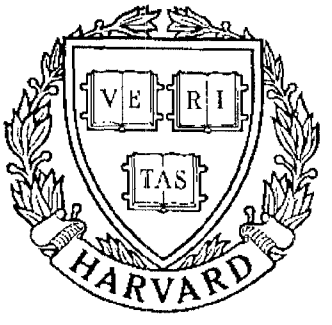
TECHNICAL RESEARCH REPORT

SYSTEMS RESEARCH CENTER



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

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

by G-C. Yang and T. Fuja

TR 92-89

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

Guu-chang Yang and Thomas Fuja*
Department of Electrical Engineering
System Research Center
University of Maryland
College Park, MD 20742

Abstract

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 paper 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 paper 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.

Submitted to *IEEE Transactions on Information Theory.*

*Supported in part by National Science Foundation grant NCR-8957623; also by the NSF Engineering Research Centers Program, CDR-8803012. Portions of this work were presented at the 1991 International Symposium on Communications, Tainan, Taiwan, December 1991; also at the 1992 Conference on Information Sciences and Systems, Princeton, NJ, March 1992.

Index Terms

Code-division multiple access, optical networks, constant-weight codes, unequal error protection codes, spread-spectrum systems.

Captions

Table 1: The cardinality of $(n, w, 2, 1)$ OOC's constructed using the technique of Section 5.1.

Table 2: The cardinality of $(n, w, 2, 1)$ OOC's constructed using the technique of Section 5.2.

Table 3: The cardinality of $(n, w, 1, 1)$ OOC's constructed using the technique of Section 5.3. All codes are optimal.

Figure 1: Lower bounds on the number of message bits that can be protected against two errors while $\lfloor \log_2 n \rfloor$ bit are protected against single errors. The bound from Theorem 4 assumes constant weight codewords with $w = \lfloor n/2 \rfloor$,

1. Introduction

This paper 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) [1-4].

The results in this paper extend previous work on optical orthogonal codes in that 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 the same, and so considering only codes for which the constraints 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. Background and Motivation

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

2.1. Definitions and Past Work

What follows is the definition of an OOC given by Salehi *et. al.* [3].

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 $\mathbf{x}$, let $D^i\mathbf{x}$ 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 $\mathbf{x} \in \mathcal{C}$, $d^a_{\min}(\mathbf{x}) \geq 2w - 2\lambda_a$, where $d^a_{\min}(\mathbf{x})$ is the minimum distance between $\mathbf{x}$ and its cyclic shifts – i.e., $d^a_{\min}(\mathbf{x}) \triangleq \min\{d_H(\mathbf{x}, D^\tau\mathbf{x}) : \tau = 1, 2, \ldots, n-1\}$.
- (Cross-correlation) For any $\mathbf{x} \in \mathcal{C}$ and any $\mathbf{y} \in \mathcal{C}$, $d^c_{\min}(\mathbf{x}, \mathbf{y}) \geq 2w - 2\lambda_c$, 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 [1-2]. We assume that each user may transmit a physical optical pulse – a *chip* – at any time. If we assume incoherent

processing – i.e., only the intensity and not the the phase of the signal is available at the receiver – then multiple pulses transmitted simultaneously by different users sum.

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 chip durations. 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 [1][2] Salehi introduced optical orthogonal codes and computed the error probability assuming a channel where the only "noise" is interference from other users. In [3] 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 [3] 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. 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 enables 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 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 a threshold 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 demonstrate in this paper that the cardinality of the $(n, w+m, \lambda+m, \lambda)$ code can actually exceed that of the less robust codes – thus more users can be provided even better performance. Clearly, this motivates the study of such OOC's.

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.,

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

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 $M_{\mathbf{x},\lambda}$; 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,

$$R_{\mathbf{x}} = \begin{pmatrix} t_0 & t_1 & t_2 & t_3 \\ t_0 + t_1 & t_1 + t_2 & t_2 + t_3 & t_3 + t_0 \\ t_0 + t_1 + t_2 & t_1 + t_2 + t_3 & t_2 + t_3 + t_0 & t_3 + t_0 + t_1 \end{pmatrix}$$

$$= \begin{pmatrix} 3 & 1 & 4 & 5 \\ 4 & 5 & 9 & 8 \\ 8 & 10 & 12 & 9 \end{pmatrix}$$

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 $M_{\mathbf{x},\lambda}$ is given in the following three lemmas.

Lemma 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 of $R_{\mathbf{x}}$ 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 $j = 0, 1, \ldots, \lambda$

$$x_{i_j} = x_{i'_j} = 1 \quad \text{and} \quad i_j - i'_j = \tau^* \neq 0.$$

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

Lemma 2: 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 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. 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.

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 [3] 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 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 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 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, if $[a_0, a_1, \ldots, a_{\lambda-1}] \in M_{\mathbf{x},\lambda}$ then $a_0 + a_1 + \ldots + a_{\lambda-1} \leq n-1$. 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 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

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

Proof: See Appendix A.

Theorem 2: Let m be a non-negative integer. Then

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

Proof: Let $\mathcal{C}$ be an $(n, w, \lambda + m, \lambda)$ OOC such that $|\mathcal{C}| = \Phi(n, w, \lambda + m, \lambda)$. By Lemma 4, for any $\mathbf{x} \in \mathcal{C}$, $|M_{\mathbf{x},\lambda}| \geq w\binom{w-1}{\lambda}/(m+\lambda)$. Furthermore, by Lemma 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 1 to be $\binom{n-1}{\lambda}$. So:

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

which was to be proven. QED.

Examining Theorem 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. For instance, considering $(n, w, 2, 1)$ OOC's, the bound in Theorem 2 is tighter than the Johnson bound for $(n, w, 2, 2)$ codes provided $n > 2w - 2$.

We note also that Theorem 2 is only a generalization of the Johnson bound for $\lambda = 1$; for $\lambda \geq 2$ the bound on $\Phi(n, w, \lambda, \lambda)$ obtained by setting $m = 0$ in Theorem 2 is weaker than the bound in Theorem 1.

3.2. Lower Bounds

In [3][5] a lower bound on $\Phi(n, w, \lambda_a, \lambda_c)$ was derived for odd prime n. Subsequently, Victor Wei [6] derived an alternate lower bound – again for odd prime n. In what follows we use the general approach of Wei [6] to bound $\Phi(n, w, \lambda_a, \lambda_c)$ for $\lambda_a \neq \lambda_c$ and any n.

Theorem 3:

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

where

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

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

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

and

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

Proof: As in [3][5], 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 [3][5]. 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.

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 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,

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

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

Lemma 6: Let λ_a and λ_c be positive integers, and let p be a positive 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 5 tell us that if the codeword weight grows faster than $\sqrt{n}$ we will be unable to find *any* codewords for large n. Lemma 6 suggests if we let the weight grow slower than $n^{1/5}$ there is no limit to the number of codewords we can construct.

Finally, we demonstrate that when the auto- and cross-correlation constraints are growing like n^k for a constant k, we are guaranteed the existence of codes provided the codeword weight grows no faster than $\sqrt{n}$.

Lemma 7: Let p, q, and r be constants $0 < p, q, r < 1$. Then if $p < 1/2$,

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

for any positive real α, β and γ.

Proof: See Appendix C.

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 $\mathbf{x} \in \{0,1\}^k$ 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. 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 and j is called an unequal error protection (UEP) code.

There is a substantial body of literature on UEP codes. (See [7-11] for references.) However, there has been no investigation of *constant weight* UEP codes.

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, each 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 3 – i.e., $M(n, w, \lambda_a, \lambda_c) = (\binom{n}{w} - A)/B$, where A and B are given in Theorem 3.

Theorem 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 4 with existing bounds for non-constant weight UEP codes. Bassalgyo *et.al.* [11] 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 holds:

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

If we let $k_1 = \lfloor \log_2 n \rfloor$, this bound may be directly compared to the bound in Theorem 4. Figure 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 4 when we set $w = \lfloor n/2 \rfloor$. The lower bound in Theorem 4 is a clear improvement over the bound in [11].

This may at first seem surprising; the bound from Theorem 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.

5. New Constructions

In this section we present four new techniques for constructing optical orthogonal codes. Two of the new methods describe ways to construct $(n, w, 2, 1)$ codes; another describes an optimal $(n, w, 1, 1)$ code, while the last is a method for designing an $(n, w, 1, 2)$ OOC.

5.1 Construction 1 – An (n, w, 2, 1) OOC

We now demonstrate a technique for constructing an $(n, w, 2, 1)$ OOC. The method is a variation on the technique proposed by Wilson [12] to construct $(n, w, 1, 1)$ codes. We begin by considering the specific values of $w = 5, 6$, and we then generalize the technique.

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 that are non-zero modulo three and are distinct from each other modulo three.

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 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 $R_{\mathbf{x}_{i'}}$ form disjoint sets for $i \neq i'$, 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

$$\begin{aligned}
\mathbf{x}_0 &= [1100001000000000000000000000000100001] \\
\mathbf{x}_1 &= [1000000010010000000000000010010000000] \\
\mathbf{x}_2 &= [1000000000100010000000010001000000000].
\end{aligned}$$

Note that, using Theorem 2, we know that $\Phi(37,5,2,1) \leq (36 \cdot 2)/(5 \cdot 4) = 3.6$ and so this construction is optimal.

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 that are non-zero modulo three and are distinct modulo three,

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. Furthermore, these techniques 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 $n = w^2t/2 + 1$ for an integer t. 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 $n = (w^2 - 1)t/2 + 1$ for an integer t. 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.

5.2 Construction 2 – Another (n, w, 2, 1) OOC

We now demonstrate another technique for constructing an $(n, w, 2, 1)$ optical orthogonal code. The new technique is based on the approach of Hanani[13]. Codes derived from Construction 2 have codeword weights equivalent to zero or one modulo four. Construction 2 is neither a generalization nor a special case of Construction 1; for each construction one can "build" some codes for which it is impossible to build an equivalent code using the other method. However, for some values of n and w the two constructions lead to equivalent codes.

An (n, w, 2, 1) OOC for w = 4m: Let $n = w^2t/2 + 1$ be a prime number, where $w = 4m$. Furthermore, assume that α is a primitive element of GF(n) such that all of the following hold for some integer y, $1 \leq y \leq 4mt - 1$:

- $\alpha^{k4mt+y} - 1 = \alpha^{i_k}$, for $k = 0, 1, \ldots, m-1$;
- $\alpha^{k4mt} - \alpha^y = \alpha^{j_k}$, for $k = 1, 2, \ldots, m$;
- $\alpha^{k4mt} - 1 = \alpha^{r_k}$, for $k = 1, 2, \ldots, m$;
- $\alpha^y(\alpha^{k4mt} - 1) = \alpha^{s_k}$, for $k = 1, 2, \ldots, m$,

Here, the integers $i_0, i_1, \cdots, i_{m-1}$, $j_1, \cdots, j_m$, $r_1, \cdots, r_m$, and $s_1, \cdots, s_m$ are all distinct modulo $4m$. Then the blocks

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

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

An $(\mathbf{n}, \mathbf{w}, \mathbf{2}, \mathbf{1})$ OOC for $\mathbf{w = 4m + 1}$: Let $n = (w^2 - 1)t/2 + 1$ be a prime number, where $w = 4m + 1$. Furthermore, assume that α is a primitive element of GF(n) such that all of the following hold for some integer y, $1 \leq y \leq (4m + 2)t - 1$:

- $\alpha^{k(4m+2)t+y} - 1 = \alpha^{i_k}$, for $k = 0, 1, \ldots, m-1$;
- $\alpha^{k(4m+2)t} - \alpha^{y} = \alpha^{j_k}$, for $k = 1, 2, \ldots, m$;
- $\alpha^{k(4m+2)t} - 1 = \alpha^{r_k}$, for $k = 1, 2, \ldots, m$;
- $\alpha^{y}(\alpha^{k(4m+2)t} - 1) = \alpha^{s_k}$, for $k = 1, 2, \ldots, m$,

Here, the integers $y, i_0, i_1, \cdots, i_{m-1}$, $j_1, \cdots, j_m$, $r_1, \cdots, r_m$, and $s_1, \cdots, s_m$ are all distinct modulo $4m + 2$ and non-zero modulo $4m + 2$. Then the blocks

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

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

Example: Let $n = 41$ $w = 4$, and $t = 5$. Choose $\alpha = 6$ as the primitive element of GF(41) and choose $y = 3$ and so $6^y - 1 = 10 = 6^8$, $6^{20} - 6^y = 29 = 6^7$, $6^{20} - 1 = 39 = 6^6$, and $6^y(6^{20} - 1) = 19 = 6^9$ – i.e., $i_0 = 8$, $j_1 = 7$, $r_1 = 6$, and $s_1 = 9$. Furthermore, i_0, j_1, r_1, s_1 are distinct modulo 4. Then the code consists of the blocks $\{[1, 11, 30, 40], [12, 16, 25, 29], [10, 13, 28, 31], [3, 4, 37, 38], [7, 18, 23, 34]\}$ – i.e., the codewords are

$$\mathbf{x}_0 = [0100000000010000000000000000010000000001]$$
$$\mathbf{x}_1 = [0000000000001000100000001000100000000000]$$

$$\mathbf{x}_2 = [0000000000100100000000000001001000000000]$$
$$\mathbf{x}_3 = [0001100000000000000000000000000000001100]$$
$$\mathbf{x}_4 = [0000000100000000001000010000000001000000].$$

Note that, from Theorem 2, $\Phi(41,4,2,1) \leq 2 \cdot 40/12 = 6.66$, so we cannot say for sure if this code is optimal; there may be a $(41,4,2,1)$ code with six codewords.

Tables 1 and 2 show the parameters of some $(n,w,2,1)$ OOC's that can be constructed using the approaches outlined above. Also included is the upper bound on $\Phi(n,w,2,1)$ derived in Theorem 2. Theorem 2 tells us it is impossible to construct an $(n,w,2,1)$ OOC with more than $2(n-1)/(w(w-1))$ codewords; the methods above tell us how to construct $(n,w,2,1)$ OOC's with $2(n-1)/w^2$ codewords (for even w) as well as ones with $2(n-1)/(w^2-1)$ codewords (for odd w).

These constructions also illustrate the point made in Section 2.2 that, for some block-lengths, codes with $\lambda_a \neq \lambda_c$ may be preferable to codes with equal constraints. Comparing an $(n,w,1,1)$ code with an $(n,w+1,2,1)$ code, recall from Section 2.2 that the performance figures of merit for the $(n,w+1,2,1)$ code dominate those of the $(n,w,1,1)$ code. Yet we've just shown it is possible to construct an $(n,w+1,2,1)$ code with either $2(n-1)/(w+1)^2$ codewords (for even $w+1$) or $2(n-1)/w(w+2)$ codewords (for odd $w+1$); but Theorem 1 tells us it is impossible to construct an $(n,w,1,1)$ code with more than $(n-1)/(w(w-1))$ codewords. Therefore for $w \geq 6$ the $(n,w+1,2,1)$ codes offer better performance *and* more codewords than any $(n,w,1,1)$ code.

As a simple example, from Table 2 we see that it is possible to construct a $(1801,9,2,1)$ OOC with 45 codewords. But to construct an $(1801,8,1,1)$ code the Johnson bound tells us it's impossible to have more than 32 codewords.

5.3. Construction of an $(n,w,1,1)$ OOC

In [14] Bose used balanced incomplete block design (BIBD) to design an $(n,w,1,1)$ OOC for $w = 3,4$, and 5. Wilson generalized these results to arbitrary w in [12]. Hanani also used BIBD (with different parameters) to construct an $(n,6,1,1)$ in [13].

In the last two sections we have used similar approaches to construct $(n,w,2,1)$ OOC's. While doing so, we were also able to generalize Hanani's result to any w equivalent to two

or three modulo four – i.e., we have constructed $(n, w, 1, 1)$ OOC's for $w = 4m+2, 4m+3$. The resulting codes meet the Johnson bound with equality and are thus optimal.

For completeness, we first present Wilson's construction from [12]. These codes are also optimal.

An $(n, w, 1, 1)$ OOC for Odd w: 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: 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 follows. It is valid only for codeword weights that are equivalent to two or three modulo four; so at first glance it may seem more restricted than Wilson's construction. However, for some blocklengths the new construction yields codes while Wilson's construction does not.

An $(n, w, 1, 1)$ OOC for $w = 4m + 2$: Let $n = w(w-1)t + 1$ be a prime number, where $w = 4m + 2$. Furthermore, assume that α is a primitive root of GF(n) such that all of the following hold for some integer y, $1 \leq y \leq (8m+2)t - 1$:

- $\alpha^{k(8m+2)t+y} - 1 = \alpha^{i_k}$ for $k = 0, 1, \ldots, m$;
- $\alpha^{k(8m+2)t} - \alpha^y = \alpha^{j_k}$ for $k = 1, 2, \ldots, m$;
- $\alpha^{k(8m+2)t} - 1 = \alpha^{r_k}$, for $k = 1, 2, \ldots, m$;
- $\alpha^y(\alpha^{k(8m+2)t} - 1) = \alpha^{s_k}$, for $k = 1, 2, \ldots, m$,

Here, the integers $i_0, i_1, \cdots, i_m$, $j_1, \cdots, j_m$, $r_1, \cdots, r_m$, and $s_1, \cdots, s_m$ 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 an $(n, w, 1, 1)$ OOC.

An $(n, w, 1, 1)$ OOC for $w = 4m + 3$: Let $n = w(w-1)t+1$ be a prime number, where $w = 4m + 3$. Furthermore, assume that α is a primitive root of GF(n) such that all of the following hold for some integer y, $1 \leq y \leq (8m + 6)t - 1$:

- $\alpha^{k(8m+6)t+y} - 1 = \alpha^{i_k}$ for $k = 0, 1, \ldots, m$;
- $\alpha^{k(8m+6)t} - \alpha^{y} = \alpha^{j_k}$ for $k = 1, 2, \ldots, m$;
- $\alpha^{k(8m+6)t} - 1 = \alpha^{r_k}$, for $k = 1, 2, \ldots, m$;
- $\alpha^{y}(\alpha^{k(8m+2)t} - 6) = \alpha^{s_k}$, for $k = 1, 2, \ldots, m$,

Here, the integers $y, i_0, i_1, \cdots, i_m, j_1, \cdots, j_m, r_1, \cdots, r_m$, and $s_1, \cdots, s_m$ are all distinct modulo $4m + 3$. Then the blocks

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

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

Table 3 shows the parameters of some $(n, w, 1, 1)$ codes that can be constructed using the above approach. We note again that for each code $|\mathcal{C}| = (n-1)/w(w-1)$ and so the Johnson bound is met with equality, meaning the codes are all optimal.

5.4. Construction of an $(n, w, 1, 2)$ OOC

In Section 5.3 we presented four techniques for constructing optimal $(n, w, 1, 1)$ codes. In this section we demonstrate how we can use essentially the same techniques to construct $(n, w, 1, 2)$ codes with twice as many codewords as the $(n, w, 1, 1)$ codes.

Let us consider only Wilson's construction for odd weight codes; the extension to the other three cases is straightforward. Specifically, let us construct a code using the same

blocks that were used in the earlier construction, but extending the index i from 0 to $2t-1$ rather than from 0 to $t-1$; that is, consider the blocks

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

where $w = 2m+1$, α is a primitive element of GF(n), $n = w(w-1)t+1$, and $\{\log_\alpha[\alpha^{2mkt} - 1] : 1 \leq k \leq m\}$ are all distinct modulo m.

The claim is that this is an $(n, w, 1, 2)$ OOC. Keeping in mind that $\alpha^{w(w-1)t} = 1$ and $\alpha^{w(w-1)t/2} = -1$, it can be shown that the blocks corresponding to $i = t, t+1, \ldots, 2t-1$ are "coordinate reversed" images of the blocks corresponding to $i = 0, i = 1, \ldots, i = t-1$. That is, the code consists of the union of the following 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 $\mathbf{x}$ is a codeword from the first group with $\mathbf{t_x} = [t_0, t_1, \ldots, t_{w-1}]$ then there is a codeword $\mathbf{y}$ from the second group with $\mathbf{t_y} = [t_{w-1}, \ldots, t_1, t_0]$. Clearly, the coordinate reversal does not change the auto-correlation so the resulting code still has $\lambda_a = 1$. Furthermore, the inner-product of a (possibly shifted) codeword and its coordinate-reversed "image" will be two, while the inner product of any codeword and any codeword other than its reversed image will be at most one. Therefore, the resulting code is an $(n, w, 1, 2)$ OOC.

Example: Let $n = 19$ and $w = 3$; then using Wilson's construction we can design an optimal $(19, 3, 1, 1)$ code with three codewords:

$$\begin{aligned}
\mathbf{x}_0 &= [0100000100010000000] \\
\mathbf{x}_1 &= [0011000000000010000] \\
\mathbf{x}_2 &= [0000101001000000000]
\end{aligned}$$

If we now take the coordinate-reversed images of these codewords we obtain

$$\mathbf{x}_3 = [0000000100010000010]$$

$$\mathbf{x}_4 = [0000100000000001100]$$
$$\mathbf{x}_5 = [0000000001001010000]$$

and so $\mathcal{C} = \{\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_5\}$ is a $(19, 3, 1, 2)$ code.

6. Summary

In this paper we derived new bounds on the number of users that can be supported on an optical network employing code division multiple access with binary signature sequences; in addition we presented a number of new methods for designing codes with good auto- and cross-correlation properties. Unlike previous work in this area, we considered the possibility that the auto- and cross-correlation constraints might not be identical; indeed, the bounds and the constructions suggest that it may sometimes be preferable to use such "asymmetric" OOC's. Among the constructions presented, we note that the $(n, w, 2, 1)$ codes are near-optimal; their cardinality is $2(n-1)/w^2$ and we have demonstrated that it's impossible to get more than $2(n-1)/(w^2-w)$ codewords

We also noted the relationship between OOC's with unequal constraints and constant-weight unequal error protection codes with two levels of protection; the lower bound we derived for OOC's, when applied to UEP codes, was shown to be tighter than the best previously known bound.

Acknowledgement

The authors would like to thank Dr. Victor Wei for helpful suggestions regarding the lower bound.

Appendix A

Lemma 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

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

Proof: Let

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

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

$$f([i_0, i_1, \ldots, i_{\lambda-1}]) \triangleq [\sum_{k_0=0}^{i_0} t_{j \uplus k_0}, \sum_{k_1=i_0+1}^{i_1} t_{j \uplus k_1}, \sum_{k_2=i_1+1}^{i_2} t_{j \uplus k_2}, \cdots, \sum_{k_{\lambda-1}=i_{\lambda-2}+1}^{i_{\lambda-1}} t_{j \uplus k_{\lambda-1}}]$$

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

So $M_{\mathbf{x},\lambda}$ is exactly the image of $f(\cdot)$; to prove the lemma it is sufficient to show that $f(\cdot)$ maps at most $m + \lambda$ elements of $\mathcal{I}$ to the same element of $M_{\mathbf{x},\lambda}$; then we will have shown $|M_{\mathbf{x},\lambda}| \geq |\mathcal{I}|/(m+\lambda) = w\binom{w-1}{\lambda}/(\lambda + m)$.

Now define a function $g : \mathcal{I} \rightarrow Z$ such that $g(\underline{i})$ is the sum of the components of $f(\underline{i})$ – i.e., if $f(\underline{i}) = [a_0, a_1, \ldots, a_{\lambda-1}]$ then $g(\underline{i}) = a_0 + a_1 + \ldots + a_{\lambda-1}$. Obviously, if $f(\underline{i}) = f(\underline{i}')$ then $g(\underline{i}) = g(\underline{i}')$ so to prove the lemma it is sufficient to show that $g(\cdot)$ maps at most $m+\lambda$ elements of $\mathcal{I}$ to the same integer. For each $\underline{i} \in \mathcal{I}$ the integer $g(\underline{i})$ is a component of $R_{\mathbf{x}}$, and we know no component of $R_{\mathbf{x}}$ can be repeated more than $\lambda + m$ times by Lemma 1. Therefore, the function $g(\cdot)$ is (at most) $(m + \lambda)$-to-one and the Lemma holds. QED.

Appendix B

We now prove that the number A in Theorem 4 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" $\{i_0, i_1, \cdots, i_x\}$ is a set of integers modulo n 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 are 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 6: Let λ_a and λ_c be positive integers, and let p be a positive 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 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 $w = \lfloor \alpha n^p \rfloor$. Then for n sufficiently large – more precisely, for n such that $0 < \alpha n^p - 1 < n$ – the following inequality holds:

$$\begin{aligned}\binom{n}{w} &\geq \frac{n^w}{w!}[1 - \frac{\alpha^2 n^{2p}}{n - \alpha n^p + 1}] \\ &= \frac{n^w}{w!} + (\text{low order terms}),\end{aligned}$$

where we've made use of the fact that $p < 1/2$ by assumption. Similarly, for n sufficiently large we can show that

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

and

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

Since $p < \lambda_a/(2\lambda_a+3)$, A is a low order term compared with $n^w/w!$. Combining the numerator and denominator,we have

$$\begin{aligned}\Phi(n, \lfloor \alpha n^p \rfloor, \lambda_a, \lambda_c) &\geq \frac{n^{\lambda_c}(\lambda_c+1)!}{w^{2\lambda_c+3}} + (\text{low order terms}) \\ &\geq n^{\lambda_c - p(2\lambda_c+3)} \frac{(\lambda_c+1)!}{\alpha^{2\lambda_c+3}} + (\text{low order terms}).\end{aligned}$$

Since $p < \lambda_c/(2\lambda_c+3)$ we obtain the desired result. QED.

Lemma 7: Let p, q, and r be constants, $0 < p, q, r < 1$. Then if $p < 1/2$,

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

for any positive real α, β and γ.

Proof: Define $\lambda_a(n) = \lfloor \beta n^q \rfloor$ and $\lambda_c(n) = \lfloor \gamma n^r \rfloor$. Then

$$\lim_{n\to\infty} \min\{\lambda_a(n)/(2\lambda_a(n)+3), \lambda_c(n)/(2\lambda_c(n)+3)\} = 1/2.$$

Noting this, the result can be proved using an approach essentially identical to that used in proving Lemma 6. If $p < 1/2$ then for n sufficiently large $p < \min\{\lambda_a(n)/(2\lambda_a(n)+3), \lambda_c(n)/(2\lambda_c(n)+3)\}$ and the bounds employed in the proof of Lemma 6 can be invoked to obtain the desired result. QED.

References

[1] 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.

[2] 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.

[3] 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.

[4] 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.

[5] 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.

[6] V. K. Wei, private communication, January 1992.

[7] B. Masnick and J. Wolf, "On linear unequal error protection codes," *IEEE Transactions on Information Theory*, vol. 13, pp. 600–607, October 1967.

[8] 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.

[9] 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.

[10] R. H. Morelos-Zaragoza, *Multi-level Error Correcting Codes.* PhD thesis, Univ. of Hawaii, May 1992.

[11] 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.

[12] R. M. Wilson, "Cyclotomy and difference families in elementary abelian groups," *J. Number Theory*, vol. 4, pp. 17–47, 1972.

[13] H. Hanani, "The existence and constructions of balanced incomplete block designs," *Ann. Math. Statist.*, vol. 32, pp. 361–386, 1961.

[14] R. C. Bose, "On the construction of balanced incomplete block design," *Ann. Eugenics*, vol. 9, pp. 353–399, 1939.

n	w	$\|\mathcal{C}\|$	Bound from Thm. 2	n	w	$\|\mathcal{C}\|$	Bound from Thm. 2	n	w	$\|\mathcal{C}\|$	Bound from Thm. 2	n	w	$\|\mathcal{C}\|$	Bound from Thm. 2
5	3	1	1	17	4	2	2	13	5	1	1	19	6	1	1
13	3	3	4	73	4	9	12	37	5	3	3	199	6	11	13
29	3	7	9	89	4	11	14	61	5	5	6	487	6	27	32
37	3	9	12	97	4	12	16	73	5	6	6	829	6	46	55
53	3	13	17	193	4	24	32	97	5	8	9	883	6	49	58
61	3	15	20	233	4	29	38	181	5	15	18	*	*	*	*
101	3	25	33	241	4	30	40	193	5	16	19	*	*	*	*
109	3	27	36	281	4	35	46	241	5	20	24	*	*	*	*
149	3	37	49	401	4	50	66	313	5	26	31	*	*	*	*
157	3	39	52	*	*	*	*	337	5	28	33	*	*	*	*
173	3	43	57	*	*	*	*	349	5	29	34	*	*	*	*
181	3	45	60	*	*	*	*	373	5	31	37	*	*	*	*
197	3	49	65	*	*	*	*	409	5	34	40	*	*	*	*
*	*	*	*	*	*	*	*	421	5	35	42	*	*	*	*
*	*	*	*	*	*	*	*	541	5	45	54	*	*	*	*
*	*	*	*	*	*	*	*	577	5	48	57	*	*	*	*

Table 1: The cardinality of $(n, w, 2, 1)$ OOC's constructed using the technique of Section 5.1.

n	w	$\|\mathcal{C}\|$	Bound from Thm. 2	n	w	$\|\mathcal{C}\|$	Bound from Thm. 2	n	w	$\|\mathcal{C}\|$	Bound from Thm. 2	n	w	$\|\mathcal{C}\|$	Bound from Thm. 2
41	4	5	6	13	5	1	1	641	8	20	22	281	9	7	7
73	4	9	12	37	5	3	3	929	8	29	33	401	9	10	11
89	4	11	14	61	5	5	6	1217	8	38	43	521	9	13	14
97	4	12	16	73	5	6	7	1409	8	44	50	601	9	15	16
113	4	14	18	97	5	8	9	1601	8	50	57	761	9	19	21
137	4	17	22	109	5	9	10	*	*	*	*	881	9	22	24
193	4	24	32	157	5	13	15	*	*	*	*	1201	9	30	33
233	4	29	38	181	5	15	18	*	*	*	*	1361	9	34	37
241	4	30	40	193	5	16	19	*	*	*	*	1481	9	37	41
257	4	32	42	229	5	19	22	*	*	*	*	1601	9	40	44
281	4	35	46	241	5	20	24	*	*	*	*	1721	9	43	47
313	4	39	52	277	5	23	27	*	*	*	*	1801	9	45	50
337	4	42	56	313	5	26	31	*	*	*	*	*	*	*	*
353	4	44	58	337	5	28	33	*	*	*	*	*	*	*	*
401	4	50	66	349	5	29	34	*	*	*	*	*	*	*	*
*	*	*	*	373	5	31	*	*	*	*	*	*	*	*	*
*	*	*	*	397	5	33	*	*	*	*	*	*	*	*	*
*	*	*	*	409	5	34	*	*	*	*	*	*	*	*	*
*	*	*	*	421	5	35	*	*	*	*	*	*	*	*	*
*	*	*	*	541	5	45	*	*	*	*	*	*	*	*	*
*	*	*	*	577	5	48	*	*	*	*	*	*	*	*	*

Table 2: The cardinality of $(n, w, 2, 1)$ OOC's constructed using the technique of Section 5.2.

n	w	$\|\mathcal{C}\|$	n	w	$\|\mathcal{C}\|$	n	w	$\|\mathcal{C}\|$
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 3: The cardinality of $(n, w, 1, 1)$ OOC's constructed using the technique of Section 5.3. All codes are optimal

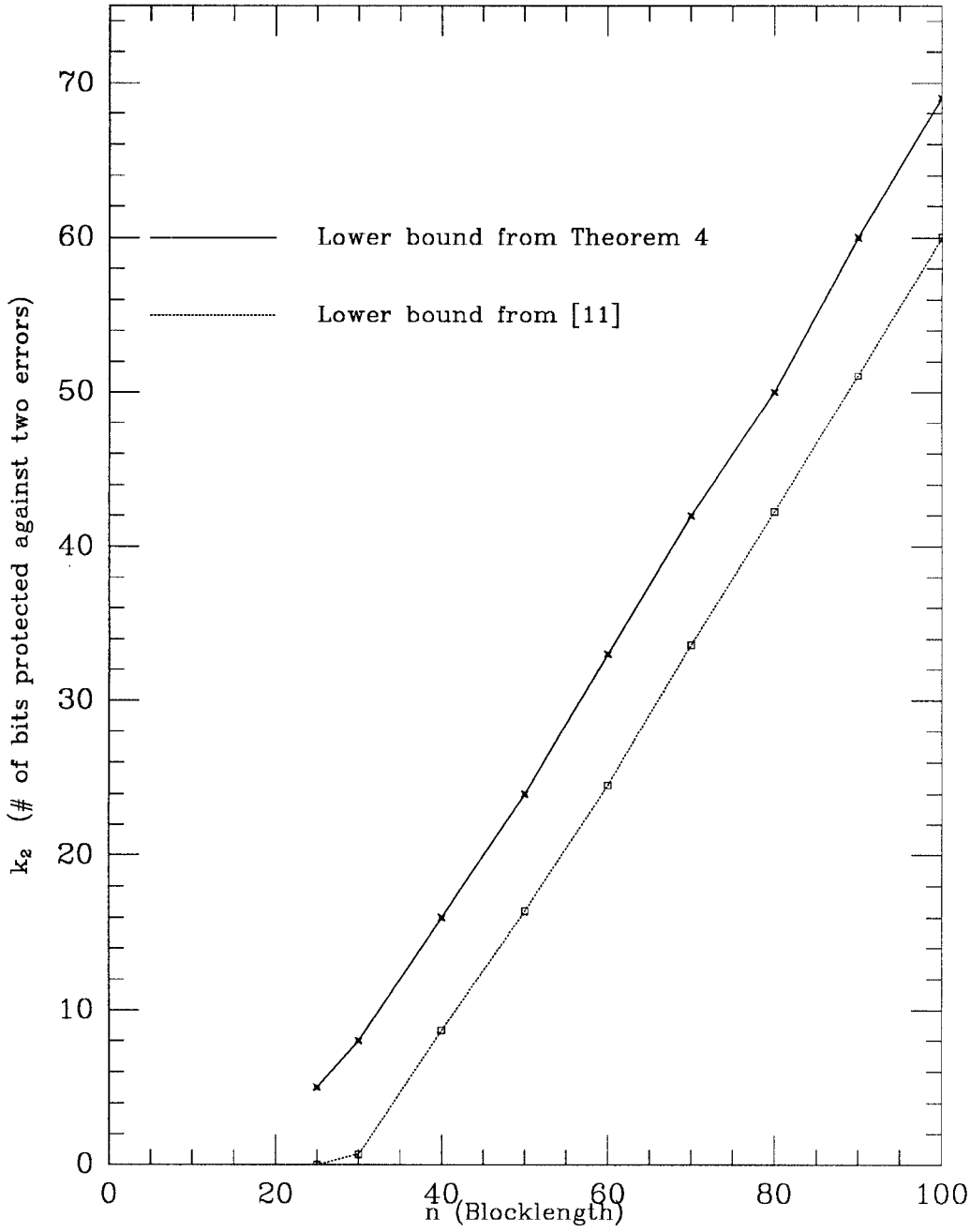


Figure 1: Lower bounds on the number of message bits that can be protected against two errors while $\lfloor \log_2 n \rfloor$ bit are protected against single errors. The bound from Theorem 4 assumes constant weight codewords with $w = \lfloor n/2 \rfloor$.