DIFFERENCE SET CODES: CODES WITH SQUARED EUCLIDEAN DISTANCE OF SIX FOR PARTIAL RESPONSE CHANNELS

Khaled A.S. Abdel-Ghaffar
University of California
Dept. Electrical & Comp. Eng.
Davis, CA 95616, USA
ghaffar@ece.ucdavis.edu

Øyvind Ytrehus
University of Bergen
Department of Informatics
N-5020, Bergen, Norway
oyvind@ii.uib.no

This work was supported by

NFR, Grants 107542/410 and 107623/420,

and by

NSF, Grant NCR 96-12354.

February 1998

Abstract

We present a new construction of block codes for the $(1-D)$-PR channel. The codewords in the code correspond to constant-sum subsets of a difference set. It is shown that at the output of a noiseless $(1-D)$-PR channel, the minimum squared Euclidean distance of such a code is at least six, compared to two for the uncoded system. This construction yields larger code rates than previously known codes with the same minimum distance for large code lengths. The construction technique also imposes upper bounds on the decoding complexity of the codes.

Index terms: Block codes, difference sets, trellis decoding complexity, partial-response channels.

I Introduction

Partial-response (PR) techniques are commonly used in modern magnetic and optical recording channels. Several coding schemes have been developed in the last decade to enhance the reliability of channels characterized by PR polynomials. This work is mainly concerned with block codes for the $(1-D)$-PR channel. These codes can also be used, more generally, for channels characterized by a polynomial of the form $1 \mp D^m$, where m is a positive integer, by suitable interleaving and transformation of symbols. An excellent exposition of these codes can be found in [13] where three general classes of codes are described: precoded binary codes with large minimum Hamming distance [14], spectral null codes [9], and concatenated codes. Codes with minimum squared Euclidean distance four are emphasized in [13]. It should be pointed out that the squared Euclidean distance according to our definition is twice that in [13]. In particular, squared Euclidean distance six in the present work corresponds to squared Euclidean distance three in [13].

We present a new class of block codes, which we call difference set codes, with minimum squared Euclidean distance of six. Except for small code lengths, the new codes have larger rates than other known codes.

In Section II, we introduce the basic notation and review some previously known results. Section III describes the difference set code construction, and contains theorems on essential code parameters. A sphere-packing argument, discussed in Section IV, provides an upper bound on the size of codes with minimum squared Euclidean distance six. Decoding is discussed in Section V. In Section VI, an alternative construction is presented which utilizes the fact that we can construct many disjoint difference set codes; these are then employed in a generalized concatenation code construction. Finally, a brief discussion of computer searches for optimum codes with minimum squared Euclidean distance six follows in Section VII.

II Preliminaries

We will use the following notation. Let $x_1, x_2, \ldots$ be a sequence, which we denote by $\mathbf{x}$. If confusion is unlikely, we may write $\mathbf{x} = x_1 x_2 \cdots$. In the present work, we say that the sequence $\mathbf{x}$ is binary, ternary, or real, if $x_i \in \{0,1\}$, $x_i \in \{0,1,-1\}$, or x_i is real, respectively, for all i. We may denote -1 in a ternary sequence by $\bar{1}$. If $x_i = 0$ for all i, we say that $\mathbf{x}$ is a zero sequence, denoted by $\mathbf{0}$. Corresponding to any binary sequence $\mathbf{x}$, we define the ternary sequence $\mathbf{x}^T = x_1^T, x_2^T, \ldots$ by

$$x_i^T = \begin{cases} 0 & \text{if } x_i = 0, \\ 1 & \text{if } x_i = 1 \text{ and } \sum_{j=1}^{i-1} x_j \equiv 0 \pmod 2, \\ -1 & \text{if } x_i = 1 \text{ and } \sum_{j=1}^{i-1} x_j \equiv 1 \pmod 2, \end{cases}$$

for $i = 1, 2, \ldots$. Thus, the ternary sequence $\mathbf{x}^T$ is obtained from the binary sequence $\mathbf{x}$ by alternating the signs of the 1's in $\mathbf{x}$ starting with $+$.

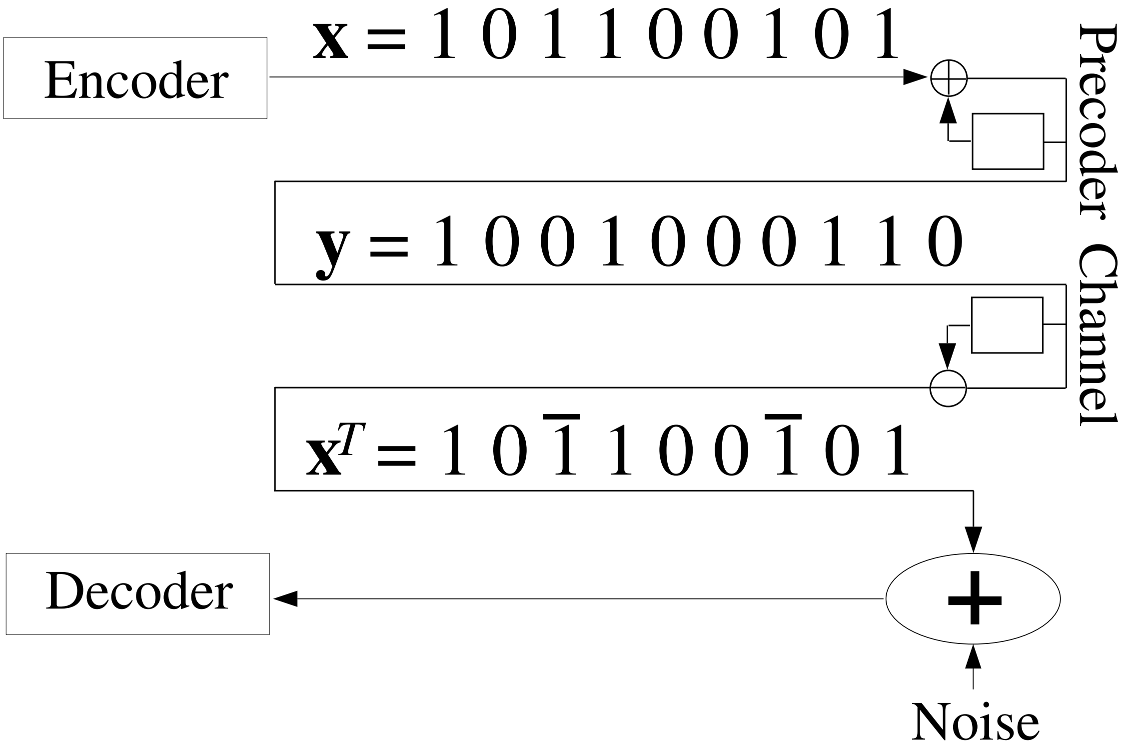
In particular, if ϵ_1 and ϵ_{-1} are the numbers of 1's and -1's in $\mathbf{x}^T$, respectively, then $\epsilon_1 = \epsilon_{-1}$ if $\mathbf{x}$ has an even number of 1's, and $\epsilon_1 = \epsilon_{-1} + 1$ if $\mathbf{x}$ has an odd number of 1's. Clearly, by taking absolute values, $\mathbf{x}$ can be recovered from $\mathbf{x}^T$. Notice that if $\mathbf{x}_i$, for $i \geq 1$, are binary sequences and $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \ldots$ denotes their concatenation, then $(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \ldots)^T = \mathbf{x}_1^T, (-1)^{s_1}\mathbf{x}_2^T, (-1)^{s_2}\mathbf{x}_3^T, \ldots$, where s_i, for $i \geq 1$, is the total number of 1's in the binary sequence $\mathbf{x}_1, \ldots, \mathbf{x}_i$. If $\mathcal{S}$ is a set of binary sequences, let $\mathcal{S}^T = \{x^T : x \in \mathcal{S}\}$. Furthermore, if $\mathcal{S}$ is a set of real sequences, let $-\mathcal{S} = \{-\mathbf{x} : \mathbf{x} \in \mathcal{S}\}$. The weight of a real sequence is the number of its nonzero components, which may be infinite. We define the squared Euclidean distance between the real sequences $\mathbf{x}$ and $\mathbf{x}'$ of the same length as $d_E^2(\mathbf{x}, \mathbf{x}') = \sum_i (x_i - x_i')^2$, provided that this sum exists.

In case the sequences are of infinite length and the sum does not converge, we write $d_E^2(\mathbf{x}, \mathbf{x}') = \infty$.

In this paper, we consider the following block coding scheme. We are given a set of M messages and with each message we associate a distinct binary sequence $\mathbf{c}$ of length n. These sequences are called codewords and the set of all codewords forms the code $\mathcal{C}$. The code rate is defined by $R = \log_2 M/n$. The code is used as follows. A sequence of messages $\mathbf{m} = m_1, m_2, \ldots$ is mapped to a binary sequence $\mathbf{x} = x_1, x_2, \ldots$ obtained by cascading the codewords associated with the messages, i.e., $\mathbf{x} = \mathbf{c}_1, \mathbf{c}_2, \ldots$, where $\mathbf{c}_i = (x_{(i-1)n+1}, \ldots, x_{in})$ is the codeword associated with message m_i for $i \geq 1$. The binary sequence $\mathbf{x}$ is then passed through a precoder that produces the precoded binary sequence $\mathbf{y} = y_1, y_2, \ldots$ given by $y_i \equiv x_i + y_{i-1} \pmod 2$ for $i \geq 1$, where $y_0 = 0$ and $y_i \in \{0, 1\}$. Notice that $y_i \equiv \sum_{j=1}^{i} x_j \pmod 2$. The precoded sequence $\mathbf{y}$ is transmitted over the $(1 - D)$ channel producing the real sequence $\mathbf{z} = z_1, z_2, \ldots$ given by $z_i = y_i - y_{i-1} + e_i$, where $\mathbf{e} = e_1, e_2, \ldots$ is a real sequence representing the noise introduced by the channel. We assume that the e_i's are independent and identically distributed zero-mean Gaussian random variables. It is easy to verify that $y_i - y_{i-1} = x_i^T$ for $i \geq 1$, i.e., $\mathbf{z} = \mathbf{x}^T + \mathbf{e}$. In particular, the noiseless output of the channel is $\mathbf{x}^T = (\mathbf{c}_1, \mathbf{c}_2, \ldots)^T = \mathbf{c}_1^T, (-1)^{s_1}\mathbf{c}_2^T, (-1)^{s_2}\mathbf{c}_3^T, \ldots$, where s_i, for $i \geq 1$, is the total number of 1's in the binary sequence $\mathbf{c}_1, \ldots, \mathbf{c}_i$. The decoder attempts to find a binary sequence $\hat{\mathbf{x}}$ obtained by cascading codewords in the code such that the squared Euclidean distance between $\mathbf{z}$ and $\hat{\mathbf{x}}^T$ is minimum. Based on these codewords, the decoder produces an estimate for the sequence of messages, cf. Figure 1.

For the purpose of analysis and decoding, it is convenient to define a channel state oriented trellis associated with the code $\mathcal{C}$ as follows. Let $\mathcal{C}_e$ and $\mathcal{C}_o$ be the sets of

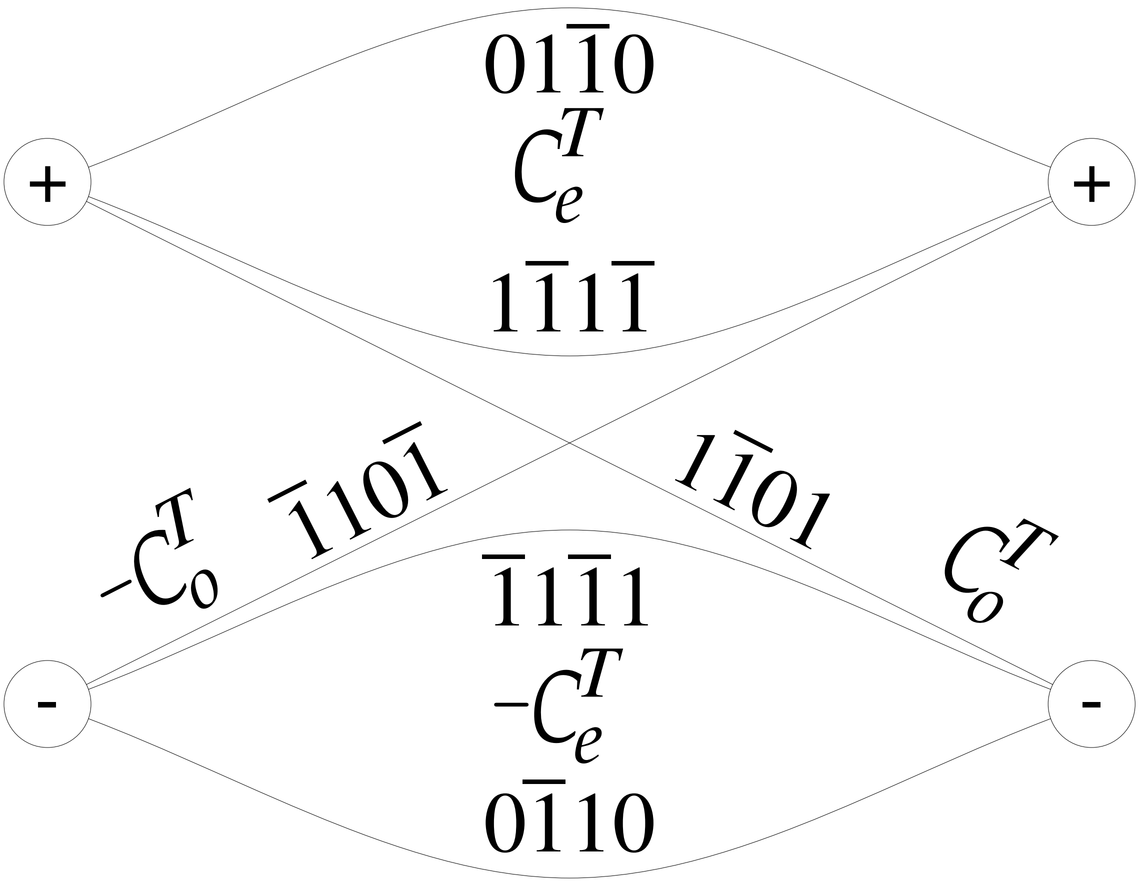
Figure 1: A precoded $1 - D$ partial response channel with an example of an input sequence $\mathbf{x}$. It is assumed that the sequence is indexed in increasing order from right to left, so that the *rightmost* symbol x_1 in $\mathbf{x}$ is the first to enter the precoder. It is further assumed that the initial symbol y_0 in the precoder and channel memory is 0, and that the input sequence is followed by a zero symbol. The sequence $\mathbf{y}$ of precoder and channel states is updated as $y_i := x_i + y_{i-1} \pmod 2$ for $i \geq 1$. The noiseless output sequence $\mathbf{x}^T$ is updated as $x_i^T := y_i - y_{i-1}$, where the subtraction is real.



codewords in $\mathcal{C}$ of even weight and odd weight, respectively. The trellis has two states denoted by '+' and '−'. Each sequence in $\mathcal{C}_e^T$ is the label of an edge from state '+' to state '+'. Similarly, sequences in $\mathcal{C}_o^T$, $-\mathcal{C}_o^T$, and $-\mathcal{C}_e^T$ are labels from states '+', '−', and '−' to states '−', '+', and '−', respectively. Since $|\mathcal{C}_e| + |\mathcal{C}_o| = |\mathcal{C}| = M$, each state has M out-going edges and M incoming edges. In order to distinguish this trellis from one that will appear in Section V, we will refer to such a trellis as a *block-oriented trellis*. Figure 2 shows a block-oriented trellis for a simple code. Notice that a label on an edge starting and ending at the same state has an even number of nonzero symbols while a label on an edge starting and ending at different states has an odd number of nonzero symbols.

Hence, a path in the block-oriented trellis starting at state '+' terminates at state '+' or state '−' if the total number of nonzero symbols in the labels on path edges is even or odd, respectively. In particular, the sequence of labels on the edges of any path in the block-oriented trellis starting at state '+' is a sequence of the form $\mathbf{c}_1^T, (-1)^{s_1}\mathbf{c}_2^T, (-1)^{s_2}\mathbf{c}_3^T, \ldots$, where $\mathbf{c}_i \in \mathcal{C}$ and s_i is the total number of 1's in the binary sequence $\mathbf{c}_1, \ldots, \mathbf{c}_i$. We conclude that there is a one-to-one correspondence between the paths in the block-oriented trellis starting at state '+' and the noiseless output sequences of the $(1 - D)$ channel. For simplicity, we will adopt the common convention of writing the squared Euclidean distance between paths when we actually mean the squared Euclidean distance between the sequences of labels on the path edges. The Viterbi algorithm may be applied to the block-oriented trellis to find a closest (sequence of labels on the edges of a) path to the noisy channel output. Once such a path is found, we take absolute values of the elements of its edge labels to obtain a sequence of codewords in $\mathcal{C}$ which are then decoded into messages.

Figure 2: A block-oriented trellis for the code $\mathcal{C} = \{0110, 1111, 1101\}$, with $\mathcal{C}_e^T = \{01\bar{1}0, 1\bar{1}1\bar{1}\}$ and $\mathcal{C}_o^T = \{1\bar{1}01\}$.



The single most important parameter used to evaluate the performance of the coding scheme is the minimum squared Euclidean distance, denoted by $d_E^2(\mathcal{C})$. It is given by the squared Euclidean distance between any pair of distinct paths starting at the state '+', minimized over all such pairs. In general, among all codes of the same rate, it is often desirable to choose a code with the largest minimum squared Euclidean distance in order to achieve a small probability of decoding error.

To characterize $d_E^2(\mathcal{C})$, it is convenient to define

$$d_=^2 = \min\{d_E^2(\mathbf{c}_1^T, \mathbf{c}_2^T) : (\mathbf{c}_1, \mathbf{c}_2) \in (\mathcal{C}_e \times \mathcal{C}_e) \cup (\mathcal{C}_o \times \mathcal{C}_o), \mathbf{c}_1 \neq \mathbf{c}_2\}, \tag{1}$$

$$d_<^2 = \min\{d_E^2(\mathbf{c}_1^T, \mathbf{c}_2^T) : (\mathbf{c}_1, \mathbf{c}_2) \in \mathcal{C}_e \times \mathcal{C}_o\}, \tag{2}$$

and

$$d_>^2 = \min\{d_E^2(\mathbf{c}_1^T, \mathbf{c}_2^T) : (\mathbf{c}_1, \mathbf{c}_2) \in \mathcal{C}_e \times (-\mathcal{C}_o)\}, \tag{3}$$

where we write $d_=^2 = \infty$ in case neither $\mathcal{C}_e$ nor $\mathcal{C}_o$ has more than one sequence, and $d_>^2 = d_<^2 = \infty$ in case $\mathcal{C}_e$ or $\mathcal{C}_o$ is empty. The parameter $d_=^2$ is the minimum squared Euclidean distance between distinct parallel edges, i.e., edges starting at the same state and terminating at the same state. The parameter $d_<^2$ is the minimum squared Euclidean distance between diverging edges, i.e., edges starting at the same state and terminating at different states. Finally, the parameter $d_>^2$ is the minimum squared Euclidean distance between converging edges, i.e., edges starting at different states and terminating at the same state. Clearly, if $\mathbf{c}_1 = (c_{11}, \ldots, c_{1n})$ and $\mathbf{c}_2 = (c_{21}, \ldots, c_{2n})$ are binary sequences, then $d_E^2(\mathbf{c}_1^T, \mathbf{c}_2^T) = \sum_{i=1}^n (c_{1i}^T - c_{2i}^T)^2$ is congruent modulo two to the number of indices i for which $c_{1i} \neq c_{2i}$. Hence, $d_E^2(\mathbf{c}_1^T, \mathbf{c}_2^T)$ is congruent modulo two to the difference of the weights of the binary sequences $\mathbf{c}_1$ and

$\mathbf{c}_2$. Since $\mathcal{C}_e$ and $\mathcal{C}_o$ consist of even and odd weight codewords, respectively, it follows that $d_=^2$ is even and both $d_<^2$ and $d_>^2$ are odd, provided that they are finite.

In particular, for an uncoded system we have $d_<^2 = d_>^2 = 1$. Hence by considering pairs of paths that converge to a given state, it follows that the minimum squared Euclidean distance is equal to two.

Notice that all nonzero labels on edges starting from state '+' have 1 as their first nonzero symbol, all nonzero labels on edges starting from state '−' have -1 as their first nonzero symbol, and there is an edge with an all zero label if and only if $\mathcal{C}$ has the zero codeword $\mathbf{0}$. This edge starts and ends at the same state. If we consider two "infinite-distinct" paths, that is two paths of infinite length starting at the same state and not passing through any common intermediate state thereafter, the squared Euclidean distance between them is finite if and only if $\mathbf{0} \in \mathcal{C}$ and, except for a finite number of (initial) edges, the two paths trace edges with zero labels. In this case, the squared Euclidean distance minimized over all such paths is $d_<^2$. We conclude that

$$d_E^2(\mathcal{C}) = \begin{cases} \min\{d_=^2, d_<^2\} & \text{if } \mathbf{0} \in \mathcal{C}, \\ \min\{d_=^2, d_<^2 + d_>^2\} & \text{if } \mathbf{0} \notin \mathcal{C}. \end{cases} \tag{4}$$

We should mention that having such "infinite-distinct" paths such that the squared Euclidean distance between them is finite does not lead to catastrophic error propagation since, except for a finite number of edges, these paths follow edges with zero labels corresponding to the same codeword and hence the same message. Indeed, although a block-oriented trellis is used for decoding, we are considering a block-type coding scheme.

Example 1 Consider the code $\mathcal{C} = \{0110, 1111, 1101\}$. We have $\mathcal{C}_e^T = \{01\bar{1}0, 1\bar{1}1\bar{1}\}$ and $\mathcal{C}_o^T = \{1\bar{1}01\}$. It can be easily checked that $d_=^2 = 10$, $d_<^2 = 5$, and $d_>^2 = 3$. Hence,

$d_E^2(\mathcal{C}) = 8$. The block-oriented trellis of this code is shown in Figure 2.

A simple construction technique for block codes is based on restricting the codewords to have even weight only. If $\mathcal{C}$ is an even weight code, then $\mathcal{C}_e = \mathcal{C}$ and $\mathcal{C}_o$ is empty. From (1), (2), and (3), it follows that $d_=^2 = \min\{d_E^2(\mathbf{c}_1^T, \mathbf{c}_2^T) : \mathbf{c}_1, \mathbf{c}_2 \in \mathcal{C}, \mathbf{c}_1 \neq \mathbf{c}_2\}$ and $d_<^2 = d_>^2 = \infty$. The minimum squared Euclidean distance, $d_E^2(\mathcal{C})$, of $\mathcal{C}$, as given by (4), simply equals $d_=^2$. Since $\mathcal{C}_o$ is empty, no path in the block-oriented trellis starting at state '+' deviates from this state. In particular, the block-oriented trellis may be reduced to a single state with parallel edges labeled by sequences in $\mathcal{C}^T$. In other words, the noiseless channel output corresponding to a codeword $\mathbf{x}$ in an even weight code $\mathcal{C}$ is $\mathbf{x}^T$, while it is either $\mathbf{x}^T$ or $-\mathbf{x}^T$, depending on the parity of the total number of 1's in all previous codewords, if $\mathcal{C}$ is not an even weight code.

Tolhuizen, Immink, and Hollmann [13] presented three construction methods of even weight block codes: precoding, spectral nulls, and concatenation. First we will explain their terminology since it differs from ours. In our work, we define the codewords to be the sequences of length n that are input to the precoder. In [13], the codewords are defined to be the output sequences of the precoder, after deleting the last bit in each sequence, which is zero for any even weight code. Thus, if $\mathbf{x} = (x_1, \ldots, x_n)$ is a codeword in an even weight code according to our definition, then $(y_1, \ldots, y_{n-1})$, where $y_i \equiv x_i + y_{i-1} \pmod 2$ and $y_0 = 0$, is the corresponding codeword according to the definition of [13]. Notice that $y_i \equiv \sum_{j=1}^{i} x_j \pmod 2$ and in particular $y_n = 0$ as mentioned above. The noiseless channel output is $\mathbf{x}^T = (x_1^T, \ldots, x_n^T)$, where $x_i^T = y_i - y_{i-1}$. In conclusion, there is a one-to-one correspondence between codes of length $n-1$ in [13] and even weight codes of length n as defined here. Another discrepancy between our terminology and the one used in [13] is related to the definition of squared Euclidean distance. Our squared Euclidean distance is twice that in [13]. To avoid confusion, we will translate the parameters of codes described in [13] to agree with the terminology developed in this paper. For example, when we refer to an even weight code of length 11 and minimum squared Euclidean distance of 6 in [13], it should be found there as a code of length 10 and minimum squared Euclidean distance of 3.

In Table I, we list the maximum size of even weight codes of minimum squared Euclidean distance 6 and length $n \leq 24$ constructed using the three methods in [13]. For the first method, precoding, this maximum size equals the maximum size of an even weight code of minimum Hamming distance 6 and length n. The values and bounds listed in the table are quoted from Table 9.1 in [6]. For the second method, spectral nulls, the maximum size equals the size of the set of even weight binary sequences $\mathbf{c} = (c_1, \ldots, c_n)$ such that $\sum_{i=1}^{n} i c_i^T = d_1$ and $\sum_{i=1}^{n} i^2 c_i^T = d_2$, maximized over all choices of d_1 and d_2. For the third method, concatenation, several alternatives are described in [13] by choosing different inner and outer codes. The results in Table I are based on codes explicitly defined in [13]. In particular, the listed maximum sizes are achieved by codes of length $2rm + 1$ and size $(2^r + 2)^m$, where $r \geq 3$ and $m \geq 1$ are integers (see Example 3 in [13]).

III Difference Set Codes

A difference set is a set $\mathcal{A}_n = \{a_1, a_2, \ldots, a_n\}$ of integers such that all the nonzero differences $a_i - a_j$ for $i \neq j$ are distinct. Without loss of generality, we will assume in the following that $0 = a_1 < a_2 < \cdots < a_n$. In Table II we list difference sets $\mathcal{A}_n$ of size $n \leq 24$. We note that the difference sets described here are sometimes called difference triangle sets. They should be distinguished from perfect systems of difference sets, where all the positive differences are distinct and consecutive, as well as other families of so-called difference sets that belong to a given group, where each nonzero group element occurs equally often as a difference [5].

We will describe now a new technique to construct block codes of minimum squared Euclidean distance (at least) 6 based on difference sets. Suppose that w_e and w_o are (possibly negative) integers such that $w_e \neq 0$, $|w_e - w_o| \notin \mathcal{A}_n$, and $|w_e + w_o| \notin \mathcal{A}_n$. The last two conditions will be expressed as $|w_e \pm w_o| \notin \mathcal{A}_n$. Let

$$\mathcal{C}_e(\mathcal{A}_n, w_e) = \{\mathbf{x} = (x_1, \ldots, x_n) \in \{0,1\}^n : \mathbf{x} \text{ has even weight and } \sum_{i=1}^{n} a_i x_i^T = w_e\} \tag{5}$$

and

$$\mathcal{C}_o(\mathcal{A}_n, w_o) = \{\mathbf{x} = (x_1, \ldots, x_n) \in \{0,1\}^n : \mathbf{x} \text{ has odd weight and } \sum_{i=1}^{n} a_i x_i^T = w_o\}. \tag{6}$$

The difference set code $\mathcal{C}(\mathcal{A}_n, w_e, w_o)$ is defined by

$$\mathcal{C}(\mathcal{A}_n, w_e, w_o) = \mathcal{C}_e(\mathcal{A}_n, w_e) \cup \mathcal{C}_o(\mathcal{A}_n, w_o). \tag{7}$$

Notice that $\mathcal{C}_e(\mathcal{A}_n, w_e)$ and $\mathcal{C}_o(\mathcal{A}_n, w_o)$ are the sets of even weight and odd weight codewords in $\mathcal{C}(\mathcal{A}_n, w_e, w_o)$, respectively. In particular, the sequences in $\mathcal{C}_e^T(\mathcal{A}_n, w_e)$ and $\mathcal{C}_o^T(\mathcal{A}_n, w_o)$ are the labels of edges from state '+' to states '+' and '−', respectively, and the sequences in $-\mathcal{C}_e^T(\mathcal{A}_n, w_e)$ and $-\mathcal{C}_o^T(\mathcal{A}_n, w_o)$ are the labels of edges from state '−' to states '−' and '+', respectively.

Example 2 From Table II, we have the difference set $\mathcal{A}_4 = \{0, 2, 5, 6\}$ for $n = 4$. In Table III we computed the values of $\sum_{i=1}^{4} a_i x_i^T$ for all binary sequences $\mathbf{x} =$

(x_1, x_2, x_3, x_4). Notice that the pair $w_e = -3$ and $w_o = 4$ is a valid choice to define a difference set code since $w_e \neq 0$ and $|w_e \pm w_o| \notin \mathcal{A}_4$. For this choice, $\mathcal{C}_e(\mathcal{A}_4, -3) = \{0110, 1111\}$, $\mathcal{C}_o(\mathcal{A}_4, 4) = \{1101\}$, and $\mathcal{C}(\mathcal{A}_4, -3, 4) = \{0110, 1111, 1101\}$. This code was examined in Example 1 where it was shown that its minimum squared Euclidean distance is 8.

In Theorem 1 we show that the minimum squared Euclidean distance of the difference set code $\mathcal{C}(\mathcal{A}_n, w_e, w_o)$ is at least 6. The proof depends on the following three lemmas.

Lemma 1 *Let $\mathbf{c}_1 = (c_{11}, \ldots, c_{1n})$ and $\mathbf{c}_2 = (c_{21}, \ldots, c_{2n})$ be distinct binary sequences whose weights are congruent modulo two and whose associated ternary sequences $\mathbf{c}_1^T = (c_{11}^T, \ldots, c_{1n}^T)$ and $\mathbf{c}_2^T = (c_{21}^T, \ldots, c_{2n}^T)$ satisfy $\sum_{i=1}^n a_i c_{1i}^T = \sum_{i=1}^n a_i c_{2i}^T$. Then, $d_E^2(\mathbf{c}_1^T, \mathbf{c}_2^T) \geq 6$.*

Proof. Suppose that $d_E^2(\mathbf{c}_1^T, \mathbf{c}_2^T) < 6$. Since the weights of $\mathbf{c}_1$ and $\mathbf{c}_2$ are congruent modulo two, then $d_E^2(\mathbf{c}_1^T, \mathbf{c}_2^T)$ is even and, therefore,

$$d_E^2(\mathbf{c}_1^T, \mathbf{c}_2^T) \in \{2, 4\}. \tag{8}$$

If $c_{1j}^T = 1$ and $c_{2j}^T = -1$ for some j, then the numbers of nonzero symbols in the subsequences $(c_{11}^T, \ldots, c_{1(i-1)}^T)$ and $(c_{21}^T, \ldots, c_{2(i-1)}^T)$ are even and odd, respectively. In particular, $d_E^2(\mathbf{c}_1^T, \mathbf{c}_2^T) > 4$, which contradicts (8). The same argument holds if $c_{1j}^T = -1$ and $c_{2j}^T = 1$. Hence, we conclude that the sequence $\mathbf{u} = (u_1, \ldots, u_n) = \mathbf{c}_1^T - \mathbf{c}_2^T$ is ternary. From (8), $\mathbf{u}$ has either a single 1 and a single -1 or two 1's and two -1's. In the first case $\sum_{i=1}^n a_i u_i$ is a difference $a_{i_1} - a_{i_2}$, and in the second case $\sum_{i=1}^n a_i u_i$ is a difference between two differences $(a_{i_1} - a_{i_2}) - (a_{i_3} - a_{i_4})$, where i_1, i_2, i_3, and i_4, are distinct. Since $\sum_{i=1}^n a_i c_{1i}^T = \sum_{i=1}^n a_i c_{2i}^T$ and $\mathbf{u} = \mathbf{c}_1^T - \mathbf{c}_2^T$, it follows that $\sum_{i=1}^n a_i u_i = 0$. This contradicts the assumption that $\mathcal{A}_n = \{a_1, \ldots, a_n\}$ is a difference set. Q.E.D.

Lemma 2 *Let $\mathbf{c}_1 = (c_{11}, \ldots, c_{1n})$ and $\mathbf{c}_2 = (c_{21}, \ldots, c_{2n})$ be distinct binary sequences whose weights are incongruent modulo two and whose associated ternary sequences $\mathbf{c}_1^T = (c_{11}^T, \ldots, c_{1n}^T)$ and $\mathbf{c}_2^T = (c_{21}^T, \ldots, c_{2n}^T)$ satisfy $|\sum_{i=1}^n a_i c_{1i}^T - \sum_{i=1}^n a_i c_{2i}^T| \notin \mathcal{A}_n$. Then, $d_E^2(\mathbf{c}_1^T, \mathbf{c}_2^T) \geq 3$.*

Proof. Suppose that $d_E^2(\mathbf{c}_1^T, \mathbf{c}_2^T) < 3$. Since the weights of $\mathbf{c}_1$ and $\mathbf{c}_2$ are incongruent modulo two, then $d_E^2(\mathbf{c}_1^T, \mathbf{c}_2^T)$ is odd and, therefore, $d_E^2(\mathbf{c}_1^T, \mathbf{c}_2^T) = 1$. It follows that there is a unique j for which $(c_{1j}^T, c_{2j}^T) \in \{(\pm 1, 0), (0, \pm 1)\}$ and $c_{1i}^T = c_{2i}^T$ for all $i \neq j$. Hence,

$$|\sum_{i=1}^n a_i c_{1i}^T - \sum_{i=1}^n a_i c_{2i}^T| = a_j \in \mathcal{A}_n,$$

contradicting the assumption that the left hand side does not belong to $\mathcal{A}_n$. Q.E.D.

Lemma 3 *Let $\mathbf{c}_1 = (c_{11}, \ldots, c_{1n})$ and $\mathbf{c}_2 = (c_{21}, \ldots, c_{2n})$ be distinct binary sequences whose weights are incongruent modulo two and whose associated ternary sequences $\mathbf{c}_1^T = (c_{11}^T, \ldots, c_{1n}^T)$ and $\mathbf{c}_2^T = (c_{21}^T, \ldots, c_{2n}^T)$ satisfy $|\sum_{i=1}^n a_i c_{1i}^T + \sum_{i=1}^n a_i c_{2i}^T| \notin \mathcal{A}_n$. Then, $d_E^2(\mathbf{c}_1^T, -\mathbf{c}_2^T) \geq 3$.*

The proof is similar to that of Lemma 2.

Theorem 1 *The minimum squared Euclidean distance of a difference set code is at least equal to 6.*

Proof. Let $\mathcal{C} = \mathcal{C}(\mathcal{A}_n, w_e, w_o)$ be a difference set code. From (5), (6), and Lemma 1 it follows that if $\mathbf{c}_1$ and $\mathbf{c}_2$ are distinct sequences such that both belong to either

$\mathcal{C}_e = \mathcal{C}_e(\mathcal{A}_n, w_e)$ or $\mathcal{C}_o = \mathcal{C}_o(\mathcal{A}_n, w_o)$, then $d_E^2(\mathbf{c}_1^T, \mathbf{c}_2^T) \geq 6$. Hence, $d_=^2$, as given in (1), is at least equal to 6. Also, from (5), (6), and Lemmas 2 and 3 it follows that if $\mathbf{c}_1$ belongs to $\mathcal{C}_e(\mathcal{A}_n, w_e)$ and $\mathbf{c}_2$ belongs to either $\mathcal{C}_o(\mathcal{A}_n, w_o)$ or $-\mathcal{C}_o(\mathcal{A}_n, w_o)$, then $d_E^2(\mathbf{c}_1^T, \mathbf{c}_2^T) \geq 3$. Hence, $d_<^2$ and $d_>^2$, as given in (2) and (3), are at least equal to 3. Finally, notice that the zero sequence $\mathbf{0}$ does not belong to $\mathcal{C}_e(\mathcal{A}_n, w_e)$ since $w_e \neq 0$. From (4), we conclude that $d_E^2(\mathcal{C}(\mathcal{A}_n, w_e, w_o)) \geq 6$. Q.E.D.

Notice that the set $\mathcal{C}_e(\mathcal{A}_n, w_e)$, where w_e may be zero, defines an even weight difference set code of minimum squared Euclidean distance (at least) 6.

Let $M_e(\mathcal{A}_n, w_e)$, $M_o(\mathcal{A}_n, w_o)$, and $M(\mathcal{A}_n, w_e, w_o)$ be the sizes of the sets $\mathcal{C}_e(\mathcal{A}_n, w_e)$, $\mathcal{C}_o(\mathcal{A}_n, w_o)$, and $\mathcal{C}(\mathcal{A}_n, w_e, w_o)$, respectively. From (7) we have

$$M(\mathcal{A}_n, w_e, w_o) = M_e(\mathcal{A}_n, w_e) + M_o(\mathcal{A}_n, w_o). \tag{9}$$

For a given difference set $\mathcal{A}_n$, the maximum sizes of a difference set code and an even weight difference set code are given by

$$M(\mathcal{A}_n) = \max_{\substack{w_e, w_o \\ w_e \neq 0, |w_e \pm w_o| \notin \mathcal{A}_n}} M(\mathcal{A}_n, w_e, w_o) \tag{10}$$

and

$$M_e(\mathcal{A}_n) = \max_{w_e} M_e(\mathcal{A}_n, w_e), \tag{11}$$

respectively.

Example 3 From Table III we know that in the case $n = 4$ and $\mathcal{A}_4 = \{0, 2, 5, 6\}$

$$M_o(\mathcal{A}_4, w) = M_e(\mathcal{A}_4, -w) = \begin{cases} 1 & \text{if } w \in \{0,1,2,4,5,6\}, \\ 2 & \text{if } w = 3, \\ 0 & \text{if otherwise.} \end{cases}$$

Note that $M(\mathcal{A}_4) \leq 3$ since $M_e(\mathcal{A}_4, w_e) + M_o(\mathcal{C}_4, w_o) \geq 4$ if and only if $w_e = -3$ and $w_o = 3$ and, hence, $|w_e + w_o| = 0 \in \mathcal{A}_4$. On the other hand, $M(\mathcal{A}_4, -3, 4) =$

$M_e(\mathcal{A}_4, -3) + M_o(\mathcal{A}_4, 4) = 3$ and, thus, $M(\mathcal{A}_4) = 3$. The code $\mathcal{C}(\mathcal{A}_4, -3, 4)$ is described in Example 2.

The fact that $M_o(\mathcal{A}_4, w) = M_e(\mathcal{A}_4, -w)$ and the symmetry of these two as functions of w is not peculiar to Example 3 as the following two lemmas demonstrate.

Lemma 4 *For any difference set $\mathcal{A}_n = \{a_1, \ldots, a_n\}$ and any integer w,*

$$M_o(\mathcal{A}_n, w) = M_e(\mathcal{A}_n, a_1 - w).$$

Proof. Let $\mathbf{x} = (x_1, \ldots, x_n)$ be a binary sequence of length n and odd weight such that $\sum_{i=1}^n a_i x_i^T = w$. Let $\mathbf{y} = (y_1, \ldots, y_n)$ be the binary sequence of length n defined by

$$y_i = \begin{cases} x_1 + 1 \pmod 2 & \text{if } i = 1, \\ x_i & \text{if } 2 \le i \le n. \end{cases}$$

Clearly $\mathbf{y}$ has even weight. Since

$$x_i^T = (-1)^{\sum_{j=1}^{i-1} x_j} x_i,$$

where $1 \le i \le n$, and a similar expression holds for y_i^T, we have

$$y_i^T = \begin{cases} 1 - x_1^T & \text{if } i = 1, \\ -x_i^T & \text{if } 2 \le i \le n. \end{cases}$$

Therefore,

$$\sum_{i=1}^n a_i y_i^T = a_1 - \sum_{i=1}^n a_i x_i^T = a_1 - w.$$

As $\mathbf{x}$ runs over all odd weight sequences for which $\sum_{i=1}^n a_i x_i^T = w$, $\mathbf{y}$ runs over all even weight sequences for which $\sum_{i=1}^n a_i y_i^T = a_1 - w$. Hence, the numbers of these sequences are equal. Q.E.D.

Lemma 5 *For any difference set $\mathcal{A}_n = \{a_1, \ldots, a_n\}$ and any integer w,*

$$M_e(\mathcal{A}_n, w) = M_e(\mathcal{A}_n, a_1 - a_n - w)$$

and

$$M_o(\mathcal{A}_n, w) = M_o(\mathcal{A}_n, a_1 + a_n - w).$$

Proof. An argument similar to the one used in Lemma 4 proves the result. Q.E.D.

The following result gives bounds on the maximum sizes of difference set codes for a given $\mathcal{A}_n$.

Theorem 2 *For any difference set $\mathcal{A}_n = \{a_1, a_2, \ldots, a_n\}$, where $0 = a_1 < a_2 < \cdots < a_n$,*

$$\frac{2^{n-1}}{a_n + 1} \leq M_e(\mathcal{A}_n) \leq M(\mathcal{A}_n) \leq 2M_e(\mathcal{A}_n).$$

Proof. The second and third inequalities follow readily from (9), (10), (11), and Lemma 4. For the first inequality, notice that $-a_n \leq \sum_{i=1}^{n} a_i c_i^T \leq 0$ and, therefore, $M_e(\mathcal{A}_n, w_e) \neq 0$ only if $-a_n \leq w_e \leq 0$. Hence,

$$M_e(\mathcal{A}_n) = \max_{w_e} M_e(\mathcal{A}_n, w_e) \geq \frac{\sum_{w_e=-a_n}^{0} M_e(\mathcal{A}_n, w_e)}{a_n + 1} = \frac{2^{n-1}}{a_n + 1}.$$

Q.E.D.

Table I lists the numbers $M(\mathcal{A}_n)$ and $M_e(\mathcal{A}_n)$ which give the maximum sizes of difference set codes and even weight difference set codes, respectively, for the difference sets $\mathcal{A}_n$ given in Table II. For these numbers, the upper bound on $M(\mathcal{A}_n)$ stated in Theorem 2 is much sharper than the lower bound.

Theorem 2 demonstrates that constructing difference sets with small values of a_n guarantees good difference set codes. Bose (see [8]) presented the following straightforward construction of difference sets of size n for which $a_n \leq q^2 - 3$, where q is a prime power not less than n. Let α be a primitive element in the finite field $\mathrm{GF}(q^2)$. The set $\{a : 0 \le a < q^2 - 1, \alpha^a + \alpha \in \mathrm{GF}(q)\}$, of q elements, has the property that the differences $a' - a''$ are distinct modulo $q^2 - 1$ for distinct pairs of numbers $a' \neq a''$ in this set. This can be shown as follows. For each a in the set, let $c_a = \alpha^a + \alpha \in \mathrm{GF}(q)$. If $a_1' - a_1'' \equiv a_2' - a_2'' \pmod{q^2 - 1}$ for $a_1' \neq a_1'', a_2' \neq a_2''$ in the set, then the polynomial

$$(x - c_{a_1'})(x - c_{a_2''}) - (x - c_{a_2'})(x - c_{a_1''})$$

has α as a root. Since this polynomial over $\mathrm{GF}(q)$ has degree at most one and $\alpha \notin \mathrm{GF}(q)$, it follows that the polynomial is identically zero. Since any quadratic, and in particular $(x - c_{a_1'})(x - c_{a_2''})$, has exactly two distinct roots or one repeated root, we have $\{c_{a_1'}, c_{a_2''}\} = \{c_{a_2'}, c_{a_1''}\}$. This implies that $\{a_1', a_2''\} = \{a_2', a_1''\}$ and therefore $a_1' = a_2'$, $a_1'' = a_2''$. This proves the property. Deleting $q - n$ elements from the set and subtracting the smallest number, which is positive, among the n remaining numbers from each of them yields a difference set $\mathcal{A}_n = \{a_1, a_2, \ldots, a_n\}$, where $0 = a_1 < a_2 < \cdots < a_n \le q^2 - 3$. A similar method for constructing difference sets, based also on finite fields, and which actually predates the Bose method, is due to Singer (see [8]). The difference sets listed in Table II are obtained by simple modifications of the Bose and the Singer constructions [1], [2].

Although Theorem 2 shows that difference sets with small value of a_n are good, it does not imply that difference sets with minimum value of a_n, which are often called optimal difference sets (or Golomb rulers), maximize the code size. For example in case $n = 16$, the difference set $\mathcal{A}_{16}$ in Table II gives $M(\mathcal{A}_{16}) = 866$ and $M_e(\mathcal{A}_{16}) = 434$. On the other hand, if $\mathcal{A}_{16}$ is the optimal difference set given by

$$\{0, 1, 4, 11, 26, 32, 56, 68, 76, 115, 117, 134, 150, 163, 168, 177\},$$

which is unique apart from a trivial transformation [12], then $M(\mathcal{A}_{16}) = 818$ and $M_e(\mathcal{A}_{16}) = 409$. Actually, except for the case $n = 16$, all the listed difference sets in Table II have the smallest known a_n. In fact, for $n \leq 19$ and $n \neq 16$, the difference sets are optimal (see [7], [10] and their references). We should mention that we did not attempt to optimize the difference sets listed in Table II with respect to the code size. Rather, if we checked more than one set, which we rarely did, we selected a set that maximizes (both) the numbers $M(\mathcal{A}_n)$ and $M_e(\mathcal{A}_n)$.

It is shown in [2] that the Singer construction yields difference sets of size n for which $a_n \leq n^2 + nt$, where $q = n + t \geq n$ is a prime power. By restricting q to be a prime, bounds on a_n can be readily obtained from bounds on the differences between consecutive primes. Indeed, [2] exploits a result of Huxley (1972) which states that $t \leq n^{7/12}$ for large n to obtain $a_n \leq n^2 + n^{19/12}$. Using a more recent result due to Baker and Harman [3] which states that $t \leq n^{0.535}$ for large n we get $a_n \leq n^2 + n^{1.535}$. Combining this with Theorem 2, we conclude that for any $\epsilon > 0$, there are even weight difference set codes $\mathcal{C}$ of length n such that $\log_2 |\mathcal{C}| > n - 2\log_2 n - 1 - \epsilon$ provided that n is sufficiently large.

IV An Upper Bound

In this section, we present a sphere-packing upper bound on the size of any code whose minimum squared Euclidean distance is at least six. Consider a binary sequence $\mathbf{x} = (x_1, \ldots, x_n)$ of length n. Let $\mathcal{I}_{\mathbf{x}} = \{i_1, i_2, \ldots, i_t\} = \{i : x_i = 1\}$, where $i_1 < i_2 < \cdots < i_t$. Here t is the weight of $\mathbf{x}$. Clearly, if $t = 0$, then $\mathcal{I}_{\mathbf{x}}$ is empty. For convenience, let $i_0 = 0$ and $i_{t+1} = n + 1$. We define

$$\mathcal{S}'_{\mathbf{x}} = \{\mathbf{v} \in \{0,1\}^n : \mathcal{I}_{\mathbf{v}} = (\mathcal{I}_{\mathbf{x}} \setminus \{i_j\}) \cup \{i\} \text{ for some } i_j < i < i_{j+1} \text{ and } 1 \leq j \leq t\}.$$

In words, $\mathcal{S}'_{\mathbf{x}}$ is the set of all binary sequences obtained from $\mathbf{x}$ by shifting a single 1 in $\mathbf{x}$, if there is any, to the right to any of the consecutive positions occupied by 0's that lie to the right of this 1 in $\mathbf{x}$. Notice that $\mathcal{S}'_{\mathbf{x}}$ is empty if $t = 0$.

We also define

$$\mathcal{S}''_{\mathbf{x}} = \{\mathbf{v} \in \{0,1\}^n : \mathcal{I}_{\mathbf{v}} = \mathcal{I}_{\mathbf{x}} \cup \{i', i''\} \text{ for some } i_j < i' < i'' < i_{j+1} \text{ and } 0 \leq j \leq t\}.$$

In words, $\mathcal{S}''_{\mathbf{x}}$ is the set of all binary sequences obtained from $\mathbf{x}$ by inserting two 1's in any pair of positions occupied by 0's that are not separated by 1's.

Finally, define $\mathcal{S}_{\mathbf{x}} = \{\mathbf{x}\} \cup \mathcal{S}'_{\mathbf{x}} \cup \mathcal{S}''_{\mathbf{x}}$ to be the sphere centered around $\mathbf{x}$.

Example 4 If $\mathbf{x} = 00010010$, then

$$\mathcal{S}'_{\mathbf{x}} = \{00001010, 00000110, 00010001\},$$

$$\mathcal{S}''_{\mathbf{x}} = \{11010010, 10110010, 01110010, 00011110\},$$

and $\mathcal{S}_{\mathbf{x}} = \{\mathbf{x}\} \cup \mathcal{S}'_{\mathbf{x}} \cup \mathcal{S}''_{\mathbf{x}}$ consists of the above eight sequences.

Next, we determine and bound the volume of the sphere $\mathcal{S}_{\mathbf{x}}$ for a given $\mathbf{x}$.

Lemma 6 *Let* $\mathbf{x} = (x_1, \ldots, x_n)$ *be a binary sequence of length* n *and* $\{i_1, \ldots, i_t\} = \{i : x_i = 1\}$, *where* $0 = i_0 < i_1 < \cdots < i_t < i_{t+1} = n + 1$. *Then,*

$$|\mathcal{S}_{\mathbf{x}}| = 1 + \binom{i_1 - 1}{2} + \sum_{j=1}^{t} \binom{i_{j+1} - i_j}{2} \geq n - t.$$

Proof. From the definitions of $\mathcal{S}'_{\mathbf{x}}$, $\mathcal{S}''_{\mathbf{x}}$, and $\mathcal{S}_{\mathbf{x}}$, we have

$$|\mathcal{S}'_{\mathbf{x}}| = \sum_{j=1}^{t}(i_{j+1} - i_j - 1),$$

$$|\mathcal{S}''_{\mathbf{x}}| = \sum_{j=0}^{t}\binom{i_{j+1} - i_j - 1}{2},$$

and $|\mathcal{S}_{\mathbf{x}}|$ is as stated above. Since $\binom{i}{2} \geq i - 1$ for any integer i, we have

$$|\mathcal{S}_{\mathbf{x}}| \geq 1 + i_1 - 2 + \sum_{j=1}^{t}(i_{j+1} - i_j - 1) = n - t.$$

Q.E.D.

We observe the following for $\mathbf{v} \in \mathcal{S}_{\mathbf{x}}$.

1. If $\mathbf{v} \neq \mathbf{x}$, then $\mathbf{v}$ and $\mathbf{x}$ differ in exactly two positions.

2. If $x_i = 1$, then $\sum_{j=1}^{i-1} x_j \equiv \sum_{j=1}^{i-1} v_j \pmod 2$.

The following lemma shows that spheres centered around sequences of equal length whose associated ternary sequences are separated by squared Euclidean distance of six or more are disjoint.

Lemma 7 *If $\mathbf{x}$ and $\mathbf{y}$ are sequences of length n and $d_E^2(\mathbf{x}^T, \mathbf{y}^T) \geq 6$, then $\mathcal{S}_{\mathbf{x}}$ and $\mathcal{S}_{\mathbf{y}}$ are disjoint.*

Proof. Suppose that $\mathbf{v} \in \mathcal{S}_{\mathbf{x}} \cap \mathcal{S}_{\mathbf{y}}$. The first observation says that $\mathbf{v}$ and each of $\mathbf{x}$ and $\mathbf{y}$ differ in at most two positions. Hence, the binary sequences $\mathbf{x}$ and $\mathbf{y}$, and therefore their associated ternary sequences $\mathbf{x}^T$ and $\mathbf{y}^T$, differ in at most four positions. Since $d_E^2(\mathbf{x}^T, \mathbf{y}^T) > 4$, it follows that $x_i^T = -y_i^T \neq 0$ for some $1 \leq i \leq n$. Therefore, $x_i = y_i = 1$ and $\sum_{j=1}^{i-1} x_j \not\equiv \sum_{j=1}^{i-1} y_j \pmod 2$. However, this contradicts the second observation which states that $\sum_{j=1}^{i-1} x_j \equiv \sum_{j=1}^{i-1} v_j \pmod 2$ as $\mathbf{v} \in \mathcal{S}_{\mathbf{x}}$ and $\sum_{j=1}^{i-1} y_j \equiv \sum_{j=1}^{i-1} v_j \pmod 2$ as $\mathbf{v} \in \mathcal{S}_{\mathbf{y}}$. Q.E.D.

Next, we use the sphere-packing argument to derive a bound on the size of codes of squared Euclidean distance of 6.

Theorem 3 *Let $\mathcal{C}$ be a block code of length n such that $d_E^2(\mathcal{C}) \geq 6$. For any $\epsilon > 0$, $\log_2 |\mathcal{C}| < n - \log_2 n + 1 + \epsilon$ if n is sufficiently large. Furthermore, if $\mathcal{C}$ is an even weight code, then $\log_2 |\mathcal{C}| < n - \log_2 n + \epsilon$.*

Proof. First, assume that $\mathcal{C}$ is an even weight code. Let $\mathcal{C}_t$ be the set of codewords of weight t in $\mathcal{C}$. For any $0 < \mu < 0.5$, we have

$$\sum_{(1-\mu)n < t \leq n} |\mathcal{C}_t| \leq \sum_{(1-\mu)n < t \leq n} \binom{n}{t} \leq 2^{nH_2(\mu)},$$

where H_2 is the binary entropy function (see Chapter 10, Section 11 in [11]). From the definition of $\mathcal{S}_{\mathbf{x}}$ and the fact that any codeword $\mathbf{x} \in \mathcal{C}$ has even weight, it follows that $\mathcal{S}_x$ consists of even weight sequences, i.e., $|\cup_{\mathbf{x} \in \mathcal{C}} \mathcal{S}_{\mathbf{x}}| \leq 2^{n-1}$. Since, by Lemma 7, the sets $\mathcal{S}_{\mathbf{x}}$'s are disjoint for different $\mathbf{x}$'s,

$$\begin{aligned} \sum_{0 \leq t \leq (1-\mu)n} |\mathcal{C}_t| &\leq \frac{2^{n-1}}{\min\{\mathcal{S}_{\mathbf{x}} : \mathbf{x} \text{ has weight at most } (1-\mu)n\}} \\ &\leq \frac{2^{n-1}}{n - (1-\mu)n} = \frac{2^{n-1}}{\mu n}, \end{aligned}$$

where the second inequality follows from Lemma 6. Thus, we conclude that

$$|\mathcal{C}| = \sum_{0 \leq t \leq (1-\mu)n} |\mathcal{C}_t| + \sum_{(1-\mu)n < t \leq n} |\mathcal{C}_t| \leq \frac{2^{n-1}}{\mu n} + 2^{nH_2(\mu)}.$$

Since μ is strictly less than 0.5, we have $H_2(\mu) < 1$. Then for any $\epsilon > 0$, we can select μ such that $-\log_2 \mu < 1 + \epsilon/2$. Furthermore, for sufficiently large n, it holds that

$\log_2 |\mathcal{C}| < n - 1 + \epsilon/2 - \log n - \log \mu$, and the stated upper bound on $|\mathcal{C}|$ holds in case $\mathcal{C}$ is an even weight code. The same argument shows that the same bound holds if $\mathcal{C}$ consists only of odd weight codewords. Hence, if $\mathcal{C}$ is any block code with even and odd weight subcodes $\mathcal{C}_e$ and $\mathcal{C}_o$, respectively, then

$$\log_2 |\mathcal{C}| = \log_2(|\mathcal{C}_e| + |\mathcal{C}_o|) < n - \log_2 n + 1 + \epsilon$$

provided that n is sufficiently large. Q.E.D.

Theorem 3 gives the upper bound $1 - (\log_2 n)/n + O(1/n)$ on the rate of any block code of length n and minimum squared Euclidean distance of at least six. It is interesting to notice that there are difference set codes with rates lower bounded by $1 - 2(\log_2 n)/n - O(1/n)$, as argued in the end of Section III.

V A Symbol-Oriented Trellis Description

Difference set codes introduced in Section III can be decoded by a Viterbi decoder. In order to simplify the decoder it can be designed to operate on a symbol-oriented trellis, that is, a trellis with one (ternary) symbol per trellis branch. In this perspective, the code corresponds to all paths (from left to right) through the trellis. Clearly, the amount of work performed by the decoder depends on the numbers of states and branches in the trellis.

V.1 Decoding of $\mathcal{C}(\mathcal{A}_n, w_e, w_o)$

Figure 3 describes the codewords of a 9-codeword code with $n = 7$ (see Tables I and II), at the output of a noiseless $(1 - D)$-PR channel. The symbol-oriented trellis is a directed graph that contains nodes (states), and edges, each with a single (ternary) symbol label. The code corresponds to all edge-labeled sequences associated with paths from the leftmost state at time 0 through the trellis to any of the two terminal states at time instant $n = 7$. Associated with each state s_j of the trellis at time instant j is a state label

$$\lambda(s_j) = \sum_{i=1}^{j} x_i^T a_i,$$

where $\mathbf{x}$ is a codeword and $\mathbf{x}^T$ passes through s_j. Because the elements a_i of the difference set $\mathcal{A}_n$ are increasing with i, the sign of $\lambda(s_j)$ expresses the channel state. Hence, every horizontal edge in the figure is labelled by a zero symbol, an upward sloping edge has a -1 label, while a downward sloping edge has a $+1$ label. The values of the state labels $\lambda(s_j)$ are indicated in the right column of Figure 3.

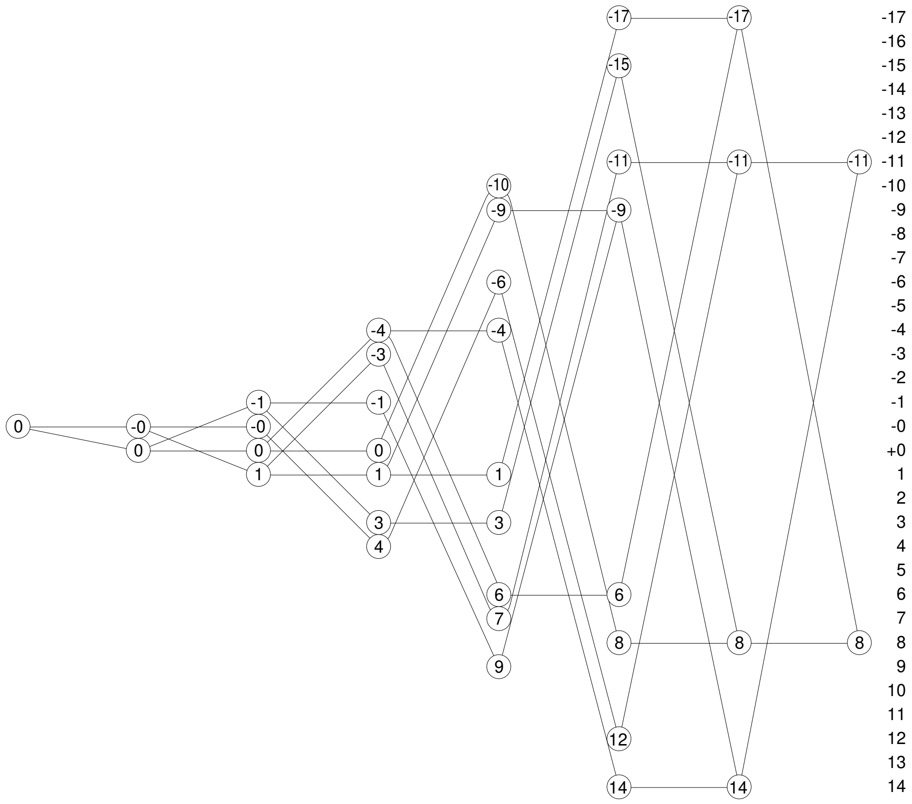
We assume that the channel is originally in the '+' state. The complete Viterbi decoder also needs a copy of this trellis for the case where the channel is originally in the '−' state. At the end of each codeword, the two '+' final states are merged into one '+' state with one backwards pointer for the survivor path, and the two final '−' states are merged into one '−' state with one backwards pointer. Then the Viterbi algorithm can proceed with the next codeword. The decoding decision on a given codeword is delayed for a few codeword lengths. The actual delay is determined by, among other things, the actual code used. Observe that any pair of paths through the trellis will build up a squared Euclidean distance of at least six within four codeword lengths after and including the codeword where they diverge.

In general, the number of decoder states at time instant i, numbered from 0 to n, is at most

$$\min\{2 \cdot (\text{Number of codewords}), 2^{i+1}, 2^{n-i+1}, 4(a_i + 1)\}.$$

The first term is obvious, since each of the two versions (starting in the '+' or the

Figure 3: Trellis of code for $n = 7$.



'$-$' state) of a codeword can pass through at most one state at each time instant. The second term arises since the trellis has two states, '+' and '$-$', at time instant zero, and the number of states at time instant i is at most twice the number of states at time instant $i-1$ for $i \geq 1$. The third term is a result of a similar argument applied backwards from time instant n. Finally, the last term follows since the partial difference set sum can be used to label the states, cf. Figure 3.

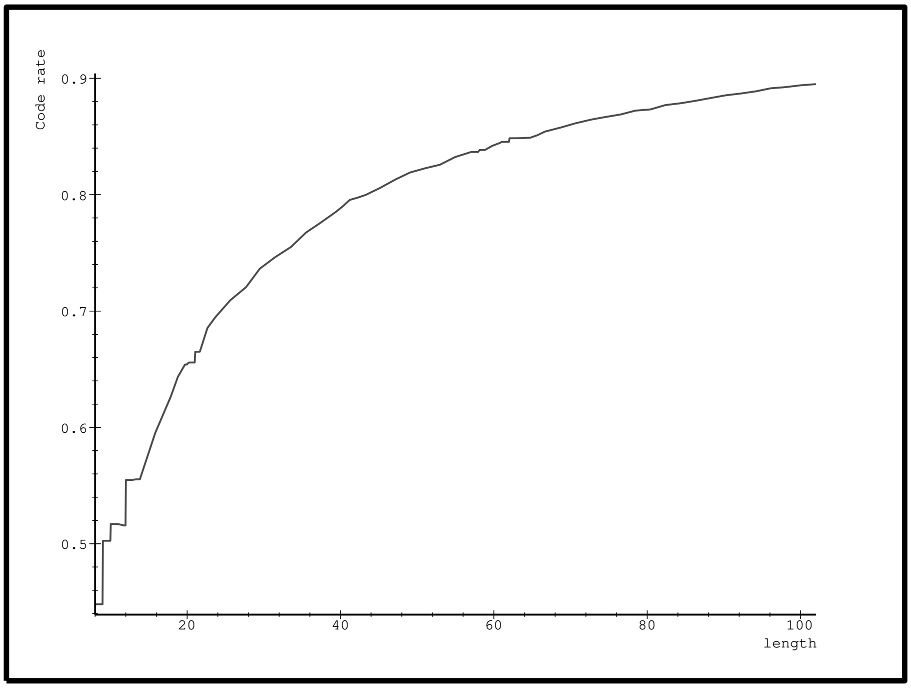
V.2 Codeword enumeration

The number of codewords, or trellis paths, can be easily calculated from the symbol-oriented trellis. If $N(\mathcal{A}_n, i, s)$ is the number of paths from the initial state at time zero to the state s at time i, then $N(\mathcal{A}_n, i, s)$ is either equal to $N(\mathcal{A}_n, i-1, s')$ where s' is the single predecessor of s, or $N(\mathcal{A}_n, i, s)$ is equal to $N(\mathcal{A}_n, i-1, s') + N(\mathcal{A}_n, i-1, s'')$ where s' and s'' are the only two predecessors of s. In order to select the code in the first place, we can work on a trellis with all the possible states at time instant n as terminal nodes; use a recursive procedure to calculate $N(\mathcal{A}_n, n, s)$ for each possible terminal state, and select the two terminal states such that the code size is maximized. Using this procedure we obtain code rates as shown in Figure 4 for $n \leq 102$.

VI Generalized Concatenation

The difference set construction can also be applied towards the construction of generalized concatenated codes of squared Euclidean distance of six [15]. Let $\{\mathcal{C}_e(\mathcal{A}_{n_1}, w) : w = w_0 \ldots w_{2^r-1}\}$ be a set of 2^r disjoint even weight difference set codes, each of length n_1 and each with at least 2^{k_1} codewords. Also, let $\mathcal{C}_{\text{outer}}$ be a 2^r-ary code of length n_2 with $2^{k_{2r}}$ codewords and minimum Hamming distance 3. Then a gener-

Figure 4: Code rate of difference set codes for $n \leq 102$.



alized concatenated code with code rate $\frac{(k_2 r)+(k_1 n_2)}{n_1 n_2}$ can be encoded as follows: Let $k_2 r$ input bits select a codeword $\mathbf{c} = (c_1, \ldots, c_{n_2})$ in $\mathcal{C}_{\text{outer}}$. In the second stage of the encoding process, each of the n_1 coordinates in $\mathbf{c}$ determines a difference set code $\mathcal{C}_e(\mathcal{A}_{n_1}, c_i)$, where we abuse the notation by assuming a natural one-to-one mapping from the 2^r-ary alphabet of $\mathcal{C}_{\text{outer}}$ onto the 2^r integers $w_0 \ldots w_{2^r-1}$. Finally, for each of the n_2 coordinates in $\mathbf{c}$, another k_1 input bits select a codeword in $\mathcal{C}_e(\mathcal{A}_{n_1}, c_i)$.

As an example, consider $n_1 = 7$. It turns out that we can find at least four (actually six: select $w \in \{-8, -9, -11, -14, -16, -17\}$) disjoint even-weight difference set codes each with at least four codewords, so in this case we can use $r = k_1 = 2$. Let $\mathcal{C}_{\text{outer}}$ be a quaternary linear [21,18,3] Hamming code [4]. Thus we can construct a code with rate $78/147 = 0.53\cdots$.

The Viterbi algorithm can also be used as the first stage of a two-stage decoder for the generalized concatenated code. The goal of this first stage is to provide, for each length-n_1 codeword, an estimate for the appropriate subcode $\mathcal{C}_e(\mathcal{A}_{n_1}, w)$. This subcode determines the value of one coordinate c_i^* of a vector $\mathbf{c}^* \in V_{n_2}(2^r)$. Furthermore, the Viterbi decoder should also determine the most likely codeword within each subcode $\mathcal{C}_e(\mathcal{A}_{n_1}, j)$, for $j \in \{w_0, \ldots, w_{2^r-1}\}$. This can be achieved by having one terminal node for each of the weights $w_0, \ldots, w_{2^r-1}$.

Arguing as in Section V, we find that the number of decoder states of the Viterbi decoder at time instant i is at most

$$\min\{\text{Total number of codewords}, 2^i, 2^{n_1-i+r}, 2(a_i+1)\}.$$

The second stage of the decoder finds the codeword $\mathbf{c} = (c_1, \ldots, c_{n_2}) \in \mathcal{C}_{\text{outer}}$ which is closest to $\mathbf{c}^*$. Finally, for each of the n_2 coordinates c_i, the most likely codeword for the appropriate subcode $\mathcal{C}_e(\mathcal{A}_{n_1}, c_i)$ is obtained from the Viterbi decoder in the first stage.

Note that this two-stage decoder is not a maximum-likelihood (ML) decoder, and that it may be hard to design an ML decoder for these codes.

VII Computer Search

In order to compare our results with optimum codes, we have carried out a computer search for $n \leq 8$. The results are listed in Table I. The table shows the largest size of any block code of minimum squared Euclidean distance six. In most cases, the largest code has $d_<^2 = d_>^2 = 3$. However, for $n = 6$ the largest code with $d_<^2 = d_>^2 = 3$ has only 9 codewords, whereas there exists a code with $d_<^2 = 5, d_>^2 = 1$, that has 10 codewords.

References

[1] M. D. Atkinson and A. Hassenklover, "Sets of integers with distinct differences," Report SCS-TR-63, School of Computer Science, Carleton University, Ottawa, Canada, October 1984.

[2] M. D. Atkinson, N. Santoro, and J. Urrutia, "Integer sets with distinct sums and differences and carrier frequency assignments for nonlinear repeaters," *IEEE Trans. Commun.*, vol. COM-34, no. 6, pp. 614–617, June 1986.

[3] R. C. Baker and G. Harman, "The difference between consecutive primes," *Proc. London Math. Soc.*, vol. 72, no. 3, pp. 261–280, March 1996.

[4] A. E. Brouwer, "On-Line Linear Code Bound Tables," *http://www.win.tue.nl/math/dw/voorlincod.html.*

[5] C. J. Colbourn and J. H. Dinitz, *The CRC Handbook of Combinatorial Designs.* Boca Raton, Florida: CRC Press, 1996.

[6] J. H. Conway and N. J. A. Sloane, *Sphere Packings, Lattices, and Groups.* New York: Springer-Verlag, 1993.

[7] A. Dollas, W. T. Rankin, and D. McCracken, "A new algorithm for Golomb ruler derivation and proof of the 19 marker ruler," *IEEE Trans. Inform. Theory*, vol. 44, no. 1, pp. 379–382, January 1998.

[8] H. Halberstam and K. F. Roth, *Sequences.* New York: Springer-Verlag, 1983.

[9] R. Karabed and P. H. Siegel, "Matched spectral-null codes for partial-response channels," *IEEE Trans. Inform. Theory*, vol. 37, no. 3, pp. 818–855, May 1991.

[10] T. Kløve, "Bounds and construction for difference triangle sets," *IEEE Trans. Inform. Theory*, vol. 35, no. 4, pp. 879–886, July 1989.

[11] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes.* Amsterdam, The Netherlands: North-Holland, 1977.

[12] J. B. Shearer, "Some new optimum Golomb rulers," *IEEE Trans. Inform. Theory*, vol. 36, no. 1, pp. 183–184, January 1990.

[13] L. M. G. M. Tolhuizen, K. A. S. Immink, and H. D. L. Hollmann, "Constructions and properties of block codes for partial-response channels," *IEEE Trans. Inform. Theory*, vol. 41, no. 6, pp. 2019–2026, November 1995.

[14] J. K. Wolf and G. Ungerboeck, "Trellis coding for partial-response channels," *IEEE Trans. Commun.*, vol. COM-34, no. 8, pp. 765-773, Aug. 1986.

[15] V. A. Zinov'ev, "Generalized cascade codes." *Problemy Peredachi Informatsii*, vol. 12, no. 1, pp. 5-15, 1976.

Table I: Sizes of codes of length $n \leq 24$ and squared Euclidean distance 6. Columns 2–4 are obtained by construction methods in [13]. Columns 5 and 6 give maximal sizes of difference set codes based on $\mathcal{A}_n$ of Table II. Column 7 gives maximum sizes of codes from computer search.

n	Precoding	Spectral Nulls	Concatenation	$M(\mathcal{A}_n)$	$M_e(\mathcal{A}_n)$	Maximum
1	1	1		1	1	1
2	1	1		1	1	2
3	1	1		2	1	3
4	1	1		3	2	4
5	1	2		4	2	6
6	2	2		6	3	10
7	2	3	10	9	5	16
8	2	5		13	7	25
9	4	8	18	20	10	
10	6	12		36	18	
11	12	20	34	58	29	
12	24	32		101	51	
13	32	58	100	170	86	
14	64	94		292	147	
15	128	169	130	503	252	
16	256	289		866	434	
17	$256 - 340$	526	324	1607	804	
18	$512 - 680$	910		3061	1531	
19	$1024 - 1288$	1667	1000	5504	2754	
20	$2048 - 2372$	2934		8884	4442	
21	$2560 - 4096$	5448	1156	16010	8006	
22	$4096 - 6942$	9686		34617	17312	
23	$8192 - 13774$	18084	2050	64312	32162	
24	$16384 - 24106$	32540		113584	56793	

Table II: Difference sets $\mathcal{A}_n$ for $n \leq 24$.

n	$\mathcal{A}_n$
1	0
2	0, 1
3	0, 1, 3
4	0, 2, 5, 6
5	0, 3, 4, 9, 11
6	0, 1, 4, 10, 15, 17
7	0, 1, 4, 10, 18, 23, 25
8	0, 1, 4, 9, 15, 22, 32, 34
9	0, 1, 5, 12, 25, 27, 35, 41, 44
10	0, 1, 6, 10, 23, 26, 34, 41, 53, 55
11	0, 1, 4, 13, 28, 33, 47, 54, 64, 70, 72
12	0, 2, 6, 24, 29, 40, 43, 55, 68, 75, 76, 85
13	0, 2, 5, 25, 37, 43, 59, 70, 85, 89, 98, 99, 106
14	0, 4, 6, 20, 35, 52, 59, 77, 78, 86, 89, 99, 122, 127
15	0, 6, 7, 15, 28, 40, 51, 75, 89, 92, 94, 121, 131, 147, 151
16	0, 6, 19, 40, 58, 67, 78, 83, 109, 132, 133, 162, 165, 169, 177, 179
17	0, 5, 7, 17, 52, 56, 67, 80, 81, 100, 122, 138, 159, 165, 168, 191, 199
18	0, 2, 10, 22, 53, 56, 82, 83, 89, 98, 130, 148, 153, 167, 188, 192, 205, 216
19	0, 4, 13, 15, 42, 56, 59, 77, 93, 116, 126, 138, 146, 174, 214, 221, 240, 245, 246
20	0, 1, 8, 11, 68, 77, 94, 116, 121, 156, 158, 179, 194, 208, 212, 228, 240, 253, 259, 283
21	0, 2, 24, 56, 77, 82, 83, 95, 129, 144, 179, 186, 195, 255, 265, 285, 293, 296, 310, 329, 333
22	0, 3, 15, 26, 65, 86, 93, 103, 133, 152, 177, 197, 228, 232, 234, 250, 286, 313, 342, 347, 355, 356
23	0, 6, 22, 24, 43, 56, 95, 126, 137, 146, 172, 173, 201, 213, 258, 273, 281, 306, 311, 355, 365, 369, 372
24	0, 9, 33, 37, 38, 97, 122, 129, 140, 142, 152, 191, 205, 208, 252, 278, 286, 326, 332, 353, 368, 384, 403, 425

Table III: $\sum_{i=1}^{4} a_i x_i^T$, where $a_i = 0, 2, 5, 6$ for $i = 1, 2, 3, 4$ (Example 2).

$\mathbf{x} = (x_1, x_2, x_3, x_4)$	$\mathbf{x}^T = (x_1^T, x_2^T, x_3^T, x_4^T)$	$\sum_{i=1}^{4} a_i x_i^T$
0000	0000	0
0001	0001	6
0010	0010	5
0011	$001\bar{1}$	-1
0100	0100	2
0101	$010\bar{1}$	-4
0110	$01\bar{1}0$	-3
0111	$01\bar{1}1$	3
1000	1000	0
1001	$100\bar{1}$	-6
1010	$10\bar{1}0$	-5
1011	$10\bar{1}1$	1
1100	$1\bar{1}00$	-2
1101	$1\bar{1}01$	4
1110	$1\bar{1}10$	3
1111	$1\bar{1}1\bar{1}$	-3