NEW EXTREMAL BINARY SELF-DUAL CODES OF LENGTH 68 FROM GENERALIZED NEIGHBORS

JOE GILDEA, ABIDIN KAYA, ADRIAN KORBAN AND BAHATTIN YILDIZ

Abstract. In this work, we use the concept of distance between self-dual codes, which generalizes the concept of a neighbor for self-dual codes. Using the k-neighbors, we are able to construct extremal binary self-dual codes of length 68 with new weight enumerators. We construct 143 extremal binary self-dual codes of length 68 with new weight enumerators including 42 codes with $\gamma = 8$ in their $W_{68,2}$ and 40 with $\gamma = 9$ in their $W_{68,2}$. These examples are the first in the literature for these γ values. This completes the theoretical list of possible values for γ in $W_{68,2}$.

1. Introduction

Self-dual codes are a special class of linear codes. Because of the many interesting properties that they have and the many different fields that they are connected with, they have attracted a considerable interest in coding theory research community.

One of the most active research areas in the field of self-dual codes is the construction and classification of extremal binary self-dual codes. Type I extremal binary self-dual codes of lengths such as 64, 66, 68, etc. have parameters in their weight enumerators, which have not all been found to exist. Hence, the recent years have seen a surge of activity in finding extremal binary self-dual codes of various lengths with new weight enumerators. Many different techniques have been employed in constructing these extremal binary self-dual codes such as constructions over certain rings, constructions through automorphism groups, neighboring constructions, shadows, extensions, etc. [1], [4], [7], [8], [10], [13], [14] are just a sample of the works that contain these ideas and their applications in finding new extremal binary self-dual codes.

In this work, we use the concept of "distance" between self-dual codes. After proving some theoretical results about the distance we observe that the neighbor can be defined in terms of the distance and this leads to the concept of "k-range neighbors" or "k-neighbors", which generalize the concept of neighbors of self-dual codes. We then use these k-neighbors to construct extremal self-dual codes of length 68 from a given self-dual code. In particular we construct 139 new extremal binary self-dual codes of length 68 with new weight enumerators, including the first examples with $\gamma = 8, 9$ in $W_{68,2}$ in the literature. This completes the list of possible γ values that can be found in $W_{68,2}$. Forty two of the codes we have constructed have $\gamma = 8$ in their weight enumerator, while forty of them have $\gamma = 9$ in their weight enumerators.

The rest of the work is organized as follows: In Section 2, we give the preliminaries about self-dual codes and the neighbor construction.In Section 3, we introduce the concept of distance and define the related generalization of the neighbors. In Section 4, we apply the generalized neighbors to construct extremal binary self-dual codes of length 68 with new weight enumerators. We finish the work with concluding remarks and directions for possible future research.

2. Preliminaries

2.1. Self-dual codes.

For $\overline{x} = (x_1, x_2, \dots, x_n)$ and $\overline{y} = (y_1, y_2, \dots, y_n) \in \mathbb{F}_2^n$, we define

$$\langle \overline{x}, \overline{y} \rangle = x_1y_1 + x_2y_2 + \cdots + x_ny_n.$$

This inner product leads to the following definition:

2010 *Mathematics Subject Classification.* Primary 94B05; Secondary 11T71.

Key words and phrases. extremal self-dual codes, neighbor, distance of self-dual codes, kth neighbor, weight enumerator.

Definition 2.1. Let C be a binary linear code over of length n, then we define the dual of C as

$$C^{\perp} := \{\overline{y} \in \mathbb{F}_2^n | \langle \overline{y}, \overline{x} \rangle = 0, \quad \forall \overline{x} \in C\}.$$

Note that, if C is a linear $[n,k]$ code, then $C^{\perp}$ is a linear $[n, n-k]$-code.

Definition 2.2. If $C \subseteq C^{\perp}$, then C is called *self-orthogonal* and it is called *self-dual* if $C = C^{\perp}$.

Definition 2.3. Let C be a self-dual binary code. If the Hamming weights of all the codewords in C are divisible by 4, C is called **Type II** (or doubly-even), otherwise it is called **Type I** (or singly even).

The following theorem gives an upper bound for minimum distance of self-dual codes:

Theorem 2.4. ([15], [5]) *Let $d_I(n)$ and $d_{II}(n)$ be the minimum distance of a Type I and Type II binary code of length n. then*

$$d_{II}(n) \leq 4\lfloor \frac{n}{24} \rfloor + 4$$

and

$$d_I(n) \leq \begin{cases} 4\lfloor \frac{n}{24} \rfloor + 4 & \text{if } n \not\equiv 22 \pmod{24} \\ 4\lfloor \frac{n}{24} \rfloor + 6 & \text{if } n \equiv 22 \pmod{24}. \end{cases}$$

Self-dual codes that attain the bounds given in the previous theorem are called *extremal*.

2.2. **The neighbor construction.** Two self-dual codes of length n are called *neighbors* if their intersection is a code of dimension $\frac{n}{2} - 1$. This idea has been used extensively in the literature to construct new self-dual codes from an existing one. For some of the works that have used this idea, we can refer to [3], [7], [10] and references therein.

Given a self-dual code C, a vector $x \in \mathbb{F}_2^n - C$ is picked and then D is formed by letting $D = \left\langle \langle x \rangle^{\perp} \cap C, x \right\rangle$. The search for D can be made efficient by using the standard form of the generator matrix of C, which allows one to fix the first $n/2$ entries of x without loss of generality. Usually in practical applications the first $n/2$ entries of x are set to be 0.

3. Distance between self-dual codes and generalized neighbors

To generalize the notion of a neighbor, we first begin with the following definition of a distance between two self-dual codes:

Definition 3.1. Let C_1 and C_2 be two binary self-dual codes of length n. The *neighbor-distance* between C_1 and C_2 is defined as

$$d_N(C_1, C_2) = \frac{n}{2} - dim(C_1 \cap C_2).$$

Remark 3.2. We would like to observe that this concept of the distance might have been considered in the literature in the context of a graph that can be formed by taking all the self-dual codes as the vertices. Then edges are drawn between the vertices if they are neighbors. What we have done in the paper is mostly equivalent to this concept but stated in algebraic terms.

Proposition 3.3. *The distance function d_N defined above is a metric on the set of all binary self-dual codes of length n.*

Proof. Since $dim(C_1 \cap C_2) \leq dim(C_1) = dim(C_2) = \frac{n}{2}$, we have $d_N(C_1, C_2) \geq 0$ for all self-dual codes C_1, C_2.

Next, observe that if $d_N(C_1, C_2) = 0$, this means

$$dim(C_1 \cap C_2) = \frac{n}{2} = dim(C_1) = dim(C_2),$$

which implies $C_1 = C_2 = C_1 \cap C_2$. Conversely, if $C_1 = C_2$, then $d_N(C_1, C_2) = \frac{n}{2} - \frac{n}{2} = 0$.

By the definition, it is clear that $d_N(C_1, C_2) = d_N(C_2, C_1)$.

For the triangle inequality, assume that C_1, C_2, C_3 are self-dual codes. Observe that

$$(C_1 \cap C_2) \cup (C_2 \cap C_3) = C_2 \cap (C_1 \cup C_3) \subseteq C_2.$$

which implies

$$dim(C_1 \cap C_2) + dim(C_2 \cap C_3) \leq dim(C_2) = \frac{n}{2}.$$

Thus we have

$$dim(C_1 \cap C_2) + dim(C_2 \cap C_3) - dim(C_1 \cap C_3) \leq \frac{n}{2}.$$

Adding $\frac{n}{2}$ to both sides and sending $dim(C_1 \cap C_2) + dim(C_2 \cap C_3)$ over to the right side of the equation, we get

$$\begin{aligned} d_N(C_1, C_3) &= \frac{n}{2} - dim(C_1 \cap C_3) \\ &\leq \frac{n}{2} - dim(C_1 \cap C_2) + \frac{n}{2} - dim(C_2 \cap C_3) \\ &\leq d_N(C_1, C_2) + d_N(C_2, C_3). \end{aligned}$$

□

The next proposition shows that self-dual codes cannot have the maximum distance to each other:

Proposition 3.4. *Let C_1 and C_2 be two binary self-dual codes of length n. Then $d_N(C_1, C_2) < \frac{n}{2}$.*

Proof. It is well known that if C is any binary self-dual code, then $(1, 1, \ldots, 1) \in C$. thus $\overline{1} \in C_1 \cap C_2$, which implies that $dim(C_1 \cap C_2) \geq 1$. But then

$$d_N(C_1, C_2) = \frac{n}{2} - dim(C_1 \cap C_2) \leq \frac{n}{2} - 1 < \frac{n}{2}.$$

□

Question: Is there an upper bound on the distance between equivalence classes of self-dual codes? The proposition shows that the distance cannot be larger than $\frac{n}{2} - 1$. It is an open question whether this upper bound can be reduced further.

We now define *k-range neighbor* and *k-neighbor* of a code using the distance notation:

Definition 3.5. Let C_1 and C_2 be two self-dual codes. C_1 and C_2 are said to be *k-range neighbors* if $d_N(C_1, C_2) \leq k$ and they are called *k-neighbors* if $d_N(C_1, C_2) = k$.

Remark 3.6. The *neighbor* of a self-dual code is well known in the literature and it corresponds to a 1-neighbor in our context.

Remark 3.7. The concept of a k-range neighbor code can be more useful than the strict k-neighbor codes, because of the following observation:

Suppose C_1 and C_2 are self-dual binary codes with generator matrices $[I_{n/2}|M_1]$ and $[I_{n/2}|M_2]$, respectively, where

$$M_1 = \begin{bmatrix} \overline{r}_1 \\ \hline \overline{r}_2 \\ \hline \overline{r}_3 \\ \hline \vdots \\ \hline \overline{r}_{n/2} \end{bmatrix}, \qquad M_2 = \begin{bmatrix} \overline{s}_1 \\ \hline \overline{s}_2 \\ \hline \overline{s}_3 \\ \hline \vdots \\ \hline \overline{s}_{n/2} \end{bmatrix}.$$

Here $\overline{r}_i$ and $\overline{s}_j$ are the rows of M_1 and M_2 respectively. If $\overline{r}_i = \overline{s}_i$ for $i = k+1, k+2, \ldots, n/2$, then C_1 and C_2 are k-range neighbors.

Remark 3.8. As we observed above, the ordinary neighbor of a code C is a 1-neighbor. We can also observe that, the neighbor a 1-neighbor of C is a 2-range neighbor of C. This can be generalized into considering the neighbor of a neighbor of a neighbor etc. of a code as a k-range neighbor of the original code.

4. Applications of k-range neighbor codes to extremal self-dual codes

In this section, we will give an equivalent description for the k-range neighbors and use them to construct new extremal binary self-dual codes. Let $\mathcal{N}_{(0)}$ be a binary self-dual code of length $2n$. Let $x_0 \in \mathbb{F}_2^{2n} \setminus \mathcal{N}_{(0)}$, define

$$\mathcal{N}_{(i+1)} = \left\langle \langle x_i \rangle^{\perp} \cap \mathcal{N}_{(i)}, x_i \right\rangle$$

where $\mathcal{N}_{(i+1)}$ is the neighbour of $\mathcal{N}_{(i)}$ and $x_i \in \mathbb{F}_2^{2n} \setminus s\mathcal{N}_{(i)}$.

It is not hard to see that $\mathcal{N}_{(i)}$ defined in this way is an i-range neighbor of $\mathcal{N}_{(0)}$ as was observed above in Remark 3.8. In what follows, we will apply this idea to search for extremal binary self-dual codes from k-range neighbors of a known code. We use Magma Algebra System ([2]) for our searches.

4.1. **Numerical results from i-range neighbours.** The possible weight enumerator of an extremal binary self-dual code of length 68 (of parameters $[68, 34, 12]$) is in one of the following forms by [4, 12, 6]:

$$\begin{aligned} W_{68,1} &= 1 + (442 + 4\beta)\, y^{12} + (10864 - 8\beta)\, y^{14} + \cdots, 104 \leq \beta \leq 1358, \\ W_{68,2} &= 1 + (442 + 4\beta)\, y^{12} + (14960 - 8\beta - 256\gamma)\, y^{14} + \cdots \end{aligned}$$

where $0 \leq \gamma \leq 9$. Recently, Yankov et al. constructed the first examples of codes with a weight enumerator for $\gamma = 7$ in $W_{68,2}$ in [1]. Together with these, the existence of codes in $W_{68,2}$ is known for many values. In order to save space we only give the lists for $\gamma = 5$, $\gamma = 6$ and $\gamma = 7$, which are updated in this work;

$$\begin{aligned} &\gamma = 5 \text{ with } \beta \in \{101,105,109,111,...,182,187,189,191,192,193,201,202,213\} \\ &\gamma = 6 \text{ with } \beta \in \{133, 137, 139, \ldots, 174, 176, 177, 184, 192, 210\} \\ &\gamma = 7 \text{ with } \beta \in \{7m | m = 14, \ldots, 39, 42\} \end{aligned}$$

Let $\mathcal{N}_{(0)}$ be the extremal binary self-dual code of length 68 ($W_{68,2}$) with the parameters $\gamma = 5$ and $\beta = 213$ which was recently constructed in [9]. Its generating matrix is given by $(I_{34}|A)$ where

$$A = \begin{pmatrix}
0&0&0&1&0&0&0&1&1&1&1&0&1&1&1&1&1&1&0&0&1&1&1&1&0&1&0&1&0&0&0&1&1&0 \\
0&1&1&0&0&0&1&0&0&1&1&1&1&0&1&0&1&0&0&0&0&1&1&0&1&0&0&0&1&1&0&0&1&0 \\
0&1&0&0&1&0&1&0&1&1&0&1&0&0&0&0&1&0&1&1&0&1&1&1&1&0&1&0&0&0&0&0&0&1 \\
1&0&0&0&0&1&0&1&0&0&0&1&1&1&1&1&1&0&1&1&0&1&0&0&0&1&1&0&0&0&1&1&1&0 \\
0&0&1&0&1&1&1&0&0&1&1&1&1&0&0&0&1&0&1&1&1&0&1&0&1&0&0&0&0&0&0&1&1&1 \\
0&0&0&1&1&1&0&1&1&0&1&1&0&1&0&0&0&1&1&1&0&1&0&1&0&1&0&0&0&0&1&0&1&1 \\
0&1&1&0&1&0&0&0&0&0&1&1&0&1&1&1&1&0&1&0&0&0&1&1&0&0&1&0&1&0&0&1&0&0 \\
1&0&0&1&0&1&0&0&1&1&0&0&0&1&0&0&1&0&0&1&1&1&0&0&0&0&1&0&0&1&0&1&0&0 \\
1&0&1&0&0&1&0&1&0&1&1&0&0&0&0&0&0&1&0&1&0&0&1&0&0&0&0&0&1&0&1&0&1&1 \\
1&1&1&0&1&0&0&1&1&1&1&1&1&0&1&0&0&0&1&0&0&1&1&1&1&1&1&0&1&0&1&1&1&1 \\
1&1&0&0&1&1&1&0&1&1&1&0&0&0&1&1&0&1&1&1&0&0&1&1&0&1&1&1&0&1&0&1&1&0 \\
0&1&1&1&1&1&1&0&0&1&0&0&0&1&1&0&1&1&1&1&1&0&1&0&0&1&1&0&0&1&1&1&0&1 \\
1&1&0&1&1&0&0&0&0&1&0&0&1&0&1&0&1&1&1&1&0&0&0&1&0&1&0&1&1&0&0&1&0&1 \\
1&0&0&1&0&1&1&1&0&0&0&1&0&0&0&0&1&0&1&1&1&0&1&1&0&1&1&1&1&0&1&1&1&0 \\
1&1&0&1&0&0&1&0&0&1&1&1&1&0&1&0&0&0&0&1&0&1&0&0&1&1&0&1&0&1&1&0&0&1 \\
1&0&0&1&0&0&1&0&0&0&1&0&0&0&0&0&0&1&1&0&0&0&0&1&0&0&1&1&0&1&0&0&1&0 \\
1&1&1&1&1&0&1&1&0&0&0&1&1&1&0&0&1&0&0&1&1&1&0&0&1&1&1&0&0&1&0&1&1&1 \\
1&0&0&0&0&1&0&0&1&0&1&1&1&0&0&1&0&0&1&0&0&1&0&1&0&0&0&0&0&1&1&1&1&1 \\
0&0&1&1&1&1&1&0&0&1&1&1&1&1&0&1&0&1&0&0&0&1&1&1&0&0&1&1&1&0&0&0&1&0 \\
0&0&1&1&1&1&0&1&1&0&1&1&1&1&1&0&1&0&0&0&1&0&1&1&0&0&1&1&0&1&0&0&0&1 \\
1&0&0&0&1&0&0&1&0&0&0&1&0&1&0&0&1&0&0&1&1&1&0&1&0&0&1&0&0&1&0&0&0&1 \\
1&1&1&1&0&1&0&1&0&1&0&0&0&0&1&0&1&1&1&0&1&0&0&0&1&1&1&1&0&1&1&0&1&0 \\
1&1&1&0&1&0&1&0&1&1&1&1&0&1&0&0&0&0&1&1&0&0&0&1&0&1&0&1&0&1&1&0&1&1 \\
1&0&1&0&0&1&1&0&0&1&1&0&1&1&0&1&0&1&1&1&1&0&1&1&0&1&1&1&0&1&0&0&1&1 \\
0&1&1&0&1&0&0&0&0&1&0&0&0&0&1&0&1&0&0&0&0&1&0&0&1&1&1&1&1&1&1&1&1&1 \\
1&0&0&1&0&1&0&0&0&1&1&1&1&1&1&0&1&0&0&0&0&1&1&1&1&1&0&0&1&1&0&0&1&1 \\
0&0&1&1&0&0&1&1&0&1&1&1&0&1&0&1&1&0&1&1&1&1&0&1&1&0&0&1&0&1&1&0&1&1 \\
1&0&0&0&0&0&0&0&0&0&1&0&1&1&1&1&0&0&1&1&0&1&1&1&1&0&1&1&0&1&0&0&1&1 \\
0&1&0&0&0&0&1&0&1&1&0&0&1&1&0&0&0&0&1&0&0&0&0&0&1&1&0&0&1&1&1&0&1&0 \\
0&1&0&0&0&0&0&1&0&0&1&1&0&0&1&1&1&1&0&1&1&1&1&1&1&1&1&1&1&1&1&0&0&1 \\
0&0&0&1&0&1&0&0&1&1&0&1&0&1&1&0&0&1&0&0&0&1&1&0&1&0&1&0&1&1&0&0&0&1 \\
1&0&0&1&1&0&1&1&0&1&1&1&1&1&0&0&1&1&0&0&0&0&0&0&1&0&0&0&0&0&0&1&1&0 \\
1&1&0&1&1&1&0&0&1&1&1&0&0&1&0&1&1&1&1&0&0&1&1&1&1&1&1&1&1&0&0&1&0&1 \\
0&0&1&0&1&1&0&0&1&1&0&1&1&0&1&0&1&1&0&1&1&0&1&1&1&1&1&1&0&1&1&0&1&0
\end{pmatrix}.$$

Implementing the formula described above to this code $\mathcal{N}_{(0)}$, we obtain:

Table 1. i-range neighbour of $\mathcal{N}_{(0)}$

i	$\mathcal{N}_{(i+1)}$	x_i	$\lvert Aut(\mathcal{N}_{(i+1)})\rvert$	γ	β
0	$\mathcal{N}_{(1)}$	(1100000101101111011001110100000100)	1	6	210
1	$\mathcal{N}_{(2)}$	(0111111111110110010100110111001100)	1	**7**	**212**
2	$\mathcal{N}_{(3)}$	(0111010010010101001000101110011001)	1	**8**	**221**
3	$\mathcal{N}_{(4)}$	(1000000111110011101001110001110000)	1	**9**	**221**

4.2. **Neighbours of Neighbours.** In this section, we separately consider neighbours of $\mathcal{N}_{(0)}$, $\mathcal{N}_{(1)}$, $\mathcal{N}_{(2)}$, $\mathcal{N}_{(3)}$ and $\mathcal{N}_{(4)}$.

Table 2. Neighbours of $\mathcal{N}_{(0)}$

$\mathcal{C}_i$	$(x_{35}, x_{36}, ..., x_{68})$	$\lvert Aut(\mathcal{N}_{68,i})\rvert$	γ	β
$\mathcal{C}_1$	(1001100000001010001010000111100011)	2	**5**	**195**
$\mathcal{C}_2$	(1000010001011010000011010000011010)	1	**5**	**198**
$\mathcal{C}_3$	(0111101000110110001011101100010000)	1	**5**	**200**
$\mathcal{C}_4$	(0111001101010010011001000101101010)	1	**5**	**202**
$\mathcal{C}_5$	(0100101101000111111110110101110111)	2	**5**	**211**
$\mathcal{C}_6$	(0011011100110001100010000000100100)	1	**6**	**198**
$\mathcal{C}_7$	(0111011111101001111101101111001000)	1	**6**	**204**

Table 3. Neighbours of $\mathcal{N}_{(1)}$

$\mathcal{C}_i$	$(x_{35}, x_{36}, ..., x_{68})$	γ	β	$\mathcal{C}_i$	$(x_{35}, x_{36}, ..., x_{68})$	γ	β
$\mathcal{C}_8$	(1001010111010111110011100111000011)	**6**	**175**	$\mathcal{C}_9$	(0001011110111110011101001111111100)	**6**	**177**
$\mathcal{C}_{10}$	(1011110110111010111010010111101111)	**6**	**179**	$\mathcal{C}_{11}$	(1011011001101100010101001010001111)	**6**	**181**
$\mathcal{C}_{12}$	(0111001000010101110001001100111100)	**6**	**182**	$\mathcal{C}_{13}$	(0111111111011111101100100100001110)	**6**	**183**
$\mathcal{C}_{14}$	(1011111001001110011110000010100011)	**6**	**185**	$\mathcal{C}_{15}$	(1010100011111100010011111101001101)	**6**	**186**
$\mathcal{C}_{16}$	(1011010111110011001011000100111011)	**6**	**187**	$\mathcal{C}_{17}$	(0011110011110111111101101100110100)	**6**	**188**
$\mathcal{C}_{18}$	(0000000010010101010011001010001011)	**6**	**189**	$\mathcal{C}_{19}$	(1011101110011101111110100101011100)	**6**	**190**
$\mathcal{C}_{20}$	(1111000000001011000111100111101010 1)	**6**	**191**	$\mathcal{C}_{21}$	(1010111010101011100011100011001111)	**6**	**193**
$\mathcal{C}_{22}$	(1000001101010100110101000011000101)	**6**	**194**	$\mathcal{C}_{23}$	(0101011110100101000000011010001111)	**6**	**195**
$\mathcal{C}_{24}$	(1111011101111110111011010001000011 1)	**6**	**196**	$\mathcal{C}_{25}$	(1111011010010110011101100001000110)	**6**	**197**
$\mathcal{C}_{26}$	(1011110001001000100001000110100000)	**6**	**199**	$\mathcal{C}_{27}$	(1001010010110100100000001100000101)	**6**	**200**
$\mathcal{C}_{28}$	(1010010010100011111100011100111010)	**6**	**201**	$\mathcal{C}_{29}$	(0100010011101100010110001010110000)	**6**	**202**
$\mathcal{C}_{30}$	(0100100000001111000000100110000101 10)	**6**	**206**	$\mathcal{C}_{31}$	(0001110111010001111011010001011111)	**6**	**207**
$\mathcal{C}_{32}$	(0010110010011001111110101000011110)	**7**	**184**	$\mathcal{C}_{33}$	(0011001100100010111110000000011001)	**7**	**185**

Table 4. Neighbours of $\mathcal{N}_{(2)}$

$\mathcal{C}_i$	$(x_{35}, x_{36}, ..., x_{68})$	γ	β	$\mathcal{C}_i$	$(x_{35}, x_{36}, ..., x_{68})$	γ	β
$\mathcal{C}_{34}$	(1101111111101111011110011101101110)	**7**	**174**	$\mathcal{C}_{35}$	(1001101001000111010110110111111111)	**7**	**177**
$\mathcal{C}_{36}$	(1011010010001010111100010010000100)	**7**	**178**	$\mathcal{C}_{37}$	(1101111001010110110111010110001111)	**7**	**179**
$\mathcal{C}_{38}$	(1011101100100101100110111101101111)	**7**	**181**	$\mathcal{C}_{39}$	(1001111111011111110110010001001110)	**7**	**183**
$\mathcal{C}_{40}$	(1101010001010110001001111001100010)	**7**	**186**	$\mathcal{C}_{41}$	(0110111111000000011011000001110001)	**7**	**187**
$\mathcal{C}_{42}$	(0010011000010000000011111111111010)	**7**	**188**	$\mathcal{C}_{42}$	(1001101101101011111111010100101101)	**7**	**190**
$\mathcal{C}_{43}$	(0110000000101101001100111111010001 0)	**7**	**191**	$\mathcal{C}_{44}$	(1011000011111010100011111011100011)	**7**	**192**
$\mathcal{C}_{45}$	(0100001110010111010110101010011110)	**7**	**193**	$\mathcal{C}_{46}$	(0000001101111111010000110000100000 0)	**7**	**194**
$\mathcal{C}_{47}$	(0100110110001011101001011000110001)	**7**	**195**	$\mathcal{C}_{48}$	(1111010100000001011110010010111010 1)	**7**	**197**
$\mathcal{C}_{49}$	(1100100000000100110011001011111111 1)	**7**	**198**	$\mathcal{C}_{50}$	(0000110011100001111010110100110001)	**7**	**199**

Table 5. Neighbours of $\mathcal{N}_{(3)}$

$\mathcal{C}_i$	$(x_{35}, x_{36}, ..., x_{68})$	γ	β	$\mathcal{C}_i$	$(x_{35}, x_{36}, ..., x_{68})$	γ	β
$\mathcal{C}_{51}$	(0101000010100010000111100101011100)	**7**	**171**	$\mathcal{C}_{52}$	(0000011101001011001010111100001110)	**7**	**173**
$\mathcal{C}_{53}$	(0000011101101010010000110000101000)	**7**	**176**	$\mathcal{C}_{54}$	(1011010010100001010000111011100110)	**7**	**180**
$\mathcal{C}_{55}$	(1011110011111111111100111110100111)	**8**	**181**	$\mathcal{C}_{56}$	(1101111111010111111111100010110111)	**8**	**186**
$\mathcal{C}_{57}$	(0000110100100011011001001101111010)	**8**	**187**	$\mathcal{C}_{58}$	(1101000111000011001010010001000000)	**8**	**189**
$\mathcal{C}_{59}$	(0000001011100100110100101111000100)	**8**	**190**	$\mathcal{C}_{60}$	(0111011011011011010010101101100011)	**8**	**192**
$\mathcal{C}_{61}$	(0011011011110001000111111111011110)	**8**	**191**	$\mathcal{C}_{62}$	(1010111011000111100110111001110111)	**8**	**193**
$\mathcal{C}_{63}$	(0010010011101001110101011010111100)	**8**	**194**	$\mathcal{C}_{64}$	(0001001001010000111101111001110111)	**8**	**195**
$\mathcal{C}_{65}$	(0001101011101101100010111110110011)	**8**	**196**	$\mathcal{C}_{66}$	(0100100001010000001010001111100010)	**8**	**197**
$\mathcal{C}_{67}$	(1011100101000101100011110111101011)	**8**	**198**	$\mathcal{C}_{68}$	(0011011111101011011011111011110011)	**8**	**199**
$\mathcal{C}_{69}$	(1111000101100111100010101010000001)	**8**	**200**	$\mathcal{C}_{70}$	(0011000110010100110010000110000001)	**8**	**201**
$\mathcal{C}_{71}$	(1110101001000010010100101000011100)	**8**	**202**	$\mathcal{C}_{72}$	(0110111010011110110001011011101001)	**8**	**203**
$\mathcal{C}_{73}$	(1001100101111110111101011001101110)	**8**	**204**	$\mathcal{C}_{74}$	(0000100111111101000010110011001001)	**8**	**205**
$\mathcal{C}_{75}$	(1011001000010010011100101011000100)	**8**	**206**	$\mathcal{C}_{76}$	(0101111110001111110000111111111011)	**8**	**207**
$\mathcal{C}_{77}$	(0110101010100001110101011010110110)	**8**	**208**	$\mathcal{C}_{78}$	(0001111000110101011111001111101111)	**8**	**209**
$\mathcal{C}_{79}$	(1111011101111110000011100111111011)	**8**	**210**	$\mathcal{C}_{80}$	(1101010100000100000001110100010001)	**8**	**211**
$\mathcal{C}_{81}$	(0011110101110001000001111001110000)	**8**	**212**	$\mathcal{C}_{82}$	(1100011110110111110101000101011111)	**8**	**213**
$\mathcal{C}_{83}$	(0101011001011011111001010100001000)	**8**	**214**	$\mathcal{C}_{84}$	(0000011110101100110001010101100011)	**8**	**215**
$\mathcal{C}_{85}$	(1110100101011111001101011011011110)	**8**	**216**	$\mathcal{C}_{86}$	(0001111000001111100010100011011010)	**8**	**217**
$\mathcal{C}_{87}$	(1110011000000001010010100010101010110)	**8**	**218**	$\mathcal{C}_{88}$	(0100011111001011000000000010000011)	**8**	**220**

Table 6. Neighbours of $\mathcal{N}_{(4)}$

$\mathcal{C}_i$	$(x_{35}, x_{36}, ..., x_{68})$	γ	β	$\mathcal{C}_i$	$(x_{35}, x_{36}, ..., x_{68})$	γ	β
$\mathcal{C}_{89}$	(0110011110100110101110111111001110)	**7**	**163**	$\mathcal{C}_{90}$	(1010011101011110101011111111011110)	**7**	**166**
$\mathcal{C}_{91}$	(1001110111100010010000100001111010)	**7**	**169**	$\mathcal{C}_{92}$	(0011101011111100101001110010011011)	**7**	**170**
$\mathcal{C}_{93}$	(1101101101010111100010000101001101)	**7**	**172**	$\mathcal{C}_{94}$	(1001001100010110100011110011101101)	**8**	**180**
$\mathcal{C}_{95}$	(0110000111110011101010000111110111)	**8**	**182**	$\mathcal{C}_{96}$	(1110000100101011001100000100001101)	**8**	**183**
$\mathcal{C}_{97}$	(1110111101111011001110111111010111)	**8**	**184**	$\mathcal{C}_{98}$	(0111001000001101101001110011010010)	**8**	**185**
$\mathcal{C}_{99}$	(0010010100101110111101101011111111)	**8**	**219**	$\mathcal{C}_{100}$	(1101000000110100011000000110101000)	**8**	**188**
$\mathcal{C}_{101}$	(0111101011001101101011010011001011)	**9**	**186**	$\mathcal{C}_{102}$	(0001001011000000110111110000010110)	**9**	**187**
$\mathcal{C}_{103}$	(1011011001100001001110011100101101)	**9**	**188**	$\mathcal{C}_{104}$	(1000000111000110111000100010000001)	**9**	**189**
$\mathcal{C}_{105}$	(1111010100111110101110110000011111)	**9**	**190**	$\mathcal{C}_{106}$	(1000100000000111110001110000010010)	**9**	**192**
$\mathcal{C}_{107}$	(0111011010010110011110110001000110)	**9**	**193**	$\mathcal{C}_{108}$	(0010011100001000000010001111011000)	**9**	**194**
$\mathcal{C}_{109}$	(0001101100111010100010011110101000)	**9**	**195**	$\mathcal{C}_{110}$	(1000011011111111111010001110010001)	**9**	**196**
$\mathcal{C}_{111}$	(0111010111111001111101011000101110)	**9**	**198**	$\mathcal{C}_{112}$	(0101101101001100001001110011010010)	**9**	**199**
$\mathcal{C}_{113}$	(1111011011111111111010100100111001)	**9**	**200**	$\mathcal{C}_{114}$	(0011111100000101110110110111011111)	**9**	**201**
$\mathcal{C}_{115}$	(0100111111101001101001110001101011)	**9**	**202**	$\mathcal{C}_{116}$	(1111101111000110001100111111101100)	**9**	**203**
$\mathcal{C}_{117}$	(0101111110001110100001110110011011)	**9**	**204**	$\mathcal{C}_{118}$	(0111110111111110111101000001110100)	**9**	**205**
$\mathcal{C}_{119}$	(1110110111101011000110100111111100)	**9**	**206**	$\mathcal{C}_{120}$	(0000000111010010100010010001011001)	**9**	**207**
$\mathcal{C}_{121}$	(0001001101110101011111001000101101)	**9**	**208**	$\mathcal{C}_{122}$	(0100001111001011001010000111010011)	**9**	**209**
$\mathcal{C}_{123}$	(0101000110111111010111000111000100)	**9**	**210**	$\mathcal{C}_{124}$	(0110110111011011011110111101001100)	**9**	**211**
$\mathcal{C}_{125}$	(0001110001110001001001110010111010)	**9**	**213**	$\mathcal{C}_{126}$	(0001010100001110010110011101111101)	**9**	**214**
$\mathcal{C}_{127}$	(0101010110001011110111000001101110)	**9**	**215**	$\mathcal{C}_{128}$	(0010011111011010100011110101011011)	**9**	**216**
$\mathcal{C}_{129}$	(0111100100111001111101100111110101)	**9**	**217**	$\mathcal{C}_{130}$	(1101110110110011011001111111011011)	**9**	**218**
$\mathcal{C}_{131}$	(1101011010110011000111101000101100)	**9**	**219**	$\mathcal{C}_{132}$	(0110101110111110101011011111101011)	**9**	**220**
$\mathcal{C}_{133}$	(1110000011001101000110000000101110)	**9**	**222**	$\mathcal{C}_{134}$	(1001111010110000000101110100000100)	**9**	**223**
$\mathcal{C}_{135}$	(1001000111100111010011111100111001)	**9**	**224**	$\mathcal{C}_{136}$	(1011111011110111101111011111011100)	**9**	**225**
$\mathcal{C}_{137}$	(0011111100110101110101101110110101)	**9**	**226**	$\mathcal{C}_{138}$	(1011010011100011110000011000001011)	**9**	**228**
$\mathcal{C}_{139}$	(0101001011001111001010011001000011)	**9**	**230**				

5. Conclusion

We introduced the concept of a distance between self-dual codes. This generalizes the notion of a neighbor in self-dual codes, which leads to a new way of constructing new self-dual codes from a known one. Applying these ideas to an extremal binary self-dual code of length 68 we were able to construct 143 new extremal binary self-dual codes of length 68 with new weight enumerators, including the first examples with $\gamma = 8, 9$ in $W_{68,2}$ in the literature. Thus, we have now completed the theoretical list of possible γ values that can be found in $W_{68,2}$. Generator matrices for some of the new codes are available online at [11]. Out of the codes we have constructed, 42 have $\gamma = 8$ in their weight enumerator, while 40 of them have $\gamma = 9$ in their weight enumerators. In particular, we have been able to construct the codes that have the following parameters:

$$
\begin{aligned}
&(\gamma = 5, \quad \beta = \{195, 198, 200, 202, 211\}),\\
&(\gamma = 6, \quad \beta = \{175, 177, 179, 181, 182, 183, 185, 186, 187, 188, 189, 190, 191, 193, 194, 195, 196, 197, 198, 199,\\
&\qquad\qquad 200, 201, 202, 204, 206, 207\}),\\
&(\gamma = 7, \quad \beta = \{163, 166, 169, 170, 171, 172, 173, 174, 176, 177, 178, 179, 180, 181, 183, 184, 185, 186, 187, 188,\\
&\qquad\qquad 190, 191, 192, 193, 194, 195, 197, 198, 199, 212\}),\\
&(\gamma = 8, \quad \beta = \{180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 195, 196, 197, 198, 199,\\
&\qquad\qquad 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 218, 219,\\
&\qquad\qquad 220, 221\}),\\
&(\gamma = 9, \quad \beta = \{186, 187, 188, 189, 190, 192, 193, 194, 195, 196, 198, 199, 200, 201, 202, 203, 204, 205, 206, 207,\\
&\qquad\qquad 208, 209, 210, 211, 213, 214, 215, 216, 217, 218, 219, 220, 221, 222, 223, 224, 225, 226, 228, 230\})
\end{aligned}
$$

The strength of this new approach has been demonstrated by the number of new weight enumerators that we have been able to obtain by applying it to a single code. We believe this will open up new venues in the search and classification of new extremal binary self-dual codes.

References

[1] D. Anev, M. Harada and N. Yankov, "New extremal binary self-dual codes of lengths 64 and 66", *J. Alg. Comb. Disc. Structures and Appl.*, vol. 5, no.3, pp. 143–151, 2018.

[2] W. Bosma, J. Cannon, and C. Playoust, "The Magma algebra system. I. The user language", *J. Symbolic Comput.*, Vol. 24, pp. 235–265, 1997.

[3] R.A. Brualdi and V.S. Pless, "Weight Enumerators of Self-Dual Codes", *IEEE Trans. Infrom. Theory*, vol. 37, no. 4, pp. 1222-1225, 1991.

[4] S. Buyuklıeva, I. Boukliev, "Extremal self-dual codes with an automorphism of order 2", *IEEE Trans. Inform. Theory*, Vol. 44, pp. 323–328, 1998.

[5] J.H. Conway, N.J.A. Sloane, "A new upper bound on the minimal distance of self-dual codes", *IEEE Trans. Inform. Theory*, Vol. 36, 6, 1319–1333, 1990.

[6] S.T. Dougherty, T.A. Gulliver, M. Harada,"Extremal binary self dual codes", *IEEE Trans. Inform. Theory*, Vol. 43 pp. 2036-2047, 1997.

[7] J. Gildea, H. Hamilton, A. Kaya and B. Yildiz, "Modified quadratic residue constructions and new extremal binary self-dual codes of lengths 64, 66 and 68", *Inf. Proces. Letters*, vol. 157, 2020.

[8] J. Gildea., A. Kaya, R. Taylor, and B. Yildiz, "Constructions for self-dual codes induced from group rings", *Finite Fields Appl.*, vol. 51, pp. 71–92, 2018.

[9] J. Gildea, A. Korban and A. Kaya, "Self-dual codes using bisymmetric designs and group rings", **submitted**.

[10] J. Gildea, A. Kaya and B. Yildiz, "An altered four circulant construction for self-dual codes from group rings and new extremal binary self-dual codes I", *Discrete Mathematics.*, vol. 342, no. 12, 111620, 2019.

[11] J. Gildea, A. Kaya, A. Korban, B. Yildiz, "Binary generator matrices for some new extremal binary self-dual codes of length 68", available online at `http://abidinkaya.wixsite.com/math/neighbor`.

[12] M. Harada, A. Munemasa, "Some restrictions on weight enumerators of singly even self-dual codes", *IEEE Trans. Inform. Theory*, Vol. 52, pp. 1266–1269, 2006.

[13] A. Kaya and B. Yildiz "Various constructions for self-dual codes over rings and new binary self-dual codes", *Discrete Math*, Vol. 339, No. 2 pp. 460–469, 2016.

[14] A. Kaya, B. Yildiz, A. Paşa "New extremal binary self-dual codes from a modified four circulant construction", *Discrete Math*, Vol. 339, No. 3 pp. 1086–1094, 2016.

[15] E.M. Rains, "Shadow Bounds for Self Dual Codes", *IEEE Trans. Inf. Theory*, Vol.44, pp.134–139, 1998.

Department of Mathematical and Physical Sciences, Faculty of Science and Engineering, University of Chester, England, UK

Email address: `j.gildea@chester.ac.uk`

Department of Mathematics Education, Sampoerna University, 12780, Jakarta, Indonesia
Email address: abidin.kaya@sampoernauniversity.ac.id

Department of Mathematical and Physical Sciences, Faculty of Science and Engineering, University of Chester, England, UK
Email address: adrian3@windowslive.com

Department of Mathematics & Statistics, Northern Arizona University, Flagstaff, AZ 86001, USA
Email address: bahattinyildiz@gmail.com