-1\}$$

is an $(n, w, 1, 1)$ OOC.

An (n, w, 1, 1) OOC for Even w:[34] Let $w = 2m$ and choose n to be a prime number such that $n = w(w-1)t + 1$. Let α be a primitive element of GF(n) such that $\{\log_\alpha[\alpha^{2mkt} - 1] : 1 \leq k \leq m-1\}$ are all distinct modulo m and non-zero modulo m. Then the code consisting of the blocks

$$\{[0, \alpha^{mi}, \alpha^{mi+2mt}, \alpha^{mi+4mt}, \ldots, \alpha^{mi+4m(m-1)t}] : i = 0, 1, \ldots, t-1\}$$

is an $(n, w, 1, 1)$ OOC.

The new construction is as follows.

An $(\mathbf{n}, \mathbf{w}, \mathbf{1}, \mathbf{1})$ OOC for $\mathbf{w = 4m + 2}$: Let $w = 4m+2$, and $n = w(w-1)t+1$, n is prime, and if α is a primitive root of GF(n) such that $\alpha^{k(8m+2)t+y} - 1 = \alpha^{i_k}$, for $0 \leq k \leq m$, $\alpha^{k(8m+2)t} - \alpha^y = \alpha^{j_k}$, for $1 \leq k \leq m$, $\alpha^{k(8m+2)t} - 1 = \alpha^{r_k}$, for $1 \leq k \leq m$, $\alpha^y(\alpha^{k(8m+2)t} - 1) = \alpha^{s_k}$, for $1 \leq k \leq m$, y is an integer between 1 and $(8m + 2)t - 1$ and if $i_0, i_1, \cdots, i_m$, $j_1, \cdots, j_m$, $r_1, \cdots, r_m$, $s_1, \cdots, s_m$ are all distinct modulo $4m + 1$, then the blocks

$$(\alpha^{(4m+1)i}, \alpha^{y+(4m+1)i}, \alpha^{(8m+2)t+(4m+1)i}, \alpha^{(8m+2)t+y+(4m+1)i}, \cdots,$$
$$\alpha^{(8m+2)2mt+(4m+1)i}, \alpha^{(8m+2)2mt+y+(4m+1)i)}) : i = 0, 1, \cdots, t-1$$

are the codewords of $(n, w, 1, 1)$ OOC.

An $(\mathbf{n}, \mathbf{w}, \mathbf{1}, \mathbf{1})$ OOC for $\mathbf{w = 4m + 3}$: Let $w = 4m+3$, and $n = w(w-1)t+1$, n is prime, and if α is a primitive root of GF(n) such that $\alpha^{k(8m+6)t+y} - 1 = \alpha^{i_k}$, for $0 \leq k \leq m$, $\alpha^{k(8m+6)t} - \alpha^y = \alpha^{j_k}$, for $1 \leq k \leq m$, $\alpha^{k(8m+6)t} - 1 = \alpha^{r_k}$, for $1 \leq k \leq m$, $\alpha^y(\alpha^{k(8m+6)t} - 1) = \alpha^{s_k}$, for $1 \leq k \leq m$, y is an integer between 1 and $(8m + 6)t - 1$ and if $y, i_0, i_1, \cdots, i_m$, $j_1, \cdots, j_m$, $r_1, \cdots, r_m$, $s_1, \cdots, s_m$ are all distinct modulo $4m + 3$ and non-zero modulo $4m + 3$, then the blocks

$$(0, \alpha^{(4m+3)i}, \alpha^{y+(4m+3)i}, \alpha^{(8m+6)t+(4m+3)i}, \alpha^{(8m+6)t+y+(4m+3)i}, \cdots,$$
$$\alpha^{(8m+6)2mt+(4m+3)i}, \alpha^{(8m+6)2mt+y+(4m+3)i)}) : i = 0, 1, \cdots, t-1$$

are the codewords of $(n, w, 1, 1)$ OOC.

There is one point we want to mention. Wilson's construction appears more general than our new construction in that the weights are arbitrary; however, for some blocklengths n Wilson's construction fails to construct a code but our new construction succeeds. This can be seen in Table 2.3-2.5 when $w = 6, 7$.

2.5.3 An (n, w, 2, 1) OOC (Construction 2)

As indicated in Section 2.5.1.2, we use the the same technique to construct an $(n, w, 2, 1)$ OOC from the construction in Section 2.5.2 for an $(n, w, 1, 1)$ OOC.

An (n, w, 2, 1) OOC for w = 4m: Let $w = 4m$, and $n = w^2t/2 + 1$, n is prime, and if α is a primitive root of GF(n) such that $\alpha^{k4mt+y} - 1 = \alpha^{i_k}$, for $0 \leq k \leq m-1$, $\alpha^{k4mt} - \alpha^y = \alpha^{j_k}$, for $1 \leq k \leq m$, $\alpha^{k4mt} - 1 = \alpha^{r_k}$, for $1 \leq k \leq m$, $\alpha^y(\alpha^{k4mt} - 1) = \alpha^{s_k}$, for $1 \leq k \leq m$, y is an integer between 1 and $4mt - 1$ and if $i_0, i_1, \cdots, i_{m-1}$, $j_1, \cdots, j_m$, $r_1, \cdots, r_m$, $s_1, \cdots, s_m$ are all distinct modulo $4m$, then the blocks

$$(\alpha^{4mi}, \alpha^{y+4mi}, \alpha^{4mt+4mi}, \alpha^{4mt+y+4mi}, \cdots,$$
$$\alpha^{4m(2m-1)t+4mi}, \alpha^{4m(2m-1)t+y+4mi}) : i = 0, 1, \cdots, t-1$$

are the codewords of an $(n, w, 2, 1)$ OOC.

An (n, w, 2, 1) OOC for w = 4m + 1: Let $w = 4m+1$, and $n = (w^2-1)t/2+1$, n is prime, and if α is a primitive root of GF(n) such that $\alpha^{k(4m+2)t+y} - 1 = \alpha^{i_k}$, for $0 \leq k \leq m-1$, $\alpha^{k(4m+2)t} - \alpha^y = \alpha^{j_k}$, for $1 \leq k \leq m$, $\alpha^{k(4m+2)t} - 1 = \alpha^{r_k}$, for $1 \leq k \leq m$, $\alpha^y(\alpha^{k(4m+2)t} - 1) = \alpha^{s_k}$, for $1 \leq k \leq m$, y is an integer between 1 and $(4m+2)t - 1$ and if $y, i_0, i_1, \cdots, i_{m-1}$, $j_1, \cdots, j_m$, $r_1, \cdots, r_m$, $s_1, \cdots, s_m$ are all distinct modulo $4m+2$ and non-zero modulo $4m+2$, then the blocks

$$(0, \alpha^{(4m+2)i}, \alpha^{y+(4m+2)i}, \alpha^{(4m+2)t+(4m+2)i}, \alpha^{(4m+2)t+y+(4m+2)i}, \cdots,$$
$$\alpha^{(4m+2)(2m-1)t+(4m+2)i}, \alpha^{(4m+2)(2m-1)t+y+(4m+2)i}) : i = 0, 1, \cdots, t-1$$

are the codewords of an $(n, w, 2, 1)$ OOC.

The numbers of codewords in $(n, w, 2, 1)$ OOC's are listed in Table 2.1–2.2. The numbers of codewords in $(n, w, 1, 1)$ OOC's are listed in Table 2.3–2.5.

Table 2.1-2.2 shows the cardinality of the codes constructed using the above techniques for $w = 3, 4, 5, 6, 8, 9$ and moderate blocklength. It should be noted that in some cases the construction of an $(n, w, 2, 1)$ code yields more codewords than of be possible with an $(n, w-1, 1, 1)$ OOC – even though the performance of the $(n, w, 2, 1)$ would be at least as good. For instance, the cardinalities of the $(n, 6, 2, 1)$ codes in Table 2.1 exceed the upper bound on $\Phi(n, 5, 1, 1)$ derived in Section 2.3.1. This means we could support more users using such "asymmetric" codes and reinforces the idea that they're worth examining.

n	w	t	n	w	t	n	w	t	n	w	t
5	3	1	17	4	2	13	5	1	19	6	1
13	3	3	73	4	9	37	5	3	199	6	11
29	3	7	89	4	11	61	5	5	487	6	27
37	3	9	97	4	12	73	5	6	829	6	46
53	3	13	193	4	24	97	5	8	883	6	49
61	3	15	233	4	29	181	5	15	*	*	*
101	3	25	241	4	30	193	5	16	*	*	*
109	3	27	281	4	35	241	5	20	*	*	*
149	3	37	401	4	50	313	5	26	*	*	*
157	3	39	*	*	*	337	5	28	*	*	*
173	3	43	*	*	*	349	5	29	*	*	*
181	3	45	*	*	*	373	5	31	*	*	*
197	3	49	*	*	*	409	5	34	*	*	*
*	*	*	*	*	*	421	5	35	*	*	*
*	*	*	*	*	*	541	5	45	*	*	*
*	*	*	*	*	*	577	5	48	*	*	*

Table 2.1: The cardinality of codes constructed according to the technique of Section 2.5.1.2 for $t \leq 50$

n	w	t	n	w	t	n	w	t	n	w	t
41	4	5	13	5	1	641	8	20	281	9	7
73	4	9	37	5	3	929	8	29	401	9	10
89	4	11	61	5	5	1217	8	38	521	9	13
97	4	12	73	5	6	1409	8	44	601	9	15
113	4	14	97	5	8	1601	8	50	761	9	19
137	4	17	109	5	9	*	*	*	881	9	22
193	4	24	157	5	13	*	*	*	1201	9	30
233	4	29	181	5	15	*	*	*	1361	9	34
241	4	30	193	5	16	*	*	*	1481	9	37
257	4	32	229	5	19	*	*	*	1601	9	40
281	4	35	241	5	20	*	*	*	1721	9	43
313	4	39	277	5	23	*	*	*	1801	9	45
337	4	42	313	5	26	*	*	*	*	*	*
353	4	44	337	5	28	*	*	*	*	*	*
401	4	50	349	5	29	*	*	*	*	*	*
*	*	*	373	5	31	*	*	*	*	*	*
*	*	*	397	5	33	*	*	*	*	*	*
*	*	*	409	5	34	*	*	*	*	*	*
*	*	*	421	5	35	*	*	*	*	*	*
*	*	*	541	5	45	*	*	*	*	*	*
*	*	*	577	5	48	*	*	*	*	*	*

Table 2.2: The cardinality of codes constructed according to the technique of Section 2.5.3 for $t \leq 50$

2.5.4 An $(n, w, 1, 2)$ OOC

In general, an $(n, w, 1, 2)$ OOC can be constructed from an $(n, w, 1, 1)$ OOC by looking at its *reverse shifted adjacent relative delay vector.*

It's possible to construct $(n, w, 1, 2)$ OOC's by using any of the four designs for $(n, w, 1, 1)$ OOC's in Section 2.5.2 and extending i from $t-1$ to $2t-1$. Therefore, there are $2t$ codewords for the $(n, w, 1, 2)$ OOC compared with t codewords for the $(n, w, 1, 1)$ OOC.

Proof: We only prove the case for odd w in Wilson's construction. The extension to the other three cases is straightforward.

If we extend i from $t-1$ to $2t-1$ in Wilson's construction in Section 2.5.2, then for an $(n, w, 1, 2)$ OOC, the code consists of the blocks

$$\{[\alpha^{mi}, \alpha^{mi+2mt}, \alpha^{mi+4mt}, \ldots, \alpha^{mi+4m^2t}] : i = 0, 1, \ldots, 2t-1\}.$$

Keeping in mind that $\alpha^{w(w-1)t} = 1$ and $\alpha^{w(w-1)t/2} = -1$, then we can view the code as consisting of the blocks

$$\{[\alpha^{mi}, \alpha^{mi+2mt}, \alpha^{mi+4mt}, \ldots, \alpha^{mi+4m^2t}] : i = 0, 1, \ldots, t-1\},$$

and

$$\{[-\alpha^{mi}, -\alpha^{mi+2mt}, -\alpha^{mi+4mt}, \ldots, -\alpha^{mi+4m^2t}] : i = 0, 1, \ldots, t-1\}.$$

So if we use an *adjacent relative delay vector* $\mathbf{t}_{\mathbf{x}_i} = [t_0, t_1, \cdots, t_{w-1}]$ associated with the codeword $\mathbf{x}_i$ of the first group of an $(n, w, 1, 2)$ OOC, $\forall i \in \{0, 1, \ldots, t-1\}$, then the *adjacent relative delay vector* $\mathbf{t}_{\mathbf{y}_i}$ associated with the codeword $\mathbf{y}_i$ of the second group of an $(n, w, 1, 2)$ OOC $\forall i \in \{0, 1, \ldots, t-1\}$ is $[t_{w-1}, t_{w-2}, \cdots, t_0]$. This completes the proof.

2.5.5 A Recursive Construction of an $(n, w, 2, 1)$ OOC (Construction 3)

Chen *et. al* proposed a new construction for an $(n, w, 1, 1)$ OOC[37]. The idea was from Colbourns' construction for disjoint difference sets[38]. Here, we will show that their construction can be generalized to recursively construct a large blocklength $(n, w, 2, 1)$ OOC using our two constructions for smaller blocklength OOC's presented in Sections 2.5.1 and 2.5.3, respectively. Throughout this section, we use A_l to denote a set $\{0, 1, \ldots, l-1\}$.

Construction A: Consider an $(n, w, 2, 1)$ OOC with t codewords. Then a $(vn, w, 2, 1)$ OOC with vt codewords can be constructed if v is a positive integer and v and $(w-1)!$ are co-prime. This can be done as follows. Construct an *adjacent relative delay vector* $\mathbf{t}_{\mathbf{x}_i} = [t_0, t_1, \cdots, t_{w-1}]$ associated with the codeword $\mathbf{x}_i$ of an $(n, w, 2, 1)$ OOC, $\forall i \in A_t$. For each $\mathbf{t}_{\mathbf{x}_i}$, identify some $j \in A_w$ such that $t_j \neq t_k \ \forall k \in A_w \backslash \{j\}$. Right shift the vector $\mathbf{t}_{\mathbf{x}_i}$ cyclicly until t_j is in the last coordinate of $\mathbf{t}_{\mathbf{x}_i}$. Denote by $\mathbf{t'}_{\mathbf{x}_i}$ this cyclically shifted version $\mathbf{t}_{\mathbf{x}_i}$. From $\mathbf{t'}_{\mathbf{x}_i}$ write the associated block of the codeword as $(0, d_{i1}, d_{i2}, \ldots, d_{i(w-1)})$. Then the new codewords can be denoted by the blocks $(0, d_{i1} + jn, d_{i2} + 2jn, \ldots, d_{i(w-1)} + (w-1)jn)$, $\forall i \in A_t$ and $\forall j \in A_v$, where the addition is performed modulo vn. In all there are vt codewords. Furthermore, if $v \geq n$ then there are t additional codewords represented by the blocks $(0, nd_{i1}, nd_{i2}, \ldots, nd_{i(w-1)})$, $\forall i \in A_t$. Hence the total number of codewords is $(v+1)t$.

The proof for Construction A is similar to Theorem 2.1 in [38] and is omitted here.

2.6 Summary

In this chapter we derived new upper bounds on the number of users that can be supported on an optical network employing code division multiple access with OOC signature sequences. Unlike previous work in this area, we considered the possibility that the auto- and cross-correlation constraints might not be identical; indeed, the bounds derived suggest that it may sometimes be preferable to use such "asymmetric" OOC's. We then used the upper and lower bounds to illustrate some asymptotic properties of the cardinality of an OOC. We also draw the relationship between OOC's and two-level UEP codes. Finally we demonstrated three techniques for constructing $(n, w, 2, 1)$ OOC's that yield more codewords than could be possible with the parameters $(n, w-1, 1, 1)$ but which perform at least as well. Moreover, we have constructed $(n, w, 1, 2)$ OOC's which have twice as many codewords than $(n, w, 1, 1)$ OOC's.

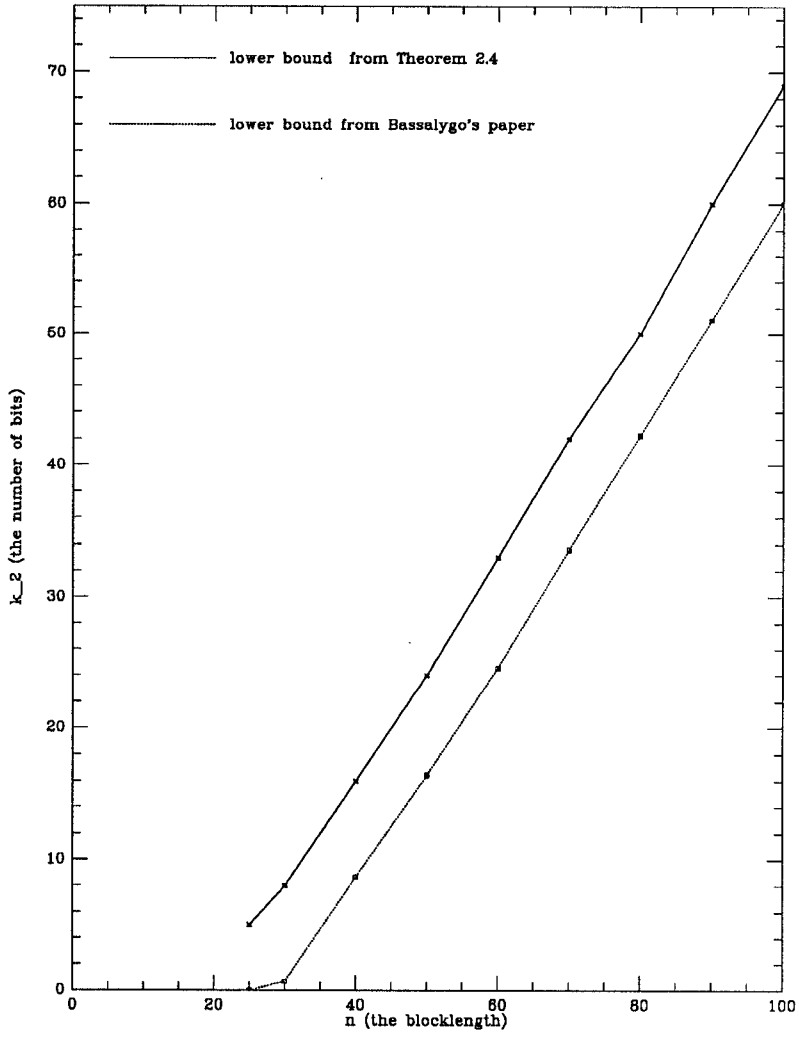


Figure 2.1: A comparison between the lower bound from Theorem 2.4 and the lower bound from Bassalygo's paper.

n	w	t	n	w	t	n	w	t	n	w	t
7	3	1	13	4	1	41	5	2	181	6	6
13	3	2	73	4	6	61	5	3	211	6	7
19	3	3	97	4	8	241	5	12	241	6	8
31	3	5	109	4	9	281	5	14	631	6	21
37	3	6	181	4	15	421	5	21	691	6	23
43	3	7	229	4	19	601	5	30	*	*	*
61	3	10	241	4	20	641	5	32	*	*	*
67	3	11	277	4	23	661	5	33	*	*	*
73	3	12	337	4	28	701	5	35	*	*	*
79	3	13	409	4	34	821	5	41	*	*	*
97	3	16	421	4	35	881	5	44	*	*	*
103	3	17	457	4	38	*	*	*	*	*	*
109	3	18	541	4	45	*	*	*	*	*	*
127	3	21	*	*	*	*	*	*	*	*	*
139	3	23	*	*	*	*	*	*	*	*	*
151	3	25	*	*	*	*	*	*	*	*	*
157	3	26	*	*	*	*	*	*	*	*	*
163	3	27	*	*	*	*	*	*	*	*	*
181	3	30	*	*	*	*	*	*	*	*	*
193	3	32	*	*	*	*	*	*	*	*	*
199	3	33	*	*	*	*	*	*	*	*	*
211	3	35	*	*	*	*	*	*	*	*	*
223	3	37	*	*	*	*	*	*	*	*	*
229	3	38	*	*	*	*	*	*	*	*	*
241	3	40	*	*	*	*	*	*	*	*	*
271	3	45	*	*	*	*	*	*	*	*	*
277	3	46	*	*	*	*	*	*	*	*	*
283	3	47	*	*	*	*	*	*	*	*	*

Table 2.3: The cardinality of codes constructed according to the technique of Section 2.5.2 from Wilson's construction for $t \leq 50$

n	w	t	n	w	t	n	w	t
337	7	8	1009	8	18	73	9	1
421	7	10	*	*	*	1153	9	16
463	7	11	*	*	*	1873	9	26
883	7	21	*	*	*	2017	9	28
1723	7	41	*	*	*	*	*	*

Table 2.4: The cardinality of codes constructed according to the technique of Section 2.5.2 from Wilson's construction for $t \leq 50$

n	w	t	n	w	t	n	w	t
7	3	1	31	6	1	337	7	8
13	3	2	151	6	5	379	7	9
19	3	3	181	6	6	421	7	10
31	3	5	211	6	7	463	7	11
37	3	6	241	6	8	547	7	13
43	3	7	271	6	9	631	7	15
61	3	10	331	6	11	673	7	16
67	3	11	421	6	14	967	7	23
73	3	12	541	6	18	1009	7	24
79	3	13	571	6	19	1051	7	25
97	3	16	601	6	20	1093	7	26
103	3	17	631	6	21	1303	7	31
109	3	18	661	6	22	1429	7	34
127	3	21	691	6	23	1471	7	35
139	3	23	751	6	25	1723	7	41
151	3	25	811	6	27	1933	7	46
157	3	26	991	6	33	2017	7	48
163	3	27	1021	6	34	*	*	*
181	3	30	1051	6	35	*	*	*
193	3	32	1171	6	39	*	*	*
199	3	33	1201	6	40	*	*	*
211	3	35	1231	6	41	*	*	*
223	3	37	1291	6	43	*	*	*
229	3	38	1321	6	44	*	*	*
241	3	40	1381	6	46	*	*	*
271	3	45	1471	6	49	*	*	*
277	3	46	*	*	*	*	*	*
283	3	47	*	*	*	*	*	*

Table 2.5: The cardinality of codes constructed according to the technique of Section 2.5.2 from the new construction for $t \leq 50$

Chapter 3

Variable Weight Optical Orthogonal Codes for CDMA Networks with Multiple Performance Requirements

Up to now all work on optical orthogonal codes (OOC's) has assumed that the weight of each codeword is the same. In this chapter we develop bounds on the size of OOC's when this assumption is removed. In addition, we demonstrate construction techniques for building such "variable weight" OOC's. The results demonstrate that it is possible to assign codewords with different weights among the users. Changing the weight of a user's signature sequence affects that user's performance; therefore this approach is useful for CDMA optical fiber networks with multiple performance requirements among the users.

3.1 Background and Motivation

As the demand for personal communication services continues to rise, multiple access techniques become ever more important. Code-division multiple access (CDMA) is a kind of spread spectrum technology that enables many users to share the same channel without interference by employing a unique signature

sequence to distinguish different users' transmission.

Optical orthogonal codes (OOC's) were introduced by Salehi *et. al* as a means of obtaining code division multiple access among asynchronous users on optical fiber networks. An OOC is a collection of binary sequences with good auto- and cross-correlation properties. Up to now all work on OOC's[6][7][8][9][10][11][12] have assumed that the weight of each codeword is the same. In this paper we develop bounds on the size of OOC's when this assumption is removed. In addition, we demonstrate construction techniques for building such "variable weight" OOC's. The results demonstrate that it is possible to assign codewords with different weights among the users.

Throughout this chapter, we use W, L, and Q, to denote the sets $\{w_0, w_1, \ldots, w_p\}$, $\{\lambda_a^0, \lambda_a^1, \ldots, \lambda_a^p\}$, and $\{q_0, q_1, \ldots, q_p\}$, respectively.

Definition: An (n, W, L, λ_c, Q) variable weight optical orthogonal code $\mathcal{C}$ is a collection of binary n-tuples such that the following two properties hold:

- (Weight Distribution) Every n-tuple in $\mathcal{C}$ has a Hamming weight contained in the set W; furthermore, there are exactly $q_i \cdot |\mathcal{C}|$ codeword of weight w_i – i.e., q_i indicates the fraction of codewords of weight w_i.
- (Auto-correlation Property) For any $\mathbf{x} = [x_0, x_1, \ldots, x_{n-1}] \in \mathcal{C}$ with Hamming weight $w_i \in A$ and any integer τ, $0 < \tau < n$,

$$\sum_{t=0}^{n-1} x_t x_{t\oplus\tau} \leq \lambda_a^i.$$

- (Cross-correlation Property) For any $\mathbf{x} = [x_0, x_1, \ldots, x_{n-1}] \in \mathcal{C}$ and any $\mathbf{y} = [y_0, y_1, \ldots, y_{n-1}] \in \mathcal{C}$ such that $\mathbf{x} \neq \mathbf{y}$ and any integer τ,

$$\sum_{t=0}^{n-1} x_t y_{t\oplus\tau} \leq \lambda_c.$$

Note: OOC's were defined in terms of periodic correlation; thus the

addition in the subscripts above – denoted "$\oplus$" – is all modulo-n.

The definition of a variable weight OOC is a generalization of the definition for OOC given in [6][7].

The use of OOC's for multiple access is described in [6][7][8].

- The auto-correlation constraint guarantees that each signature sequence is unlike cyclic shifts of itself. This property is used to enable the receiver to obtain *synchronization.*
- The cross-correlation constraint guarantees that each signature sequence is unlike cyclic shifts of the *other* signature sequences. This property is used to enable the receiver to estimate its message in the presence of interference from other users.

A reasonable "figure of merit" for a code is the number of interfering users necessary to cause the code to fail. For instance, assume synchronization has been achieved; then the only errors the i^{th} user can make are $0 \rightarrow 1$ errors, and they can only occur when enough *other* users interfere to make the correlation at the i^{th} receiver exceed $w_{\pi(i)}$ ($w_{\pi(i)}$ is the Hamming weight of i^{th} user's codeword.). Since each of those other users can contribute at most λ_c to the correlation, the performance "figure of merit" is $w_{\pi(i)}/\lambda_c$. In a similar vein, the synchronization "figure of merit" is $(w_{\pi(i)} - \lambda_a^{\pi(i)})/\lambda_c$ ($\lambda_a^{\pi(i)}$ is the auto-correlation constraint associated with $w_{\pi(i)}$.) for multiple-access synchronization and $w_{\pi(i)} - \lambda_a^{\pi(i)}$ for single-user synchronization.

From above, we can see that the weight of the codeword will affect the user's performance. Therefore, by assigning codewords with different weights we are able to accommodate multiple performance requirements among the network's users.

This chapter, which investigates variable weight OOC's, consists of four parts. First we derive new upper and lower bounds on the size of OOC's with different weights among the codewords. We then demonstrate techniques for constructing such OOC's. Two of the constructions reach the upper bound and hence they are optimal. Finally, we generalize the analysis of the performance for the constant weight OOC[39] to the variable weight OOC.

3.2 Some New Bounds on Optical Orthogonal Codes

Define $\Phi(n, W, L, \lambda_c, Q)$ to be the cardinality of an optimal variable weight optical orthogonal code with the given parameters – i.e.,

$$\Phi(n, W, L, \lambda_c, Q) \triangleq \max\{|\,\mathcal{C}\,|: \mathcal{C} \text{ is an } (n, W, L, \lambda_c, Q) \text{ variable weight OOC}\}.$$

In order to derive the upper bounds on $\Phi(n, W, L, \lambda_c, Q)$ we need to use Lemma 2.1-2.4 in Chapter 2. We rewrite them here for convenience.

Lemma 2.1: Let $\mathbf{x} = [x_0, x_1, \cdots, x_{n-1}]$ be a binary n-tuple. Then the inequality

$$\sum_{t=0}^{n-1} x_t x_{t\oplus\tau} \leq \lambda$$

holds for all $1 \leq \tau \leq n-1$ if and only if no component in $R_{\mathbf{x}}$ is repeated more than λ times.

Lemma 2.2: Let $\mathbf{x} = [x_0, x_1, \ldots, x_{n-1}]$ and $\mathbf{y} = [y_0, y_1, \ldots, y_{n-1}]$ be binary n-tuples. Then the inequality

$$\sum_{t=0}^{n-1} x_t y_{t\oplus\tau} \leq \lambda$$

holds for all $0 \leq \tau \leq n-1$ if and only if $M_{\mathbf{x},\lambda}$ and $M_{\mathbf{y},\lambda}$ are disjoint.

Lemma 2.3: Let $\mathbf{x} = [x_0, x_1, \ldots, x_{n-1}]$ be a binary n-tuple. Then the inequality

$$\sum_{t=0}^{n-1} x_t x_{t+\tau} \leq \lambda$$

holds for all $\tau = 1, 2 \cdots, n-1$ if and only if $|M_{\mathbf{x},\lambda}| = w\binom{w-1}{\lambda}$ – i.e., if and only if the vectors

$$\{[\sum_{k_0=0}^{i_0} t_{j \oplus k_0}, \sum_{k_1=i_0+1}^{i_1} t_{j \oplus k_1}, \sum_{k_2=i_1+1}^{i_2} t_{j \oplus k_2}, \cdots, \sum_{k_{\lambda-1}=i_{\lambda-2}+1}^{i_{\lambda-1}} t_{j \oplus k_{\lambda-1}}] : 0 \leq i_0 < i_1 < \cdots, < i_{\lambda-1} \leq w-2, j = 0, 1, \cdots, w-1\}$$

are all distinct.

Lemma 2.4: Let $\mathbf{x} \in \mathcal{C}$, where $\mathcal{C}$ is an $(n, w, \lambda + m, \lambda)$ optical orthogonal code. (Assume $m \geq 0$ is an integer.) Then if $w > 2\lambda - 2$,

$$|M_{\mathbf{x},\lambda}| \geq \frac{w\binom{w-1}{\lambda}}{m+1}.$$

Now we will use the characterizations developed above to provide upper bounds on $\Phi(n, W, L, \lambda_c, Q)$ when $\lambda_a^i \geq \lambda_c$ for all $\lambda_a^i \in L$).

The first bound is for the case $\lambda_a^i = \lambda_a = \lambda_c = \lambda$ for all $\lambda_a^i \in L$. It is a generalization of bound in [10].

Theorem 3.1: The following inequality holds:

$$\Phi(n, W, \{\lambda, \ldots, \lambda\}, \lambda, Q) \leq \frac{(n-1)(n-2)\ldots(n-\lambda)}{\sum_{i=0}^{p} q_i w_i (w_i - 1)\ldots(w_i - \lambda)}.$$

Proof: Let $\mathcal{C}$ be an optimal $(n, W, \{\lambda, \ldots, \lambda\}, \lambda, Q)$ OOC – i.e., $|\mathcal{C}| = \Phi(n, W, \{\lambda, \ldots, \lambda\}, \lambda, Q)$. From Lemma 2.3 we know that for every $\mathbf{x} \in \mathcal{C}$ with weight w_i the set $M_{\mathbf{x},\lambda}$ consists of $w_i\binom{w_i-1}{\lambda}$ distinct integer λ-tuples. Furthermore, from

Lemma 2.2 we know that for $\mathbf{x}, \mathbf{y} \in \mathcal{C}$, $\mathbf{x} \neq \mathbf{y}$, the sets $M_{\mathbf{x},\lambda}$ and $M_{\mathbf{y},\lambda}$ are disjoint. Therefore the union of $M_{\mathbf{x},\lambda}$ as $\mathbf{x}$ varies over all $\mathbf{x} \in \mathcal{C}$ consists of $\sum_{i=0}^{p} q_i \Phi(n, W, \{\lambda, \ldots, \lambda\}, \lambda, Q) \cdot w_i \binom{w_i - 1}{\lambda}$ distinct integer λ-tuples. However, by definition if $[a_0, a_1, \ldots, a_{\lambda-1}] \in M_{\mathbf{x},\lambda}$ then $a_0 + a_1 + \ldots + a_{\lambda-1} \leq n - 1$. But the number of ways to select λ positive a_i's that sum to no more than $n - 1$ is just the number of compositions of n with $\lambda + 1$ positive parts – and that is equal to $\binom{n-1}{\lambda}$. We have thus shown that

$$\sum_{i=0}^{p} q_i \Phi(n, W, \{\lambda, \ldots, \lambda\}, \lambda, Q) \cdot w_i \binom{w_i - 1}{\lambda} \leq \binom{n-1}{\lambda},$$

which was to be proven. QED.

For fixed λ, w_0, w_1 it is interesting to look at how the upper bound in Theorem 3.1 varies with q_0, q_1. For example if $q_0 = 0, q_1 = 1$ we are looking at a constant weight OOC with weight w_1, and if $q_0 = 1/2, q_1 = 1/2$, we are looking at an OOC with half the codewords with weight w_0 and half the codewords with weight w_1. In Figure 3.1, we can see the factors of q_0, q_1 as we vary the blocklength n when $w_0 = 3, w_1 = 4$.

Our next goal is to bound $\Phi(n, W, L, \lambda_c, Q)$ for $\lambda_a^i \geq \lambda_c$ ($\lambda_a^i \in L$).

Theorem 3.2: Let m be a non-negative integer. Then if $w_i > 2\lambda - 2$ (for all $w_i \in W$),

$$\Phi(n, W, \{\lambda + m, \ldots, \lambda + m\}, \lambda, Q) \leq \frac{(n-1)(n-2)\ldots(n-\lambda)(m+1)}{\sum_{i=0}^{p} q_i w_i (w_i - 1)(w_i - 2)\ldots(w_i - \lambda)}.$$

Proof: Let $\mathcal{C}$ be an $(n, W, \{\lambda+m, \ldots, \lambda+m\}, \lambda, Q)$ OOC such that $|\mathcal{C}| = \Phi(n, W, \{\lambda + m, \ldots, \lambda + m\}, \lambda, Q)$. By Lemma 2.4, for any $\mathbf{x} \in \mathcal{C}$ with weight w_i,

$$|M_{\mathbf{x},\lambda}| \geq \frac{w_i \binom{w_i - 1}{\lambda}}{m + 1}.$$

Furthermore, by Lemma 2.2, $M_{\mathbf{x},\lambda}$ and $M_{\mathbf{y},\lambda}$ are disjoint for $\mathbf{x}, \mathbf{y} \in \mathcal{C}$ and $\mathbf{x} \neq \mathbf{y}$; therefore $|M_{\mathbf{x},\lambda}|$ summed over all $\mathbf{x} \in \mathcal{C}$ cannot exceed the total number of integer λ-tuples that are "allowable" as elements of $M_{\mathbf{x},\lambda}$ – a number shown in the proof of Theorem 3.1 to be $\binom{n-1}{\lambda}$. Thus,

$$\begin{aligned}
\binom{n-1}{\lambda} &\geq \sum_{\mathbf{x}\in\mathcal{C}} |M_{\mathbf{x},\lambda}| \\
&\geq \sum_{i=0}^{p} q_i \Phi(n, W, \{\lambda+m, \ldots, \lambda+m\}, \lambda, Q) \cdot \min\{|M_{\mathbf{x},\lambda}| : \mathbf{x} \in \mathcal{C}\} \\
&\geq \sum_{i=0}^{p} q_i \Phi(n, W, \{\lambda+m, \ldots, \lambda+m\}, \lambda, Q) \cdot \frac{w_i \binom{w_i - 1}{\lambda}}{m+1}
\end{aligned}$$

which was to be proven. QED.

Theorem 3.3: Let $\lambda_a^i \geq \lambda$ ($\lambda_a^i \in L$). Then if $w_i > 2\lambda - 2$ (for all $w_i \in W$),

$$\Phi(n, W, L, \lambda, Q) \leq \frac{(n-1)(n-2)\ldots(n-\lambda)}{\sum_{i=0}^{p} q_i w_i (w_i-1)(w_i-2)\ldots(w_i-\lambda)/(\lambda_a^i - \lambda + 1)}.$$

Proof: Proof completely analogous to that of Theorem 3.2.

Finally, we derive a new lower bound on the cardinality of an (n, W, L, λ_c, Q) variable weight OOC using Theorem 2.3 for an $(n, w, \lambda_a, \lambda_c)$ OOC.

Theorem 3.4:

$$\Phi(n, W, L, \lambda_c, Q) \geq \frac{\sum_{i=0}^{p} \left[\binom{n}{w_i} - F \right]}{G},$$

where

$$\begin{aligned}
F = \sum_{\substack{\delta=1 \\ \delta | n}}^{\lfloor n/2 \rfloor} \sum_{N=0}^{\lfloor w_i \delta / n \rfloor} \Bigg[& \triangle(N - w_i\delta/n) \binom{\delta}{N} + \sum_{c=\lceil w_i \delta/n \rceil}^{w_i - \lambda_a^i + N - 1} \binom{c}{N} \binom{w_i - Nn/\delta - 1}{c - N - 1} \\
& \binom{\delta}{N} \binom{n - Nn/\delta}{c - N} \Bigg] + |\{\delta : 1 \leq \delta \leq \lfloor n/2 \rfloor, \delta \not| n\}| \sum_{c=1}^{w_i - \lambda_a^i - 1} \binom{w_i - 1}{c-1} \binom{n}{c},
\end{aligned}$$

$$\triangle(x) = \begin{cases} 1, & \text{if } x = 0; \\ 0, & \text{otherwise,} \end{cases}$$

and

$$G = n \sum_{i=0}^{p} \sum_{j=\lambda_c+1}^{w_i} q_i \binom{n-w_i}{w_i - j} \binom{w_i}{j}.$$

Proof: The proof is analogous to Theorem 2.4 in Chapter 2 and is omitted.

3.3 Constructions of OOC's

3.3.1 Constructions of an $(\mathbf{n}, \{\mathbf{w_0}, \mathbf{w_1}\}, \{\mathbf{1}, \mathbf{1}\}, \mathbf{1}, \mathbf{Q})$ OOC (Construction 1)

We now demonstrate a technique for constructing an $(n, \{w_0, w_1\}, \{1, 1\}, 1, \{t_0/(t_0 + t_1), t_1/(t_0 + t_1)\})$ OOC. Basically, we want to combine two optimal constructions – one for an $(n, w, 1, 1)$ OOC and one for an $(n, w+1, 1, 1)$ OOC – to construct an $(n, \{w+1, w\}, \{1, 1\}, 1, \{t_0/(t_0+t_1), t_1/(t_0+t_1)\})$ OOC. Since the cardinality of the resulting code reaches the upper bound in Theorem 3.1, it is an optimal construction. We begin by considering specific values of w_i ($w_0 = 4$ and $w_1 = 3$) and then generalizing the technique.

3.3.1.1 Some Specific Examples

An $(\mathbf{n}, \{\mathbf{4}, \mathbf{3}\}, \{\mathbf{1}, \mathbf{1}\}, \mathbf{1}, \{\mathbf{1/2}, \mathbf{1/2}\})$ OOC: Let n be a prime number such that $n = 18t + 1$ for an integer t. Let α be a primitive element of the field GF(n) such that $\alpha^q = \alpha^{6t} - 1$ and $\alpha^r = \alpha^y(\alpha^{6t} - 1)$, where y is any integer between 1 and $3t - 1$, q and r are integers satisfying one of the following two conditions:

- $q \equiv 1 \pmod 3$ and $r \equiv 2 \pmod 3$;
- $q \equiv 2 \pmod 3$ and $r \equiv 1 \pmod 3$.

Then we can construct an $(n, \{4,3\}, \{1,1\}, 1, \{1/2, 1/2\})$ OOC $\mathcal{C}$ with cardinality $|\mathcal{C}| = 2t$ (t codewords for $(n,4,1,1)$ OOC and t codewords for $(n,3,1,1)$ OOC) as follows. The i^{th} weight-four codeword $\mathbf{x}_i$ contains a "1" in positions 0, α^{3i}, α^{6t+3i}, and α^{12t+3i} and a "0" everywhere else for an $(n,4,1,1)$ OOC; the i^{th} weight-three codeword $\mathbf{y}_i$ contains a "1" in positions α^{y+3i}, $\alpha^{y+6t+3i}$, and $\alpha^{y+12t+3i}$ and a "0" everywhere else for an $(n,3,1,1)$ OOC. This holds for $i = 0, 1, \ldots, t-1$. (Note: We say that the code consists of the "blocks" $\{[0, \alpha^{3i}, \alpha^{6t+3i}, \alpha^{12t+3i}]$ and $\{[\alpha^{y+3i}, \alpha^{y+6t+3i}, \alpha^{y+12t+3i}] : i = 0, 1, \ldots, t-1\}$.)

To see that this construction yields an $(n, \{4,3\}, \{1,1\}, 1), \{1/2, 1/2\})$ code let $R_{\mathbf{x}_0}$ denote the array consisting of all the relative delays between pairs of 1's in $\mathbf{x}_0$ and $R_{\mathbf{y}_0}$ for $\mathbf{y}_0$. Keeping in mind that $x^{18t} = 1$ and $x^{9t} = -1$, simple algebra reveals that

$$R_{\mathbf{x}_0} = \begin{pmatrix} 1 & \alpha^{6t}-1 & \alpha^{6t}(\alpha^{6t}-1) & \alpha^{3t} \\ \alpha^{6t} & \alpha^{3t}(\alpha^{6t}-1) & \alpha^{15t} & \alpha^{12t}(\alpha^{6t}-1) \\ \alpha^{12t} & \alpha^{9t} & \alpha^{9t}(\alpha^{6t}-1) & \alpha^{15t}(\alpha^{6t}-1) \end{pmatrix}.$$

$$R_{\mathbf{y}_0} = \begin{pmatrix} \alpha^{y}(\alpha^{6t}-1) & \alpha^{6t+y}(\alpha^{6t}-1) & \alpha^{12t+y}(\alpha^{6t}-1) \\ \alpha^{3t+y}(\alpha^{6t}-1) & \alpha^{9t+y}(\alpha^{6t}-1) & \alpha^{15t+y}(\alpha^{6t}-1) \end{pmatrix}.$$

Every component of $R_{\mathbf{x}_0}$ and $R_{\mathbf{y}_0}$ is of the form $\beta\alpha^{3jt}$ where $\beta \in \{1, \alpha^{6t}-1, \alpha^{y}(\alpha^{6t}-1)\}$ and $j \in \{0,1,2,3,4,5\}$. Therefore as long as the base-α logarithms of $\alpha^{6t}-1$ and $\alpha^{y}(\alpha^{6t}-1)$ are not equivalent to each other mod 3 and are not equivalent to zero mod 3,

$$\beta_1 \neq \beta_2 \text{ or } j_1 \neq j_2 \Rightarrow \beta_1\alpha^{3j_1 t} \neq \beta_2\alpha^{3j_2 t}.$$

And so no element of $R_{\mathbf{x}_0}$ and $R_{\mathbf{y}_0}$ is repeated; this implies the auto-correlation constraint is met.

To check the cross-correlation constraint, we note as before that $M_{\mathbf{x},1}$ and $M_{\mathbf{y},1}$ are sets whose "vectors" are just the 1-tuples from $R_{\mathbf{x}}$ and $R_{\mathbf{y}}$. Therefore

as long as the components of $R_{\mathbf{x}_i}$, $R_{\mathbf{y}_i}$ and the components of $R_{\mathbf{x}_j}$,$R_{\mathbf{y}_j}$ form disjoint sets for $i \neq j$, we will have proven that the cross-correlation constraint is met. But to obtain $R_{\mathbf{x}_i}$ and $R_{\mathbf{y}_i}$ we just multiply $R_{\mathbf{x}_0}$ and $R_{\mathbf{y}_0}$ through by α^{3i}; thus, the components of $R_{\mathbf{x}_i}$ and $R_{\mathbf{y}_i}$ are of the form $\beta\alpha^{3(jt+i)}$. As long as $0 \leq i \leq t-1$, the components form disjoint sets and so the cross-correlation constraint is met.

Example: Let $n = 37$ and $t = 2$. Choose $\alpha = 2$ as the primitive element of GF(37) and so $2^{6t}-1 = 25 = 2^{10}$ while $y = 1$ - i.e., $q = 10$ and $r = 11$. Then the code consists of the blocks $\{[0,1,10,26],[0,6,8,23],[2,15,20],[9,12,16]\}$ and so the four codewords are

$$\begin{aligned}
\mathbf{x}_0 &= [11000000001000000000000010000000000] \\
\mathbf{x}_1 &= [10000010100000000000000100000000000] \\
\mathbf{y}_0 &= [00100000000000010000100000000000000] \\
\mathbf{y}_1 &= [00000000010010001000000000000000000].
\end{aligned}$$

A $(\mathbf{n},\{4,3\},\{1,1\},1,\{\mathbf{t_0}/(\mathbf{t_0}+\mathbf{t_1}),\mathbf{t_1}/(\mathbf{t_0}+\mathbf{t_1})\})$ OOC: Let n be a prime number such that $n = 12t_0 + 6t_1 + 1$ for integer t_0 and t_1 (Assume $t_0|t_1$). Let α be a primitive element of the field GF(n) such that $\alpha^q = \alpha^{4t_0+2t_1} - 1$ and $\alpha^{r_j} = \alpha^{y_j}(\alpha^{4t_0+2t_1} - 1)(j = 0,\cdots,(t_1/t_0)-1)$, where y_j are integers between 1 and $2t_0 + t_1 - 1$, and q and r_j are integers that are all distinct modulo $2 + t_1/t_0$ and non-zero modulo $2 + t_1/t_0$.

Then we can construct an $(n,\{4,3\},\{1,1\},1,\{t_0/(t_0+t_1),t_1/(t_0+t_1)\})$ OOC $\mathcal{C}$ with cardinality $|\mathcal{C}| = t_0+t_1$ (t_0 codewords for an $(n,4,1,1)$ OOC and t_1 codewords for an $(n,3,1,1)$ OOC) with the blocks $\{[0,\alpha^{(2+t_1/t_0)i},\alpha^{4t_0+2t_1+(2+t_1/t_0)i},$ $\alpha^{8t_0+4t_1+(2+t_1/t_0)i}] : i = 0,1,\ldots,t_0-1\}$ and $\{[\alpha^{(2+t_1/t_0)i+y_j},\alpha^{4t_0+2t_1+(2+t_1/t_0)i+y_j},$

$\alpha^{8t_0+4t_1+(2+t_1/t_0)i+y_j}] : i = 0, 1, \ldots, t_0 - 1, j = 0, \cdots, (t_1/t_0) - 1\}$.

The proof that this construction yields an $(n, \{4, 3\}, \{1, 1\}, 1, \{t_0/(t_0 + t_1), t_1/(t_0 + t_1)\})$ code when the number of the codewords of weight three and four are different is analogous to the result for the previous example and is omitted.

3.3.1.2 Generalizing the Examples

The techniques described in Section 3.3.1.1 are easily generalized to different values of w_0 and w_1; the details are given below.

An $(n, \{w + 1, w\}, \{1, 1\}, 1, \{t_0/(t_0 + t_1), t_1/(t_0 + t_1)\})$ OOC for odd w: Let $w = 2m + 1$ and choose n to be a prime number such that $n = (w + 1)wt_0 + w(w - 1)t_1 + 1$. If r denotes the greatest common divisor of t_0 and t_1, let α be a primitive element of GF(n) such that

- $\{\log_\alpha[\alpha^{x_j}(\alpha^{[(w+1)t_0+(w-1)t_1]k} - 1)] : 1 \leq k \leq m, j = 0, \cdots, (t_1/r) - 1\}$,
- $\{\log_\alpha[\alpha^{z_l}(\alpha^{[(w+1)t_0+(w-1)t_1]k} - 1)] : 1 \leq k \leq m, l = 0, \cdots, (t_0/r) - 1\}$,
- $\{z_l : l = 0, \cdots, (t_0/r) - 1\}$

are all distinct modulo $((m+1)t_0+mt_1)/r$, where x_j and z_l are integers between 0 and $(m + 1)t_0 + mt_1 - 1$. Then the code consisting of the blocks

$$\{[0, \alpha^{[(m+1)t_0+mt_1]i/r+z_l}, \alpha^{[(m+1)t_0+mt_1]i/r+z_l+(2m+2)t_0+2mt_1}, \cdots,$$
$$\alpha^{[(m+1)t_0+mt_1]i/r+z_l+(2m+2)2mt_0+4m^2t_1}] : i = 0, \ldots, r - 1, l = 0, \cdots, (t_0/r) - 1\}$$

and the blocks

$$\{[\alpha^{[(m+1)t_0+mt_1]i/r+x_j}, \alpha^{[(m+1)t_0+mt_1]i/r+x_j+(2m+2)t_0+2mt_1}, \cdots,$$
$$\alpha^{[(m+1)t_0+mt_1]i/r+x_j+(2m+2)2mt_0+4m^2t_1}] : i = 0, \ldots, r - 1, j = 0, \cdots, (t_1/r) - 1\}$$

form an $(n, \{w+1, w\}, \{1,1\}, 1, \{t_0/(t_0+t_1), t_1/(t_0+t_1)\})$ OOC.

Some of constructions in Section 3.3.1.2 are in Table 3.1. All the constructions in Table 3.1 are optimal. For example consider a $(163, \{4,3\}, \{1,1\}, 1, \{\frac{1}{2}, \frac{1}{2}\})$ OOC, the cardinality is equal to the bound in Theorem 3.1.

code	t_0	t_1	primitive α	x_0	x_1
$(19, \{4,3\}, \{1,1\}, 1, \{\frac{1}{2}, \frac{1}{2}\})$	1	1	2	2	–
$(37, \{4,3\}, \{1,1\}, 1, \{\frac{1}{2}, \frac{1}{2}\})$	2	2	2	1	–
$(109, \{4,3\}, \{1,1\}, 1, \{\frac{1}{2}, \frac{1}{2}\})$	6	6	6	2	–
$(127, \{4,3\}, \{1,1\}, 1, \{\frac{1}{2}, \frac{1}{2}\})$	7	7	3	2	–
$(163, \{4,3\}, \{1,1\}, 1, \{\frac{1}{2}, \frac{1}{2}\})$	9	9	2	1	–
$(181, \{4,3\}, \{1,1\}, 1, \{\frac{1}{2}, \frac{1}{2}\})$	10	10	2	1	–
$(199, \{4,3\}, \{1,1\}, 1, \{\frac{1}{2}, \frac{1}{2}\})$	11	11	3	2	–
$(73, \{4,3\}, \{1,1\}, 1, \{\frac{1}{3}, \frac{2}{3}\})$	3	6	5	1	2
$(193, \{4,3\}, \{1,1\}, 1, \{\frac{1}{3}, \frac{2}{3}\})$	8	16	5	1	3
$(101, \{6,5\}, \{1,1\}, 1, \{\frac{1}{2}, \frac{1}{2}\})$	2	2	2	3	–
$(151, \{6,5\}, \{1,1\}, 1, \{\frac{1}{2}, \frac{1}{2}\})$	3	3	6	4	–
$(701, \{6,5\}, \{1,1\}, 1, \{\frac{1}{2}, \frac{1}{2}\})$	14	14	2	1	–
$(751, \{6,5\}, \{1,1\}, 1, \{\frac{1}{2}, \frac{1}{2}\})$	15	15	3	2	–
$(71, \{6,5\}, \{1,1\}, 1, \{\frac{1}{3}, \frac{2}{3}\})$	1	2	7	1	6

Table 3.1: The cardinality of codes constructed according to the technique of Section 3.3.1.2

3.3.2 Constructions of an $(\mathbf{n}, \{\mathbf{w_0}, \mathbf{w_1}\}, \{\mathbf{1,1}\}, \mathbf{1}, \mathbf{Q})$ OOC (Construction 2)

In this section, we generalize the constructions in Section 2.5.2 for an $(n, w, 1, 1)$ OOC to construct an $(n, \{w+1, w\}, \{1,1\}, 1, \{t_0/(t_0+t_1), t_1/(t_0+t_1)\})$ OOC as in Section 3.3.1.

An $(\mathbf{n}, \{\mathbf{w}+\mathbf{1}, \mathbf{w}\}, \{\mathbf{1,1}\}, \mathbf{1}, \{\mathbf{t_0}/(\mathbf{t_0}+\mathbf{t_1}), \mathbf{t_1}/(\mathbf{t_0}+\mathbf{t_1})\})$ OOC for $\mathbf{w} = \mathbf{4m}+\mathbf{2}$: Let $w = 4m+2$ and choose n to be a prime number such that

$n = (w+1)wt_0 + w(w-1)t_1 + 1$. If r denotes the greatest common divisor of t_0 and t_1, let α be a primitive element of GF(n) such that all of the following,

- $\{\log_\alpha[\alpha^{x_j}(\alpha^{[2(w+1)t_0+2(w-1)t_1]k+y} - 1)] : 0 \le k \le m, j = 0, \cdots, (t_1/r) - 1\}$;
- $\{\log_\alpha[\alpha^{x_j}(\alpha^{[2(w+1)t_0+2(w-1)t_1]k} - \alpha^y)] : 1 \le k \le m, j = 0, \cdots, (t_1/r) - 1\}$;
- $\{\log_\alpha[\alpha^{x_j}(\alpha^{[2(w+1)t_0+2(w-1)t_1]k} - 1)] : 1 \le k \le m, j = 0, \cdots, (t_1/r) - 1\}$;
- $\{\log_\alpha[\alpha^{x_j+y}(\alpha^{[2(w+1)t_0+2(w-1)t_1]k} - 1)] : 1 \le k \le m, j = 0, \cdots, (t_1/r) - 1\}$;
- $\{\log_\alpha[\alpha^{z_l}(\alpha^{[2(w+1)t_0+2(w-1)t_1]k+y} - 1)] : 0 \le k \le m, l = 0, \cdots, (t_0/r) - 1\}$;
- $\{\log_\alpha[\alpha^{z_l}(\alpha^{[2(w+1)t_0+2(w-1)t_1]k} - \alpha^y)] : 1 \le k \le m, l = 0, \cdots, (t_0/r) - 1\}$;
- $\{\log_\alpha[\alpha^{z_l}(\alpha^{[2(w+1)t_0+2(w-1)t_1]k} - 1)] : 1 \le k \le m, l = 0, \cdots, (t_0/r) - 1\}$;
- $\{\log_\alpha[\alpha^{z_l+y}(\alpha^{[2(w+1)t_0+2(w-1)t_1]k} - 1)] : 1 \le k \le m, l = 0, \cdots, (t_0/r) - 1\}$;
- $\{z_l : l = 0, \cdots, (t_0/r) - 1\}$;
- $\{z_l + y : l = 0, \cdots, (t_0/r) - 1\}$,

are all distinct modulo $((w+1)t_0 + (w-1)t_1)/r$, where x_j and z_l are integers between 0 and $(w+1)t_0 + (w-1)t_1 - 1$ and y is an integer between 1 and $2(w+1)t_0 + 2(w-1)t_1 - 1$. Then the code consisting of the blocks

$$\{[0, \alpha^{[(w+1)t_0+(w-1)t_1]i/r+z_l}, \alpha^{[(w+1)t_0+(w-1)t_1]i/r+z_l+y}, \alpha^{[(w+1)t_0+(w-1)t_1]i/r+z_l+2(w+1)t_0+2(w-1)t_1}, \cdots, \alpha^{[(w+1)t_0+(w-1)t_1]i/r+z_l+y+4(w+1)mt_0+4(w-1)mt_1}] : i = 0, 1, \ldots, r-1, l = 0, \cdots, (t_0/r) - 1\}$$

and the blocks

$$\{[\alpha^{[(w+1)t_0+(w-1)t_1]i/r+x_j}, \alpha^{[(w+1)t_0+(w-1)t_1]i/r+x_j+y}, \alpha^{[(w+1)t_0+(w-1)t_1]i/r+x_j+2(w+1)t_0+2(w-1)t_1}, \cdots,$$

$$\alpha^{[(w+1)t_0+(w-1)t_1]i/r+x_j+y+4(w+1)mt_0+4(w-1)mt_1}] :$$

$$i = 0, 1, \ldots, r-1, j = 0, \cdots, (t_1/r) - 1\}$$

form an $(n, \{w+1, w\}, \{1,1\}, 1, \{t_0/(t_0+t_1), t_1/(t_0+t_1)\})$ OOC.

3.3.3 Constructions of an $(n, \{w_0, w_1\}, \{2,2\}, 1, Q)$ OOC (Construction 1)

As indicated in Section 3.3.1, we use the the same technique to construct an $(n, \{w+1, w\}, \{2,2\}, 1, \{t_0/(t_0+t_1), t_1/(t_0+t_1)\})$ OOC from the construction in Section 2.5.1.2 for an $(n, w, 2, 1)$ OOC.

An $(n, \{w+1, w\}, \{2,2\}, 1, \{t_0/(t_0+t_1), t_1/(t_0+t_1)\})$ OOC for even w: Let $w = 2m$ and choose n to be a prime number such that $n = [(w+1)^2 - 1]/2t_0 + w^2/2t_1 + 1$. If r denotes the greatest common divisor of t_0 and t_1, let α be a primitive element of GF(n) such that

- $\{\log_\alpha[\alpha^{x_j}(\alpha^{[(m+1)t_0+mt_1]k} - 1)] : 1 \le k \le m, j = 0, \cdots, (t_1/r) - 1\}$,
- $\{\log_\alpha[\alpha^{z_l}(\alpha^{[(m+1)t_0+mt_1]k} - 1)] : 1 \le k \le m, l = 0, \cdots, (t_0/r) - 1\}$,
- $\{z_l : l = 0, \cdots, (t_0/r) - 1\}$

are all distinct modulo $((m+1)t_0 + mt_1)/r$, where x_j and z_l are integers between 0 and $(m+1)t_0 + mt_1 - 1$. Then the code consisting of the blocks

$$\{[0, \alpha^{[(m+1)t_0+mt_1]i/r+z_l}, \alpha^{[(m+1)t_0+mt_1]i/r+z_l+(m+1)t_0+mt_1}, \cdots,$$

$$\alpha^{[(m+1)t_0+mt_1]i/r+z_l+(m+1)(2m-1)t_0+m(2m-1)t_1}] :$$

$$i = 0, \ldots, r-1, l = 0, \cdots, (t_0/r) - 1\}$$

and the blocks

$$\{[\alpha^{[(m+1)t_0+mt_1]i/r+x_j}, \alpha^{[(m+1)t_0+mt_1]i/r+x_j+(m+1)t_0+mt_1}, \ldots,$$

$$\alpha^{[(m+1)t_0+mt_1]i/r+x_j+(m+1)(2m-1)t_0+m(2m-1)t_1]}:$$

$$i=0,\ldots,r-1, j=0,\cdots,(t_1/r)-1\}$$

form an $(n,\{w+1,w\},\{2,2\},1,\{t_0/(t_0+t_1),t_1/(t_0+t_1)\})$ OOC.

Some of constructions in Section 3.3.3 are in Table 3.2.

code	t_0	t_1	primitive α	x_0
$(41,\{5,4\},\{2,2\},1,\{\frac{1}{2},\frac{1}{2}\})$	2	2	6	1
$(61,\{5,4\},\{2,2\},1,\{\frac{1}{2},\frac{1}{2}\})$	3	3	2	1
$(101,\{5,4\},\{2,2\},1,\{\frac{1}{2},\frac{1}{2}\})$	5	5	2	1
$(181,\{5,4\},\{2,2\},1,\{\frac{1}{2},\frac{1}{2}\})$	9	9	2	1
$(281,\{5,4\},\{2,2\},1,\{\frac{1}{2},\frac{1}{2}\})$	14	14	3	4

Table 3.2: The cardinality of codes constructed according to the technique of Section 3.3.3

3.3.4 Constructions of an $(n,\{w_0,w_1\},\{2,2\},1,Q)$ OOC (Construction 2)

As indicated in Section 3.3.1, we use the the same technique to construct an $(n,\{w+1,w\},\{2,2\},1,\{t_0/(t_0+t_1),t_1/(t_0+t_1)\})$ OOC from the construction in Section 2.5.3 for an $(n,w,2,1)$ OOC.

An $(n,\{w+1,w\},\{2,2\},1,\{t_0/(t_0+t_1),t_1/(t_0+t_1)\})$ OOC for $w=4m$: Let $w=4m$ and choose n to be a prime number such that $n=[(w+1)^2-1]/2t_0+w^2/2t_1+1$. If r denotes the greatest common divisor of t_0 and t_1, let α be a primitive element of GF(n) such that all the following,

- $\{\log_\alpha[\alpha^{x_j}(\alpha^{[(w+2)t_0+wt_1]k+y}-1)]:0\le k\le m-1, j=0,\cdots,(t_1/r)-1\}$;
- $\{\log_\alpha[\alpha^{x_j}(\alpha^{[(w+2)t_0+wt_1]k}-\alpha^y)]:1\le k\le m, j=0,\cdots,(t_1/r)-1\}$;
- $\{\log_\alpha[\alpha^{x_j}(\alpha^{[(w+2)t_0+wt_1]k}-1)]:1\le k\le m, j=0,\cdots,(t_1/r)-1\}$;

- $\{\log_\alpha[\alpha^{x_j+y}(\alpha^{[(w+2)t_0+wt_1]k}-1)] : 1 \le k \le m, j = 0, \cdots, (t_1/r)-1\}$;
- $\{\log_\alpha[\alpha^{z_l}(\alpha^{[(w+2)t_0+wt_1]k+y}-1)] : 0 \le k \le m-1, l = 0, \cdots, (t_0/r)-1\}$;
- $\{\log_\alpha[\alpha^{z_l}(\alpha^{[(w+2)t_0+wt_1]k}-\alpha^y)] : 1 \le k \le m, l = 0, \cdots, (t_0/r)-1\}$;
- $\{\log_\alpha[\alpha^{z_l}(\alpha^{[(w+2)t_0+wt_1]k}-1)] : 1 \le k \le m, l = 0, \cdots, (t_0/r)-1\}$;
- $\{\log_\alpha[\alpha^{z_l+y}(\alpha^{[(w+2)t_0+wt_1]k}-1)] : 1 \le k \le m, l = 0, \cdots, (t_0/r)-1\}$;
- $\{z_l : l = 0, \cdots, (t_0/r)-1\}$;
- $\{z_l + y : l = 0, \cdots, (t_0/r)-1\}$,

are all distinct modulo $((w+2)t_0+wt_1)/r$, where x_j and z_l are integers between 0 and $(w+2)t_0+wt_1-1$ and y is an integer between 1 and $(w+2)t_0+wt_1-1$. Then the code consisting of the blocks

$$\{[0, \alpha^{[(w+2)t_0+wt_1]i/r+z_l}, \alpha^{[(w+2)t_0+wt_1]i/r+z_l+y}, \alpha^{[(w+2)t_0+wt_1]i/r+z_l+(w+2)t_0+wt_1}, \cdots, \alpha^{[(w+2)t_0+wt_1]i/r+z_l+y+(w+2)mt_0+wmt_1}] : i = 0, 1, \ldots, r-1, l = 0, \cdots, (t_0/r)-1\}$$

and the blocks

$$\{[\alpha^{[(w+2)t_0+wt_1]i/r+x_j}, \alpha^{[(w+2)t_0+wt_1]i/r+x_j+y}, \alpha^{[(w+2)t_0+wt_1]i/r+x_j+(w+2)t_0+wt_1}, \cdots, \alpha^{[(w+2)t_0+wt_1]i/r+x_j+y+(w+2)mt_0+wmt_1}] : i = 0, 1, \ldots, r-1, j = 0, \cdots, (t_1/r)-1\}$$

form an $(n, \{w+1, w\}, \{2, 2\}, 1, \{t_0/(t_0+t_1), t_1/(t_0+t_1)\})$ OOC.

3.3.5 Constructions of an $(n, \{w_0, w_1\}, \{2, 1\}, 1, Q)$ OOC

Instead of constructing an OOC with the same auto-correlation as in previous constructions, we now demonstrate the technique to construct an $(n, \{2w, w\}, \{2, 1\}, 1, \{t_0/(t_0 + t_1), t_1/(t_0 + t_1)\})$ OOC, where $(n, \{2w, w\}, \{2, 1\}, 1, \{t_0/(t_0 + t_1), t_1/(t_0 + t_1)\})$ OOC means two kinds of OOC's: one, an $(n, 2w, 2, 1)$ OOC, and the other, an $(n, w, 1, 1)$ OOC.

3.3.5.1. Constructions of an $(n, \{2w, w\}, \{2, 1\}, 1, Q)$ OOC for odd w

An $(n, \{2w, w\}, \{2, 1\}, 1, \{t_0/(t_0 + t_1), t_1/(t_0 + t_1)\})$ OOC for odd w: Let $w = 2m + 1$ and choose n to be a prime number such that $n = 2w^2 t_0 + w(w - 1)t_1 + 1$. If r denotes the greatest common divisor of t_0 and t_1, let α be a primitive element of GF(n) such that

- $\{\log_\alpha[\alpha^{x_j}(\alpha^{[(2m+1)t_0+mt_1]k} - 1)] : 1 \leq k \leq m, j = 0, \cdots, (t_1/r) - 1\}$,
- $\{\log_\alpha[\alpha^{z_l}(\alpha^{[(2m+1)t_0+mt_1]k} - 1)] : 1 \leq k \leq 2m + 1, l = 0, \cdots, (t_0/r) - 1\}$

are all distinct modulo $((2m+1)t_0+mt_1)/r$, where x_j and z_l are integers between 0 and $(2m + 1)t_0 + mt_1 - 1$. Then the code consisting of the blocks

$$\{[\alpha^{[(2m+1)t_0+mt_1]i/r+z_l}, \alpha^{[(2m+1)t_0+mt_1]i/r+z_l+(2m+1)t_0+mt_1}, \cdots,$$
$$\alpha^{[(2m+1)t_0+mt_1]i/r+z_l+(2m+1)(4m+1)t_0+m(4m+1)t_1}] :$$
$$i = 0, \ldots, r - 1, l = 0, \cdots, (t_0/r) - 1\}$$

and the blocks

$$\{[\alpha^{[(2m+1)t_0+mt_1]i/r+x_j}, \alpha^{[(2m+1)t_0+mt_1]i/r+x_j+2(2m+1)t_0+2mt_1}, \cdots,$$
$$\alpha^{[(2m+1)t_0+mt_1]i/r+x_j+(2m+1)(4m+1)t_0+m(4m+1)t_1}] :$$
$$i = 0, \ldots, r - 1, j = 0, \cdots, (t_1/r) - 1\}$$

form an $(n, \{2w, w\}, \{2,1\}, 1, \{t_0/(t_0+t_1), t_1/(t_0+t_1)\})$ OOC.

Some of constructions in Section 3.3.5.1 are in Table 3.3.

code	t_0	t_1	primitive α	x_0
$(97, \{6,3\}, \{2,1\}, 1, \{\frac{1}{2}, \frac{1}{2}\})$	4	4	5	1
$(241, \{6,3\}, \{2,1\}, 1, \{\frac{1}{2}, \frac{1}{2}\})$	10	10	7	1
$(409, \{6,3\}, \{2,1\}, 1, \{\frac{1}{2}, \frac{1}{2}\})$	17	17	21	3
$(457, \{6,3\}, \{2,1\}, 1, \{\frac{1}{2}, \frac{1}{2}\})$	19	19	13	1
$(211, \{10,5\}, \{2,1\}, 1, \{\frac{1}{2}, \frac{1}{2}\})$	3	3	2	1

Table 3.3: The cardinality of codes constructed according to the technique of Section 3.3.5.1

3.3.5.2. Constructions of an $(n, \{2w+1, w\}, \{2,1\}, 1, Q)$ OOC for odd w

An $(n, \{2w+1, w\}, \{2,1\}, 1, \{t_0/(t_0+t_1), t_1/(t_0+t_1)\})$ OOC for odd w: Let $w = 2m+1$ and choose n to be a prime number such that $n = (2w^2 + 2w)t_0 + w(w-1)t_1 + 1$. If r denotes the greatest common divisor of t_0 and t_1, let α be a primitive element of GF(n) such that

- $\{\log_\alpha[\alpha^{x_j}(\alpha^{[(2m+2)t_0+mt_1]k} - 1)] : 1 \le k \le m, j = 0, \cdots, (t_1/r) - 1\}$,
- $\{\log_\alpha[\alpha^{z_l}(\alpha^{[(2m+2)t_0+mt_1]k} - 1)] : 1 \le k \le 2m+1, l = 0, \cdots, (t_0/r) - 1\}$,
- $\{z_l : l = 0, \ldots, (t_0/r - 1)\}$

are all distinct modulo $((2m+2)t_0+mt_1)/r$, where x_j and z_l are integers between 0 and $(2m+2)t_0 + mt_1 - 1$. Then the code consisting of the blocks

$$\{[0, \alpha^{[(2m+2)t_0+mt_1]i/r+z_l}, \alpha^{[(2m+2)t_0+mt_1]i/r+z_l+(2m+2)t_0+mt_1}, \cdots,$$

$$\alpha^{[(2m+2)t_0+mt_1]i/r+z_l+(2m+2)(4m+1)t_0+m(4m+1)t_1}] :$$

$$i = 0, \ldots, r-1, l = 0, \cdots, (t_0/r) - 1\}$$

and the blocks

$$\{[\alpha^{[(2m+2)t_0+mt_1]i/r+x_j}, \alpha^{[(2m+2)t_0+mt_1]i/r+x_j+2(2m+2)t_0+2mt_1}, \cdots,$$

$$\alpha^{[(2m+2)t_0+mt_1]i/r+x_j+(2m+2)(4m+1)t_0+m(4m+1)t_1}] :$$

$$i = 0, \ldots, r-1, j = 0, \cdots, (t_1/r) - 1\}$$

form an $(n, \{2w+1, w\}, \{2,1\}, 1, \{t_0/(t_0+t_1), t_1/(t_0+t_1)\})$ OOC.

3.3.6 Constructions of an $(n, \{w_0, w_1\}, \{2,2\}, 1, Q)$ OOC

We demonstrate some constructions for different w_i.

3.3.6.1. Constructions of an $(n, \{2w, w\}, \{2,2\}, 1, Q)$ OOC for even w

An $(n, \{2w, w\}, \{2,2\}, 1, \{t_0/(t_0+2t_1), 2t_1/(t_0+2t_1)\})$ OOC for even w: Let $w = 2m$ and choose n to be a prime number such that $n = 2w^2t_0 + w^2t_1 + 1$. If r denotes the greatest common divisor of t_0 and t_1, let α be a primitive element of GF(n) such that

- $\{\log_\alpha[\alpha^{x_j}(\alpha^{[2mt_0+mt_1]k} - 1)] : 1 \le k \le m, j = 0, \cdots, (t_1/r) - 1\}$,
- $\{\log_\alpha[\alpha^{z_l}(\alpha^{[2mt_0+mt_1]k} - 1)] : 1 \le k \le 2m, l = 0, \cdots, (t_0/r) - 1\}$

are all distinct modulo $(2mt_0 + mt_1)/r$, where x_j and z_l are integers between 0 and $2mt_0 + mt_1 - 1$. Then the code consisting of the blocks

$$\{[\alpha^{[2mt_0+mt_1]i/r+z_l}, \alpha^{[2mt_0+mt_1]i/r+z_l+2mt_0+mt_1}, \cdots,$$

$$\alpha^{[2mt_0+mt_1]i/r+z_l+2m(4m-1)t_0+m(4m-1)t_1}] : i = 0, \ldots, r-1, l = 0, \cdots, (t_0/r) - 1\}$$

and the blocks

$$\{[\alpha^{[2mt_0+mt_1]i/r+x_j}, \alpha^{[2mt_0+mt_1]i/r+x_j+4mt_0+2mt_1}, \cdots,$$

$$\alpha^{[2mt_0+mt_1]i/r+x_j+2m(4m-2)t_0+m(4m-2)t_1}] : i = 0, \ldots, r-1, j = 0, \cdots, (t_1/r) - 1\}$$

and the blocks

$$\{[\alpha^{[2mt_0+mt_1]i/r+x_j+2mt_0+mt_1}, \alpha^{[2mt_0+mt_1]i/r+x_j+6mt_0+3mt_1}, \cdots,$$

$$\alpha^{[2mt_0+mt_1]i/r+x_j+2m(4m-1)t_0+m(4m-1)t_1}] : i = 0, \ldots, r-1, j = 0, \cdots, (t_1/r)-1\}$$

form an $(n, \{2w, w\}, \{2,2\}, 1, \{t_0/(t_0+2t_1), 2t_1/(t_0+2t_1)\})$ OOC.

Some of constructions in Section 3.3.6.1 are in Table 3.4.

code	t_0	$2t_1$	primitive α	x_0
$(2017, \{8,4\}, \{2,2\}, 1, \{\frac{1}{3}, \frac{2}{3}\})$	42	84	5	2
$(2593, \{8,4\}, \{2,2\}, 1, \{\frac{1}{3}, \frac{2}{3}\})$	54	108	7	4

Table 3.4: The cardinality of codes constructed according to the technique of Section 3.3.6.1

3.3.6.2. Constructions of an $(n, \{2w+1, w\}, \{2,2\}, 1, Q)$ OOC for even w

An $(n, \{2w+1, w\}, \{2,2\}, 1, \{t_0/(t_0+2t_1), 2t_1/(t_0+2t_1)\})$ OOC for even w: Let $w = 2m$ and choose n to be a prime number such that $n = 2(w^2 + w)t_0 + w^2t_1 + 1$. If r denotes the greatest common divisor of t_0 and t_1, let α be a primitive element of GF(n) such that

- $\{\log_\alpha[\alpha^{x_j}(\alpha^{[(2m+1)t_0+mt_1]k} - 1)] : 1 \le k \le m, j = 0, \cdots, (t_1/r)-1\}$,
- $\{\log_\alpha[\alpha^{z_l}(\alpha^{[(2m+1)t_0+mt_1]k} - 1)] : 1 \le k \le 2m, l = 0, \cdots, (t_0/r)-1\}$,
- $\{z_l : l = 0, \cdots, (t_0/r)-1\}$

are all distinct modulo $((2m+1)t_0+mt_1)/r$, where x_j and z_l are integers between 0 and $(2m+1)t_0 + mt_1 - 1$. Then the code consisting of the blocks

$$\{[0, \alpha^{[(2m+1)t_0+mt_1]i/r+z_l}, \alpha^{[(2m+1)t_0+mt_1]i/r+z_l+(2m+1)t_0+mt_1}, \cdots,$$

$$\alpha^{[(2m+1)t_0+mt_1]i/r+z_l+(2m+1)(4m-1)t_0+m(4m-1)t_1}]:$$

$$i=0,\ldots,r-1, l=0,\cdots,(t_0/r)-1\}$$

and the blocks

$$\{[\alpha^{[(2m+1)t_0+mt_1]i/r+x_j}, \alpha^{[(2m+1)t_0+mt_1]i/r+x_j+(4m+2)t_0+2mt_1},\cdots,$$

$$\alpha^{[(2m+1)t_0+mt_1]i/r+x_j+(2m+1)(4m-2)t_0+m(4m-2)t_1}]:$$

$$i=0,\ldots,r-1, j=0,\cdots,(t_1/r)-1\}$$

and the blocks

$$\{[\alpha^{[(2m+1)t_0+mt_1]i/r+x_j+(2m+1)t_0+mt_1}, \alpha^{[(2m+1)t_0+mt_1]i/r+x_j+(6m+3)t_0+3mt_1},\cdots,$$

$$\alpha^{[(2m+1)t_0+mt_1]i/r+x_j+(2m+1)(4m-1)t_0+m(4m-1)t_1}]:$$

$$i=0,\ldots,r-1, j=0,\cdots,(t_1/r)-1\}$$

form an $(n,\{2w+1,w\},\{2,2\},1,\{t_0/(t_0+2t_1),2t_1/(t_0+2t_1)\})$ OOC.

Some of constructions in Section 3.3.6.2 are in Table 3.5.

code	t_0	$2t_1$	primitive α	x_0
$(337,\{9,4\},\{2,2\},1,\{\frac{1}{3},\frac{2}{3}\})$	6	12	10	1
$(449,\{9,4\},\{2,2\},1,\{\frac{1}{3},\frac{2}{3}\})$	8	16	3	10
$(2521,\{9,4\},\{2,2\},1,\{\frac{1}{3},\frac{2}{3}\})$	45	90	17	6

Table 3.5: The cardinality of codes constructed according to the technique of Section 3.3.6.2

3.3.6.3. Constructions of an $(n,\{2w+1,w+1\},\{2,2\},1,Q)$ OOC for even w

An $(n,\{2w+1,w+1\},\{2,2\},1,\{t_0/(t_0+2t_1),2t_1/(t_0+2t_1)\})$ OOC for even w: Let $w=2m$ and choose n to be a prime number such that $n=$

$2(w^2+w)t_0+(w^2+2w)t_1+1$. If r denotes the greatest common divisor of t_0 and t_1, let α be a primitive element of GF(n) such that

- $\{\log_\alpha[\alpha^{x_j}(\alpha^{[(2m+1)t_0+(m+1)t_1]k}-1)] : 1 \le k \le m, j = 0, \cdots, (t_1/r)-1\}$,
- $\{x_l : l = 0, \cdots, (t_1/r)-1\}$,
- $\{\log_\alpha[\alpha^{z_l}(\alpha^{[(2m+1)t_0+(m+1)t_1]k}-1)] : 1 \le k \le 2m, l = 0, \cdots, (t_0/r)-1\}$,
- $\{z_l : l = 0, \cdots, (t_0/r)-1\}$

are all distinct modulo $((2m+1)t_0+(m+1)t_1)/r$, where x_j and z_l are integers between 0 and $(2m+1)t_0+(m+1)t_1-1$. Then the code consisting of the blocks

$$\{[0, \alpha^{[(2m+1)t_0+(m+1)t_1]i/r+z_l}, \alpha^{[(2m+1)t_0+(m+1)t_1]i/r+z_l+(2m+1)t_0+(m+1)t_1}, \cdots,$$
$$\alpha^{[(2m+1)t_0+(m+1)t_1]i/r+z_l+(2m+1)(4m-1)t_0+(m+1)(4m-1)t_1}] :$$
$$i = 0, \ldots, r-1, l = 0, \cdots, (t_0/r)-1\}$$

and the blocks

$$\{[0, \alpha^{[(2m+1)t_0+(m+1)t_1]i/r+x_j}, \alpha^{[(2m+1)t_0+(m+1)t_1]i/r+x_j+(4m+2)t_0+2(m+1)t_1}, \cdots,$$
$$\alpha^{[(2m+1)t_0+(m+1)t_1]i/r+x_j+(2m+1)(4m-2)t_0+(m+1)(4m-2)t_1}] :$$
$$i = 0, \ldots, r-1, j = 0, \cdots, (t_1/r)-1\}$$

and the blocks

$$\{[0, \alpha^{[(2m+1)t_0+(m+1)t_1]i/r+x_j+(2m+1)t_0+(m+1)t_1},$$
$$\alpha^{[(2m+1)t_0+(m+1)t_1]i/r+x_j+(6m+3)t_0+3(m+1)t_1}, \cdots,$$
$$\alpha^{[(2m+1)t_0+(m+1)t_1]i/r+x_j+(2m+1)(4m-1)t_0+(m+1)(4m-1)t_1}] :$$
$$i = 0, \ldots, r-1, j = 0, \cdots, (t_1/r)-1\}$$

form an $(n, \{2w+1, w+1\}, \{2,2\}, 1, \{t_0/(t_0+2t_1), 2t_1/(t_0+2t_1)\})$ OOC.

Some of constructions in Section 3.3.6.3 are in Table 3.6.

code	t_0	$2t_1$	primitive α	x_0
$(41, \{5,3\}, \{2,2\}, 1, \{\frac{1}{3}, \frac{2}{3}\})$	2	4	6	4
$(61, \{5,3\}, \{2,2\}, 1, \{\frac{1}{3}, \frac{2}{3}\})$	3	6	2	4
$(101, \{5,3\}, \{2,2\}, 1, \{\frac{1}{3}, \frac{2}{3}\})$	5	10	2	4
$(181, \{5,3\}, \{2,2\}, 1, \{\frac{1}{3}, \frac{2}{3}\})$	9	18	2	4
$(769, \{9,5\}, \{2,2\}, 1, \{\frac{1}{3}, \frac{2}{3}\})$	12	24	11	2

Table 3.6: The cardinality of codes constructed according to the technique of Section 3.3.6.3

3.3.7 Constructions of an $(\mathbf{n}, \{\mathbf{w_0}, \mathbf{w_1}, \mathbf{w_2}\}, \{\mathbf{2,2,2}\}, \mathbf{1}, \mathbf{Q})$ OOC

Instead of combining two kinds of OOC's, what we show here is combining the three kinds of OOC's to construct an $(n, \{w_0, w_1, w_2\}, \{2,2,2\}, 1, \{t_0/(t_0+2t_1+2t_2), 2t_1/(t_0+2t_1+2t_2), 2t_2/(t_0+2t_1+2t_2))$ OOC.

An $(\mathbf{n}, \{\mathbf{2w+1, w+1, w}\}, \{\mathbf{2,2,2}\}, \mathbf{1}, \{\mathbf{t_0/(t_0+2t_1+2t_2), 2t_1/(t_0+2t_1+2t_2), 2t_2/(t_0+2t_1+2t_2)})$ OOC for even w: Let $w = 2m$ and choose n to be a prime number such that $n = 2(w^2+w)t_0 + (w^2+2w)t_1 + w^2t_2 + 1$. If r denotes the greatest common divisor of t_0, t_1 and t_2, let α be a primitive element of GF(n) such that all the following,

- $\{\log_\alpha[\alpha^{s_j}(\alpha^{[(2m+1)t_0+(m+1)t_1+mt_2]k} - 1)] : 1 \le k \le m, j = 0, \cdots, (t_2/r) - 1\}$;
- $\{\log_\alpha[\alpha^{x_j}(\alpha^{[(2m+1)t_0+(m+1)t_1+mt_2]k} - 1)] : 1 \le k \le m, j = 0, \cdots, (t_1/r) - 1\}$;
- $\{x_l : l = 0, \cdots, (t_1/r) - 1\}$;
- $\{\log_\alpha[\alpha^{z_l}(\alpha^{[(2m+1)t_0+(m+1)t_1+mt_2]k} - 1)] : 1 \le k \le 2m, l = 0, \cdots, (t_0/r) - 1\}$;
- $\{z_l : l = 0, \cdots, (t_0/r) - 1\}$,

are all distinct modulo $((2m+1)t_0 + (m+1)t_1 + mt_2)/r$, where s_j, x_j, and z_l are integers between 0 and $(2m+1)t_0 + (m+1)t_1 + mt_2 - 1$. Then the code

consisting of the blocks

$$\{[0, \alpha^{[(2m+1)t_0+(m+1)t_1+mt_2]i/r+z_l},$$

$$\alpha^{[(2m+1)t_0+(m+1)t_1+mt_2]i/r+z_l+(2m+1)t_0+(m+1)t_1+mt_2}, \cdots,$$

$$\alpha^{[(2m+1)t_0+(m+1)t_1+mt_2]i/r+z_l+(2m+1)(4m-1)t_0+(m+1)(4m-1)t_1+m(4m-1)t_2}] :$$

$$i = 0, \ldots, r-1, l = 0, \cdots, (t_0/r) - 1\}$$

and the blocks

$$\{[0, \alpha^{[(2m+1)t_0+(m+1)t_1+mt_2]i/r+x_j},$$

$$\alpha^{[(2m+1)t_0+(m+1)t_1+mt_2]i/r+x_j+(4m+2)t_0+2(m+1)t_1+2mt_2}, \cdots,$$

$$\alpha^{[(2m+1)t_0+(m+1)t_1+mt_2]i/r+x_j+(2m+1)(4m-2)t_0+(m+1)(4m-2)t_1+m(4m-2)t_2}] :$$

$$i = 0, \ldots, r-1, j = 0, \cdots, (t_1/r) - 1\}$$

and the blocks

$$\{[0, \alpha^{[(2m+1)t_0+(m+1)t_1+mt_2]i/r+x_j+(2m+1)t_0+(m+1)t_1+mt_2},$$

$$\alpha^{[(2m+1)t_0+(m+1)t_1+mt_2]i/r+x_j+(6m+3)t_0+3(m+1)t_1+3mt_2}, \cdots,$$

$$\alpha^{[(2m+1)t_0+(m+1)t_1+mt_2]i/r+x_j+(2m+1)(4m-1)t_0+(m+1)(4m-1)t_1+m(4m-1)t_2}] :$$

$$i = 0, \ldots, r-1, j = 0, \cdots, (t_1/r) - 1\}$$

and the blocks

$$\{[\alpha^{[(2m+1)t_0+(m+1)t_1+mt_2]i/r+s_j},$$

$$\alpha^{[(2m+1)t_0+(m+1)t_1+mt_2]i/r+s_j+(4m+2)t_0+2(m+1)t_1+2mt_2}, \cdots,$$

$$\alpha^{[(2m+1)t_0+(m+1)t_1+mt_2]i/r+s_j+(2m+1)(4m-2)t_0+(m+1)(4m-2)t_1+m(4m-2)t_2}] :$$

$$i = 0, \ldots, r-1, j = 0, \cdots, (t_2/r) - 1\}$$

and the blocks

$$\{[\alpha^{[(2m+1)t_0+(m+1)t_1+mt_2]i/r+s_j+(2m+1)t_0+(m+1)t_1+mt_2},$$

$$\alpha^{[(2m+1)t_0+(m+1)t_1+mt_2]i/r+s_j+(6m+3)t_0+3(m+1)t_1+3mt_2}, \cdots,$$

$$\alpha^{[(2m+1)t_0+(m+1)t_1+mt_2]i/r+s_j+(2m+1)(4m-1)t_0+(m+1)(4m-1)t_1+m(4m-1)t_2]}:$$

$$i = 0, \ldots, r-1, j = 0, \cdots, (t_2/r) - 1\}$$

form an $(n, \{2w+1, w+1, w\}, \{2,2,2\}, 1, \{t_0/(t_0+2t_1+2t_2), 2t_1/(t_0+2t_1+2t_2), 2t_2/(t_0+2t_1+2t_2))$ OOC.

Some of constructions in Section 3.3.7 are in Table 3.7.

code	t_0	$2t_1$	$2t_2$	primitive α	s_0	x_0
$(313, \{5,3,2\}, \{2,2,2\}, 1, \{\frac{1}{5}, \frac{2}{5}, \frac{2}{5}\})$	13	26	26	10	1	3
$(337, \{5,3,2\}, \{2,2,2\}, 1, \{\frac{1}{5}, \frac{2}{5}, \frac{2}{5}\})$	14	28	28	10	1	3
$(409, \{5,3,2\}, \{2,2,2\}, 1, \{\frac{1}{5}, \frac{2}{5}, \frac{2}{5}\})$	17	34	34	21	1	3

Table 3.7: The cardinality of codes constructed according to the technique of Section 3.3.7

3.4 Performance Analysis

In [39], Azizoğlu *et. al* analyzed the performance of an $(n, w, 1, 1)$ OOC in a CDMA network by considering the probability distribution of the interference pattern. They showed the probability of error in the hard-limiting case when the threshold is w is given by

$$P_e = \frac{1}{2}\sum_{i=0}^{w}(-1)^i \binom{w}{i}(1-\frac{qi}{w})^{M-1}, \tag{3.1}$$

where M is the total number of users and $q = \frac{w^2}{2n}$ is the probability that a pulse belonging to a particular user overlaps with one of the pulses of the desired user.

Here, we will generalize equation (3.1) to obtain the performance analysis of an $(n, \{A\}, \{1, \ldots, 1\}, 1, Q)$ variable weight OOC for the hard-limiting case.

Claim: The probability of error for the hard-limiting case for user j with weight w_j is,

$$P_{e_j} = \frac{1}{2}\{\sum_{i=0}^{w_j-1}(-1)^i\binom{w_j}{i}[(1-\frac{q_j i}{w_j})]^{M_j-1}\prod_{\substack{l=0\\l\neq j}}^{p}[(1-\frac{q_l i}{w_j})]^{M_l} + (-1)^{w_j}(1-q_j)^{M_j-1}\prod_{\substack{l=0\\l\neq j}}^{p}(1-q_l)^{M_l}\}, \quad (3.2)$$

where $M_l(l \in \{0,1,\ldots,p\})$ is the total number of users with weight w_l and $q_l = \frac{w_l w_j}{2n}(l \in \{0,1,\ldots,p\})$ is the probability that a pulse belonging to a user with weight l overlaps with one of the pulses of the desired user with weight w_j.

Proof: See Appendix E.

Figure 3.2 shows the error probability as a function of w (=threshold) for an $(n, \{w+1, w\}, \{1,1\}, 1, \{1/2, 1/2\})$ variable weight OOC with different values of M, where $M_0 = M_1 = M/2$.

3.5 Conclusions

In this chapter we derived new upper and lower bounds on the number of users that can be supported on an optical network employing code division multiple access with OOC signature sequences. Unlike previous work in this area, we considered the possibility that the weight of each codeword might not be identical. We then demonstrated techniques for constructing $(n, \{w+1, w\}, \{1,1\}, 1, Q)$, $(n, \{w+1, w\}, \{2,2\}, 1, Q)$, $(n, \{2w, w\}, \{2,1\}, 1, Q)$, $(n, \{2w+1, w\}, \{2,1\}, 1, Q)$, $(n, \{2w, w\}, \{2,2\}, 1, Q)$, $(n, \{2w+1, w\}, \{2,2\}, 1, Q)$, $(n, \{2w+1, w+1\}, \{2,2\}, 1, Q)$, and $(n, \{2w+1, w+1, w\}, \{2,2,2\}, 1, Q)$ OOC's. Furthermore, we have analyzed the performance of these OOC's. The results allow us to assign codewords with different weights among the users. Changing the weight of a user's

signature sequence affects that user's performance; therefore this approach is useful for CDMA fiber optic networks with multiple performance requirements among the users.

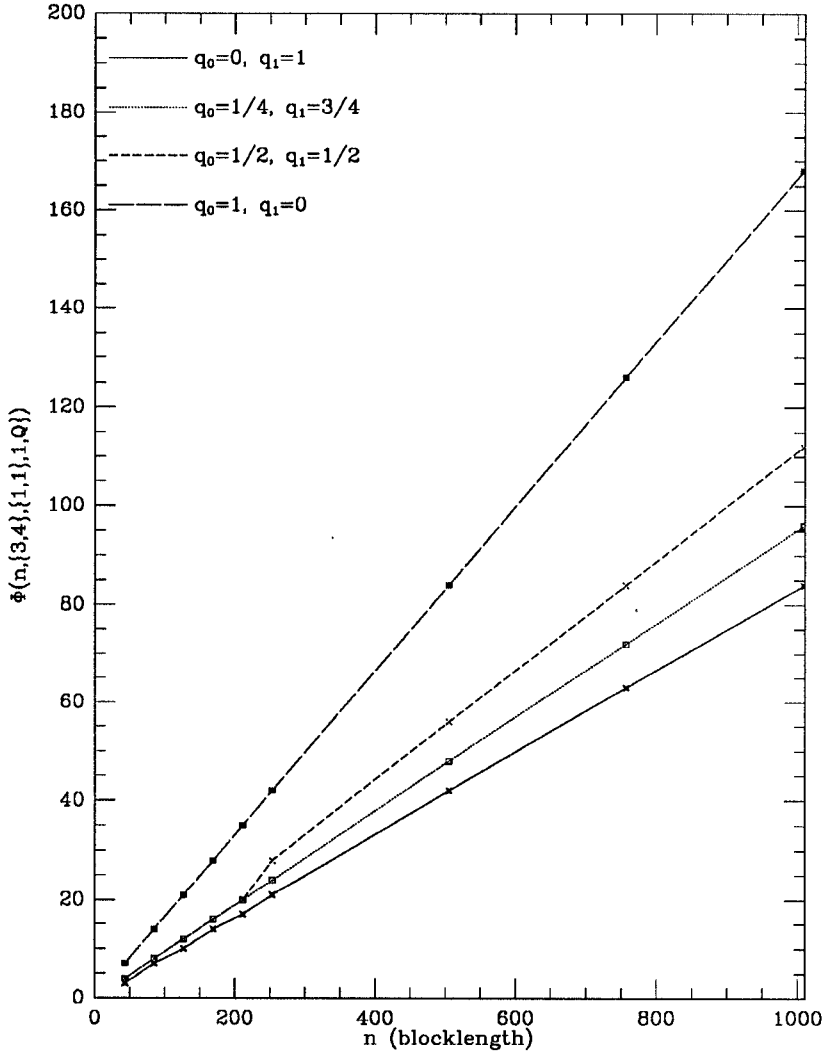


Figure 3.1: The upper bound on the $\Phi(n, \{3,4\}, \{1,1\}, 1, \{q_0, q_1\})$ variable weight OOC with different q_0, q_1.

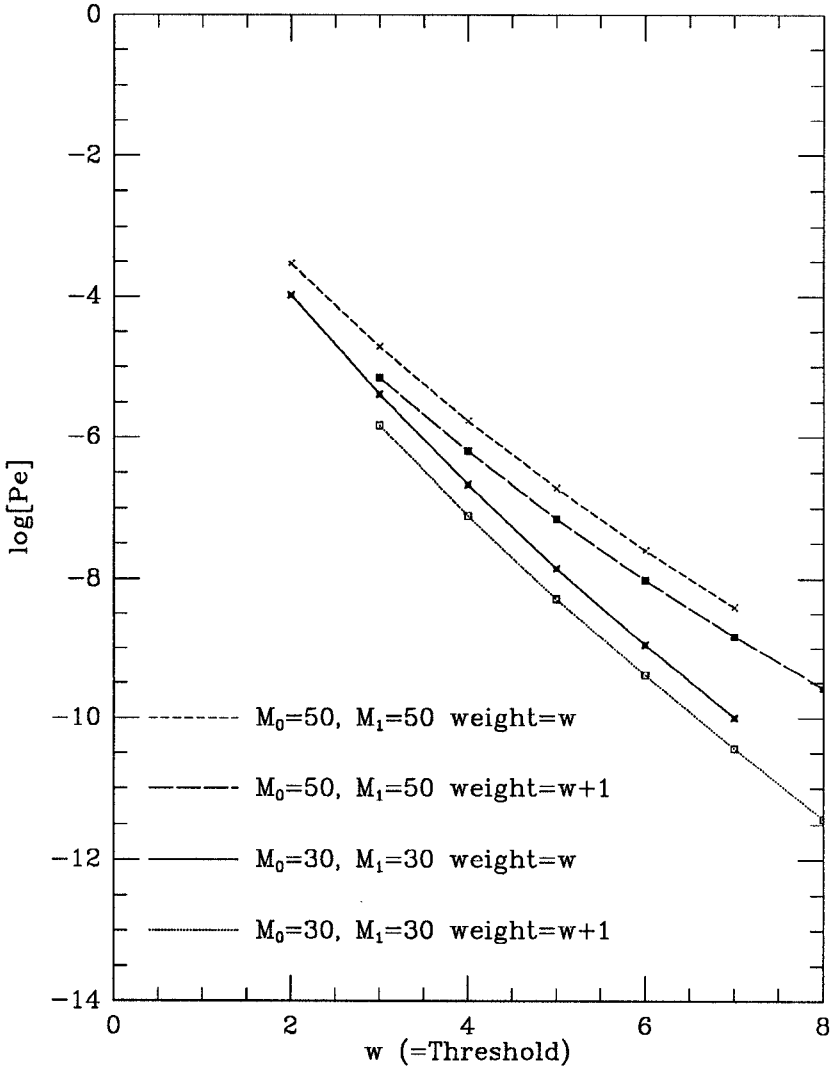


Figure 3.2: The error probability as a function of w for an $(n, \{w+1, w\}, \{1, 1\}, 1, \{1/2, 1/2\})$ variable weight OOC with different values of M.

Chapter 4

Concatenated Coding Scheme for Code Division Multiple Access on Optical Networks

In this chapter we consider a concatenated coding scheme consisting of a CDMA inner code and a channel code—possibily different for each user—as an outer code. We demonstrate how such a configuration offers novel tradeoffs when compared with a single CDMA code – tradeoffs including performance/rate and multiple performance requirements among the network's users.

4.1 Background and Motivation

In taking advantage of the abundant bandwidth of the optical fiber, a fiber-optical code division multiple access (CDMA) system maps the low-rate information signals onto high-rate optical sequences to achieve random, asynchronous communication access among users. The success of the CDMA systems depends on the signature sequence of each user.

The performance of the system will depend on the maximum auto- and cross-correlation, as well as the blocklength and weight of the OOC sequences as mentioned in Chapter 2. In a noiseless CDMA system, all errors will come from the interference of the other users; therefore one means of enhancing a

user's performance is to increase his codeword weight while maintaining the same auto- and cross-correlation constraints. (This was investigated in Chapter 3.) However, if we use $\Phi(n, w, \lambda_a, \lambda_c)$ to denote the cardinality of an optimal optical orthogonal code with given auto-correlation λ_a and cross-correlation λ_c, then it was shown in Chapter 2 that $\Phi(n, w, \lambda_a, \lambda_c) \leq \frac{(n-1)...(n-\lambda_c)(\lambda_a-\lambda_c+1)}{w(w-1)...(w-\lambda_c)}$. Therefore, when $\lambda_a = \lambda_c = 1$, $n \geq w(w-1)\Phi(n, w, 1, 1) + 1$; i.e., blocklength n will increase at least as fast as $O(w^2)$ with weight w. In this chapter, we consider a concatenated coding scheme which can enhance the performance of the system without increasing the weight (and thus the blocklength) of the OOC. By considering the original CDMA system as the inner code and adding a channel encoder for each user as an outer code, we will show this concatenated coding scheme can increase the information rate, compared with the original system. Furthermore, by using a different outer code for each user, we can meet the performance requirements for users with different priorities as in Chapter 3.

4.2 Performance Analysis

Suppose we use a rate $R = k/n$ error-correcting code as an outer code and an $(n_1 = w(w-1)t+1, w, 1, 1)$ OOC as an inner code. The rate R_c of the resulting system is given by

$$R_c = \frac{k}{nn_1} = \frac{k}{n[w(w-1)t+1]},$$

where t is the number of users in the system. A special case of this is an "uncoded" system – i.e., one in which no error control code is present, in which case the overall rate is

$$R_u = \frac{1}{n_1} = \frac{1}{w(w-1)t+1}.$$

4.2.1 Uncoded System

The block diagram of a CDMA optical system is shown in Figure 4.1.

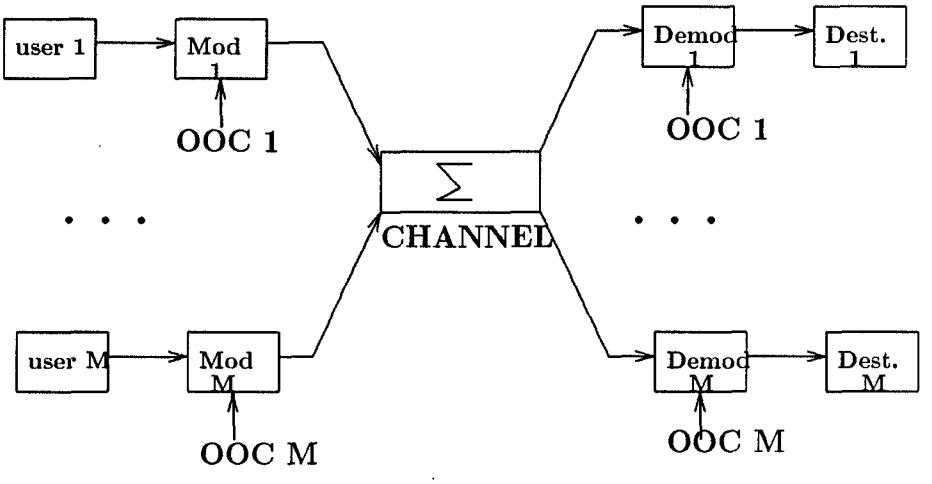


Figure 4.1: The uncoded CDMA optical system

In [39], Azizoğlu *et. al.* analyzed of the performance of an $(n, w, 1, 1)$ OOC in a CDMA network by considering the probability distribution of the interference pattern. They showed the probability of error in a hard-limiting case when the threshold is w is given by

$$P_b = \frac{1}{2}\sum_{i=0}^{w}(-1)^i\binom{w}{i}(1-\frac{qi}{w})^{M-1}, \tag{4.1}$$

where M is the total number of users and $q = \frac{w^2}{2n}$ is the probability that a pulse belonging to a particular user overlaps with one of the pulses of the desired user.

We note that this same formula is appropriate for an $(n, w, 2, 1)$ OOC. The constraint on the auto-correlation λ_a is only useful for the synchronization process. If we don't count the error probability due to the synchronization process, then the error probability of the system will be the same as $(n, w, 1, 1)$ OOC. Let t denote the number of codewords of the $(n, w, 2, 1)$ OOC described in Chapter

2, then the block length n is $n = w^2t/2+1$ for an even w and $n = (w^2-1)t/2+1$ for an odd w.

4.2.2 Coded System

The block diagram of a concatenated CDMA optical system is shown in Figure 4.2.

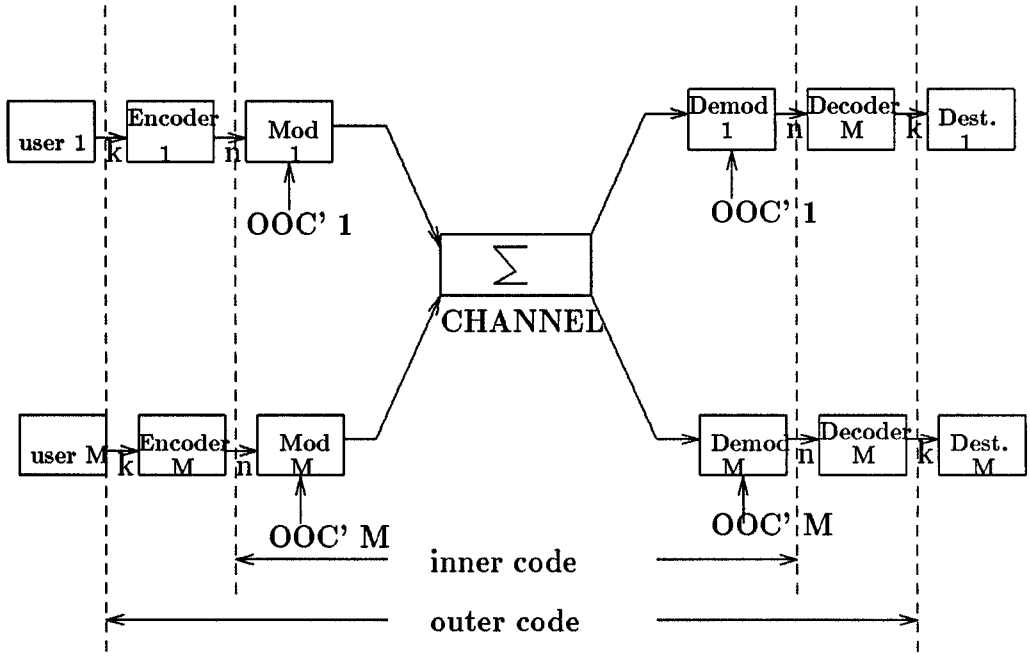


Figure 4.2: The coded CDMA optical system

4.2.2.1 (n, k) t-Error Correcting Block Code

If we use an (n, k) t-error correcting code as an outer code, then the performance of the whole system can be derived as follows. Let P_e be the error probability for the received bit in the inner code. Then, P_w, the probability of receiving an

n-bit codeword in error if we assume bound-distance decoding is given by

$$P_w = \sum_{i=t+1}^{n} \binom{n}{i} P_e^i (1 - P_e)^{n-i}.$$

The bit error probability of the system can be approximated by the post-decoding bit error rate P_b which assumes that all incorrect decoding events produce $i + t$ errors[40]. Then

$$P_b \approx \frac{1}{n} \sum_{i=t+1}^{n} (i + t) \binom{n}{i} P_e^i (1 - P_e)^{n-i}.$$

4.2.2.2 (n, k) t-Asymmetric Error Correcting Code

Since we assume the only errors come from the interference of the other users, all channel errors must be of the $0 \rightarrow 1$ variety. In other words, we only consider a so called 0-type error. The more natural role for the outer code is therefore an (n, k) t-asymmetric error correcting code. Let $\mathcal{C}$ be a binary block code of length n. The asymmetric distance $d_A(\mathbf{x}, \mathbf{y})$ between $\mathbf{x} = (x_1, x_2, \ldots, x_n) \in \mathcal{C}$ and $\mathbf{y} = (y_1, y_2, \ldots, y_n) \in \mathcal{C}$ is determined by

$$d_A(\mathbf{x}, \mathbf{y}) = \max\{N(\mathbf{x}, \mathbf{y}), N(\mathbf{y}, \mathbf{x})\},$$

where $N(\mathbf{x}, \mathbf{y}) = |\{i | x_i = 1 \text{ and } y_i = 0\}|$. The relation between the Hamming distance $d_H(\mathbf{x}, \mathbf{y})$ and the asymmetric distance $d_A(\mathbf{x}, \mathbf{y})$ is

$$2d_A(\mathbf{x}, \mathbf{y}) = d_H(\mathbf{x}, \mathbf{y}) + |w(\mathbf{x}) - w(\mathbf{y})|,$$

where $w(\mathbf{x})$ and $w(\mathbf{y})$ are the weights of $\mathbf{x}$ and $\mathbf{y}$. The asymmetric distance $\triangle$ of a code $\mathcal{C}$ is defined by

$$\triangle = \min\{d_A(\mathbf{x}, \mathbf{y}) | \mathbf{x}, \mathbf{y} \in \mathcal{C}; \mathbf{x} \neq \mathbf{y}\}.$$

It has been proven that a code $\mathcal{C}$ of asymmetric distance $\triangle$ can correct $\triangle - 1$ or fewer asymmetric errors[41].

Thus, P_w, the probability of receiving n-bits codeword in errors, is

$$P_w \leq \frac{1}{|\mathcal{C}|} \sum_{j=0}^{n} |\mathcal{A}_j| \sum_{i=t+1}^{n-j} \binom{n-j}{i} P_e^i (1-P_e)^{n-j-i},$$

where $\mathcal{A}_j = \{\mathbf{x} \in \mathcal{C} : wt(\mathbf{x}) = j\}$ is the set of codewords in $\mathcal{C}$ with weight equal to j.

The bit error probability of the system can be approximated by the post-decoding bit error rate P_b which assumes that all incorrect decoding events produce $i + t$ errors[40]. Then

$$P_b \leq \frac{1}{n|\mathcal{C}|} \sum_{j=0}^{n} |\mathcal{A}_j| \sum_{i=t+1}^{n-j} (i+t) \binom{n-j}{i} P_e^i (1-P_e)^{n-j-i}.$$

Example: An (2,1,1) asymmetric code

We encode information bit 0 to 00, and information bit 1 to 11. It is a one error correcting asymmetric code. Thus,

$$P_b = P_w = P_e^2/2.$$

4.2.2.3 (n, k) Convolutional Code

Consider the use of convolutional codes with maximum likelihood decoding using the Viterbi algorithm with hard decisions. We only consider the $(2, 1)$ convolutional codes with constraint lengths 5 and 7[42]. Thus, for constraint length 5,

$$\begin{aligned} P_b \leq\ & 4D^7 + 12D^8 + 20D^9 + 72D^{10} + 225D^{11} + 500D^{12} + 1324D^{13} \\ & + 3680D^{14} + 8967D^{15} + 22270D^{16}, \end{aligned}$$

and, for constraint length 7,

$$P_b = 36D^{10} + 211D^{12} + 1404D^{14} + 11633D^{16} + 76628D^{18} + 469991D^{20},$$

where $D = 2\sqrt{P_e(1-P_e)}$.

4.2.2.4 Different Priority

Instead of using (n, W, L, λ_c, Q) variable weight OOC to achieve the performance requirements for users with different priorities as in Chapter 3, we may use different outer codes for different users in the concatenated coding scheme and keep the inner codes OOCs with the same weight.

4.3 Numerical Results

In Figure 4.3, we compare an uncoded system using an $(n, 4, 1, 1)$ OOC with two coded systems. For the coded system, we use a $(7, 4)$ one-error and a $(21, 12)$ two-error correcting BCH codes as outer codes and an $(n, 3, 1, 1)$ OOC as an inner code. The performances of both coded systems are better than the uncoded system; furthermore, the ratio of the rate of a coded system to the uncoded system is

$$\frac{k}{n} \times \frac{12t+1}{6t+1} \approx \frac{2k}{n}.$$

Since $k/n > 1/2$ for each outer code, the overall rate is better. Thus the coded systems outperform the uncoded system in Figure 4.3.

In Figure 4.4, we use an $(n, 5, 1, 1)$ OOC for the uncoded system. For the coded system, we use $(23, 12)$ and $(15, 5)$ three-error correcting codes as outer codes and $(n, 3, 1, 1)$ OOC as an inner code. Once again the performances of

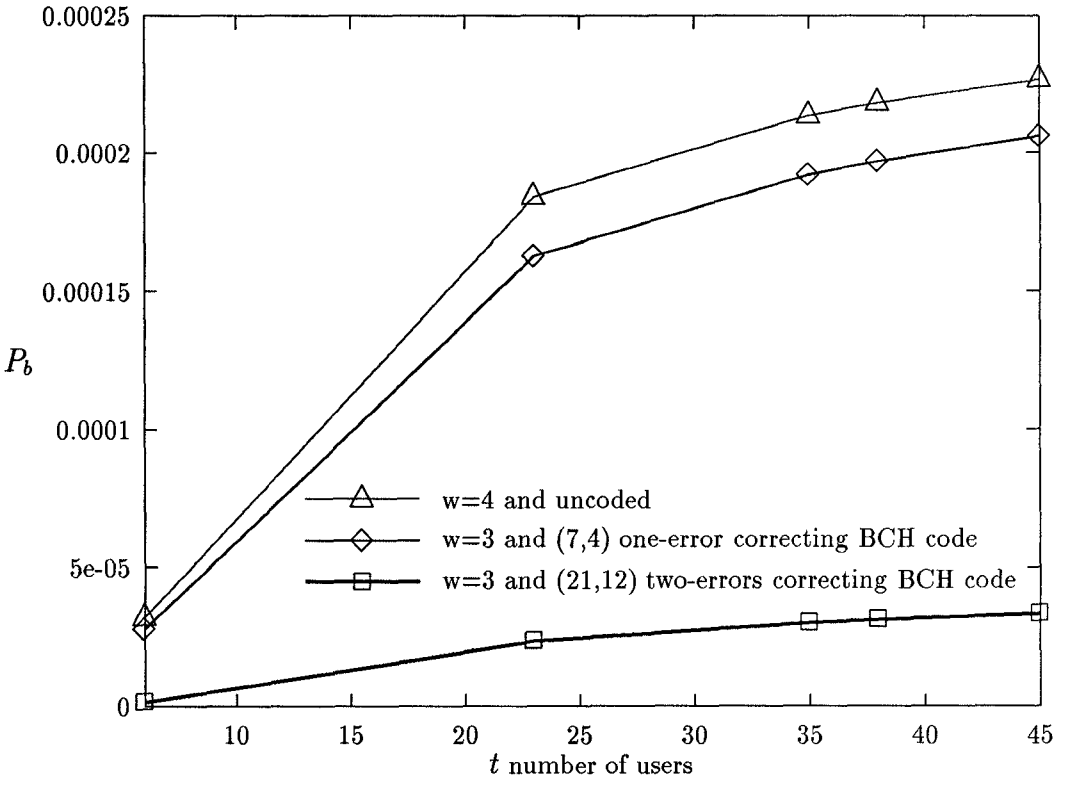


Figure 4.3: Comparison between uncoded system and symmetric BCH coded systems

both coded systems are better than the uncoded system, and the rate ratios are

$$\frac{k}{n} \times \frac{20t+1}{6t+1} \approx \frac{10k}{3n} > 1.$$

For both cases in Figure 4.4, the coded systems outperform the uncoded system.

In Figure 4.5, we use an $(n, 6, 2, 1)$ OOC for the uncoded system. For the coded system, we use $(21, 12)$ two-error and $(23, 12)$ three-error correcting BCH codes as outer codes and $(n, 4, 2, 1)$ OOC as an inner code. Again the performances of both coded systems are better than the uncoded system, and the rate ratios are

$$\frac{k}{n} \times \frac{18t+1}{8t+1} \approx \frac{9k}{4n} > 1.$$

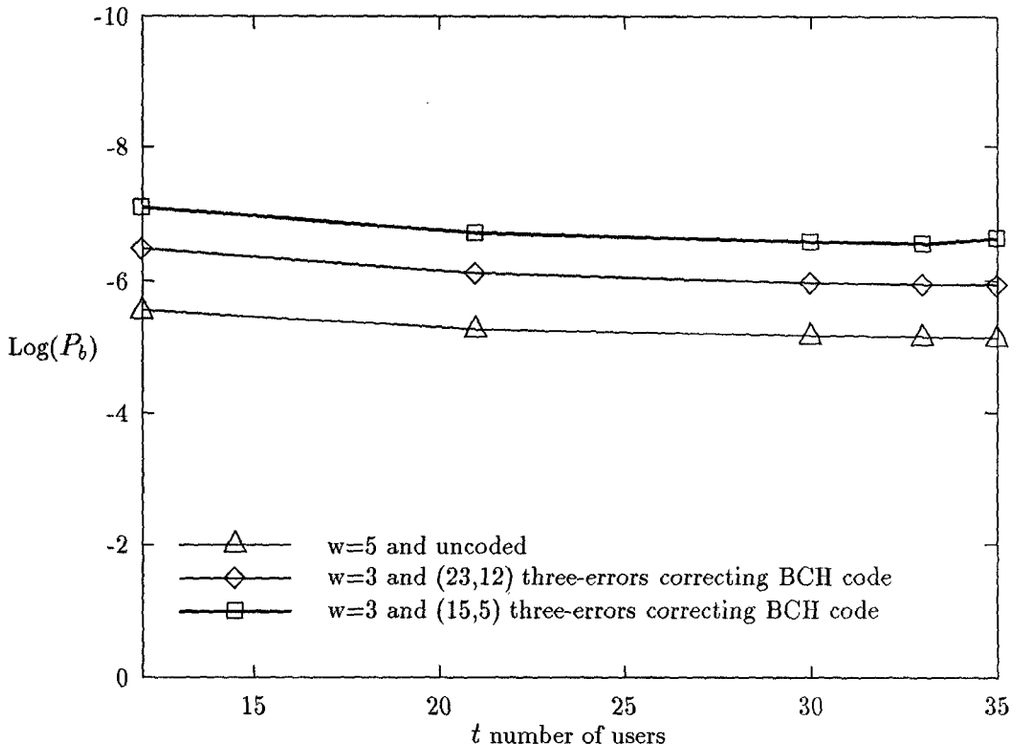


Figure 4.4: Comparison between uncoded system and symmetric BCH coded systems

Then the coded systems outperform the uncoded system in Figure 4.5.

In Figure 4.6, we use an $(n, 6, 2, 1)$ OOC for the uncoded system. For the coded system, we use $(2, 1)$ one-asymmetric error corecting code as an outer code and $(n, 4, 2, 1)$ OOC as an inner code. The performance of the coded system is better than the uncoded system, and the rate ratio is

$$\frac{k}{n} \times \frac{18t+1}{8t+1} \approx \frac{9k}{4n} > 1.$$

Then the coded system outperforms the uncoded system.

In Figure 4.7, we use $(n, 4, 1, 1)$ and $(n, 5, 1, 1)$ OOCs for the uncoded systems. For the coded systems, we use $(2, 1)(N = 5)$ and $(2, 1)(N = 7)$ con-

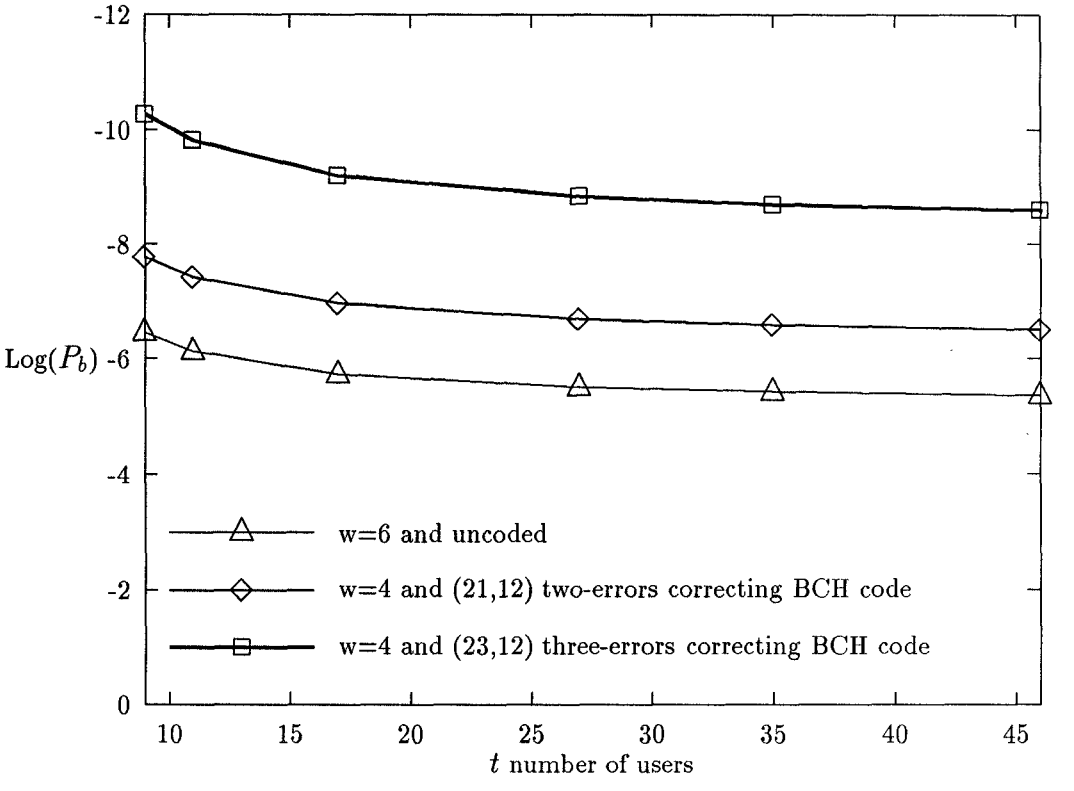


Figure 4.5: Comparison between uncoded system and symmetric BCH coded systems

volutional codes as outer codes and $(n, 3, 1, 1)$ OOC as an inner code. The performances of both coded systems are better than the uncoded system using an $(n, 5, 1, 1)$ OOC, and the rate ratios are

$$\frac{k}{n} \times \frac{20t+1}{6t+1} \approx \frac{10k}{3n} > 1.$$

Then the coded systems outperform the uncoded system. The performances of both coded systems are better than the uncoded system using an $(n, 4, 1, 1)$ OOC, and the rate ratios are

$$\frac{k}{n} \times \frac{12t+1}{6t+1} \approx \frac{2k}{n} \approx 1.$$

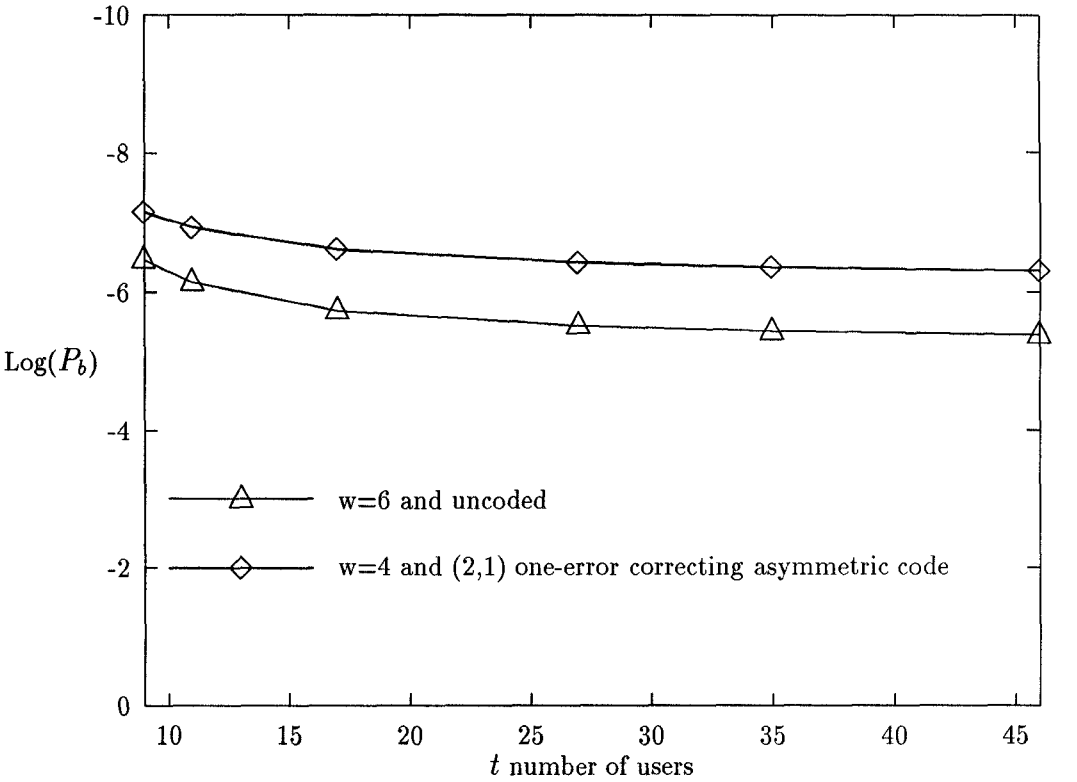


Figure 4.6: Comparison between uncoded system and asymmetric coded system

Then the coded systems outperform the uncoded system.

4.4 Conclusion

In this chapter, we have presented a concatenated coding scheme which may perform better than an uncoded CDMA optical system. Furthermore, we have demonstrated the simplicity of adjusting the system to satisfy the users with different priorities. The concatenated coding scheme is useful for many CDMA optical systems.

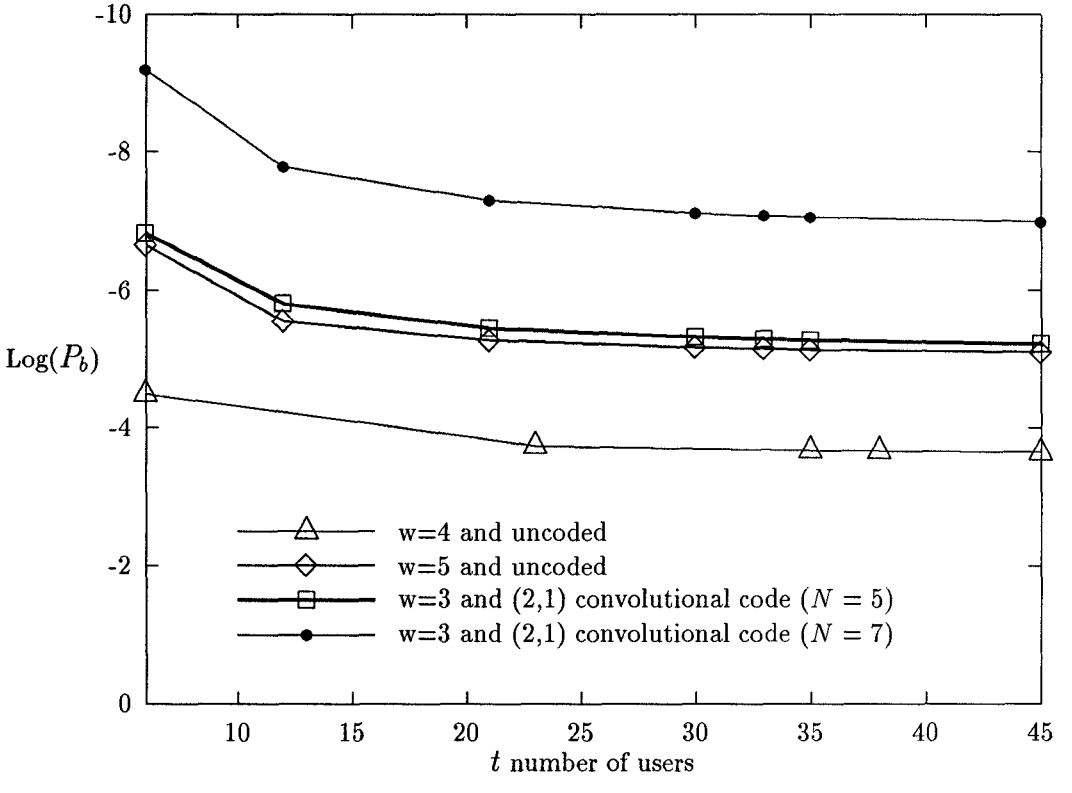


Figure 4.7: Comparison between uncoded systems and symmetric convolutional coded systems

Chapter 5

Signature Pairs for Synchronization and CDMA on Optical Fiber Networks Using Pulse Position Modulation

This chapter considers a new approach to code division multiple access (CDMA) on fiber-optical networks employing pulse position modulation (PPM). We propose assigning to each user in the network two signature sequences – a synchronization sequence that the user will use to establish frame synchronization and a data sequence that will be used to convey information once frame synchronization is established.

Previous work on CDMA for optical channels has assumed the assignment of a single signature sequence to each user. In some designs – e.g., prime sequences – the auto-correlation of each signature sequences is high, creating possible problems with respect to synchronization. In other designs – e.g., optical orthogonal codes – the auto-correlation of sequences is kept low; however, this property is useless once frame synchronization is maintained, and it limits the number of signature sequences that can be constructed.

Our approach requires a set of synchronization sequences with good auto-

correlation properties as well as a set of data sequences with good cross-correlation properties; in addition, good "global" correlation properties – e.g. constraints on the cross-correlation between data and synchronization sequences – is necessary. We demonstrate how this can be attained using extended quadratic congruence (QC) sequences for synchronization and extended prime sequences as data sequences.

5.1 Background and Motivation

The increasing use of optical communications has fostered a renewed interest in both pulse position modulation (PPM) and code division multiple access (CDMA). This paper describes a new approach to designing and using binary sequences for CDMA-PPM.

In a "typical" optical CDMA system[6][7], each user is assigned a distinct n-bit "signature sequence" that the user transmits when a logical "1" is to be conveyed; when a logical "0" is to be conveyed, an all-zero sequence is transmitted. This set of signature sequences is designed to satisfy two constraints:

- **Auto-correlation property:** Each sequence can easily be distinguished from a shifted version of itself. This property is used to enable the receiver to obtain *synchronization* – that is, to find the beginning of its message so that subsequent frame boundaries can be located.
- **Cross-correlation property:** Each sequence can be easily distinguished from a shifted version of every other sequence in the set. This property enables the receiver to estimate its message in the presence of interference from other users. Thus, the cross-correlation constraint aids both synchronization in the presence of multiple users *and* permits each receiver to "track" its message after synchronization is achieved.

Therefore, the auto-correlation constraint contributes only to synchronization, while the cross-correlation constraint affects both synchronization *and* operation. Furthermore, we observe that once synchronization is obtained – something that needs to be done relatively rarely – the auto-correlation property of a sequence becomes useless.

This suggests that a "natural" way to implement a CDMA system is to assign *two* different signature sequences to each user: one that the user will employ when attempting to obtain synchronization ("synchronization sequence"), and another that will be used when synchronization is already obtained and the user wishes to convey a message ("data sequence"). The necessary correlation properties can be described as follows:

(1) The set of synchronization sequences must have good auto-correlation properties.

(2) The set of data sequences must have good cross-correlation properties.

(3) The cross-correlation between any synchronization sequence and any data sequence must be low, to permit synchronization in the presence of multiple users.

(4) The cross-correlation between two different synchronization sequences may be of secondary importance, since this plays a role only when two different users are attempting to obtain synchronization simultaneously – possibly a relatively rare occurrence.

(5) The auto-correlation properties of the data sequences are unimportant.

Our model of an optical network is the one found in [6][7]. We assume that each user transmits a *logical* bit by transmitting a frame of n *channel* bits (or *chips*); if a user wishes to send a logical "1" then that user's n-bit signature sequence is transmitted, while if a logical "0" is to be conveyed then a frame of n

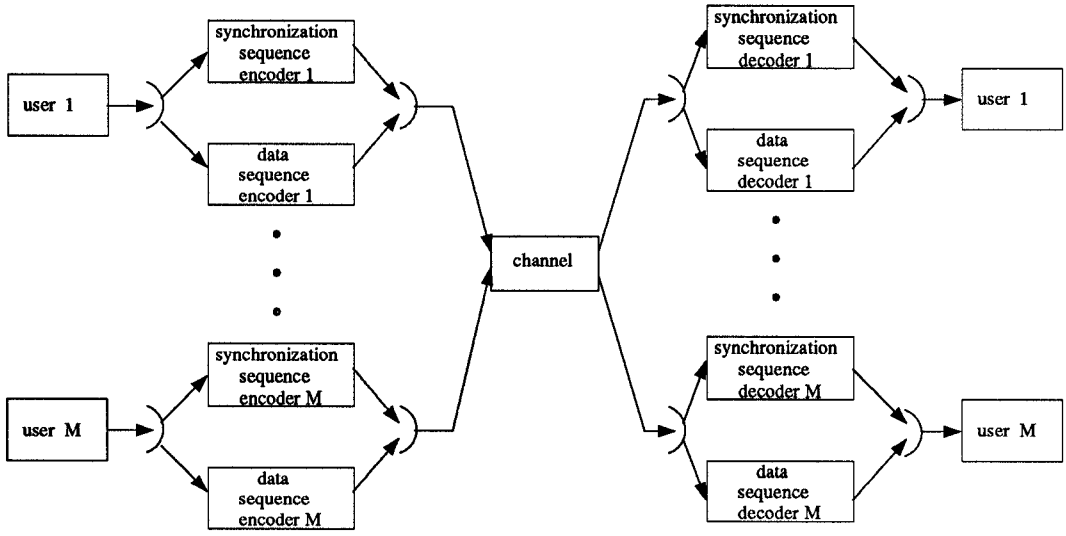


Figure 5.1: The system block diagram for two sequence scheme.

zeroes is sent. We assume that the optical signals are processed asynchronously, which means that the receiver can only determine the intensity of the received signals; this in turn means that the channel acts like a "sum channel", with the output intensity at any time being directly proportional to the number of users transmitting a channel bit at that time.

In the case of CDMA with pulse position modulation, there is one additional constraint. Each logical "1" is transmitted as a frame of n channel bits; we assume that $n = w \cdot m$, with each frame consisting of w sub-frames of m bits each. Furthermore, we require that only one bit in each sub-frame can be a "1"; this constraint describes PPM, in which information is conveyed by the position of the pulse (i.e. – the "1") within the sub-frame.

The whole system block diagram is shown in Figure 5.1.

The challenge, then, is to find a set of synchronization sequences and a set of data sequences – each a set of binary n-tuples, where $n = w \cdot m$ – that

have the necessary correlation constraints *and* have a single "1" in each m-bit sub-block. This paper demonstrates that an extension of Shaar and Davies "prime sequences" is a good candidate for the set of data sequences while Maric's extended quadratic codes make suitable synchronization sequences.

5.2 Some Upper Bounds

We begin by deriving the upper bound on the cardinality of the number of users which can be supported in our system.

Denote by $\mathcal{Z}(n, w_1, w_2, \lambda_a^1, \lambda_a^2, \lambda_c^1, \lambda_c^2, \lambda_c^3)$ the maximum number of users we can support if we assign each user an $(n, w_1, \lambda_a^1, \lambda_c^1)$ synchronization sequence and an $(n, w_2, \lambda_a^2, \lambda_c^2)$ data sequence, and in addition we require that the cross-correlation between any synchronization sequence and any data sequence be at most λ_c^3 – i.e.,

$$\mathcal{Z}(n, w_1, w_2, \lambda_a^1, \lambda_a^2, \lambda_c^1, \lambda_c^2, \lambda_c^3) \triangleq \max\{\min(|\mathcal{X}|, |\mathcal{Y}|): \mathcal{X} \text{ is an } (n, w_1, \lambda_a^1, \lambda_c^1) \text{ OOC}, \mathcal{Y} \text{ is an } (n, w_2, \lambda_a^2, \lambda_c^2) \text{ OOC, cross-correlation between any } x \in \mathcal{X} \text{ and } y \in \mathcal{Y} \text{ is at most } \lambda_c^3\}$$

Let $|\mathcal{X}^*|$ and $|\mathcal{Y}^*|$ be the OOC's that are optimal – i.e., they're the codes that achieve the maximum above. From Theorem 2.3, we can bound $|\mathcal{X}^*|$ and $|\mathcal{Y}^*|$ as follows:

$$|\mathcal{X}^*| \leq \frac{(n-1)(n-2)\ldots(n-\lambda_c^1)(\lambda_a^1 - \lambda_c^1 + 1)}{w_1(w_1-1)(w_1-2)\ldots(w_1-\lambda_c^1)}, \tag{5.1}$$

and

$$|\mathcal{Y}^*| \leq \frac{(n-1)(n-2)\ldots(n-\lambda_c^2)(\lambda_a^2 - \lambda_c^2 + 1)}{w_2(w_2-1)(w_2-2)\ldots(w_2-\lambda_c^2)}. \tag{5.2}$$

Furthermore, the requirement that the cross-correlation between any two element of $\mathcal{X}$ and $\mathcal{Y}$ cannot exceed λ_c^3 yields the following inequality:

$$\begin{aligned} |\mathcal{X}^*| * \frac{w_1(w_1-1)(w_1-2)\ldots(w_1-\lambda_c^3)}{\lambda_a^1-\lambda_c^3+1} \\ + |\mathcal{Y}^*| * \frac{w_2(w_2-1)(w_2-2)\ldots(w_2-\lambda_c^3)}{\lambda_a^2-\lambda_c^3+1} \\ \leq (n-1)(n-2)\ldots(n-\lambda_c^3). \end{aligned} \tag{5.3}$$

When $\lambda_c^1 = \lambda_c^2 = \lambda_c^3 = \lambda$, the three inequalities degenerate into one inequality, which is

$$\mathcal{Z}(n, w_1, w_2, \lambda_a^1, \lambda_a^2, \lambda, \lambda, \lambda) \leq \frac{(n-1)(n-2)\ldots(n-\lambda)}{\frac{w_1(w_1-1)(w_1-2)\ldots(w_1-\lambda)}{\lambda_a^1-\lambda+1} + \frac{w_2(w_2-1)(w_2-2)\ldots(w_2-\lambda)}{\lambda_a^2-\lambda+1}} \tag{5.4}$$

Note that this bound is not specific to PPM – i.e., we have not required that each sub-frame contain exactly one "1".

5.3 Sequence Design

In this section we begin by defining a code for pulse position modulation.

Definition: An $(n, w, \lambda_a, \lambda_c)$ PPM code $\mathcal{C}$ is a collection of binary n-tuples, each of Hamming weight w where $w|n$, such that the following three properties hold:

- (PPM Property) For any $\mathbf{x} = [x_0, x_1, \ldots, x_{n-1}] \in \mathcal{C}$ and any $k = 0, 1, \ldots, w-1$, the Hamming weight of the vector $[x_{\frac{kn}{w}}, x_{\frac{kn}{w}+1}, \ldots, x_{\frac{kn}{w}+\frac{n}{w}-1}]$ is one.

- (Auto-correlation Property) For any $\mathbf{x} = [x_0, x_1, \ldots, x_{n-1}] \in \mathcal{C}$ and any integer τ, $0 < \tau < n$,

$$\sum_{t=0}^{n-1} x_t x_{t\oplus\tau} \leq \lambda_a.$$

- (Cross-correlation Property) For any $\mathbf{x} = [x_0, x_1, \ldots, x_{n-1}] \in \mathcal{C}$ and any $\mathbf{y} = [y_0, y_1, \ldots, y_{n-1}] \in \mathcal{C}$ such that $\mathbf{x} \neq \mathbf{y}$ and any integer τ,

$$\sum_{t=0}^{n-1} x_t y_{t\oplus\tau} \leq \lambda_c.$$

Note: Following [6][7], we define PPM sequences in terms of periodic correlation; thus the addition in the subscripts above – denoted "$\oplus$" – is all modulo-n.

5.3.1 Construction of Prime Sequences

In [15] Shaar and Davies demonstrated a technique for constructing a $(p^2, p, p, 2)$ PPM code for any prime number p; since it is our intention to use an extension of the Shaar/Davies code as our set of data sequences, we briefly review their construction.

The approach is to first construct a prime sequence $S_i = (s_{i,0}, s_{i,1}, \ldots, s_{i,p-1})$ consisting of p prime numbers and then use this sequence to place p 1's inside the p sub-blocks. Let $\mathcal{F}(p) = \{0, 1, \ldots, j, \ldots, p-1\}$, denote the Galois Field with p elements; then

$$s_{i,j} = i \cdot j \in \mathcal{F}(p),$$

where "$\cdot$" denotes multiplication modulo p. Each prime sequence S_i is then mapped onto a binary code sequence $C_i = (c_{i0}, c_{i1}, \ldots, c_{ij}, \ldots, c_{i(p^2-1)})$ by the rule

$$c_{ij} = \begin{cases} 1 & \text{if } j = s_{ik} + kp, \text{ for some } k \in \{0, 1, \ldots, p-1\}, \\ 0 & \text{otherwise.} \end{cases}$$

There are p binary prime sequences of length p^2 generated by this rule.

An example $p = 5$ is shown in Table 5.1.

i	$s_{i0}s_{i1}s_{i2}s_{i3}s_{i4}$	Sequence	Binary Code Sequence
0	00000	S_0	C_0 = 10000 10000 10000 10000 10000
1	01234	S_1	C_1 = 10000 01000 00100 00010 00001
2	02413	S_2	C_2 = 10000 00100 00001 01000 00010
3	03142	S_3	C_3 = 10000 00010 01000 00001 00100
4	04321	S_4	C_4 = 10000 00001 00010 00100 01000

Table 5.1: Prime sequences S_i and binary code sequences C_i for $GF(5)$

5.3.2 Construction of Extended Prime Sequences

We now propose a new "extended prime sequence" which reduces the peak of the cross-correlation constraint to one, at the expense of increasing the blocklength p^2 of a prime sequence to $p(2p-1)$; thus, the extended prime (EP) sequence is a $(p(2p-1), p, p, 1)$ PPM sequence.

Starting with Galois field $\mathcal{F}(p)$, construct a prime sequence $S_i = (s_{i,0}, s_{i,1}, \ldots, s_{i,p-1})$ as in Section 5.2.1. Each prime sequence S_i is then mapped onto a binary code sequence $C_i = (c_{i0}, c_{i1}, \ldots, c_{ij}, \ldots, c_{i(p(2p-1))})$ by the rule

$$c_{ij} = \begin{cases} 1 & \text{if } j = s_{ik} + k(2p-1), \text{ for some } k \in \{0, 1, \ldots, p-1\}, \\ 0 & \text{otherwise.} \end{cases}$$

So the extended prime sequences are obtained be padding $p-1$ 0's to the end of each sub-block in a prime sequence.

Example $p = 5$ is shown in Table 5.2.

i	$s_{i0}s_{i1}s_{i2}s_{i3}s_{i4}$	Sequence	Binary Code Sequence
0	00000	S_0	C_0 = 100000000 100000000 100000000 100000000 100000000
1	01234	S_1	C_1 = 100000000 010000000 001000000 000100000 000010000
2	02413	S_2	C_2 = 100000000 001000000 000010000 010000000 000100000
3	03142	S_3	C_3 = 100000000 000100000 010000000 000010000 001000000
4	04321	S_4	C_4 = 100000000 000010000 000100000 001000000 010000000

Table 5.2: Extended prime sequences S_i and binary code sequences C_i for $GF(5)$

Theorem 5.1: An extended prime sequence is a $(p(2p-1), p, p, 1)$ PPM code.

Proof: For the proof we need a definition of a placement difference function[16] and two lemmas.

Let

$$y_i(k) = ik(mod\ p),\ k = 0, 1, \ldots, p-1, \tag{5.5}$$

be a placement operator. Then the i^{th} extended prime sequence $(0 \leq i \leq p-1)$ can be thought of as a two dimensional $(2p-1) \times p$ array with a "1" in the $y_i(k)$ row of the k-th column. The associated sequence is obtained by "reading" the array column by column.

The array associated with the sequence $S_2 = [02413]$ from Table 5.2 is shown in Figure 5.2.

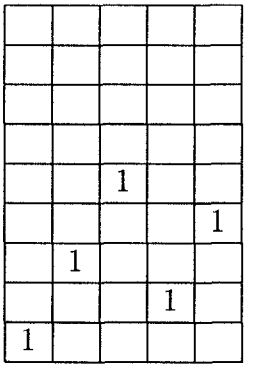


Figure 5.2: The two dimensional interpretation for EP sequence S_2 for $GF(5)$

Definition: The placement difference function for two placement operators $y_{i_1}(k)$ and $y_{i_2}(k)$ is denoted $(y_{i_1}(k) \triangle y_{i_2}(k))_{x,z}$ and is the vertical distance between the "1" in the k^{th} column of array i_2 and the "1" in the $(k+x)$ column of array i_1 after it's been shifted left x times and vertically z times, i.e.,

$$(y_{i_1}(k) \triangle y_{i_2}(k))_{x,z} = y_{i_1}(k+x) - y_{i_2}(k) - z \quad |x|, |z| \leq p-1, \tag{5.6}$$

where x and z denote an integer horizontal and vertical shift, respectively. If the value of the difference function is zero, there is a "hit".

Lemma 5.1: Two placement operators from two different extended prime sequences have at most one hit for any horizontal-vertical shift pair, (x, z) $-p+1 \leq x, z \leq p-1$.

Proof: Substituting equation (5.5) into equation (5.6) for two different extended prime sequences we see a hit occurs if and only if

$$[i_1(k+x)](mod\ p) - [i_2 k](mod\ p) - z = 0. \tag{5.7}$$

But a necessary condition for equation (5.7) to hold is

$$(i_1 - i_2)k + i_1 x - z = 0\ (mod\ p). \tag{5.8}$$

For fixed i_1, i_2, x, and z, this is a first order equation in k over $GF(p)$. Hence the placement operators of two extended prime sequences from two different extended prime sequences can have at most one hit for any two dimensional shift.

Lemma 5.2: Two placement operators from the same extended prime sequence have a maximum of p hits for any horizontal-vertical shift pair, (x, z) $-p+1 \leq x, z \leq p-1$.

Proof: This proof follows immediately from the proof of Lemma 5.1 by putting $i_1 = i_2$ in equation (5.8). Hence the placement operators of two extended prime sequences from a same extended prime sequence can have at most p hits for any two dimensional shift.

Now we are ready to prove Theorem 5.1. Each particular shift s of one binary extended prime sequence with respect to the other, can be expressed through a unique vertical and horizontal shift in the placement operator two dimensional

array – i.e.:

$$s = x \times (2p-1) + z \quad -p+1 \leq x, z \leq p-1. \tag{5.9}$$

Therefore, from Lemmas 5.1 and 5.2, we can complete the proof for Theorem 5.1. QED.

The next two sections review QC and EQC codes.

5.3.3 Constructions of QC Codes

Starting with Galois field $\mathcal{F}(p) = \{0, 1, \ldots, j, \ldots, p-1\}$, s_{ij} of a QC codeword $S_i = (s_{i0}, s_{i1}, \ldots, s_{ij}, \ldots, s_{i(p-1)})$ is constructed by

$$s_{i,j} = i \cdot \frac{j(j+1)}{2} \in \mathcal{F}(p),$$

where "$\cdot$" denotes multiplication modulo p and $i \in \mathcal{F}(p) \backslash \{0\}$. Each QC codeword S_i is then mapped into a binary code sequence $C_i = (c_{i0}, c_{i1}, \ldots, c_{ij}, \ldots, c_{i(p^2-1)})$ by the rule

$$c_{ij} = \begin{cases} 1 & \text{if } j = s_{ik} + kp, \text{ for some } k \in \{0, 1, \ldots, p-1\}, \\ 0 & \text{otherwise.} \end{cases}$$

There are $p-1$ binary QC sequences with length p^2 generated by this rule.

Example $p = 5$ is shown in Table 5.3.

i	$s_{i0}s_{i1}s_{i2}s_{i3}s_{i4}$	Sequence	Binary Code Sequence
1	01310	S_1	C_1 = 10000 01000 00010 01000 10000
2	02120	S_2	C_2 = 10000 00100 01000 00100 10000
3	03430	S_3	C_3 = 10000 00010 00001 00010 10000
4	04240	S_4	C_4 = 10000 00001 00100 00001 10000

Table 5.3: QC codewords S_i and binary code sequences C_i for $GF(5)$

Let

$$y_i(k) = i\frac{k(k+1)}{2} (mod\ p),\ k = 0, 1, \ldots, p-1, \tag{5.10}$$

be a placement operator for QC code. The i^{th} QC codeword $(1 \leq i \leq p-1)$ can also be thought of as a two dimensional $p \times p$ array with a "1" in the $y_i(k)$ row of the k-th column.

Marić has shown a QC code is a $(p^2, p, 2, 4)$ PPM code[16].

5.3.4 Constructions of EQC Codes

Starting with Galois field $GF(p)$, construct an EQC code $S_i = (s_{i0}, s_{i1}, \ldots, s_{ij}, \ldots, s_{i(p-1)})$ as in Section 5.3.3. Now each EQC code S_i is then mapped onto a binary code sequence $C_i = (c_{i0}, c_{i1}, \ldots, c_{ij}, \ldots, c_{i(p(2p-1))})$ by the rule

$$c_{ij} = \begin{cases} 1 & \text{if } j = s_{ik} + k(2p-1), \text{ for some } k \in \{0, 1, \ldots, p-1\}, \\ 0 & \text{otherwise.} \end{cases}$$

Once again the EQC sequences are obtained be padding $p-1$ 0's to the end of each sub-block in a prime sequence.

Example $p = 5$ is shown in Table 5.4.

i	$s_{i0}s_{i1}s_{i2}s_{i3}s_{i4}$	Sequence	Binary Code Sequence
1	01310	S_1	C_1 = 100000000 010000000 000100000 010000000 100000000
2	02120	S_2	C_2 = 100000000 001000000 010000000 001000000 100000000
3	03430	S_3	C_3 = 100000000 000100000 000010000 000100000 100000000
4	04240	S_4	C_4 = 100000000 000010000 001000000 000010000 100000000

Table 5.4: EQC codewords S_i and binary code sequences C_i for $GF(5)$

Let

$$y_i(k) = i\frac{k(k+1)}{2}(mod\ p),\ k = 0, 1, \ldots, p-1, \tag{5.11}$$

be a placement operator for an EQC code. Like the other codes, the i^{th} EQC codeword $(1 \leq i \leq p-1)$ can be thought of as a two dimensional $(2p-1) \times p$ array with a "1" in the $y_i(k)$ row of the k-th column.

Marić has shown an EQC code is a $(p(2p-1), p, 1, 2)$ PPM code[16].

In the next section, we are going to show the maximum cross-correlation between an EP sequence and an EQC sequence is two; while for a prime sequence and a QC sequence, it is four.

5.3.5 Cross-Correlation between Synchronization and Data Signature Sequences

Theorem 5.2: The maximum cross-correlation between an EP sequence and an EQC sequence is two.

Proof: Two placement operators – one from an EP sequence and one from an EQC sequence – have at most two hits for any horizontal-vertical shift pair, (x, z) $-p+1 \leq x, z \leq p-1$. This is because by substituting equations (5.5) and (5.11) into equation (5.6) for two sequences from EP sequences and EQC sequences we see a hit occurs if and only if

$$[i_1 \frac{(k+x)(k+x+1)}{2}](mod\ p) - [i_2 k](mod\ p) - z = 0. \tag{5.12}$$

But a necessary condition for equation (5.12) to hold is

$$(i_1 \frac{k^2}{2} + (i_1 x + \frac{i1}{2} - i_2)k + \frac{i_1 x^2}{2} + \frac{i_1 x}{2}) - z = 0\ (mod\ p). \tag{5.13}$$

For fixed i_1, i_2, x, and z, this is a second order equation in k over $GF(p)$. Hence two placement operators for two sequences from EP sequences and EQC sequences can have at most two hits for any two dimensional shift.

Each particular shift s of one binary extended prime sequence with respect to one binary EQC code, can be expressed through a unique vertical and horizontal shift in the placement operator two dimensional array as in equation (5.9), i.e.,

$$s = x \times (2p-1) + z \quad -p+1 \leq x, z \leq p-1. \tag{5.14}$$

Therefore we can complete the proof for Theorem 5.2. QED.

The placement operator for a prime sequence is defined the same as in equation (5.5),

$$y_i(k) = ik(mod\ p),\ k = 0, 1, \ldots, p-1. \tag{5.15}$$

Then a i^{th} prime sequence ($0 \leq i \leq p-1$) can be thought as two dimensional $p \times p$ array with a "1" in the $y_i(k)$ row of the k-th column.

Theorem 5.3: The maximum cross-correlation between a prime sequence and a QC sequence is four.

Proof: Two placement operators – one from a prime sequence and one from a QC sequence – have at most two hits for any horizontal-vertical shift pair, (x, z) $-p+1 \leq x, z \leq p-1$. This is because by substituting equations (5.15) and (5.10) into equation (5.6) for two sequences from prime sequence and QC code we have:

$$[i_1 \frac{(k+x)(k+x+1)}{2}](mod\ p) - [i_2 k](mod\ p) - z = 0. \tag{5.16}$$

But a necessary condition for equation (5.16) to hold is

$$i_1 \frac{k^2}{2} + (i_1 x + \frac{i_1}{2} - i_2)k + \frac{i_1 x^2}{2} + \frac{i_1 x}{2} - z = 0\ \ (mod\ p). \tag{5.17}$$

For fixed i_1, i_2, x, and z, this is a second order equation in k over $GF(p)$. Hence two placement operators for two sequences from prime sequences and QC sequences can have at most two hits for any two dimensional shift.

Each particular shift s of one binary prime sequence with respect to one binary QC codeword, can be expressed through two vertical and horizontal shifts in the placement operator two dimensional array, i.e.,

$$s = x \times p + z\ \ 0 \leq z \leq p-1, \tag{5.18}$$

and

$$s = (x+1) \times p + z' \quad -p+1 \leq z' \leq 0. \tag{5.19}$$

Furthermore, the inner product of the prime sequence/QC codeword pair will be equal to the number of distinct "hits" encountered in the two corresponding two-dimensional shifts – a number that is at worst four. Thus we complete the proof for Theorem 5.3. QED.

In the next section we are going to derive some upper bounds on the cardinalities of PPM code.

5.4 Some Upper Bounds for PPM Code

In Chapter 2 we derived upper bounds on the cardinalities of optical orthogonal code by defining two kinds of sets with respect to the auto- and cross-correlation properties then placing a constraint on the cross-correlation-associated set[11]. Here, we want to use the same technique to derive upper bounds for PPM codes. Since there is one more constraint for the PPM modulation scheme compared with on-off keying, i.e., a single "1" per subframe, we are dealing with a more complex problem than in Chapter 2.

Define $\phi(n, w, \lambda_a, \lambda_c)$ as the cardinality of an optimal PPM code with the given parameters – i.e.,

$$\phi(n, w, \lambda_a, \lambda_c) \triangleq \max\{|\,\mathcal{C}\,|: \mathcal{C} \text{ is an } (n, w, \lambda_a, \lambda_c) \text{ PPM code}\}.$$

Throughout this section, we use q to denote the subframe length. Obviously, $n = wq$.

Definition: Let $\mathbf{x}$ be a binary n-tuple of weight w in a PPM code and let $\mathbf{t_x} = [t_0, t_1, \ldots, t_{w-1}]$ be its adjacent relative delay vector.

The PPM requirement places constraints on the delay vectors expressed as follows.

$$1 \le t_i \le 2q-1, \text{ for } i = 0, \cdots, w-1,$$

$$q+1 \le t_i + t_{i\uplus 1} \le 3q-1, \text{ for } i = 0, \cdots, w-1,$$

$$\vdots$$

$$(w-2)q+1 \le t_i + t_{i\uplus 1} + \cdots + t_{i\uplus w-2} \le wq-1, \text{ for } i = 0, \cdots, w-1.$$

(Note: The subscript addition above is modulo w – denoted "$\uplus$".)

The first bound, for the case $\lambda_c = 1$, is identical to the bound in Theorem 2.2 for OOC.

Theorem 5.4: The following inequality holds:

$$\phi(n, w, \lambda, 1) \le \frac{(n-1)\lambda}{w(w-1)}.$$

Proof: Let $\mathcal{C}$ be an $(n, w, \lambda, 1)$ PPM code such that $|\mathcal{C}| = \phi(n, w, \lambda, 1)$. By Lemma 2.4, for any $\mathbf{x} \in \mathcal{C}$, $|M_{\mathbf{x},1}| \ge w(w-1)/\lambda$. Furthermore, by Lemma 2.2 $M_{\mathbf{x},1}$ and $M_{\mathbf{y},1}$ are disjoint for $\mathbf{x}, \mathbf{y} \in \mathcal{C}$ and $\mathbf{x} \ne \mathbf{y}$. Therefore the union of $M_{\mathbf{x},1}$ as $\mathbf{x}$ varies over all $\mathbf{x} \in \mathcal{C}$ consists of at least $\phi(n, w, \lambda, \lambda) \cdot w(w-1)/\lambda$ distinct integer one-tuple. However, by definition if $[a_0] \in M_{\mathbf{x},1}$ then $a_0 \le n-1$. The number of ways to select a_0 that is no more than $n-1$ and satisfy the PPM requirement is as follows. Recall that

$$a_0 = \sum_{k=0}^{i} t_{j\uplus k} \; 0 \le i \le w-2, j = 0, 1, \cdots, w-1.$$

Considering the constraints placed above on the delay vector $[t_0, t_1, \ldots, t_{w-1}]$, it means

$$\begin{aligned} a_0 \in & \{1, 2, \ldots, 2q-1\} \cup \{q+1, q+2, \ldots, 3q-1\} \cup \ldots \\ & \cup \{(w-2)q+1, (w-2)q+2, \ldots, wq-1\} \\ \in & \{1, 2, \ldots, wq-1\}. \end{aligned}$$

Therefore the number of ways to choose a_0 is $n-1$. So:

$$\begin{aligned} n-1 &\geq \sum_{\mathbf{x}\in\mathcal{C}} |M_{\mathbf{x},1}| \\ &\geq \phi(n,w,\lambda,1)\cdot \min\{|M_{\mathbf{x},1}| : \mathbf{x}\in\mathcal{C}\} \\ &\geq \phi(n,w,\lambda,1)\cdot \frac{w(w-1)}{\lambda} \end{aligned}$$

which was to be proven. QED.

The second bound is for the case $\lambda_c = 2$.

Theorem 5.5: If $\lambda \geq 2$, then the following inequality holds:

$$\phi(n,w,\lambda,2) \leq \frac{\{(n-1)(n-2)-3q(q-1)\}(\lambda-1)}{w(w-1)(w-2)}.$$

Proof: Let $\mathcal{C}$ be an $(n,w,\lambda,2)$ PPM code such that $|\mathcal{C}| = \phi(n,w,\lambda,2)$. By Lemma 2.4, for any $\mathbf{x}\in\mathcal{C}$, $|M_{\mathbf{x},2}| \geq w(w-1)(w-2)/2(\lambda-1)$. Furthermore, by Lemma 2.2 $M_{\mathbf{x},2}$ and $M_{\mathbf{y},2}$ are disjoint for $\mathbf{x},\mathbf{y}\in\mathcal{C}$ and $\mathbf{x}\neq\mathbf{y}$. Therefore the union of $M_{\mathbf{x},2}$ as $\mathbf{x}$ varies over all $\mathbf{x}\in\mathcal{C}$ consists of at least $\phi(n,w,\lambda,2)\cdot w(w-1)(w-2)/2\lambda$ distinct integer two-tuples. However, by definition if $[a_0,a_1]\in M_{\mathbf{x},2}$ then $a_0+a_1\leq n-1$. The number of ways to select $[a_0,a_1]$ such that a_0+a_1 is no more than $n-1$ and the PPM requirement is satisfied is as follows.

$$[a_0,a_1] = [\sum_{k_0=0}^{i_0} t_{j\uplus k_0}, \sum_{k_1=i_0+1}^{i_1} t_{j\uplus k_1}]\ 0\leq i_0<i_1\leq w-2, j=0,1,\cdots,w-1$$

with respect to the constraints placed on the delay vector t_is, it means a_0 and a_1 must satisfy the following four equations.

$$a_0+a_1 \geq q+1, \tag{5.20}$$

$$n-a_0 \geq q+1, \tag{5.21}$$

$$n-a_1 \geq q+1, \tag{5.22}$$

$$n - a_0 - a_1 \geq 1. \tag{5.23}$$

From equation (5.21) for fixed $a_0 \in \{1, 2, \ldots, n-q-1\}$ the number of ways to choose a_1 is as follows.

$$\begin{aligned}
a_0 = 1, &\quad n - 2q \text{ ways to choose } a_1, \\
a_0 = 2, &\quad n - 2q + 1 \text{ ways to choose } a_1, \\
&\vdots \\
a_0 = q-1, &\quad n - q - 2 \text{ ways to choose } a_1, \\
a_0 = q, &\quad n - q - 1 \text{ ways to choose } a_1, \\
a_0 = q+1, &\quad n - q - 2 \text{ ways to choose } a_1, \\
&\vdots \\
a_0 = n-q-1, &\quad q \text{ ways to choose } a_1.
\end{aligned}$$

Therefore the number of ways to select a_0 and a_1 is

$$\frac{(n-2q+n-q-2)(q-1)}{2} + n - q - 1 + \frac{(n-q-2+q)(n-2q-1)}{2} = \frac{(n-1)(n-2) - 3q(q-1)}{2}.$$

So:

$$\begin{aligned}
\frac{(n-1)(n-2) - 3q(q-1)}{2} &\geq \sum_{\mathbf{x} \in \mathcal{C}} |M_{\mathbf{x},2}| \\
&\geq \phi(n, w, \lambda, 2) \cdot \min\{|M_{\mathbf{x},2}| : \mathbf{x} \in \mathcal{C}\} \\
&\geq \phi(n, w, \lambda, 2) \cdot \frac{w(w-1)(w-2)}{2(\lambda - 1)}
\end{aligned}$$

which was to be proven. QED.

From Theorem 5.4, we know the maximum number of users of a $(p(2p-1), p, 1, 1)$ PPM code is less than or equal to $(p(2p-1)-1)/(p(p-1))$ (≈ 2).

However, we can construct a $(p(2p-1), p, p, 1)$ EP sequence with p codewords and a $(p(2p-1), p, 1, 2)$ EQC code with $p-1$ codewords. The number of users which can be supported by the networks using our scheme, therefore, is at least $(p-1)/2$ times than one sequence system. (Since the number of codewords for synchronization signature sequences from an EQC code is $p-1$, we arbitrarily discard a codeword from an EP sequence.) This amount will be large when p is large. This can be seen in Figure 5.3.

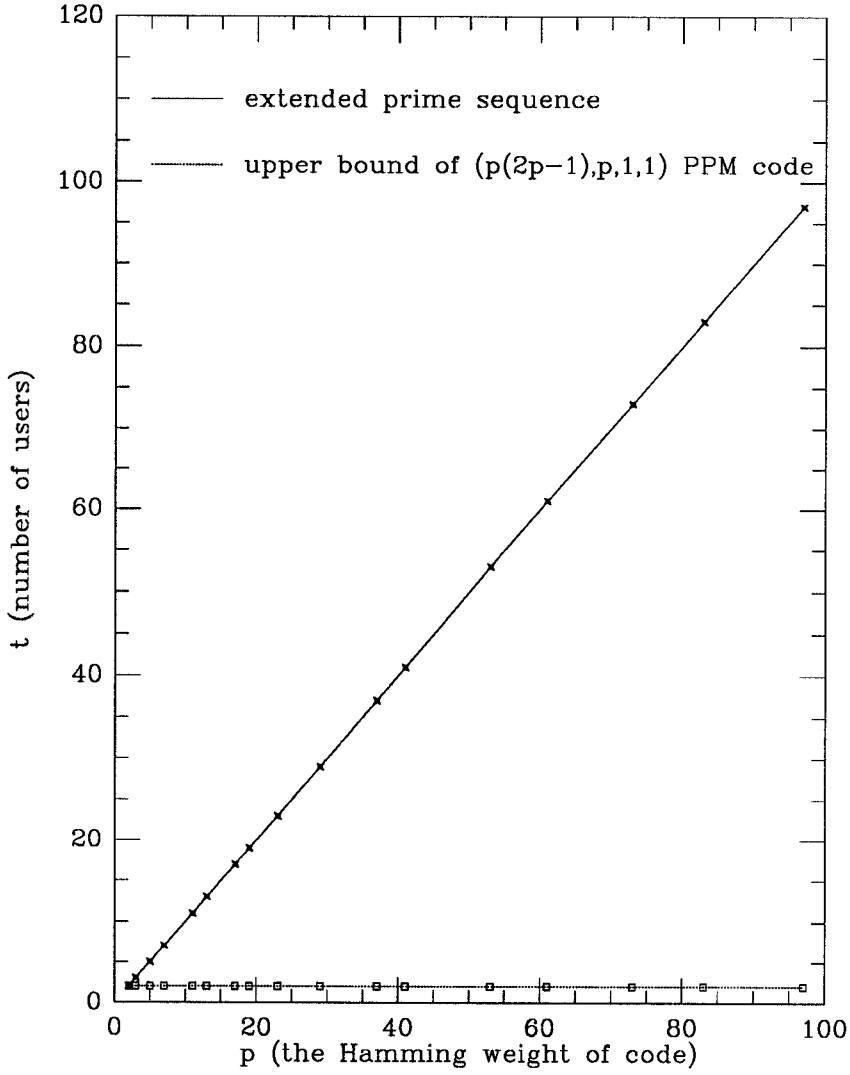


Figure 5.3: The comparison between EP sequence and the upper bound of $(p(2p-1), p, 1, 1)$ PPM code.

If we relax the maximum cross-correlation between a synchronization sequence and a data sequence to four, then we use a QC code as synchronization signature sequences and a prime sequence as data signature sequences. Compared with systems that use EQC code and EP sequence, systems using QC code and prime sequence can accommodate about $\sqrt{2}$ times more users than the former. This is because for fixed blocklength n there are $\sqrt{n}$ sequences for a prime code and $\sqrt{n/2}$ sequences for a extended prime code. This demonstrates a tradeoff between the number of users which can be supported in a system and the performance.

5.5 Discussion and Conclusion

In this chapter, we have proposed the use of two signature sequences for each user: a synchronization signature sequence to find the boundary of the codeword, and a data signature sequence to transmit data. Since a synchronization signature sequence doesn't carry any information, it is overhead—an acquisition sequence. However, depending on the ability of a system to sustain the synchronization between the transmitter and the receiver, one synchronization signature sequence usually will be accompanied by many data signature sequences. Thus, by adding an acquisition sequence, increase in the total length of sequences is very small. But we can see that the number of users that can be accommodated with this scheme is far many more than the scheme with one signature sequence for each user. This suggests that it may indeed be preferable to assign two signature sequences to each user rather than asking a single sequence to "do it all".

By mapping multibits of a user's data into a shifted version of that user's data

signature sequence as Kwon proposed for an OOC CDMA system[43], the data rate can increase sharply. By merely cyclic-shifting an EP sequence or a prime sequence, as many as $p(2p-1)$ or p^2 code sequences (except $S_0 = (0, 0, \ldots, 0)$) could be generated respectively, which would result in a significant increase in data rate to $\lfloor log_2 p(2p-1) \rfloor$ or $\lfloor log_2 p^2 \rfloor$ respectively. However, since not every cyclic-shifted version of a EP sequence or a prime sequence is a valid sequence under PPM constraints, the number of valid cyclic-shifted version of a EP sequence or a prime sequence is p. Then, the increase in data rate is $\lfloor log_2 p \rfloor$. The p cyclic-shifted versions of a EP sequence or a prime sequence are derived as follows. Each EP sequence or prime sequence S_i is taken as a seed from which a group of new code sequences can be generated. A valid cyclic-shifted version of a sequence $S_{im} = (s_{im0}, s_{im1}, \ldots, s_{imj}, \ldots, s_{im(p-1)})$ is constructed by adding every element s_{ij} from S_i by m, an element from $GF(p)$, and then reducing modulo p[44]. Each S_{im} is then mapped onto a binary code sequence by the similar rules in Section 5.3.1 for prime sequence or Section 5.3.2 for EP sequence.

Example $p = 5$ for prime sequences is shown in Table 5.5.

i	$s_{im0}s_{im1}s_{im2}s_{im3}s_{im4}$	Sequence	Binary Code Sequence
0	00000	S_{00}	C_{00} = 10000 10000 10000 10000 10000
	11111	S_{01}	C_{01} = 01000 01000 01000 01000 01000
	22222	S_{02}	C_{02} = 00100 00100 00100 00100 00100
	33333	S_{03}	C_{03} = 00010 00010 00010 00010 00010
	44444	S_{04}	C_{04} = 00001 00001 00001 00001 00001
1	01234	S_{10}	C_{10} = 10000 01000 00100 00010 00001
	12340	S_{11}	C_{11} = 01000 00100 00010 00001 10000
	23401	S_{12}	C_{12} = 00100 00010 00001 10000 01000
	34012	S_{13}	C_{13} = 00010 00001 10000 01000 00100
	40123	S_{14}	C_{14} = 00001 10000 01000 00100 00010
2	02413	S_{20}	C_{20} = 10000 00100 00001 01000 00010
	13024	S_{21}	C_{21} = 01000 00010 10000 00100 00001
	24130	S_{22}	C_{22} = 00100 00001 01000 00010 10000
	30241	S_{23}	C_{23} = 00010 10000 00100 00001 01000
	41302	S_{24}	C_{24} = 00001 01000 00010 10000 00100
3	03142	S_{30}	C_{30} = 10000 00010 01000 00001 00100
	14203	S_{31}	C_{31} = 01000 00001 00100 10000 00010
	20314	S_{32}	C_{32} = 00100 10000 00010 01000 00001
	31420	S_{33}	C_{33} = 00010 01000 00001 00100 10000
	42031	S_{34}	C_{34} = 00001 00100 10000 00010 01000
4	04321	S_{40}	C_{40} = 10000 00001 00010 00100 01000
	10432	S_{41}	C_{41} = 01000 10000 00001 00010 00100
	21043	S_{42}	C_{42} = 00100 01000 10000 00001 00010
	32104	S_{43}	C_{43} = 00010 00100 01000 10000 00001
	43210	S_{44}	C_{44} = 00001 00010 00100 01000 10000

Table 5.5: The cyclic-shifted versions of prime sequences S_{im} and binary code sequences C_{im} for $GF(5)$

Chapter 6

Conclusions and Suggestions for Future Research

This thesis concerned the design and implementation of CDMA for optical fiber networks. A variety of issues concerning the design of $(0,1)$ sequences with good correlation properties were investigated. In this chapter we review our findings and draw conclusions.

6.1 Conclusions

We started by deriving new upper and lower bounds on the number of users that can be supported on an optical network employing CDMA with OOC signature sequences. Unlike previous work in this area, we considered the possibility that the auto- and cross-correlation constraints might not be identical; indeed, it may sometimes be preferable to use such "asymmetric" OOC's to support more users in the networks. Thus more users can be provided even better performance by relaxing the auto-correlation constraint and increasing the weight of the codewords at the same time. We demonstrated three techniques for constructing $(n, w, 2, 1)$ OOC's that yield more codewords than could be possible with the parameters $(n, w-1, 1, 1)$ but which perform at least as well. The recursive construction technique used the codes which were constructed by the other two

construction techniques to provide infinite families of codes. It provides more options on the blocklength n for constructing OOC's.

We also considered the possibility that the weights of the different codewords might not be identical. We derived new upper and lower bounds on the cardinalities of such "variable weight" OOC's. We then demonstrated techniques for constructing a variety of OOC's with varying weights. Furthermore, we have analyzed the performance of these variable weight OOC's. The results allow us to assign codewords with different weights among the users to achieve the multiple performance requirements in the networks.

We have examined a concatenated coding scheme – using an error control code together with a multiple-access code – which will sometimes perform better than a "pure" CDMA optical fiber system. Using an error control code as an outer code permits us to reduce codeword weights – and blocklength – without giving up performance. Such a scheme also raises the possibility of offering multiple levels of performance by varying the outer code.

We have proposed a new CDMA scheme for use with PPM modulation. We have assigned two signature sequences to each user: a synchronization signature sequence to find the boundary of the codeword; and a data signature sequence to transmit data. The results have shown that the number of users this scheme can accommodate is – for some constraints – far larger than can be accommodated with one signature sequence for each user. Upper bounds on the cardinality of PPM codes are also derived when $\lambda_c \leq 2$. Furthermore, by mapping multiple bits of a user's data into a shifted version of the data signature sequence assigned to that user the data rate can increase sharply.

6.2 Suggestions for Future Research

There are several problems related to our work in the thesis can be extended:

- Find general constructions for $(n, w, \lambda_a, 1)$ OOC's.
- Generalize the upper and lower bounds on OOC's for codes with aperiodic correlation constraints.
- Find general sequence designs for systems which assign two signature sequences to each user.
- Find general constructions for variable weight OOC's.

Appendix A

Lemma 2.4: Let $\mathbf{x} \in \mathcal{C}$, where $\mathcal{C}$ is an $(n, w, \lambda + m, \lambda)$ optical orthogonal code. (Assume $m \geq 0$ is an integer.) Then if $w > 2\lambda - 2$,

$$|M_{\mathbf{x},\lambda}| \geq \frac{w\binom{w-1}{\lambda}}{m+1}.$$

If $w \leq 2\lambda - 2$,

$$|M_{\mathbf{x},\lambda}| \geq \frac{w\binom{w-1}{\lambda}}{\lambda+m}.$$

Proof: Let

$$\mathcal{I} = \{\underline{i} = (i_0, i_1, \ldots, i_{\lambda-1}, j) : \ 0 \leq i_0 < i_1 < \cdots < i_{\lambda-1} \leq w-2, 0 \leq j \leq w-1\}.$$

Define a function $f : \mathcal{I} \rightarrow Z^{\lambda}$ by

$$f([i_0, i_1, \ldots, i_{\lambda-1}, j]) \ \triangleq \ [\sum_{k_0=0}^{i_0} t_{j \ominus k_0}, \sum_{k_1=i_0+1}^{i_1} t_{j \ominus k_1}, \cdots, \sum_{k_{\lambda-1}=i_{\lambda-2}+1}^{i_{\lambda-1}} t_{j \ominus k_{\lambda-1}}]$$

where $\mathbf{t_x} = [t_0, t_1, \ldots, t_{x-1}]$

Therefore $M_{\mathbf{x},\lambda}$ is exactly the image of $f(\cdot)$. From Lemma 2.3, we know that f is one-to-one if and only if the inner product between $\mathbf{x}$ and any cyclic shift of $\mathbf{x}$ is no more than λ.

Define another function $g : \mathcal{I} \rightarrow Z$ such that $g(\underline{i})$ is the sum of the components of $f(\underline{i})$ – i.e., if $f(\underline{i}) = [a_0, a_1, \ldots, a_\lambda]$ then $g(\underline{i}) = a_0 + a_1 + \ldots + a_{\lambda-1}$.

Now partition $\mathcal{I}$ into sets consisting of either one or two elements; the sets with two elements we call "pairs" and the sets with one element we call "singles". Two elements can be in the same pair only if $f(\cdot)$ maps them into the same element of $M_{\mathbf{x},\lambda}$; thus if we let $f^{-1}(m)$ denote the inverse image of $m \in M_{\mathbf{x},\lambda}$ – i.e., $f^{-1}(m) = \{\underline{i} : f(\underline{i}) = m\}$ – then

$$\text{Number of pairs in the partition } = \sum_{m \in M_{\mathbf{x},\lambda}} \lfloor |f^{-1}(m)|/2 \rfloor$$

and

$$\text{Number of singles in the partition } = w\binom{w-1}{\lambda} - 2 \cdot \sum_{m \in M_{\mathbf{x},\lambda}} \lfloor |f^{-1}(m)|/2 \rfloor.$$

Each pair corresponds to two different sets of $\lambda + 1$ non-zero components that can be "lined up" via cyclic shifts. Suppose, for instance, that $(\underline{i}, \underline{i}')$ form a pair. By definition, this means that there exists an $\mathbf{a} = [a_0, a_1, \ldots, a_{\lambda-1}] \in M_{\mathbf{x},\lambda}$ such that $f(\underline{i}) = f(\underline{i}') = \mathbf{a}$. But this means there exists two different sets of $\lambda+1$ integers – call them $\{\ell_0, \ldots, \ell_\lambda\}$ and $\{\ell'_0, \ldots, \ell'_\lambda\}$ – such that

$$x_{\ell_k} = x_{\ell'_k} = 1$$

and

$$\ell_k - \ell'_k = \tau \neq 0$$

for all $k = 0, 1, \ldots, \lambda$.

So associated with this pair there are $\lambda + 1$ "copies" of τ in $R_{\mathbf{x}}$ – *and* $\lambda + 1$ copies of $n - \tau$. Furthermore, there exist j_1, i_1, j_2, i_2 such that

$$\tau = \sum_{k=0}^{i_1} t_{j_1 \uplus k} \qquad \text{and} \qquad n - \tau = \sum_{k=0}^{i_2} t_{j_2 \uplus k}.$$

Let

$$\tau^* = \begin{cases} \tau & \text{if } i_1 \geq i_2, \\ n - \tau & \text{otherwise.} \end{cases}$$

So associated with each pair is a number τ^* that appears at least $\lambda+1$ times in $R_{\mathbf{x}}$. As we look at the set of all pairs we thus generate a set of numbers; suppose there are u of them and we label them $\tau_1, \tau_2, \ldots, \tau_u$. So associated with each τ_k is the pair that generates τ_k; in fact there may be more than one pair associated with a given τ_k, since more than one pair may generate the same number.

Now for each $k \in \{1, 2, \ldots, u\}$ define $\mathcal{V}_k$ as follows:

$$\mathcal{V}_k = \{\underline{i} \in \mathcal{I} : \underline{i} \text{ is in a single and } g(\underline{i}) = \tau_k, \text{ or } \underline{i} \text{ is in a pair associated with } \tau_k\}.$$

Then $\mathcal{V}_1, \mathcal{V}_2, \ldots, \mathcal{V}_u$ form non-overlapping subsets of $\mathcal{I}$. Furthermore, if we let p_k equal the number of pairs associated with τ_k, then

$$\begin{aligned} |\mathcal{V}_k| &\leq 2p_k + (\lambda + m) - p_k(\lambda + 1) \\ &= m - (p_k - 1)(\lambda - 1) + 1 \\ &\leq m + 1, \end{aligned}$$

We now explain the first inequality. Clearly, each pair contributes exactly two elements to $\mathcal{V}_k$. Furthermore, the number of elements due to singles can be no more than $(\lambda+m)-p_k(\lambda+1)$. This is because each element from a single lying in $\mathcal{V}_k$ will cause another occurrence of τ_k in $R_{\mathbf{x}}$; by the auto-correlation property of $R_{\mathbf{x}}$ there can be no more than $\lambda + m$ occurrences of τ_k in $R_{\mathbf{x}}$ and $p_k(\lambda + 1)$ of them will be "used up" by the presence of p_k pairs. Thus the first inequality follows.

Let $\mathcal{V} = \mathcal{V}_1 \cup \mathcal{V}_2 \ldots \cup \mathcal{V}_u$. Then $|\mathcal{V}| = |\mathcal{V}_1| + |\mathcal{V}_2| + \ldots + |\mathcal{V}_u| \leq u(m+1)$. Furthermore, define

$$\mathcal{U}_k = \{\underline{i} \in \mathcal{V} : \ g(\underline{i}) = k\},$$

for all $k \in \{1, 2, \ldots, u\}$. If $w > 2\lambda - 2$ then $\max\{i_1, i_2\} \geq \lambda$, and so $|\mathcal{U}_k| \geq 1$.

This in turn means that

$$|\mathcal{U}| = |\mathcal{U}_1| + |\mathcal{U}_2| + \ldots + |\mathcal{U}_u| \geq u.$$

Since

$$u \geq \frac{|\mathcal{V}|}{m+1},$$

then

$$|\mathcal{U}| \geq \frac{|\mathcal{V}|}{m+1}.$$

Finally, if we define $\mathcal{D} = \mathcal{I} - \mathcal{V}$, then

$$\begin{aligned} w\binom{w-1}{\lambda} &= |\mathcal{D}| + |\mathcal{V}| \\ &\leq |\mathcal{D}| + (m+1)|\mathcal{U}| \\ &\leq (m+1)|\mathcal{D}| + (m+1)|\mathcal{U}| \\ &= (m+1)(|\mathcal{D}| + |\mathcal{U}|) \\ &\leq (m+1)|M_{\mathbf{x},\lambda}|. \end{aligned}$$

This completes the proof. The proof for $w \leq 2\lambda - 2$ is analogous the proof above. QED.

Appendix B

We now prove that the number A in Theorem 2.3 is an upper bound on the number of binary n-tuples of weight w with auto-correlation exceeding λ_a.

Associate the w-set $S = \{s_1, s_2, \ldots, s_w\}$ with the binary n-tuple containing ones in positions $s_1, s_2, \ldots s_w$ and zeroes everywhere else. We wish to (over) count the number of w-sets associated with n-tuples that violate the auto-correlation constraint.

Fix δ to be a positive integer. Then a "chain" is a set of integers modulo n $\{i_0, i_1, \cdots, i_x\}$ where $i_j = i_{j-1} + \delta$ for $1 \leq j \leq x$; the length of this chain is $1 + x$. By convention, a cycle (i.e. – $i_0 = i_x + \delta$) is considered a chain of length $x + 1$ with an arbitrary starting point. A maximal chain is one not contained in another chain.

Now suppose the w-set $S = \{s_1, s_2, \cdots, s_w\}$ can be partitioned into c maximal chains whose lengths are $1 + x_1, 1 + x_2, \ldots, 1 + x_c$. Then clearly

$$w = c + \sum_{i=1}^{c} x_i.$$

Realize that the w-set S is specified exactly by:

1. The value δ upon which the partitioning chains shall be based;
2. The number c of maximal chains into which it can be partitioned;
3. The lengths of the maximal chains – i.e., $x_1, x_2, \ldots, x_c$;
4. The chain "heads" – i.e., the starting point of each chain.

The auto-correlation of the n-tuple associated with S after δ cyclic shifts is $w - c + N$, where N is the number of chains that are cycles. This is because a cycle of length $x_i + 1$ adds $x_i + 1$ to the auto-correlation, whereas a non-cyclic chain of length $x_i + 1$ adds only x_i. Our approach, therefore, will be to count the number of w-sets with the property that $w - c + N > \lambda_a$; in doing so we will let δ vary from 1 to $\lfloor n/2 \rfloor$. (For $\delta > n/2$ we shall have already counted the associated w-sets with $\delta' = n - \delta$.)

Once n, w, and δ are fixed the only way a chain can be a cycle is if $\delta | kn$ for some integer k. Furthermore, if $\delta | kn$ but $\delta \not| n$ then there is some other value δ' such that $\delta' | \delta$ and $\delta' | n$ and the cycle associated with δ is identical to a cycle associated with δ'. Therefore, in looking for cycles we need only look at values of δ that divide n; when $\delta \not| n$ we will not find any cycles that have not already been accounted for.

So how many cycles can there be ? Each cycle "uses up" n/δ of the w ones; therefore there can be anywhere from zero to $\lfloor w\delta/n \rfloor$ cycles – i.e., $0 \leq N \leq \lfloor w\delta/n \rfloor$.

So, suppose we've fixed n, w, δ, and N. We're now going to specify the chains. How many ways are there to pick c non-negative integers $x_1, x_2, \ldots, x_c$ such that $x_1 + x_2 + \ldots + x_c = w - c$ and $w - c + N > \lambda_a$. Note that c must be at least $\lceil w\delta/n \rceil$; this is because the smallest number of chains is caused by making each chain as large as possible, and the largest chain is a cycle. Thus c is smallest when there are $\lfloor w\delta/n \rfloor$ cycles and (possibly) one "leftover" non-cyclic chain for a total of $\lceil w\delta/n \rceil$ chains.

So now suppose we've fixed n, w, δ, N, and c. Then there are $\binom{c}{N}$ ways to pick the chains that are cycles. Furthermore once we've picked those N cycles we've fixed exactly N of the x_i's to be equal to $(n/\delta) - 1$; thus it remains to pick

the remaining $c-N$ x_i's to add up to $w-c-N((n/\delta)-1)$. For $c > N$ there are $\binom{w-Nn/\delta-1}{c-N-1}$ ways to pick these $c-N$ x_i's. Note that the term $\triangle(N - w\delta/n)$ is included to count the case where there are $c = N = w\delta/n$ chains and they are all cycles. For the case $c = N$ we must have $c = N = w\delta/n$, since all of the 1's are used up in chains, and in this case the question of picking the "remaining" x_i's becomes vacuous.

The last terms in the first sum are used to count the number of ways to pick the heads of the chains. We know that any cycle must begin in one of the first δ positions; we know furthermore that the $c - N$ non-cyclic chains may begin in any of the $n - Nn/\delta$ positions not taken up by cycles.

The second sum counts the number of w-sets violating the auto-correlation constraint when $\delta \not| n$ and so there can be no cycles. There are $\binom{w-1}{c-1}$ ways to pick these c x_i's to add up to $w - c$ and $\binom{n}{c}$ ways to pick c chain heads.

Therefore, A represents an upper bound on the number of binary n-tuples that violate an auto-correlation constraint of λ_a. QED.

Appendix C

Lemma 2.6: Let λ_a and λ_c be positive integers, and let p be a constant such that $p < \min\{\lambda_a/(2\lambda_a+3), \lambda_c/(2\lambda_c+3)\}$. Then

$$\lim_{n\to\infty} \Phi(n, \lfloor \alpha n^p \rfloor, \lambda_a, \lambda_c) = \infty,$$

for any positive real α.

Proof: We will demonstrate that for n prime the lower bound in Theorem 2.3 grows unbounded with increasing n. For n prime the bound becomes

$$\Phi(n, w, \lambda_a, \lambda_c) \geq \frac{\binom{n}{w} - A}{B},$$

where

$$A = \lfloor n/2 \rfloor \sum_{c=1}^{w-\lambda_a-1} \binom{w-1}{c-1}\binom{n}{c},$$

and

$$B = n \sum_{i>\lambda_c} \binom{n-w}{w-i}\binom{w}{i}.$$

Let $w \triangleq \lfloor \alpha n^p \rfloor$. Then for n sufficiently large the following inequalities hold:

$$\binom{n}{w} \geq \frac{n^w}{w!}[1 - \frac{\alpha^2 n^{2p}}{n - \alpha n^p + 1}],$$

$$A \leq \frac{n^w}{w!} n^{p(2\lambda_a+3)-\lambda_a} \frac{\alpha^{2\lambda_a+3}}{2(\lambda_a+1)!},$$

$$B \leq n^{w-\lambda_c} w^{\lambda_c+2} \frac{1}{(w-\lambda_c-1)!(\lambda_c+1)!}.$$

Since $p < \lambda_a/(2\lambda_a + 3)$, A is a lower order term compared with $n^w/w!$.

Combining the numerator and denominator,we have

$$\begin{aligned}\Phi(n, \lfloor \alpha n^p \rfloor, \lambda_a, \lambda_c) &\geq \frac{n^{\lambda_c}(\lambda_c + 1)!}{w^{2\lambda_c+3}} + (\text{lower order terms}) \\ &\geq n^{\lambda_c - p(2\lambda_c+3)} \frac{(\lambda_c + 1)!}{\alpha^{2\lambda_c+3}} + (\text{lower order terms}).\end{aligned}$$

Since $p < \min\{\lambda_a/(2\lambda_a + 3), \lambda_c/(2\lambda_c + 3)\}$, then

$$\lim_{n\to\infty} \Phi(n, \lfloor \alpha n^p \rfloor, \lambda_a, \lambda_c) = \infty.$$

Appendix D

Lemma 2.7: Let p, q, and r be positive constants such that $p < (1/2)$, then

$$\lim_{n\to\infty} \Phi(n, \lfloor \alpha n^p \rfloor, \lfloor \beta n^q \rfloor, \lfloor \gamma n^r \rfloor) = \infty,$$

for any positive real α,β and γ.

Proof: As in Appendix C we will demonstrate that for n prime the lower bound in Theorem 2.3 grows unbounded with increasing n. For n prime the bound becomes

$$\Phi(n, w, \lambda_a, \lambda_c) \geq \frac{\binom{n}{w} - A}{B},$$

where

$$A = \lfloor n/2 \rfloor \sum_{c=1}^{w-\lambda_a-1} \binom{w-1}{c-1}\binom{n}{c},$$

and

$$B = n \sum_{i>\lambda_c} \binom{n-w}{w-i}\binom{w}{i}.$$

For n sufficiently large – i.e., for n such that $0 < \alpha n^p - 1 < n$ – it can be shown that the following inequality holds:

$$\binom{n}{w} \geq \frac{n^w}{w!}[1 - \frac{\alpha^2 n^{2p}}{n - \alpha n^p + 1}],$$

where $w = \lfloor \alpha n^p \rfloor$.

Furthermore, for n sufficiently large, it can be shown that

$$A \leq \frac{n^w}{w!} n^{p(2\lambda_a+3)-\lambda_a} \frac{\alpha^{2\lambda_a+3}}{2(\lambda_a+1)!}.$$

Using Stirling's approximation to $n!$, which is

$$\sqrt{2n\pi}(n/e)^n < n!$$

, then

$$\begin{aligned}\frac{\alpha^{2\lambda_a+3}}{2(\lambda_a+1)!} &< \frac{\alpha^{2\lambda_a+3}}{2\sqrt{2(\lambda_a+1)\pi}((\lambda_a+1)/e)^{\lambda_a+1}} \\ &= \left(\frac{\alpha\sqrt{e}}{\sqrt{\lambda_a+1}}\right)^{2(\lambda_a+1)} \frac{\alpha}{2\sqrt{2(\lambda_a+1)\pi}} \\ &< 1,\end{aligned}$$

where $\lambda_a = \lfloor \beta n^q \rfloor$.

Since $p < \lambda_a/(2\lambda_a+3)$, A is a low order term compared with $n^w/w!$.

Finally, again for n sufficiently large the denominator can be bounded by

$$B \leq n^{w-\lambda_c} w^{\lambda_c+2} \frac{1}{(w-\lambda_c-1)!(\lambda_c+1)!}.$$

Combining the numerator and denominator,we have

$$\begin{aligned}\Phi(n, \lfloor \alpha n^p \rfloor, \lambda_a, \lambda_c) &\geq \frac{n^{\lambda_c}(\lambda_c+1)!}{w^{2\lambda_c+3}} + (\text{low order terms}) \\ &\geq n^{\lambda_c - p(2\lambda_c+3)} \frac{(\lambda_c+1)!}{\alpha^{2\lambda_c+3}} + (\text{low order terms}).\end{aligned}$$

Again using Stirling's approximation to $(\lambda_c+1)!$, then

$$\begin{aligned}\frac{(\lambda_c+1)!}{\alpha^{2\lambda_c+3}} &> \frac{\sqrt{2(\lambda_c+1)\pi}((\lambda_c+1)/e)^{\lambda_c+1}}{\alpha^{2\lambda_c+3}} \\ &= \left(\frac{\lambda_c+1}{\alpha^2 e}\right)^{\lambda_c+1} \frac{\sqrt{2(\lambda_c+1)\pi}}{\alpha}\end{aligned}$$

where $\lambda_c = \lfloor \gamma n^r \rfloor$.

Therefore, as long as $p < \min\{\lambda_a/(2\lambda_a+3), \lambda_c/(2\lambda_c+3)\}$ for n sufficiently large, which is $1/2$, then

$$\lim_{n\to\infty} \Phi(n, \lfloor \alpha n^p \rfloor, \lfloor \beta n^q \rfloor, \lfloor \gamma n^r \rfloor) = \infty.$$

Appendix E

The probability of error for hard-limiting case for user j with weight w_j given M $(= M_0 + M_1 + \ldots + M_p)$ users is,

$$P_{e_j} = \frac{1}{2}\sum_{l_0=0}^{M_0}\cdots\sum_{l_j=0}^{M_j-1}\cdots\sum_{l_p=0}^{M_p} P_r\{\text{the probability of } l\ (= l_0 + l_1 + \ldots + l_p) \text{ interfering users}\} * P_r\{\text{the probability of error given such } l \text{ interfering users}\}. \quad \text{(E.1)}$$

For

$$P_r\{\text{the probability of } l\ (= l_0 + l_1 + \ldots + l_p) \text{ interfering users}\} = \binom{M_j - 1}{l_j} q_j^{l_j}(1 - q_j)^{M_j - 1 - l_j} \prod_{\substack{k=0 \\ k \neq j}}^{p} \binom{M_k}{l_k} q_k^{l_k}(1 - q_k)^{M_k - l_k}. \quad \text{(E.2)}$$

And for

$$P_r\{\text{the probability of error given } l \text{ interfering users}\} = \sum_{i=0}^{w_j - 1} (-1)^i \binom{w_j}{i} (1 - \frac{i}{w_j})^{l_0 + \ldots + l_j + \ldots + l_p}. \quad \text{(E.3)}$$

The detailed derivation of equation $(E.3)$ is in [39]. Therefore, substituting equations $(E.2)$ and $(E.3)$ into equation $(E.1)$, we get

$$P_{e_j} = \frac{1}{2}\sum_{l_0=0}^{M_0}\cdots\sum_{l_j=0}^{M_j-1}\cdots\sum_{l_p=0}^{M_p} \binom{M_0}{l_0}\cdots\binom{M_j - 1}{l_j}\cdots\binom{M_p}{l_p} q_0^{l_0}(1 - q_0)^{M_0 - l_0}\cdots q_j^{l_j}(1 - q_j)^{M_j - 1 - l_j}\cdots q_p^{l_p}(1 - q_p)^{M_p - l_p} *$$

$$\sum_{i=0}^{w_j-1}(-1)^i\binom{w_j}{i}(1-\frac{i}{w_j})^{l_0+\ldots+l_j+\ldots+l_p}$$

$$-\frac{1}{2}(1-q_0)^{M_0}\ldots(1-q_j)^{M_j-1}\ldots(1-q_p)^{M_p}\sum_{i=0}^{w_j-1}(-1)^i\binom{w_j}{i}$$

$$= \frac{1}{2}\sum_{i=0}^{w_j-1}(-1)^i\binom{w_j}{i}\sum_{l_0=0}^{M_0}\binom{M_0}{l_0}[q_0(1-\frac{i}{w_j})]^{l_0}(1-q_0)^{M_0-l_0}\ldots$$

$$\sum_{l_j=0}^{M_j-1}\binom{M_j-1}{l_j}[q_j(1-\frac{i}{w_j})]^{l_j}(1-q_j)^{M_j-1-l_j}\ldots\sum_{l_p=0}^{M_p}[q_p(1-\frac{i}{w_j})]^{l_p}$$

$$\cdot(1-q_p)^{M_p-l_p}-\frac{1}{2}(1-q_0)^{M_0}\ldots(1-q_j)^{M_j-1}\ldots(1-q_p)^{M_p}*$$

$$\sum_{i=0}^{w_j-1}(-1)^i\binom{w_j}{i}$$

$$= \frac{1}{2}\{\sum_{i=0}^{w_j-1}(-1)^i\binom{w_j}{i}[(1-\frac{q_j i}{w_j})]^{M_j-1}\prod_{\substack{l=0\\l\neq j}}^{p}[(1-\frac{q_l i}{w_j})]^{M_l}$$

$$+(-1)^{w_j}(1-q_j)^{M_j-1}\prod_{\substack{l=0\\l\neq j}}^{p}(1-q_l)^{M_l}\}.$$

Where the second term is owing to $l=0$.

Bibliography

[1] D. Bertsekas and R. Gallager, *Data Networks.* Prentice-Hall, Inc., 1987.

[2] K. Gilhousen, I. Jacobs, R. Padovani, A. Viterbi, L. Weaver Jr., and C. W. III, "On the capacity of a cellular CDMA system," *IEEE Transactions on Vehicular Technology*, vol. 40, pp. 303–312, May 1991.

[3] J. K. Holmes, *Coherent Spread Spectrum Systems.* John Wiley & Sons Inc., 1982.

[4] D. V. Sarwate and M. B. Pursley, "Crosscorrelation properties of pseudorandom and related sequences," *Proceedings of the IEEE*, vol. 68, pp. 593–619, May 1980.

[5] R. E. Ziemer and R. L. Peterson, *Digital Communications and Spread Sprectrum Systems.* 1985.

[6] J. A. Salehi, "Code division multiple access techniques in optical fiber networks-part I: Fundamental principles," *IEEE Transactions on Communications*, pp. 824–833, August 1989.

[7] J. A. Salehi and C. A. Brackett, "Code division multiple access techniques in optical fiber networks-part II: Systems performance analysis," *IEEE Transactions on Communications*, pp. 834–850, August 1989.

[8] F. R. K. Chung, J. A. Salehi, and V. K. Wei, "Optical orthogonal codes: Design, analysis, and applications," *IEEE Transactions on Information Theory*, vol. 35, pp. 595–604, May 1989.

[9] H. Chung and P. V. Kumar, "Optical orthogonal codes-new bounds and an optimal construction," *IEEE Transactions on Information Theory*, vol. 36, pp. 866–873, July 1990.

[10] G. C. Yang and T. Fuja, "Bounds and constructions for optical orthogonal codes: Code division multiple access for optical networks," in *International Symposium on Communication*, (Taiwan), pp. 76–79, December 1991.

[11] G. C. Yang and T. Fuja, "Optical orthogonal codes with unequal auto- and cross-correlation constraints," in *Proceedings of The Twenty-Sixth Annual Conference on Information Sciences and Systems*, (Princeton, New Jersey), March 1992.

[12] G. C. Yang and T. Fuja, "Optical orthogonal codes with unequal auto- and cross-correlation constraints: Code division multiple access for optical networks," *submitted to IEEE Transactions on Information Theory*, 1992.

[13] C. N. Georghiades, "On PPM sequences with good autocorrelation properties," *IEEE Transactions on Information Theory*, vol. 34, pp. 571–576, May 1988.

[14] R. Gagliardi, J. Robbins, and H. Taylor, "Acquisition sequences in PPM communications," *IEEE Transactions on Information Theory*, vol. 33, pp. 738–744, September 1987.

[15] A. A. Shaar and P. A. Davies, "Prime sequences: Quasi-optimal sequences for OR channel code division multiplexing," *Electronics Letters*, vol. 19, pp. 888–890, October 1983.

[16] S. V. Marić, "A new family of algebraically designed optical orthogonal codes with ideal auto- and cross-correlation functions," in *Proceedings of The Twenty-Sixth Annual Conference on Information Sciences and Systems*, (Princeton, New Jersey), March 1992.

[17] B. Masnick and J. Wolf, "On linear unequal error protection codes," *IEEE Transactions on Information Theory*, vol. 13, pp. 600–607, October 1967.

[18] W. J. V. Gils, "Two topics on linear unequal error protection codes: Bounds on their length and cyclic code classes," *IEEE Transactions on Information Theory*, vol. 29, pp. 866–876, November 1983.

[19] M. C. Lin and S. Lin, "Cyclic unequal error protection codes constructed from cyclic codes of composite length," *IEEE Transactions on Information Theory*, vol. 34, pp. 867–871, July 1988.

[20] A. W. Lam and D. V. Sarwate, "On optimum time-hopping patterns," *IEEE Transactions on Communications*, vol. 36, pp. 380–382, March 1988.

[21] T. Kløve, "Bounds on the size of optimal difference triangle sets," *IEEE Transactions on Information Theory*, vol. 34, pp. 355–361, March 1988.

[22] T. Kløve, "Bounds and construction for difference triangle sets," *IEEE Transactions on Information Theory*, vol. 35, pp. 879–886, July 1989.

[23] F. Khansefid, *Sets of* $(0,1)$*-sequences with application to optical-fibre networks*. PhD thesis, Univ. of Southern California, August 1988.

[24] P. V. Kumar, "On the existence of square dot-matrix patterns having a specific three-valued periodic-correlation function," *IEEE Transactions on Information Theory*, vol. 34, pp. 271–277, March 1988.

[25] S. W. Golomb and H. Taylor, "Two-dimensional synchronization patterns for minimum ambiguity," *IEEE Transactions on Information Theory*, vol. 28, pp. 600–605, July 1982.

[26] W. H. Kautz and R. C. Singleton, "Nonrandom binary superimposed codes," *IEEE Transactions on Information Theory*, vol. 10, pp. 363–377, October 1964.

[27] A. G. D'yachkov and V. V. Rykov, "Bounds on the length of disjunctive codes," *Problemy Peredachi Informatsii*, vol. 18, pp. 7–13, July-Sep. 1982.

[28] A. J. Shakir, *Disjunctive codes for the multiple access OR-channel.* PhD thesis, Department of Electrical Engineering, Linköping University, S-581 83 Linköping, Sweden, 1991.

[29] J. A. Salehi, "Emerging optical code-division multiple access communications systems," *IEEE Network*, pp. 31–39, March 1989.

[30] F. R. K. Chung, J. A. Salehi, and V. K. Wei, "Correction to optical orthogonal codes: Design, analysis, and applications," *submitted to IEEE Transactions on Information Theory*, 1992.

[31] V. K. Wei, "Private communication." Jan. 1992.

[32] R. H. Morelos-Zaragoza, *Multi-level Error Correcting Codes.* PhD thesis, Univ. of Hawaii, May 1992.

[33] 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, pp. 40–49, July-September 1979.

[34] R. M. Wilson, "Cyclotomy and difference families in elementary abelian groups," *J. Number Theory*, vol. 4, pp. 17–47, 1972.

[35] H. Hanani, "The existence and constructions of balanced incomplete block designs," *Ann. Math. Statist.*, vol. 32, pp. 361–386, 1961.

[36] R. C. Bose, "On the construction of balanced incomplete block design," *Ann. Eugenics*, vol. 9, pp. 353–399, 1939.

[37] Z. Chen, P. Fan, and J. Fan, "Disjoint difference sets, difference triangle sets, and related codes," *IEEE Transactions on Information Theory*, vol. 38, pp. 518–522, March 1992.

[38] M. J. Colbourn and C. J. Colbourn, "Recursive constructions for cyclic block designs," *J. Statistical Planning and Inference*, vol. 10, pp. 97–103, 1984.

[39] M. Y. Azizoğlu, J. A. Salehi, and Y. Li, "Optical CDMA via temporal codes," *will appear in IEEE Transactions on Communications.*

[40] J. G. C. Clark and J. B. Cain, *Error-Correcting Coding for Digital Communications.* New York: Plenum, 1981.

[41] T. R. N. Rao and A. S. Chawla, "Asymmetric error codes for some LSI semi-conductor memories," in *Proc. Ann. Southeastern Symp. Syst. Theory*, pp. 170–171, 1975.

[42] A. M. Michelson and A. H. Levesque, *Error Control Techniques for Digital Communication.* New York: Wiley, 1985.

[43] H. M. Kwon, "Optical orthogonal codes division multiple access system part II: Multibits/sequence-period OOCDMA," ICC, 1991.

[44] W. C. Kwong, P. A. Perrier, and P. R. Prucnal, "Performance comparison of asychronous and synchronous code-division multiple-access techniques for fiber-optical local area networks," *IEEE Transactions on Communications*, vol. 39, pp. 1625–1634, November 1991.