Designs, Codes and Cryptography, 24, 15–36, 2001

Trellis Structure and Higher Weights of Extremal Self-Dual Codes

HOUSHOU CHEN houshou@ncnu.edu.tw
Department of Electrical Engineering, National Chi-Nan University, Nantou, Taiwan 545

JOHN T. COFFEY scoffey@eecs.umich.edu
Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, Michigan 48109-2122

Communicated by: T. Helleseth

Received September 16, 1999; Revised June 13, 2000; Accepted June 13, 2000

Abstract. A method for demonstrating and enumerating uniformly efficient (permutation-optimal) trellis decoders for self-dual codes of high minimum distance is developed. Such decoders and corresponding permutations are known for relatively few codes.

The task of finding such permutations is shown to be substantially simplifiable in the case of self-dual codes in general, and for self-dual codes of sufficiently high minimum distance it is shown that it is frequently possible to deduce the existence of these permutations directly from the parameters of the code.

A new and tighter link between generalized Hamming weights and trellis representations is demonstrated: for some self-dual codes, knowledge of one of the generalized Hamming weights is sufficient to determine the entire optimal state complexity profile.

These results are used to characterize the permutation-optimal trellises and generalized Hamming weights for all [32,16,8] binary self-dual codes and for several other codes. The numbers of uniformly efficient permutations for several codes, including the [24,12,8] Golay code and both [24,12,9] ternary self-dual codes, are found.

Keywords: chain condition, Conway-Pless codes, double chain condition, generalized Hamming weights

1. Introduction

Representations of block codes by trellises allow computationally efficient soft decision decoding for the codes via the Viterbi algorithm. Given a code, there are very many different trellises that represent it, of widely varying complexity. It thus becomes important to choose the trellis that represents the code most efficiently, i.e., that minimizes the complexity.

For any fixed linear block code, i.e., if we take the code to be distinct from codes that are equivalent to it, an essentially complete solution to this problem is known. A unique "minimal trellis" can be found efficiently from any generator matrix, and this trellis minimizes complexity under a wide range of possible complexity criteria.

For many purposes, however, there is no useful distinction between two equivalent codes. This is particularly so in the main case where a trellis representation is of interest: using the trellis for Viterbi decoding over a memoryless channel. In this case the performance of the code is identical to that of any equivalent code. Thus the important question becomes how

to choose the permutation of the code whose minimal trellis has the smallest complexity. This is a very much harder problem, and remains open. Optimal permutations, in the strong sense we consider in this paper (uniform efficiency), are known for very few codes.

The derivation of optimal trellis structure is strongly related to the purely combinatorial problem of determining the generalized Hamming weights: the most common way to demonstrate that a trellis is permutation-optimal is to find the generalized Hamming weights and then to demonstrate chains of subcodes ordered by inclusion that have parameters determined by the generalized Hamming weights.

In this paper, we consider the permutation problem for self-dual codes of high minimum distance. Numerous simplifications of the general problem appear. We demonstrate a new simplification that in some cases allows us to determine and characterize uniformly efficient permutations quickly, and to count the number of such permutations. In particular, we derive uniformly efficient permutations for each of the eight inequivalent extremal binary self-dual codes of length 32. Of these, such permutations for the Reed-Muller code r_{32} and the quadratic residue code q_{32} were previously known, though without proof of uniform efficiency in the case of q_{32}. Uniformly efficient permutations are also derived for several other extremal self-dual codes of length up to 48. Where they are not already known, the full generalized Hamming weight hierarchies are also determined for these codes.

An outline of the paper is as follows. In Sections 2 and 3 we summarize previous work in this area, and review the necessary definitions and facts about the trellis structure of block codes. In Section 4 we demonstrate results that substantially simplify the problem for the special case of self-dual codes of high minimum distance. In Section 5 we apply these results to various classes of self-dual codes, including a complete classification of the optimal trellises for extremal self-dual codes of length 32, and a new characterization of uniformly efficient permutations for the [48,24,12] quadratic residue code.

2. Background

The representation of block codes by a trellis was introduced in Bahl et al. [1] in 1974, and has subsequently attracted an enormous amount of interest. For a full history we refer to the tutorial paper of Kiely et al. [21], the chapter by Vardy [35] and the book by Lin et al. [25].

The permutation problem above, i.e., the problem of finding a permutation that minimizes a given element of the state complexity profile for a general linear block code, was shown to be NP-hard by Horn and Kschischang [18]. Optimum (uniformly efficient) permutations are known for some codes, e.g., the [24,12,8] Golay code [15], the [48,24,12] quadratic residue code [2,10], and the [16,7,6] lexicode [23]. The most general result is that all binary Reed-Muller codes have as one uniformly efficient permutation the natural binary ordering [20]; this and the family of maximum distance separable are the only nontrivial infinite families of codes for which uniformly efficient permutations are known. (MDS codes have complexity independent of the coordinate ordering [27].) Berger and Be'ery [2] give a construction that is applicable to BCH and quadratic residue codes, find a permutation for the [32,16,8] quadratic residue code that we will demonstrate is uniformly efficient, and find a new uniformly efficient permutation for the [48,24,12] quadratic residue code, among others.

Encheva and Cohen [13,14] have found uniformly efficient permutations where these exist for binary and ternary self orthogonal codes of length up to 20 and binary self-dual codes of length up to 24.

The problem of determining the state complexity profile is strongly related to, though not exactly equivalent to, the problem of determining the generalized Hamming weights of the code, in a sense we review in Section 2.3. The problem of generalized Hamming weights has received much independent attention, which is reviewed and summarized in [29,32]. For the codes we consider where these are unknown, we find the full set of generalized Hamming weights.

2.1. Trellises

We assume that the reader is familiar with the basic facts about trellises; a comprehensive description is given by Vardy [35]. We follow the notation used in this reference throughout.

Given any generator matrix for C, a minimal trellis may be constructed efficiently using any of several different constructions; we refer again to Vardy [35] for details. Trellis diagrams will not therefore be given explicitly in what follows.

2.2. Dimension/Length and Length/Dimension Profiles

The ith past subcode and future subcode are denoted by P_i and F_i, respectively, and their dimensions are denoted by p_i and f_i, respectively. The code consisting of all codewords that are zero outside a set J is denoted C_J.

For any code C, the following relation holds [31]:

$$\dim((C^{\perp})_{I-J}) = n - k - |J| + \dim(C_J). \tag{1}$$

The state complexity of a code, with ordering assumed fixed, is given by [35]

$$s_i = k - p_i - f_i. \tag{2}$$

The dimension/length profile (DLP) of C is the sequence [15] $\mathbf{k}(C) = \{k_i(C), 0 \le i \le n\}$ whose ith component $k_i(C)$ is the maximum dimension of any subcode C_J of C with $|J| = i$: $k_i(C) = \max_{J:|J|=i}\{\dim(C_J)\}$, $0 \le i \le n$. The length/dimension profile (LDP) of C is the sequence [15] $\mathbf{d}(C) = \{d_r(C), 1 \le r \le k\}$ whose rth component $d_r(C)$ is the minimum support size of any subcode C_J of C with dimension r: $d_r(C) = \min_{J:\dim(C_J)=r}\{|J|\}$, $0 \le r \le k$. These numbers are also called the generalized Hamming weights of the code by Wei [36].

The LDPs of the code and its dual code are related according to Wei's duality relation [36]

$$\{d_r(C) : 1 \le r \le k\} = \{1, 2, \ldots, n\} - \{n + 1 - d_r(C^{\perp}) : 1 \le r \le n - k\}. \tag{3}$$

Codes of different dimensions cannot have the same DLPs or LDPs. We will say that the LDPs of codes C_1 and C_2 *match* if we have $d_i(C_1) = d_i(C_2)$ for $0 \le i \le \min(\dim(C_1), \dim(C_2))$.

The generalized Hamming weights obey the generalized Griesmer bounds [29, pp. 35–36]:

$$n \geq d_r(C) + \sum_{i=1}^{k-r} \left\lceil \frac{(q-1)d_r(C)}{q^i(q^r-1)} \right\rceil . \tag{4}$$

With $r = 1$ this reduces to the Griesmer bound

$$n \geq g_q(k,d) \equiv \sum_{i=0}^{k-1} \left\lceil \frac{d}{q^i} \right\rceil . \tag{5}$$

It is known that a code that meets the Griesmer bound with equality also meets the generalized Griesmer bound with equality for all r [17], [29, pp. 36–38].

A code is said to satisfy the *chain condition* if it is equivalent to a code in which $p_i(C) = k_i(C)$ for $1 \leq i \leq n$.

Several classes of codes are known to satisfy the chain condition. These include all codes that meet the Griesmer bound with equality, and all codes of length one greater than the Griesmer bound [17], [29, p. 40].

A code is said to satisfy the *double chain condition* if it is equivalent to a code in which $p_i(C) = k_i(C)$ and $f_i(C) = k_{n-i}(C)$ for all i.

The double chain condition is also known as the two-way chain condition (TCC) [11–13].

2.3. Trellis Complexity Over Permutations

The state complexity profile of a code $\pi(C)$ obtained by permuting the coordinates of C is not in general the same as the state complexity profile of C. A permutation is said to be *componentwise optimal,* or *uniformly efficient,*[1] if at each time the state complexity of the corresponding permuted code is no higher than for any other permutation, i.e., the permutation π^* is uniformly efficient if and only if $s_i^{\pi^*} \leq s_i^{\pi}$ for all $0 \leq i \leq n$ and all permutations π. Such a permutation may or may not exist [21]. Often the goal is to find a permutation that minimizes the maximum value of s_i over the range $0 \leq i \leq n$, the "absolute state complexity," and sometimes any permutation achieving this minimum is termed optimal. In this paper we will concentrate exclusively in the stronger definition of optimality given above, i.e., uniform efficiency.

Uniform efficiency is a relative concept, and it may be that even a uniformly efficient permutation of a given code has too high a trellis decoding complexity for a given application. On the other hand, any uniformly efficient permutation of a self-dual code also minimizes other measures of complexity, such as total number of edges or total number of mergers [21].

Our goal in counting the number of uniformly efficient permutations is primarily one of classification. However, there are some reasons why not all uniformly efficient permutations are equally desirable, including state connectivity, parallel structure, regularity and symmetry, as discussed by Lin et al. [25, Section 10.3]. For example, one desirable property is that the decoder should be reversible, i.e., symmetric about the central time index [35]. In distinguishing between decoders in the several categories that result, a first step is to know the size of the set of all uniformly efficient permutations.

Taking minima in (2) gives the DLP lower bound on state complexity [15]:

$$s_i(\pi(C)) \geq k - k_i - k_{n-i}, \tag{6}$$

for all i, where π is any permutation.

The DLP lower bound can be attained for all i if and only if the double chain condition is satisfied; a code meeting the DLP bound must have a uniformly efficient permutation. The converse is not true [21]. The state complexity profile of a uniformly efficient permutation will be called the optimum state complexity profile of C.

A code is said to be *uniformly concise* over a class Q of codes if at each time its state complexity is no higher than for any code in Q. This is a modification of the definition introduced by Kiely et al. [21].

2.4. Other Comments

2.4.1. Second Hamming Weight

Often it is possible to determine the second Hamming weight (and by extension, third and higher Hamming weights) by applying the residual code argument [33]. If C is an $[n, k, d]$ binary code (resp. binary doubly-even code), and there is no $[n - d, k - 1, \lceil d/2 \rceil + 1]$ binary code (resp. binary singly-even code), then $d_2(C) = d + \lceil d/2 \rceil$. Taking the residual code [33, p. 46] with respect to any codeword of weight d gives an $[n - d, k - 1, \geq \lceil d/2 \rceil]$ code. If the original code is doubly-even the residual code will be at least singly even. When the nonexistence condition above holds, we may conclude that the minimum distance of the residual code is exactly $\lceil d/2 \rceil$. Adjoining $\lceil d/2 \rceil$ positions holding a minimum weight codeword of the residual to the original d positions deleted to form the residual then gives a subspace of dimension 2.

2.4.2. Shortened Codes

We must often deal with codes that are shortened versions of codes whose properties with respect to Hamming weight hierarchy and chain condition are well understood. From Forney's development [15] we have the following relations in this case. Given a code C that achieves the DLP bound (6), consider the code C_r obtained by shortening C on the last r positions, when C is in uniformly efficient order. A necessary condition for C to satisfy the DLP bound is that $p_i(C) = k_i(C)$ for $0 \leq i \leq n$. The shortened code therefore has $p_i(C_r) = p_i(C) = k_i(C) = k_i(C_r)$ for $0 \leq i \leq n - r$. We can conclude that

(a) C_r satisfies the chain condition; and

(b) the LDP of C_r matches the LDP of C.

We also remark that shortening on any r positions of a code invariant under an r-fold transitive group produces a code that has properties (a) and (b).

3. Trellis Structure for Self-Dual Codes: Known Results

Several simplifications are known for the case of self-dual codes. The relation (1) becomes

$$\dim C_{[n]\setminus J} = n/2 - |J| + \dim C_J;$$

this is a version of Koch's balance theorem for self-dual codes [22,28] by which $\dim C_a - |a|/2 = \dim C_b - |b|/2$ where a and b partition n. Taking $J = [i]$, this becomes

$$f_i = n/2 - i + p_i \tag{7}$$

for $0 \leq i \leq n$.

The balance principle (7) implies that $p_{i+1} - p_i = f_i - f_{i+1}$, so with (2), we have $s_{i+1} - s_i \in \{-1, 1\}$ for a self-dual code. The state complexity profile thus starts with $s_0 = 0$ and increases or decreases by one with each time unit.

The Wei relations (3) reduce for a self-dual code to

$$\{d_r(C) : 1 \leq r \leq k\} = \{1, 2, \ldots, n\} - \{n + 1 - d_r(C) : 1 \leq r \leq n - k\}. \tag{8}$$

Thus for any self-dual code of length n, the LDP contains exactly one of the numbers $\{i, n + 1 - i\}$ for each $i \in [1, n]$.

While a code in general may have a permutation that is uniformly efficient but does not meet the DLP bound, this cannot happen for self-dual codes, since if s_i is minimized for every i, then p_i and f_i are maximized.

The following observation is a key starting point for this paper. We include it in the section on known results, as the basic elements all appear in the paper of Forney [15], and it can also be seen as an adaptation of Theorem 5.10 in Vardy's review [35]. However, we have not been able to find a definite attribution for the result in the form below.

From (7), we see that any ordering that maximizes all p_i's simultaneously will also maximize all f_i's simultaneously, and hence, from (2), minimize all s_i's simultaneously. Thus we have the lemma:

LEMMA 1. *A self-dual code satisfies the double chain condition if and only if it satisfies the chain condition. An ordering of such a code is uniformly efficient if and only if the code is in chain condition order.*

4. Main Results

From Lemma 1, to determine whether a self-dual code meets the double chain condition, we need only check for the chain condition. On the other hand, codes at or near the Griesmer bound automatically satisfy the chain condition, so in certain cases we may deduce the existence of a uniformly efficient permutation directly from the parameters of the code.

In this form the idea has limited applicability, as almost all self-dual codes have lengths that greatly exceed the Griesmer bound. Our main result, Theorem 1, extends this to include many more interesting codes, by working with past and future subcodes simultaneously. The idea is to find an ordering in which the p_i's are maximized for $0 \leq i \leq m$, and the f_i's are maximized for $m < i \leq n$, for some m. Our main result is then the following:

THEOREM 1. *Let C be a length n self-dual code. Let n_1 and n_2 be two non-negative integers such that $n_1 + n_2 = n$. Assume that some permutation of C has a generator matrix given by*

$$G = \begin{bmatrix} G_1 & 0 \\ 0 & G_2 \\ E & F \end{bmatrix}, \tag{9}$$

where $C_1 = \langle G_1 \rangle$ and $C_2 = \langle G_2 \rangle$ have lengths n_1, n_2, respectively. If C_1 and C_2 have LDPs that match the LDP of C and each satisfy the chain condition then:

(i) C has a uniformly efficient ordering.

(ii) An ordering for the code is uniformly efficient if and only if the resulting code has generator matrix of the form

$$G' = \begin{bmatrix} G'_1 & 0 \\ 0 & \tau(G'_2) \\ E' & F' \end{bmatrix} \tag{10}$$

where $C'_1 = \langle G'_1 \rangle$ and $C'_2 = \langle G'_2 \rangle$ have LDPs that match the LDP of C and are each in chain condition order, and $\tau(\cdot)$ is the reverse permutation.

Proof. We take C_1 in chain condition order and C_2 in the reverse of chain condition order. For $i \leq n_1$, since C_1 satisfies the chain condition, we have $p_i(C) \geq p_i(C_1) = k_i(C_1) = k_i(C)$, where the last equality follows from the assumption that the LDP of C_1 matches the LDP of C. Since $p_i(C) \leq k_i(C)$ by definition, we conclude that equality holds. From (7), maximizing p_i will maximize f_i, so we then have $f_i(C) = k_{n-i}(C)$. Then (2) gives $s_i = k - k_i - k_{n-i}$.

If $i \geq n_1$, since C_2 is in the reverse of chain condition order, we have $f_i(C) \geq f_i(C_2) = k_{n-i}(C_2) = k_{n-i}(C)$. Thus $f_i(C)$ attains its maximum for each $i > n_1$. Again from (7) we have $p_i(C) = k_i(C)$ and thus $s_i = k - k_i - k_{n-i}$. Thus C has a uniformly efficient permutation.

For the converse, if C'_1 fails to match the LDP of C, or is not in chain condition order, then there exists at least one time index $i \leq n_1$ such that $p_i(C) \leq k_i(C) - 1$, and hence $s_i \geq s_i^{\min} + 2$, and the permutation is not uniformly efficient; similarly for C'_2. ∎

Remark. Note that if we know any optimal split between left and right parts, we need only order the past subcode G_1 and future subcode G_2 to achieve the chain condition, and need not consider the glue vectors E and F at all.

4.1. The Griesmer Bound Subcode Case

All conditions in Theorem 1 are automatically satisfied in one special case. This is thus the easiest case to apply, involving only a check of the central component $k_{n/2}$ of the DLP. Recall that $g_q(k, d) = \sum_{i=0}^{k-1} \lceil d/q^i \rceil$ is the Griesmer bound function.

THEOREM 2. *Let C be a self-dual code over F_q. Suppose $k_{n/2}(C)$ is such that $n/2 = g_q(k_{n/2}, d)$. Then:*

(a) the code has a uniformly efficient permutation;

(b) the code achieves the DLP bound;

(c) the code is uniformly concise over the class of $[n, k, d]$ self-dual codes;

(d) the optimum state complexity profile is $s_i = i - 2g_q^{-1}(i, d)$ for $i \leq n/2$, and $s_{n-i} = s_i$, where $g_q^{-1}(i, d) = \max\{j \mid i \geq g_q(j, d)\}$;

(e) a permutation is uniformly efficient if and only if it is of the form

$$G = \begin{bmatrix} G_1 & 0 \\ 0 & \tau(G_2) \\ E & F \end{bmatrix}, \tag{11}$$

where $C_1 = \langle G_1 \rangle$ is a length $n/2$, distance d code that meets the Griesmer bound with equality, and is in chain condition order; and where G_2 generates a code with the same parameters as C_1, and is in chain condition order; and where $\tau(G_2)$ is the column reverse of G_2.

Proof. The condition on $k_{n/2}$ implies that we can permute the code to the form (10) where C_1 is an $[n/2 = g_q(k_{n/2}), k_{n/2}, d]$ code; by the balance principle (7) the future subcode C_2 must then have the same parameters. Since C_1 and C_2 meet the Griesmer bound with equality, they also meet the Griesmer bound for all generalized Hamming weights [29, pp. 36–38] and each component of the LDP of each is the minimum possible for the given minimum distance. Since a code cannot have higher generalized Hamming weights than one of its subcodes, C_1 and C_2 must therefore have LDPs that match the LDP of C. Then the properties above are consequences of Theorem 1. ■

Remark. In many cases, codes meeting the Griesmer bound with equality are unique up to equivalence, from results of van Tilborg [34] and Helleseth [16]. In this case $G_2 = G_1$ or a permutation thereof in part (c) above. In addition, the number of uniformly efficient permutations of the code, out of all $n!$, is

$$a_{k_{n/2}, n/2}\, n_c^2, \tag{12}$$

where $a_{i,j}$ is the number of subcodes of dimension i and effective length j, and n_c is the number of permutations of the $[n/2, k_{n/2}, d]$ code in chain condition order.

We may usually find n_c in terms of the parameters of the code. We assume the code is binary for simplicity. If taking a residual with respect to any minimum weight codeword of an $[n, k, d]$ code gives the same $[n - d, k - 1, d_1]$ code up to equivalence, then the number of chain condition orderings for the longer code is $a_d \cdot d! \cdot n_c^{(1)}$, where a_d is the number of codewords of minimum weight of the longer code, and $n_c^{(1)}$ is the number of chain condition orderings of the residual code. The chain of residuals formed by starting with a code at the Griesmer bound usually produces unique codes at each step, and when this applies we find

by iterating this idea that

$$n_c = \prod_{i=0}^{k-1} a_{\min}^i \cdot (\lceil d/2^i \rceil)! \tag{13}$$

where $a_{\min}^i$ is the number of minimum weight codewords in the ith code in the chain.

5. Trellis Structure of Given Self-Dual Codes

5.1. *Binary Self-Dual Codes of Length* ≤ 24

The permutation problem for self-dual codes of length up to 24 is well studied [13,20,27] In particular, Encheva and Cohen [12] establish the existence or nonexistence of a double chain condition permutation for every such code, not just for the extremal ones. Here we consider only the self-dual codes of highest minimum distance, and for this case we develop new characterizations of the resulting uniformly efficient permutations, where they exist, and determine the number of such permutations.

We illustrate the approach using the binary [12,6,4] code and then summarize the results obtained for other codes.

5.1.1. *The [12,6,4] Code*

For the [12,6,4] self-dual code, the weight enumerator is $1 + 15x^4 + 32x^6 + 15x^8 + x^{12}$. By taking residuals as in Section 2.4.1, we find that $d_2 = 6$, i.e., $k_6 = 2$, and we may apply Theorem 2 to conclude that the code has a uniformly efficient permutation, and that a permutation is uniformly efficient if and only if the permuted generator matrix of the form (10) where C_1 and C_2 are [6,2,4] codes containing the codeword 111100.

To find the number of uniformly efficient permutations, we need the coefficient $a_{2,6}$. When the past code is a [4,1,4] code, the future code is an [8,3,4] self-complementary code. Its weight enumerator is then determined as $1+6x^4+x^8$, from which the weight enumerator of its dual is $1 + 4x^2 + 22x^4 + 4x^6 + x^8$. Then, as each [6,2,4] code contains 3 codewords of weight 4, we have $a_{2,6} = 15 \cdot 4/3 = 20$. For the (unique) [6,2,4] code, the number of chain condition permutations is $3 \cdot 4! \cdot 2!$. Then (13) shows that the number of uniformly efficient permutations for the [12,6,4] code is $20(3 \cdot 4! \cdot 2!)^2 = 2^{10} \cdot 3^4 \cdot 5$. The group of this code has order $2^9 \cdot 3^2 \cdot 5$ [5], and thus there are 18 distinct equivalent codes in uniformly efficient order.

5.1.2. *Other Binary Self-Dual Codes of Length* ≤ 24

In the [12,6,4] case above, we were able to determine both the fact that $n/2 = g_2(k_{n/2}, d)$ and the coefficient $a_{k_{n/2},n/2}$ directly from the parameters of the code. This is sometimes possible for other codes also; if not we use results from known classifications of self-dual

Table 1. Binary codes of length ≤ 24.

Code	Result	n_1	Total Permutations	Distinct Codes
[8, 4, 4]	Thm. 2	4	$2^7 \cdot 3^2 \cdot 7$	6
[12, 6, 4]	Thm. 2	6	$2^{10} \cdot 3^4 \cdot 5$	18
[14, 7, 4]	Thm. 2	7	$2^9 \cdot 3^4 \cdot 7^2$	36
[16, 8, 4]	Thm. 2	8	$2^{15} \cdot 3^4 \cdot 7^2$	36 ($2e_8$)
	Thm. 1	8	$2^{15} \cdot 3^4 \cdot 5 \cdot 7$	18 (d_{16}^+)
	Thm. 1	8	$2^{15} \cdot 3^4$	36 ($2d_8^+$)
[18, 9, 4]	NA	—	—	—
[20, 10, 4]	NA	—	—	—
[22, 11, 6]	Thm. 1	10	$2^{19} \cdot 3^8 \cdot 5^3 \cdot 7 \cdot 11$	37324800
[24, 12, 6]	Thm. 1	12	—	—
[24, 12, 8]	Thm. 2	12	$2^{22} \cdot 3^8 \cdot 5^3 \cdot 7^3 \cdot 11 \cdot 23$	1219276800

codes. In Table 1 we collect results obtained from Theorems 1 and 2 for all binary self-dual codes of length up to 24.

We summarize the derivation of these results below.

I. The [8,4,4] *code.* This is one of only two binary self-dual codes for which the elementary approach using Lemma 1 works: the code meets the Griesmer bound with equality and thus must satisfy the chain condition. The LDP, and hence the state complexity, is determined from the fact that equality holds in the Griesmer bound.

II. The [14,7,4] *code.* It is known [30] that the only such code consists of two copies of the [7,3,4] simplex code glued by the all-ones word. Then $k_3 = 7$, and Theorem 2 applies. Since the Griesmer bound is met with equality, the state complexity is determined from part (d) of Theorem 2 as $s = \{0, 1, 2, 3, 2, 3, 2, 1, 2, 3, 2, 3, 2, 1, 0\}$. From the construction, we have $a_{3,7} = 2$, and thus the number of uniformly efficient permutations is $2 \cdot (7 \cdot 4! \cdot 3 \cdot 2!)^2$. Dividing by the order of the group, $168^2 \cdot 2$, we have 36 distinct codes in uniformly efficient order.

III. The [16,8,4] *codes.* There are three self-dual [16,8,4] codes. Two of these are doubly-even: $2e_8$, and d_{16}^+ [5]. The singly-even code is $2d_8^+$ [5].

By definition, $2e_8$ contains e_8, and $a_{4,8}(2e_8) = 2$. We can then apply Theorem 2 to this code, getting the state complexity profile $s(2e_8) = \{0, 1, 2, 3, 2, 3, 2, 1, 0, 1, 2, 3, 2, 3, 2, 1, 0\}$.

A [16,8,4] self-dual code containing a [7,3,4] past subcode has a [9,5,4] code as future code. This meets the Griesmer bound with equality and thus in particular has $d_4 = 8$, i.e., the code also contains e_8

From the known groups it is then easy to show, using the mass formula [5], that e_8 does not belong to d_{16}^+. Thus the code d_8, which is a subcode of d_{16} and has LDP $\{4, 6, 8\}$, must match the LDP of d_{16}^+. Also d_8 is the unique [8,3,4] code that does not contain e_7. Then we may apply Theorem 1 to conclude that d_{16}^+ has a uniformly efficient permutation, and that a permutation is uniformly efficient if and only if it is of the form (10) with C_1 and C_2 both equal to d_8 in chain condition order. The state complexity may be recovered from the LDP of d_8 as $s(d_{16}^+) = \{0, 1, 2, 3, 2, 3, 2, 3, 2, 3, 2, ...\}$. It is known [5] that $a_{3,8}(d_{16}^+) = 70$. The number of chain condition orderings of d_8 is $6 \cdot 4! \cdot 3 \cdot 2! \cdot 2!$.

For the singly-even code $2d_8^+$, it can be shown that the code does not contain e_7, and as it does contain d_8 the same reasoning as in the case of d_{16}^+ applies. We conclude that the code contains a uniformly efficient permutation with the same state complexity as d_{16}^+. It can be shown that $a_{3,8}(2d_8^+) = 2$, from which the number of uniformly efficient permutations can be derived, as in Table 1.

IV. The [18,9] *codes.* There is no [18,9,6] self-dual code. Neither Theorem 1 nor Theorem 2 applies to the [18,9,4] codes. In fact, Encheva and Cohen [13] have shown that exactly one of these two codes satisfies the double chain condition, so there can be no argument that decides this property directly from the code parameters.

The main results apply to self-dual codes with sufficiently high minimum distance. The notion of "sufficiently high" does not exactly coincide with extremality; there are extremal codes for which the results do not apply, as in the case of the [18,9,4] codes, and non-extremal codes in which they do apply, as in the case of the [24,12,6] code.

There is an [18,9,6] formally self-dual code, the extended quadratic residue code. This, however, does not suffice: the key to Theorem 1 is the balance principle (7), which does not necessarily apply to formally self-dual codes. In the case of the [18,9,6] code, the generalized residual lemma shows that $d_2 = 9$, i.e., $k_9 = 2$. Thus $n/2 = g_2(k_{n/2}, d)$ and the main condition of Theorem 2 holds. However, Kiely et al. [21] and Encheva [12] showed that this code does not satisfy the double chain condition.

The length of the code exceeds the Griesmer bound by one, and thus by the result of Helleseth et al. [17] the code must satisfy the chain condition. This shows that formally self-dual codes that satisfy the chain condition do not necessarily satisfy the double chain condition; thus Lemma 1 does not extend to this case.

V. The [22,11,6] *code.* Here $d_2 \geq 10$, as the [9,2,6] code is not self-orthogonal. Taking residuals shows that $d_2 = 10$; then with a [10,2,6] code as past code, the future code is a [12,3,6] code, which exceeds the Griesmer bound by one and thus satisfies the chain condition. Its LDP, {6, 10, 12}, matches the LDP of the overall code. Theorem 1 applies. A similar development shows that $a_{2,10} = 2310$, and the number of chain condition orders of the unique self-orthogonal [12,3,6] code is $4 \cdot 6! \cdot 3 \cdot 4! \cdot 2!$, from which the number of uniformly efficient permutations in Table 1 results.

VI. The [24,12,8] *Golay code.* The trellis structure of the [24,12,8] Golay code is particularly well understood. A uniformly efficient permutation was found by Muder [27]. A question posed by McEliece (unpublished) asks for the number of uniformly efficient permutations, which we provide in this section.

First, we note that the code exceeds the Griesmer bound by one in length. Thus, by Lemma 1 the code must satisfy the double chain condition. Note that this argument by itself does not provide the state complexity of such a permutation.

By taking residuals, we find that $d_2(C) = 12$, i.e., $k_{12} = 2$. Then using the Wei relations, the full LDP can be calculated, as in Pless et al. [29], as $\mathbf{d}(C) = \{8, 12, 14, 16, 18, 19, 20, 21, 22, 23, 24\}$; of course this was also obtained by Muder [27].

Since $12 = g(k_{12}, 8)$, we can again apply Theorem 2. We conclude that the extended Golay code has a uniformly efficient permutation, with optimum state complexity profile $\{0, 1, 2, 3, 4, 5, 6, 7, 6, 7, 8, 9, 8, \ldots\}$. A permutation is uniformly efficient if and only if it is of the form (10) in which $C_1 = C_2 =$ [12,2,8], with each containing the codeword $(1_8 0_4)$.

A set of 12 positions holding three codewords of the Golay code is an X_{12} in the notation of Conway and Sloane [8]; thus $a_{2,12}(C) = |X_{12}| = 35420$. The number of permutations of the [12,2,8] code containing the codeword $(1_8 0_4)$ is $3 \cdot 8! \cdot 4!$, so the number of uniformly efficient permutations out of all 24! is $35420(3 \cdot 8! \cdot 4!)^2$, i.e., a fraction of approximately 4.81×10^{-7} of all permutations. Dividing by $|\mathrm{Aut}(\mathcal{G}_{24})| = 244823040$ [26, p. 639] gives a total of 1219276800 distinct codes in uniformly efficient order.

An alternative approach is to apply Theorem 1 with $n_1 = 8$ and $n_2 = 16$. With C_1 the [8,1,8] code, C_2 is the [16,5,8] code. Theorem 2 applies.

5.1.3. Shortened Golay Codes

The following observations are useful in Section 5.5. From Section 2.4.2 and the results above, an i-shortened Golay code for $i \leq 5$ must have an LDP that matches the LDP of the Golay code, and must satisfy the chain condition.

There are exactly two distinct [18,6,8] codes up to equivalence, and these may obtained by shortening the [24,12,8] Golay code on either an S_6 or a U_6 [9]. Since the length exceeds the Griesmer bound by one, each code satisfies the chain condition. However, only the LDP of the S_6-shortened code matches the LDP of the Golay code; that it does so follows again from Section 2.4.2. A 5-dimensional subcode of the Golay code of support size 16 has as complement a codeword of weight 8, i.e., an S_8. To achieve $d_5 = 16$, we must shorten on a 6-dimensional subset of this set of 8 positions, i.e., on an S_6. The LDP of the U_6-shortened [18,6,8] code can be shown to be {8, 12, 14, 15, 17, 18}. (The statement in [29, p. 39], that the generalized Hamming weight hierarchies of codes that exceed the Griesmer bound by one are known, is therefore inaccurate, if 'known' is interpreted as 'completely determined.')

5.2. Ternary Self-Dual Codes of Length $\leq$ 24

Ternary self-dual codes of length $\leq$ 20 have previously been studied by Encheva and Cohen [14], who classify for each such code, again not necessarily extremal, whether there exists a uniformly efficient permutation or not. Again we provide a different derivation for all such codes of sufficiently high minimum distance, and enumerate all uniformly efficient permutations where our results apply, for codes of length up to 24. The results are summarized in Table 2. To determine the number of distinct codes in each case, we have used information on the groups of these codes from [6,24].

I. The $[12,6,6]_3$ *Golay code.* This code meets the Griesmer bound with equality. Thus from Lemma 1 there is at least one uniformly efficient permutation, with state complexity $s = \{0, 1, 2, 3, 4, 5, 4, 5, 4, 3, 2, 1, 0\}$. A permutation is uniformly efficient if and only if it contains a minimum weight codeword in the first six positions.

II. The $[16,8,6]_3$ *code.* A past $[6,1,6]_3$ code corresponds to a future $[10,3,6]_3$ code. This exceeds the Griesmer bound by one, and thus satisfies the chain condition; since there is no $[10,3,7]_3$ code, this future code matches the LDP of the $[16,8,6]_3$ code. Theorem 1 applies. Alternatively, we find further by taking residuals that $a_{2,8} = 224 \cdot 6/8 = 168$, from which Theorem 2 applies, and from which we may enumerate the uniformly efficient permutations.

Table 2. Ternary codes of length ≤ 24.

Code	Result	Total Permutations	Distinct Codes
$[12,6,6]_3$	Thm. 2	$2^{10} \cdot 3^5 \cdot 5^2 \cdot 11$	720
$[16,8,6]_3$	Thm. 2	$2^{17} \cdot 3^5 \cdot 5^2 \cdot 7$	129600
$[20,10,6]_3$	NA	—	—
$[24,12,9]_3$	Thm. 2	$2^{26} \cdot 3^{10} \cdot 5^3 \cdot 7^2 \cdot 11 \cdot 23$	$2^{20} \cdot 3^9 \cdot 5^3 \cdot 7^2$ (Q_{24}) $2^{19} \cdot 3^9 \cdot 5^2 \cdot 7^2 \cdot 23$ (P_{24})

III. The $[24,12,9]_3$ *codes.* This case is particularly interesting, as there are two codes with these parameters. We demonstrate that each has exactly the same number of uniformly efficient permutations out of all 24!; this illustrates the point that Theorems 1 and 2 yield information about uniformly efficient permutations that does not depend on detailed knowledge of the generator matrix of the code.

Any ternary self-dual code must have all codeword weights divisible by 3, and the weight enumerator of any $[24,12,9]_3$ self-dual code is therefore determined, with 4048 codewords of weight 9 [24]. With a [9,1,9] past subcode, the future code is $[15,4,9]_3$, again with all codeword weights divisible by 3, and as its dual has distance at least 3, the weight enumerator of the future code is also fixed, as $1 + 50x^9 + 30x^{12}$. Its dual has weight enumerator $1 + 40x^3 + \cdots$. Then $d_2 = 12$, and Theorem 2 applies. We have $a_{2,12} = 4048 \cdot 40/8 = 20240$, and a total of $a_{2,12} n_c^2 = 20240(4 \cdot 9! \cdot 3!)^2$ uniformly efficient permutations of each code. The groups have different sizes [24] and so the number of distinct codes in uniformly efficient order is different.

5.3. The [32,16,8] Self-Dual Codes

There are 85 inequivalent self-dual, doubly-even [32,16] codes, which have been classified by Conway and Pless [5]. Five of these have minimum weight 8, namely, an extended quadratic residue code q_{32}, a Reed-Muller code r_{32}, and three other codes, denoted as $2g_{16}^+$, $8f_4^+$, and $16f_2^+$. In addition, there are also exactly three inequivalent self-dual, singly-even [32,16,8] codes [7] and they can be obtained from r_{32}, $2g_{16}^+$ and $8f_4^+$. These will be denoted by s-r_{32}, s-$2g_{16}^+$, and s-$8f_4^+$.

It will emerge that the generalized Hamming weights of these codes differ. In this section, we demonstrate that all must have uniformly efficient permutations, though with a variety of optimal state complexity profiles. We then compute which state complexity profiles belong to which codes. We do not know the relevant $a_{i,j}$ for any of these codes, so cannot compute the number of uniformly efficient permutations.

THEOREM 3. *A [32,16,8] doubly-even self-dual code must have a uniformly efficient permutation.*

Proof. Let C be any such code. If $d_5(C) = 16$, we may apply Theorem 2 to conclude that the code has a uniformly efficient permutation; we may calculate the resulting state complexity profile from part (d) of Theorem 2.

If $d_4(C) = 15$ but $d_5(C) > 16$, we may write the generator matrix in the form (10) where G'_1 generates a [15,4,8] code and G'_2 a [17,5,8] code by the balance principle (and so $d_5 = 17$). The [17,5,8] code must have dual distance greater than 1, as 16 is not in the LDP, and then linear programming shows that the dual distance is exactly 2. Then all smaller generalized Hamming weights meet the generalized Griesmer bound with equality. Also C_1 and C_2 have lengths at and one greater than the Griesmer bound, respectively, so must each satisfy the chain condition. Then we may apply Theorem 1 to conclude that the code has a uniformly efficient permutation. The state complexity profile may be recovered from the LDP's of C_1 and C_2.

If $d_3(C) = 14$ but $d_4 > 15$, then with the [14,3,8] code as past code, the future code has parameters [18,5,8], dual distance greater than 1 (since $d_4 > 15$), and can have only 8, 12, and 16 as nonzero weights. Such a code cannot have dual distance 3, so must have dual distance exactly 2. Then we may write the generator matrix in the form (10) where G'_1 and G'_2 both generate [16,4,8] codes. Repeating the argument shows that each [16,4,8] code has LDP {8, 12, 14, 16}, which matches the LDP of C; each exceeds the Griesmer bound by one in length, and thus satisfies the chain condition. Then Theorem 1 applies, and the optimal state complexity profile is determined.

Finally, we show that C must have $d_3 = 14$. First, any [32,16,8] code has $d_2 = 12$ by taking residuals, since there is no [24,15,5] code [3]. Then a [12,2,8] code as past code yields a [20,6,8] code as future code, with possible nonzero weights 8, 12, and 16. Again such a code must have dual distance exactly 2, and adjoining two positions to the past 12 yields a [14,3,8] code. ■

The remaining problems are to decide which of the [32,16,8] doubly-even codes fit which cases, and to find how the singly-even codes behave. Here we use facts from the classification of these codes and the properties of self-dual codes [5, 22].

Both r_{32} and $2g_{16}^+$ contain the first-order Reed-Muller code g_{16} by construction [5]. These codes therefore have $d_5 = 16$.

Via the mass formula for doubly-even codes [5] and the known group sizes [5,30], it follows readily that these are the only doubly-even codes that contain $2g_{16}$ as a subcode. But g_{16} is present if and only if $2g_{16}$ is present by using the balance principle, so the other codes have $d_5 > 16$.

Now we invoke a result of Koch [22], who proved that the unique g_{15}[15,4,8] code (the simplex code of length 15) is a subcode of $16f_2^+$, but not a subcode of q_{32} or $8f_4^+$. Then $16f_2^+$ has $d_4 = 15$, while q_{32} and $8f_4^+$ have $d_4 = 16$.

A generator matrix for a uniformly efficient permutation for each of the [32,16,8] doubly-even self-dual codes is given in Tables 2–5. Uniformly efficient permutations and the corresponding state complexity profiles were already known for r_{32} [19] and q_{32} [2], though in the case of q_{32} the permutation was not known to be uniformly efficient.

5.3.1. Singly-Even Self-Dual Codes of Length 32

The three singly-even [32,16,8] codes s-r_{32}, s-$2g_{16}^+$ and s-$8f_4^+$ may be constructed by modifying the doubly-even counterpart [7]. Each of these has the same optimal state complexity

Table 3. Uniformly efficient permutations for r_{32} and s-r_{32}.

1111111100000000	0000000000000000
1111000011110000	0000000000000000
1100110011001100	0000000000000000
1010101010101010	0000000000000000
1001011001101001	0000000000000000
0000000000000000	0000000011111111
0000000000000000	0000111100001111
0000000000000000	0011001100110011
0000000000000000	0101010101010101
0000000000000000	1001011001101001
1000001000101000	0001010001000001
0100001011101000	0001011101000010
0010001010001000	0001000101000100
0001001001001000	0001001001001000
0000101010100000	0000010101010000
0000011001100000	0000011001100000

1111111100000000	0000000000000000
1111000011110000	0000000000000000
1100110011001100	0000000000000000
1010101010101010	0000000000000000
1001011001101001	0000000000000000
0000000000000000	0000000011111111
0000000000000000	0000111100001111
0000000000000000	0011001100110011
0000000000000000	0101010101010101
0000000000000000	1001011001101001
0111001000101000	0001010001000001
1011001011101000	0001011101000010
1101001010001000	0001000101000100
1110001001001000	0001001001001000
0000101010100000	0000010101010000
0000011001100000	0000011001100000

Table 4. Uniformly efficient permutations for $2g_{16}^{+}$ and s-$2g_{16}^{+}$.

1111111100000000	0000000000000000
1011100011110000	0000000000000000
1110010010101100	0000000000000000
1100101011001010	0000000000000000
1001011001101001	0000000000000000
0000000000000000	0000000011111111
0000000000000000	0000111100010111
0000000000000000	0011010100101101
0000000000000000	0101011001001110
0000000000000000	1001001101110100
1000001001001000	0001011100110101
0100001011000000	0001011101010110
0010001011101000	0001011100011000
0001001000101000	0000011001001000
0000101010001000	0001011101101100
0000011001100000	0000001101110111

1111111100000000	0000000000000000
1011100011110000	0000000000000000
1110010010101100	0000000000000000
1100101011001010	0000000000000000
1001011001101001	0000000000000000
0000000000000000	0000000011111111
0000000000000000	0000111100010111
0000000000000000	0011010100101101
0000000000000000	0101011001001110
0000000000000000	1001001101110100
1000001001001000	0001011100110101
1101010011000000	0001011101010110
1011010011101000	0001011100011000
0001001000101000	0000011001001000
1001110010001000	0001011101101100
0000011001100000	0000001101110111

profile as its doubly-even counterpart. A sketch of the proof follows; fuller details are in [4].

An elementary search shows that we may modify r_{32} and $2g_{16}^{+}$ to obtain the singly-even counterparts without changing the partition between past and future codes at the center, and thus both s-r_{32} and s-$2g_{16}^{+}$ have $d_5 = 16$ also, from which we may apply Theorem 2.

For $8f_4^{+}$ with generator matrix as in Table 5, we again have the property that we may modify the code to obtain s-$8f_4^{+}$ while still maintaining the [16,4,8] doubly-even code with LDP {8, 12, 14, 16} as past and future codes. The remaining question is whether or not this LDP matches the LDP of s-$8f_4^{+}$; if it did not, then we would have $d_4(s\text{-}8f_4^{+}) = 15$. The

Table 5. Uniformly efficient permutations for $8f_4^+$ and s-$8f_4^+$.

1111111100000000	0000000000000000
1111000011110000	0000000000000000
1100101011001100	0000000000000000
0011101011000011	0000000000000000
0000000000000000	0000000011111111
0000000000000000	0000111100001111
0000000000000000	0011011001100101
0000000000000000	1100011001101010
0100100001001000	0001010001000001
1110010011100010	0100010001000010
0111010011101000	0001010001000100
0010100001000010	0100010001001000
0110011010101010	0101000001010000
0110000011001010	0000000001100000
0000101011000000	0101010100000000
0110110011000000	0000011000000000

1111111100000000	0000000000000000
1111000011110000	0000000000000000
1100101011001100	0000000000000000
0011101011000011	0000000000000000
0000000000000000	0000000011111111
0000000000000000	0000111100001111
0000000000000000	0011011001100101
0000000000000000	1100011001101010
1011100001001000	0001010001000001
0001010011100010	0100010001000010
1000010011101000	0001010001000100
1101100001000010	0100010001001000
0110011010101010	0101000001010000
0110000011001010	0000000001100000
0000101011000000	0101010100000000
0110110011000000	0000011000000000

Table 6. Uniformly efficient permutation for $16f_2^+$.

1111111100000000	0000000000000000
1011100011110000	0000000000000000
0101110011001100	0000000000000000
0011011001101010	0000000000000000
0000000000000000	0000000011111111
0000000000000000	0000111101010101
0000000000000000	0011001101011010
0000000000000000	0101011000111100
0010111001101001	0000011001000001
1101111010001000	0001001001000010
1000001010001001	0001000001000100
0100010001101000	0000010001001000
1100011001001001	0001011001010000
0000011011001000	0000001001100000
0100100011100000	0001010100000000
1010101000101001	1000000000000000

modification of doubly-even to singly-even involves changing by 2 the weights of codewords that are not orthogonal to a given set of four positions t_1 [7], while leaving unchanged the weights of codewords that are orthogonal to t_1. If s-$8f_4^+$ has $d_4 = 15$ then it contains the unique g_{15}[15,4,8] code. This is doubly-even, and so we must have $u \cdot t_1 = 0$ for all $u \in g_{15}$. This implies that g_{15} is a subcode of $8f_4^+$, and this has already been ruled out.

A generator matrix for a uniformly efficient permutation for each of the [32,16,8] singly-even self-dual codes is given in Tables 3, 4, and 5.

In summary, we have the following theorem for [32,16,8] self-dual codes. The absolute state complexity is 9 for these eight codes. The fact that $d_3 = 14$ for each of the doubly-even codes plus the Wei relations (3) determine all Hamming weights apart from the fourth

Table 7. Uniformly efficient permutation for q_{32}.

1111111100000000	0000000000000000
1110001011110000	0000000000000000
1011100011001100	0000000000000000
1010011001001011	0000000000000000
0000000000000000	0000000011111111
0000000000000000	0000111101010101
0000000000000000	0011001101110010
0000000000000000	1101011100000110
0001111011100010	0101000001000001
0110011001000000	0000010001000010
0010001001100010	0001000001000100
1000100010001000	0001010001001000
0011110001000010	0000000001010000
1110001001101010	0100010001100000
0110000001100000	0101010100000000
0011000010001000	0101011000000000

and fifth, which could only be (15, 16), (15, 17), or (16, 18). Each of these possibilities arises.

THEOREM 4. *Every [32,16,8] binary self-dual code satisfies the double chain condition. The LDP and optimum SCP of r_{32}, s-r_{32}, $2g_{16}^{+}$, and s-$2g_{16}^{+}$ are*

$$\mathbf{d} = \{8, 12, 14, \mathbf{15}, \mathbf{16}, 20, 22, 23, 24, 26, 27, 28, 29, 30, 31, 32\},$$

$$\mathbf{s} = \{0, 1, 2, 3, 4, 5, 6, 7, 6, 7, 8, 9, 8, 9, 8, \mathbf{7}, \mathbf{6}, \mathbf{7}, 8, 9, 8, 9, 8, 7, 6, 7, 6, 5, 4, 3, 2, 1, 0\}.$$

The LDP and optimum SCP of $16f_2^{+}$ are

$$\mathbf{d} = \{8, 12, 14, \mathbf{15}, \mathbf{17}, 20, 22, 23, 24, 26, 27, 28, 29, 30, 31, 32\},$$

$$\mathbf{s} = \{0, 1, 2, 3, 4, 5, 6, 7, 6, 7, 8, 9, 8, 9, 8, \mathbf{7}, \mathbf{8}, \mathbf{7}, 8, 9, 8, 9, 8, 7, 6, 7, 6, 5, 4, 3, 2, 1, 0\}.$$

The LDP and optimum SCP of $8f_4^{+}$, s-$8f_4^{+}$, and q_{32} are

$$\mathbf{d} = \{8, 12, 14, \mathbf{16}, \mathbf{18}, 20, 22, 23, 24, 26, 27, 28, 29, 30, 31, 32\},$$

$$\mathbf{s} = \{0, 1, 2, 3, 4, 5, 6, 7, 6, 7, 8, 9, 8, 9, 8, \mathbf{9}, \mathbf{8}, \mathbf{9}, 8, 9, 8, 9, 8, 7, 6, 7, 6, 5, 4, 3, 2, 1, 0\}.$$

5.4. A [32,16,4] Code

One of the doubly-even [32,16,4] codes is worth noting, as it illustrates the main difficulty in applying the general method. This is the code obtained by taking a direct sum of the two distinct doubly-even [16,8,4] codes, $2e_8$ and d_{16}. This is code $C6$ in the classification of Conway and Pless [30]. We have seen that both of the component codes satisfy the chain

condition (in fact, the double chain condition). However, the overall code does not satisfy the double chain condition, as we may achieve either $s_8 = 0$ (by placing $2e_8$ first) or $s_{24} = 0$ (by placing $2e_8$ second) but not both simultaneously. In terms of the conditions listed in Theorem 1, the problem is that only one of the glued codes matches the LDP of the overall code.

5.5. *The Self-Dual Codes of Length* $\geq$ 40

Since there are very many extremal doubly-even self-dual codes of length 40 [5], we aim for blocklengths this long and longer to find a self-dual code with the smallest state complexity profile over all possible self-dual codes of the same parameters.

The strategy is to start with a self-orthogonal code $C_1[n/2, k_{\max}, d]$ with the smallest LDP, component by component, among all $[n/2, \leq k_{\max}, d]$ self-orthogonal codes, where $k_{\max}$ is the maximum dimension of a length $n/2$ binary self-orthogonal code with minimum distance $\geq d$, and to verify that C_1 satisfies the chain condition. We then seek appropriate glue vectors to produce a self-dual $[n, n/2, d]$ code; we must verify that it is possible to choose the glue vectors to maintain the minimum distance as d.

Assuming this is possible, we may apply Theorem 1 to conclude that the permutation with corresponding code generated by

$$G_C = \begin{bmatrix} G_1 & 0 \\ 0 & \tau(G_1) \\ E & F \end{bmatrix}, \tag{14}$$

is uniformly efficient, and in fact is uniformly concise in the class of all $[n, k, d]$ self-dual codes.

If $C_1 = \langle G_1 \rangle$ has $d_1^\perp \geq d/2$, we may choose any E and F so that $\langle G_1 \cup E \rangle = \langle \tau(G_1) \cup F \rangle = C_1^\perp$. Then the code with generator matrix of the form (14) is self-dual, and has minimum distance at least $\min\{d_1, 2d_1^\perp\}$ [26, p. 584], i.e., at least d if C_1 has the given dual distance.

5.5.1. *The [40,20,8] Self-Dual Codes:*

We can start with the optimal length 20 binary self-orthogonal code with minimum distance 8, the [20,8,8] code obtained by shortening the [24,12,8] Golay code in any 4 positions. This code is unique [9]. From Section 5.1.2 its LDP matches the LDP of the Golay code, and the code satisfies the chain condition. Its LDP is then $\mathbf{d}(g_{20}) = \{8, 12, 14, 15, 16, 18, 19, 20\}$. Each d_i is the smallest possible for any code with minimum distance 8 (up to d_5, they meet the generalized Griesmer bound; and there is no [17,6,8] code). Since $d(g_{20}^\perp) = 4$, we can choose E so that $\langle E \cup C_1 \rangle = C_1^\perp$, and $F = \tau(E)$. Then the code with generator matrix of the form (14) is a self-dual (in fact, doubly-even) code that satisfies the double chain condition, is in uniformly efficient order, and is uniformly concise in the class of all [40,20,8] binary self-dual codes, with state complexity profile

$$\mathbf{s} = \{0, 1, 2, 3, 4, 5, 6, 7, \mathbf{6}, 7, 8, 9, \mathbf{8}, 9, \mathbf{8}, \mathbf{7}, \mathbf{6}, 7, \mathbf{6}, \mathbf{5}, \mathbf{4}, \ldots\}.$$

By pairing different glue vectors, we can easily obtain a singly-even self-dual [40,20,8] code with the same properties [4].

Similar constructions produce uniformly concise permutations for the class of [42,21,8] and [44,22,8] self-dual codes. We remark that in applying Theorem 1, we are only concerned with whether each component code satisfies the chain condition, not the double chain condition. Here, for example, the [21,9,8] shortened Golay code satisfies the chain condition, but Encheva [12] has shown that it does not satisfy the double chain condition.

5.5.2. The [48,24,12] Self-Dual Code(s):

The only known [48,24,12] self-dual code is the extended quadratic residue code; whether another exists is an open problem [30]. A uniformly efficient permutation was found via a combination of heuristic and computer search by Dolinar et al. [10]. Another was found by Berger and Be'ery [2]. The resulting state complexity profile is

$$\mathbf{s} = \{0, 1, \ldots, 11, \mathbf{10}, 11, 12, 13, 14, 15, \mathbf{14}, 15, 16, \mathbf{15}, 16, \mathbf{15}, \mathbf{14}, \ldots\}.$$

Via a direct application of Theorem 2 and equation (12), we can conclude the following for any [48,24,12] self-dual code, not necessarily just the quadratic residue code:

If a [48,24,12] self-dual code contains a [24,5,12] subcode, then

— it has a uniformly efficient permutation that meets the DLP bound;

— the permutation is uniformly concise over the class of [48,24,12] self-dual codes;

— a permutation is uniformly efficient if and only if the resulting code has generator matrix of the form

$$G = \begin{pmatrix} G^{\rightarrow}_{[24,5,12]} & 0 \\ 0 & \tau\left(G^{\rightarrow}_{[24,5,12]}\right) \\ E & F \end{pmatrix}$$

where $G^{\rightarrow}_{[24,5,12]}$ is a generator matrix for the unique [24,5,12] code in chain condition order, and $\tau(\cdot)$ is the reverse permutation;

— the number of uniformly efficient permutations out of all 48! is $2^{44}3^{20}5^{6}7^{4}11^{2}a_{5,24}$, where $a_{5,24}$ is the number of distinct 5-dimensional subspaces of effective length 24.

The last part follows because taking a chain of residual codes with respect to minimum weight codewords gives [24,5,12] $\rightarrow$ [12,4,6] $\rightarrow$ [6,3,3] $\rightarrow \cdots$. Each of these codes is unique by the results of van Tilborg [34] and Helleseth [16]; the respective weight enumerators are $1 + 28x^{12} + 3x^{16}$, $1 + 12x^{6} + 3x^{8}$, and $1 + 4x^{3} + 3x^{4}$. We then apply (12).

We do not know of any purely combinatorial argument showing that the [24,5,12] code must be a subcode of [48,24,12] codes in general or the quadratic residue code in particular. In the case of the quadratic residue code, we can find this by inspection of the two known uniformly efficient generator matrices [2,10], which generate this code with the first five and last five rows in each case.

Since the number of uniformly efficient permutations must be divisible by the automorphism group of the code, and the complement of a [24,5,12] subspace has the same parameters, we find in the case of the quadratic residue code that $a_{5,24}$ must be divisible by $2 \cdot 23 \cdot 47$, and then dividing by $\mathrm{Aut}(Q_{48}) = 47(47^2 - 1)/2$ [26, pp. 491, 494], we have a total of at least 11.6×10^{31} distinct codes in uniformly efficient order. Of course, this represents a very small fraction (4.8×10^{-26}) of all distinct permutations of Q_{48}.

6. Conclusion

The optimal permutation problem for linear block codes may be simplified substantially for the class of self-dual codes. For the self-dual codes of highest minimum distance, it is often the case that the existence of a uniformly efficient permutation, and a full classification of all possible such permutations, may be deduced. This is the case when appropriately chosen subcodes have lengths sufficiently close to the Griesmer bound.

This result, among others, is used in the paper to classify the eight self-dual [32,16,8] codes according to optimal trellis structure and generalized Hamming weights. Optimal trellises and generalized Hamming weights are similarly derived for several other self-dual codes, including the lowest complexity [40,20,8], [42,21,8] and [44,22,8] binary and both [24,12,9] ternary codes. A new characterization is given of uniformly efficient permutations for the [48,24,12] quadratic residue code, and the number of distinct uniformly efficient permutations for many self-dual codes of length up to 24 is determined.

Note

1. This is the definition of uniform efficiency given by Vardy [35, p. 2058]. The definition given by Kiely et al. [21] is slightly stronger, and corresponds to meeting the DLP bound (6). Permutations satisfying the weaker condition are termed "efficient, but not uniformly so" by Kiely et al. The term "efficient coordinate ordering" [12] is synonymous with meeting the DLP bound.

References

1. L. R. Bahl, J. Cocke, F. Jelinek and J. Raviv, Optimal decoding of linear codes for minimizing symbol error rate, *IEEE Trans. Inform. Theory,* Vol. IT–20, No. 2 (1974) pp. 284–287.
2. Y. Berger and Y. Be'ery, The twisted squaring construction, trellis complexity and generalized weights of BCH and QR codes, *IEEE Trans. Inform. Theory*, Vol. IT–42, No. 6 (1996) pp. 1817–1827.
3. A. E. Brouwer and T. Verhoeff, An updated table of minimum-distance bounds for binary linear codes, *IEEE Trans. Inform. Theory*, Vol. 39, No. 2 (1993) pp. 662–677.
4. H. Chen, *Optimal encoding, trellis structure, and normalized weight of linear block codes*, PhD. Dissertation, University of Michigan, May (1999).
5. J. H. Conway and V. Pless, On the enumeration of self-dual codes, *J. Combin. Theory Ser. **A**,* Vol. 28, No. 1 (1980) pp. 26–53.
6. J. H. Conway, V. Pless, and N. J. A. Sloane, Self-dual codes over GF(3) and GF(4) of length not exceeding 16, *IEEE Trans. Inform. Theory,* Vol. IT-25 (1979) pp. 312–322.
7. J. H. Conway and N. J. A. Sloane, A new upper bound on the minimal distance of self-dual codes, *IEEE Trans. Inform. Theory,* Vol. 36, No. 6 (1990) pp. 1319–1333.

8. J. H. Conway and N. J. A. Sloane, Orbit and coset analysis of the Golay and related codes, *IEEE Trans. Inform. Theory,* Vol. IT-36, No. 5 (1990) pp. 1038–1050.
9. S. M. Dodunekov and S. B. Encheva, Uniqueness of some linear subcodes of the extended binary Golay code, *Problemy Peredachi Informatsii,* Vol. 29, No. 1 (1993) pp. 45–51. In Russian. English translation in *Problems of Information Transmission,* Vol. 29 No. 1 (1993) pp. 38–43.
10. S. Dolinar, L. Ekroot, A. B. Kiely, R. J. McEliece and W. Lin, The permutation trellis complexity of linear block codes, In *Proc. 32nd Allerton Conf. on Communications, Control, and Computing,* Monticello, IL, pp. 60–74, September (1994).
11. S. B. Encheva, On the binary linear codes which satisfy the two-way chain condition, *IEEE Trans. Inform. Theory,* Vol. IT-42, No. 3 (1996) pp. 1038–1047.
12. S. B. Encheva, On repeated-root cyclic codes and the two-way chain condition, in *Lecture Notes Comput. Sci.,* Vol. 1255, Springer-Verlag (1997) pp. 78–87.
13. S. B. Encheva and G. D. Cohen, Self-orthogonal codes and their coordinate ordering, *IEICE Trans. Fundamentals,* Vol. E80-A, No. 11 (1997) pp. 2256–2259.
14. S. B. Encheva and G. D. Cohen, On the state complexities of ternary codes, In *Proceedings of AAECC-13 (Honolulu, Hawaii),* November (1999) pp. 454–461.
15. G. D. Forney Jr., Dimension/length profiles and trellis complexity of linear block codes, *IEEE Trans. Inform. Theory,* Vol. IT–40, No. 6 (1994) pp. 1741–1752.
16. T. Helleseth, A characterization of codes meeting the Griesmer bound, *Information and Control,* Vol. 50, No. 2 (1981) pp. 128–159.
17. T. Helleseth, T. Kløve and Ø. Ytrehus, Generalized Hamming weights of linear codes, *IEEE Trans. Inform. Theory,* Vol. IT-38, No. 3 (1992) pp. 1133–1140.
18. G. B. Horn and F. R. Kschischang, On the intractability of permuting a block code to minimize trellis complexity, *IEEE Trans. Inform. Theory,* Vol. IT–42, No. 6 (1996) pp. 2042–2048.
19. T. Kasami, T. Takata, T. Fujiwara and S. Lin, On complexity of trellis structure of linear block codes, *IEEE Trans. Inform. Theory,* Vol. IT–39, No. 3 (1993) pp. 1057–1064.
20. T. Kasami, T. Takata, T. Fujiwara and S. Lin, On the optimum bit orders with respect to the state complexity of trellis diagrams for binary linear codes, *IEEE Trans. Inform. Theory,* Vol. IT–39, No. 1 (1993) pp. 242–245.
21. A. B. Kiely, S. Dolinar, R. J. McEliece, L. Ekroot and W. Lin, Trellis decoding complexity of linear block codes, *IEEE Trans. Inform. Theory*, Vol. IT–42, No. 6 (1996) pp. 1687–1697.
22. H. Koch, On self-dual, doubly-even codes of length 32, *J. Combin. Theory Ser.* $\boldsymbol{A}$, Vol. 51, No. 1 (1989) pp. 63–67.
23. F. R. Kschischang and G. B. Horn, A heuristic for ordering a linear block code to minimize trellis state complexity, In *Proc. 32nd Allerton Conf. on Communications, Control, and Computing,* Monticello, IL, September (1994) pp. 75–84.
24. J. S. Leon, V. Pless and N. J. A. Sloane, On ternary self-dual codes of length 24, *IEEE Trans. Inform. Theory,* Vol. IT-27, No. 2 (1981) pp. 176–180.
25. S. Lin, T. Kasami, T. Fujiwara and M. Fossorier, *Trellises and Trellis-Based Decoding Algorithms for Linear Block Codes*, Kluwer Academic, Boston, Massachusetts (1998).
26. F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*, North-Holland, Amsterdam, The Netherlands (1978).
27. D. J. Muder, Minimal trellises for block codes, *IEEE Trans. Inform. Theory,* Vol. IT–34, No. 5 (1988) pp. 1049–1053.
28. V. S. Pless, Coding constructions, in *Handbook of Coding Theory, Volume I*, (V. S. Pless and W. C. Huffman, eds.), North-Holland Amsterdam, The Netherlands (1998).
29. V. S. Pless, W. C. Huffman and R. Brualdi, An introduction to algebraic codes, in *Handbook of Coding Theory, Volume I,* (V. S. Pless and W. C. Huffman, eds.), North-Holland, Amsterdam, The Netherlands (1998).
30. E. M. Rains and N. J. A. Sloane, Self-dual codes, in *Handbook of Coding Theory, Volume I,* (V. S. Pless and W. C. Huffman, eds.), North-Holland, Amsterdam, The Netherlands (1998).
31. J. Simonis, The effective length of subcodes, *Appl. Algebra Engrg. Commun. Comput.,* Vol. 5, No. 6 (1994) pp. 371–377.

32. M. A. Tsfasman and S. G. Vlăduţ, Geometric approach to higher weights, *IEEE Trans. Inform. Theory,* Vol. 41, No. 6 (1995) pp. 1564–1588.
33. J. H. van Lint, *Introduction to Coding Theory*, Springer-Verlag, New York (1982).
34. H. C. A. van Tilborg, On the uniqueness resp. nonexistence of certain codes meeting the Griesmer bound, *Information and Control,* Vol. 44, No. 1 (1980) pp. 16–35.
35. A. Vardy, Trellis structure of codes, in *Handbook of Coding Theory, Volume II*, (V. S. Pless and W. C. Huffman, eds.), North-Holland, Amsterdam, The Netherlands (1998).
36. V. K. Wei, Generalized Hamming weights for linear codes, *IEEE Trans. Inform. Theory,* Vol. IT–37, No. 5 (1991) pp. 1412–1418.