

Construction of extremal Type II \mathbb{Z}_8 -codes via doubling method

Sara Ban (sban@math.uniri.hr), ORCID: 0000-0002-1837-8701

Sanja Rukavina (sanjar@math.uniri.hr), ORCID: 0000-0003-3365-7925

*Faculty of Mathematics, University of Rijeka
Radmile Matejčić 2, 51000 Rijeka, Croatia*

Abstract

Extremal Type II \mathbb{Z}_8 -codes are a class of self-dual \mathbb{Z}_8 -codes with Euclidean weights divisible by 16 and the largest possible minimum Euclidean weight for a given length. We introduce a doubling method for constructing a Type II \mathbb{Z}_{2k} -code of length n from a known Type II \mathbb{Z}_{2k} -code of length n . Based on this method, we develop an algorithm to construct new extremal Type II \mathbb{Z}_8 -codes starting from an extremal Type II \mathbb{Z}_8 -code of type $(\frac{n}{2}, 0, 0)$ with an extremal \mathbb{Z}_4 -residue code and length 24, 32 or 40.

We construct at least ten new extremal Type II \mathbb{Z}_8 -codes of length 32 and type $(15, 1, 1)$. Extremal Type II \mathbb{Z}_8 -codes of length 32 of this type were not known before. Moreover, the binary residue codes of the constructed extremal \mathbb{Z}_8 -codes are optimal $[32, 15]$ binary codes.

Keywords: Type II \mathbb{Z}_{2k} -code, extremal \mathbb{Z}_8 -code, residue code, doubling method

Mathematics Subject Classification: 94B05

1 Introduction

The discovery of good nonlinear binary codes arising via the Gray map from \mathbb{Z}_4 -linear codes motivated the study of codes over rings in general (see [14]). Construction of unimodular lattices with large minimum norm has motivated the construction of new self-dual \mathbb{Z}_{2k} -codes with large minimum Euclidean weights (see, for example, [4, 10]). Especially, \mathbb{Z}_8 -codes have received attention by many researchers. For instance, some construction methods for self-dual codes over \mathbb{Z}_8 for arbitrary length greater than 8 are given in [1].

Extremal Type II \mathbb{Z}_8 -codes are a class of self-dual \mathbb{Z}_8 -codes with Euclidean weights divisible by 16 and the largest possible minimum Euclidean weight for a given length. For

lengths 8 and 16, every Type II \mathbb{Z}_8 -code is extremal (see [15]). In [13], the previously known results on the existence of extremal Type II \mathbb{Z}_8 -codes for greater lengths are summarized: there are three such codes of length 24 (see [4, 11]), five codes of length 32, a large number of codes of length 40 and one such code of length 48 (see [8, 9]), up to equivalence. In addition, a new extremal Type II \mathbb{Z}_8 -code D_n is constructed for each of the lengths $n \in \{24, 32, 40\}$ (see [[13], Subsection 5.2]).

The doubling method for a construction of Type II \mathbb{Z}_4 -codes is introduced in [7]. It was used for the construction of extremal Type II \mathbb{Z}_4 -codes of length 32 and 40 in [2] and [3], respectively. In this paper, we introduce a doubling method for constructing a Type II \mathbb{Z}_{2^k} -code of length n from a known Type II \mathbb{Z}_{2^k} -code of length n . We also develop an algorithm that uses this method to construct a Type II \mathbb{Z}_{2^m} -code of length n from a known Type II \mathbb{Z}_{2^m} -code of length n . Finally, by specifying the method for $m = 3$ and $n \in \{24, 32, 40\}$, we obtain an algorithm that is then used to construct at least 10 new extremal Type II \mathbb{Z}_8 -codes of length 32.

The paper is organized as follows. The next section gives definitions and basic properties of codes over \mathbb{Z}_{2^k} that will be needed in our work. In Section 3, we introduce the doubling method to construct new Type II \mathbb{Z}_{2^k} -codes starting from a Type II \mathbb{Z}_{2^k} -code. Especially, we consider the codes over \mathbb{Z}_{2^m} . Finally, in the last section, we present a method to construct new extremal Type II \mathbb{Z}_8 -codes starting from an extremal Type II \mathbb{Z}_8 -code of type $(\frac{n}{2}, 0, 0)$ with an extremal \mathbb{Z}_4 -residue code and length 24, 32 or 40. Using this method, we construct 68850 extremal Type II \mathbb{Z}_8 -codes of type $(15, 1, 1)$ and length 32. We give the weight distributions of the corresponding binary residue codes. With respect to these weight distributions, all constructed extremal Type II \mathbb{Z}_8 -codes are divided into ten classes. For each of them we give a generator matrix in the standard form for one class representative.

2 Preliminaries

For terms not defined in this paper and the basic facts of coding theory we refer the reader to [5, 16, 18].

Let \mathbb{Z}_{2^k} denote the ring of integers modulo 2^k . A linear code C of length n over \mathbb{Z}_{2^k} (i.e., a \mathbb{Z}_{2^k} -code) is an additive subgroup of $\mathbb{Z}_{2^k}^n$. Two codes over \mathbb{Z}_{2^k} are *equivalent* if one can be obtained from the other by permuting the coordinates and (if necessary) changing the signs of certain coordinates. Codes differing by only a permutation of coordinates are called *permutation-equivalent*. An element of C is called a *codeword* of C . A *generator matrix* of C is a matrix whose rows generate C .

The *dual code* C^\perp of C is defined as

$$C^\perp = \{x \in \mathbb{Z}_{2^k}^n \mid \langle x, y \rangle = 0 \text{ for all } y \in C\},$$

where $\langle x, y \rangle = x_1y_1 + x_2y_2 + \cdots + x_ny_n \pmod{2^k}$ for $x = (x_1, x_2, \dots, x_n)$ and $y = (y_1, y_2, \dots, y_n)$. The code C is *self-orthogonal* when $C \subseteq C^\perp$ and *self-dual* if $C = C^\perp$.

The *Euclidean weight* of a codeword $x = (x_1, x_2, \dots, x_n) \in \mathbb{Z}_{2k}^n$ is

$$wt_E(x) = \sum_{i=1}^n \min\{x_i^2, (2k - x_i)^2\}.$$

It holds

$$wt_E(x + y) \equiv wt_E(x) + wt_E(y) + 2 \langle x, y \rangle \pmod{4k} \quad (1)$$

for all $x, y \in \mathbb{Z}_{2k}^n$ (see [4]). We denote the number of coordinates i (where $i = 0, 1, \dots, 2k-1$) in a codeword $x \in \mathbb{Z}_{2k}^n$ by $n_i(x)$.

The minimum Euclidean weight d_E of C is the smallest Euclidean weight among all nonzero codewords of C . A self-dual \mathbb{Z}_{2k} -code is called *Type II* if it has the property that every Euclidean weight is divisible by $4k$. Type II \mathbb{Z}_{2k} -codes are a remarkable class of self-dual codes related to even unimodular lattices. There is a Type II \mathbb{Z}_{2k} -code of length n if and only if n is divisible by eight [4]. For Type II \mathbb{Z}_{2k} -codes C of length n , the upper bound on the minimum Euclidean weight

$$d_E(C) \leq 4k \left\lfloor \frac{n}{24} \right\rfloor + 4k \quad (2)$$

holds for $k = 1$ and 2 , and for $k \geq 3$ it holds under the assumption that $\lfloor \frac{n}{24} \rfloor \leq k - 2$ (see [4]). We say that a Type II \mathbb{Z}_{2k} -code meeting (2) with equality is *extremal*.

If C is a \mathbb{Z}_{2^m} -code, then the code $C^{(2^k)} = \{x \pmod{2^k} \mid x \in C\}$, $1 \leq k \leq m - 1$, is the \mathbb{Z}_{2^k} -*residue code* of C . Each code C over \mathbb{Z}_{2^m} is permutation-equivalent to a code with a generator matrix in *standard form*

$$\begin{pmatrix} I_{k_1} & A_{1,2} & A_{1,3} & A_{1,4} & \cdots & \cdots & A_{1,m+1} \\ 0 & 2I_{k_2} & 2A_{2,3} & 2A_{2,4} & \cdots & \cdots & 2A_{2,m+1} \\ 0 & 0 & 4I_{k_3} & 4A_{3,4} & \cdots & \cdots & 4A_{3,m+1} \\ \vdots & \vdots & \vdots & \ddots & \ddots & & \vdots \\ \vdots & \vdots & \vdots & \ddots & \ddots & & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 2^{m-1}I_{k_m} & 2^{m-1}A_{m,m+1} \end{pmatrix},$$

where the matrix $A_{i,j}$ has elements in $\mathbb{Z}_{2^{j-1}}$. We say that C is of *type* $(k_1, k_2, k_3, \dots, k_m)$. The code C has $\prod_{j=1}^m (2^{m-j+1})^{k_j}$ codewords.

In this work, we have used computer algebra systems GAP [17] and Magma [6].

3 Method of construction

In [7], the doubling method for a construction of Type II \mathbb{Z}_4 -codes is introduced. In the next theorem, we generalize results from [7] and give the doubling method for a construction of Type II \mathbb{Z}_{2k} -codes.

Theorem 3.1. *Let $k \geq 2$. Let C be a Type II \mathbb{Z}_{2k} -code of length n and let $n_i(x)$ denote the number of coordinates i in $x \in \mathbb{Z}_{2k}^n$. Let $ku \in \mathbb{Z}_{2k}^n \setminus C$ be a codeword with all coordinates equal to 0 or k with the following property: if k is odd, $n_k(ku)$ is divisible by four, if k is even and not divisible by four, $n_k(ku)$ is even. Let $C_0 = \{v \in C \mid \langle ku, v \rangle = 0\}$. Then $\tilde{C} = C_0 \oplus \langle ku \rangle$ is a Type II \mathbb{Z}_{2k} -code.*

Proof. The Euclidean weight $wt_E(ku) = n_k(ku) \cdot k^2$ is divisible by $4k$ and $\langle ku, ku \rangle = 0$. It follows from (1) that \tilde{C} is self-orthogonal with all Euclidean weights divisible by $4k$.

The codeword $ku \notin C$, so there is a codeword $w \in C$ such that $\langle ku, w \rangle = k$. Suppose $\tilde{w} \in C \setminus C_0$. Then $\langle ku, \tilde{w} \rangle = k$. Therefore, $\tilde{w} \in C_0 + w$ and C_0 and $C_0 + w$ are the only cosets of C_0 in C . Since $|\tilde{C}| = |C_0| \cdot 2 = |C|$, \tilde{C} is a Type II \mathbb{Z}_{2k} -code. \square

When considering Type II \mathbb{Z}_{2^m} -codes, we can restrict the possible choices for $ku = 2^{m-1}u \in \mathbb{Z}_{2^m}^n \setminus C$.

Theorem 3.2. *Let $m \geq 2$. Let C be a Type II \mathbb{Z}_{2^m} -code of length n and type (k_1, k_2, \dots, k_m) . The choice of $2^{m-1}u \in \mathbb{Z}_{2^m}^n \setminus C$ in Theorem 3.1 can be limited to codewords with zeroes on the first $k_1 + k_2 + \dots + k_m$ coordinates.*

Proof. For every $2^{m-1}u \in \mathbb{Z}_{2^m}^n \setminus C$ satisfying the conditions of Theorem 3.1, there exists a unique codeword $2^{m-1}v \in C$ with all coordinates equal to 0 or 2^{m-1} such that $2^{m-1}u$ coincides with $2^{m-1}v$ on the first $k_1 + k_2 + \dots + k_m$ entries. Then $C_0 \oplus \langle 2^{m-1}u \rangle = C_0 \oplus \langle 2^{m-1}u - 2^{m-1}v \rangle$. \square

Now, we generalize the statement of [[3], Theorem 5] to Type II \mathbb{Z}_{2^m} -codes.

Theorem 3.3. *Let $m \geq 2$. Let C be a Type II \mathbb{Z}_{2^m} -code of length n and type (k_1, k_2, \dots, k_m) . Let G be a generator matrix of C in standard form and G_i the i^{th} row of G . Let $2^{m-1}u \in \mathbb{Z}_{2^m}^n \setminus C$ be a codeword with zeroes on the first $k_1 + k_2 + \dots + k_m$ coordinate positions such that $n_{2^{m-1}}(2^{m-1}u)$ is even if $m = 2$. Let $B = \{G_1, \dots, G_{k_1+k_2+\dots+k_m}\}$. The following process yields a generator matrix \tilde{G} of the \mathbb{Z}_{2^m} -code \tilde{C} obtained from C and $2^{m-1}u$ by the doubling method.*

Step 1: Let $B_E = \{G_i \in B \mid \langle G_i, 2^{m-1}u \rangle = 0\}$ and $B_O = B \setminus B_E$.

Step 2: Pick $G_i \in B_O$ arbitrarily. Define $B'_O = \{G_i + G_j \mid G_j \in B_O\}$.

Step 3: Let \tilde{G} be a matrix whose rows are the elements of the set $B'_O \cup B_E \cup \{2^{m-1}u\}$.

The resultant code \tilde{C} is of type

$$\begin{cases} (k_1 - 1, k_2 + 2), & \text{if } m = 2, \\ (k_1 - 1, k_2 + 1, k_3 + 1), & \text{if } m = 3, \\ (k_1 - 1, k_2 + 1, k_3, \dots, k_{m-1}, k_m + 1), & \text{if } m \geq 4. \end{cases}$$

The code \tilde{C} is independent of the choice of G_i in Step 2.

Proof. The set B_O is not empty because C is self-dual and $2^{m-1}u \notin C$. Further, for all $t = k_1 + 1, \dots, k_1 + k_2 + \dots + k_m$ it follows $G_t \in B_E$. For $G_i, G_j \in B_O$, we have

$$\langle G_i + G_j, 2^{m-1}u \rangle = \langle G_i, 2^{m-1}u \rangle + \langle G_j, 2^{m-1}u \rangle = 0,$$

and $G_i + G_j$ is an codeword with all even coordinates if and only if $i = j$.

Note that $\langle B'_O \cup B_E \rangle = \{v \in C \mid \langle 2^{m-1}u, v \rangle = 0\}$. It follows that \tilde{C} is of type

$$\begin{cases} (k_1 - 1, k_2 + 2), & \text{if } m = 2, \\ (k_1 - 1, k_2 + 1, k_3 + 1), & \text{if } m = 3, \\ (k_1 - 1, k_2 + 1, k_3, \dots, k_{m-1}, k_m + 1), & \text{if } m \geq 4. \end{cases}$$

The independence follows from the fact that

$$G_k + G_j = (G_i + G_k) + (G_i + G_j) + (2^{m-1} - 1)(G_i + G_i).$$

□

4 Construction of extremal Type II \mathbb{Z}_8 -codes

Here we consider an extremal Type II \mathbb{Z}_8 -code C of length $n \in \{24, 32, 40\}$ and type $(\frac{n}{2}, 0, 0)$ which has an extremal residue code $C^{(4)}$. Using the doubling method given in the previous chapter, we developed an algorithm for a construction of extremal Type II \mathbb{Z}_8 -codes \tilde{C} of length n and type $(\frac{n}{2} - 1, 1, 1)$.

Note that $wt_E(x) = n_1(x) + n_7(x) + 4(n_2(x) + n_6(x)) + 9(n_3(x) + n_5(x)) + 16n_4(x)$, for $x \in \mathbb{Z}_8^n$.

Theorem 4.1. *Let $n \in \{24, 32, 40\}$. Denote by $S_i(w)$ the set of positions with the element $i \in \mathbb{Z}_8$ in $w \in \mathbb{Z}_8^n$. Let C be an extremal Type II \mathbb{Z}_8 -code of length n and type (k_1, k_2, k_3) where $C^{(4)}$ is extremal. Suppose $4u \in \mathbb{Z}_8^n$ is a codeword with all coordinates equal to 0 or 4 such that $S_4(4u) \subseteq \{k_1 + k_2 + k_3 + 1, \dots, n\}$, where $|S_4(4u)| \geq 2$. If there is no codeword v of C that satisfies any of the following conditions:*

1. $S_3(v) \cup S_4(v) \cup S_5(v) \subseteq S_4(4u) \subseteq S_2(v) \cup S_3(v) \cup S_4(v) \cup S_5(v) \cup S_6(v)$ and

$$wt_E(v \pmod{4}) = 16,$$

2. $|S_4(4u) \setminus S_4(v)| + |S_4(v) \setminus S_4(4u)| = 1$ and $wt_E(v \pmod{4}) = 0$,

then the Type II \mathbb{Z}_8 -code \tilde{C} generated by $4u$ and C using the doubling method is extremal. These choices of $4u$ are the only candidates for the code C in the doubling method which lead to an extremal code.

Proof. It follows from Theorem 3.2 that the choices of $4u$ can be limited to codewords with zeroes on the first $k_1 + k_2 + k_3$ coordinates. Let us assume that the code \tilde{C} is not extremal. Then it contains a codeword of Euclidean weight 16 of the form $w = v + 4u$, where $v \in C$ is such a codeword that $\langle v, 4u \rangle = 0$, and

$$wt_E(w) = W + 8(n_3(w) + n_5(w)) + 16n_4(w),$$

where $W = n_1(w) + n_7(w) + n_3(w) + n_5(w) + 4(n_2(w) + n_6(w))$. It holds $W = 0$ or $W \geq 16$, since $C^{(4)}$ is extremal. There are three cases to consider.

Case 1: $W > 16$.

Then $wt_E(w) \geq 32$.

Case 2: $W = 16$.

For $wt_E(w)$ to be equal to 16, $n_3(w) = n_4(w) = n_5(w) = 0$, which is impossible because of the first condition.

Case 3: $W = 0$.

This condition implies that w and v have all coordinates equal to 0 or 4. Then, for $wt_E(w) = 16n_4(w)$ to be equal to 16, $n_4(w) = 1$, which is impossible because of the second condition.

The resulting choices for $4u$ are the only candidates for the code C in the doubling method, since the conditions of the theorem exclude all choices that lead to a code \tilde{C} which is not extremal. \square

For an extremal Type II \mathbb{Z}_8 -code C of length $n \in \{24, 32, 40\}$ and type $(\frac{n}{2}, 0, 0)$ which has an extremal residue code $C^{(4)}$, the next algorithm returns all unsuitable candidates $4u$, i.e., the candidates for which the application of the doubling method leads to a Type II \mathbb{Z}_8 -code \tilde{C} which is not extremal. Thus, performing the given steps will find all possible candidates $4u$ for code C to produce a new extremal Type II \mathbb{Z}_8 -code \tilde{C} by the doubling method.

Algorithm C

Let $n \in \{24, 32, 40\}$. Denote by $S_i(w)$ the set of positions with the element $i \in \mathbb{Z}_8$ in $w \in \mathbb{Z}_8^n$. Let C be an extremal Type II \mathbb{Z}_8 -code of length n and type $(\frac{n}{2}, 0, 0)$, where $C^{(4)}$ is extremal, with the generator matrix $G = [I_{\frac{n}{2}} \quad A]$ in the standard form.

1. Let $v = (v_1, \dots, v_n) \in C^{(4)}$ be a codeword of Euclidean weight 16.
- 1.2. Let $F_v = \{v_1, \dots, v_{\frac{n}{2}}\}$, $A_v = S_2(v) \cap F_v$ and $B_v = S_3(v) \cap F_v$.
- 1.3. Repeat the following steps on all $A \subseteq A_v$:
 - 1.3.1. Calculate $v' = v + 4s_A + 4s_{B_v}$, where s_A is the sum of rows in the generator matrix G of C with row indices in A and s_{B_v} is the sum of rows in G with row indices in B_v .

1.3.2. Let

$$\begin{aligned} O_{v'} &= (S_2(v') \cup S_6(v')) \cap \left\{ \frac{n}{2} + 1, \dots, n \right\}, \\ P_{v'} &= S_4(v') \cap \left\{ \frac{n}{2} + 1, \dots, n \right\}, \\ Q_{v'} &= (S_3(v') \cup S_5(v')) \cap \left\{ \frac{n}{2} + 1, \dots, n \right\}. \end{aligned}$$

1.3.3. Let \mathcal{B} be the collection of all sets

$$B = O \cup P_{v'} \cup Q_{v'}, \quad O \subseteq O_{v'},$$

where $|B| \geq 2$.

2. For all $i \in \{1, \dots, \frac{n}{2}\}$, do the following.

2.1. Let $O_i = S_4(4G_i) \cap \left\{ \frac{n}{2} + 1, \dots, n \right\}$, where G_i is the i^{th} row of G .

2.2. Include all O_i such that $|O_i| \geq 2$ in \mathcal{B} .

Our method of construction is based on the following theorem.

Theorem 4.2. *Let $n \in \{24, 32, 40\}$. Denote by $S_i(w)$ the set of positions with the element $i \in \mathbb{Z}_8$ in $w \in \mathbb{Z}_8^n$. Let C be an extremal Type II \mathbb{Z}_8 -code of length n and type $(\frac{n}{2}, 0, 0)$, where $C^{(4)}$ is extremal. Furthermore, let \mathcal{S} be the collection of all $S \subseteq \left\{ \frac{n}{2} + 1, \dots, n \right\}$ such that $|S| \geq 2$. Then $\mathcal{G} = \mathcal{S} \setminus \mathcal{B}$ is the set of all possible $S_4(4u)$ for the code C in the doubling method which lead to an extremal Type II \mathbb{Z}_8 -code \tilde{C} of length n and type $(\frac{n}{2} - 1, 1, 1)$, where \mathcal{B} is the set obtained by applying Algorithm C.*

Proof. The first condition in Theorem 4.1 is checked in Step 1. of Algorithm C. Since the condition requires that $S_3(v) \cup S_4(v) \cup S_5(v) \subseteq S_4(4u)$, the coefficients of the rows of G in the linear combination of v cannot be 3, 4 or 5. Step 1.3.1. generates all such codewords v' with $wt_E(v' \pmod{4}) = 16$. All subsets $B = S_4(4u)$ satisfying the first condition are included in \mathcal{B} in Step 1.3.3.

The second condition of Theorem 4.1, implies that v is one of the rows of G with coefficient 4. These codewords are considered in Step 2. □

4.1 New extremal Type II \mathbb{Z}_8 -codes of length 32

Six inequivalent extremal Type II \mathbb{Z}_8 -codes of length 32 are known: $C_{8,32,i}$, $i = 1, \dots, 5$ from [15] and D_{32} from [13]. Since all of them are of type $(16, 0, 0)$, we investigate the possibility of constructing new extremal Type II \mathbb{Z}_8 -codes of length 32 by using the introduced doubling method. The result of our analysis is given in Proposition 4.3.

Proposition 4.3. *There are at least 10 inequivalent extremal Type II \mathbb{Z}_8 -codes of length 32 and type $(15, 1, 1)$.*

Proof. The extremal Type II \mathbb{Z}_8 -codes $C_{8,32,1}$ and $C_{8,32,2}$ are of type $(16, 0, 0)$ and length 32 and have extremal \mathbb{Z}_4 -residue codes. So, we can apply Theorem 4.2. We applied Algorithm C and found 23067 candidates $4u$ for a construction of extremal Type II \mathbb{Z}_8 -codes of type $(15, 1, 1)$ and length 32 from $C_{8,32,1}$ by doubling method. Also, we found 22818 candidates $4u$ for a construction of extremal Type II \mathbb{Z}_8 -codes of type $(15, 1, 1)$ and length 32 from $C_{8,32,2}$ by doubling method.

The extremal Type II \mathbb{Z}_8 -codes $C_{8,32,i}$, $i = 3, 4, 5$ are of type $(16, 0, 0)$ and length 32 and have \mathbb{Z}_4 -residue codes of minimum Euclidean weight 8. So, we cannot apply Theorem 4.2 to obtain new extremal Type II \mathbb{Z}_8 -codes from $C_{8,32,i}$, $i = 3, 4, 5$ using the doubling method.

Further, the extremal Type II \mathbb{Z}_8 -code D_{32} is of type $(16, 0, 0)$ and length 32 and it has an extremal \mathbb{Z}_4 -residue code. So, we can apply Theorem 4.2. We applied Algorithm C and found 22965 candidates $4u$ for a construction of extremal Type II \mathbb{Z}_8 -codes of type $(15, 1, 1)$ and length 32 from D_{32} by doubling method.

We use Theorem 3.3 to obtain the generator matrices for the 68850 constructed extremal Type II \mathbb{Z}_8 -codes of type $(15, 1, 1)$ and length 32. Using Magma ([6]), we calculated the weight distributions of the corresponding 68850 binary residue codes and obtained that, with respect to the weight distribution of their binary residue codes, all constructed extremal Type II \mathbb{Z}_8 -codes are distributed into 10 classes. The corresponding weight distributions are given in Table 1.

	i	0	8	12	16	20	24	32
$C_1^{(2)}$	W_i	1	316	6912	18310	6912	316	1
$C_2^{(2)}$	W_i	1	332	6848	18406	6848	332	1
$C_3^{(2)}$	W_i	1	337	6888	18259	7000	283	0
$C_4^{(2)}$	W_i	1	305	6952	18259	6936	315	0
$C_5^{(2)}$	W_i	1	308	6944	18262	6944	308	1
$C_6^{(2)}$	W_i	1	300	6976	18214	6976	300	1
$C_7^{(2)}$	W_i	1	364	6720	18598	6720	364	1
$C_8^{(2)}$	W_i	1	380	7168	17670	7168	380	1
$C_9^{(2)}$	W_i	1	324	6880	18358	6880	324	1
$C_{10}^{(2)}$	W_i	1	340	6816	18454	6816	340	1

Table 1: Weight distributions of the binary residue codes

For each of the obtained weight distribution classes we give the generator matrix in standard form of one extremal Type II \mathbb{Z}_8 -code of length 32 and type $(15, 1, 1)$, namely, the generator matrices for the following class representatives:

$$C_1 = C_{8,32,10} \oplus \langle 4u \rangle, \quad S_4(4u) = \{17, 19, 21, 22\},$$

$$C_2 = C_{8,32,10} \oplus \langle 4u \rangle, \quad S_4(4u) = \{17, 18, 20, 21\},$$

$$\begin{aligned}
C_3 &= C_{8,32,1_0} \oplus \langle 4u \rangle, \quad S_4(4u) = \{17, 19, 21\}, \\
C_4 &= C_{8,32,1_0} \oplus \langle 4u \rangle, \quad S_4(4u) = \{17, 19, 20, 21, 22\}, \\
C_5 &= C_{8,32,1_0} \oplus \langle 4u \rangle, \quad S_4(4u) = \{17, 18, 19, 20\}, \\
C_6 &= C_{8,32,1_0} \oplus \langle 4u \rangle, \quad S_4(4u) = \{17, 18, 19, 20, 21, 22\}, \\
C_7 &= C_{8,32,1_0} \oplus \langle 4u \rangle, \quad S_4(4u) = \{18, 24, 25, 27\}, \\
C_8 &= C_{8,32,1_0} \oplus \langle 4u \rangle, \quad S_4(4u) = \{20, 25\}, \\
C_9 &= C_{8,32,2_0} \oplus \langle 4u \rangle, \quad S_4(4u) = \{17, 18, 19, 21\}, \\
C_{10} &= C_{8,32,2_0} \oplus \langle 4u \rangle, \quad S_4(4u) = \{17, 21, 24, 25\}.
\end{aligned}$$

Those are, respectively:

$$\begin{aligned}
G_1 &= \begin{pmatrix} 1000000000000003476716356020474 \\ 0100000000000001703275645602047 \\ 0010000000000002574363514560204 \\ 00010000000000010615011726731415 \\ 0000100000000001365303604145602 \\ 00000100000000013130517745777155 \\ 00000010000000012611632063324043 \\ 00000001000000011163700305167532 \\ 00000000100000003474020675235254 \\ 00000000010000002347402047523525 \\ 00000000001000000234740234752352 \\ 00000000000100013461057440750622 \\ 00000000000010011200362013622110 \\ 00000000000001011426617330117347 \\ 0000000000000100100242336536347 \\ 00000000000000022004046042646062 \\ 0000000000000004040440000000000 \end{pmatrix}, \quad G_2 = \begin{pmatrix} 1000000000000003032756356020474 \\ 01000000000000011003437173330306 \\ 0010000000000002574363514560204 \\ 00010000000000003257436341456020 \\ 0000100000000001721343604145602 \\ 00000100000000002532574360414560 \\ 0000001000000001657657406041456 \\ 00000001000000010665127256332404 \\ 00000000100000013130222123763513 \\ 00000000010000002347402047523525 \\ 000000000010000010770142562400611 \\ 00000000000100012163276311123574 \\ 00000000000010010346541764075062 \\ 00000000000001000464634753634752 \\ 0000000000000111142225713011734 \\ 00000000000000020200404664264606 \\ 0000000000000004404400000000000 \end{pmatrix}, \\
G_3 &= \begin{pmatrix} 10000000000000010034371733303061 \\ 01000000000000012301250222165434 \\ 0010000000000002574363514560204 \\ 00010000000000003257436341456020 \\ 0000100000000001365343604145602 \\ 00000100000000002532574360414560 \\ 0000001000000001213657406041456 \\ 00000001000000011163740305167532 \\ 00000000100000003474020675235254 \\ 00000000010000002347402047523525 \\ 00000000001000000234740234752352 \\ 00000000000100002063074063475235 \\ 00000000000010011200362013622110 \\ 00000000000001000020634753634752 \\ 0000000000000100100202336536347 \\ 00000000000000022004046042646062 \\ 0000000000000004040400000000000 \end{pmatrix}, \quad G_4 = \begin{pmatrix} 1000000000000003472716356020474 \\ 01000000000000012305210222165434 \\ 0010000000000002574363514560204 \\ 00010000000000003257436341456020 \\ 00001000000000012767326261420277 \\ 00000100000000013130517745777155 \\ 0000001000000001217617406041456 \\ 0000000100000000561765720604145 \\ 00000000100000003474020675235254 \\ 00000000010000013745425424006112 \\ 00000000001000000234740234752352 \\ 00000000000100002067034063475235 \\ 00000000000010011200362013622110 \\ 00000000000001011422617330117347 \\ 0000000000000100104242336536347 \\ 00000000000000022004046042646062 \\ 0000000000000004044440000000000 \end{pmatrix},
\end{aligned}$$

$$\begin{aligned}
G_5 &= \begin{pmatrix} 1000000000000003072356356020474 \\ 01000000000000011003437173330306 \\ 00100000000000002574363514560204 \\ 000100000000000013753230677104367 \\ 000010000000000011465545132673141 \\ 000001000000000002532574360414560 \\ 000000100000000011313051734577715 \\ 00000001000000000161325720604145 \\ 000000001000000003474020675235254 \\ 00000000010000002347402047523525 \\ 00000000001000010730542562400611 \\ 00000000000100012163276311123574 \\ 00000000000010000602347436347523 \\ 00000000000001000424234753634752 \\ 00000000000000111102625713011734 \\ 00000000000000020200404664264606 \\ 000000000000000444400000000000 \end{pmatrix}, \quad G_6 = \begin{pmatrix} 1000000000000003072716356020474 \\ 0100000000000001307275645602047 \\ 001000000000000012270525042216543 \\ 000100000000000003257436341456020 \\ 00001000000000001761303604145602 \\ 000001000000000002532574360414560 \\ 00000010000000001617617406041456 \\ 000000010000000010665127256332404 \\ 00000000100000003474020675235254 \\ 00000000010000002347402047523525 \\ 00000000001000000234740234752352 \\ 00000000000100002467034063475235 \\ 00000000000010010306501764075062 \\ 00000000000001010120036201362211 \\ 00000000000000111102265713011734 \\ 00000000000000020200404664264606 \\ 000000000000000444444000000000 \end{pmatrix}, \\
G_7 &= \begin{pmatrix} 1000000000000000736356436520474 \\ 01000000000000011641650322265434 \\ 00100000000000001274363114560204 \\ 000100000000000013455451236631415 \\ 00001000000000001525743364045602 \\ 00000100000000000232574170214560 \\ 0000001000000000553257246041456 \\ 00000001000000002425325130604145 \\ 000000001000000011472043462110641 \\ 00000000010000001247402747023525 \\ 00000000001000003034740664752352 \\ 0000000000010000623474443275235 \\ 00000000000010003002347646347523 \\ 00000000000001002460234433134752 \\ 00000000000000101440602576736347 \\ 00000000000000022204046442046062 \\ 000000000000000400000044040000 \end{pmatrix}, \quad G_8 = \begin{pmatrix} 100000000000000105761506345567733 \\ 01000000000000013007437113330306 \\ 001000000000000010630165062216543 \\ 000100000000000013313230677104367 \\ 000010000000000013465545112673141 \\ 000001000000000002536574320414560 \\ 00000010000000013313051714577715 \\ 000000010000000011661127206332404 \\ 000000001000000012534622173763513 \\ 000000000100000013403204365251064 \\ 00000000001000012370542542400611 \\ 00000000000100011163276321123574 \\ 00000000000010012746141744075062 \\ 00000000000001010120036201362211 \\ 000000000000000102402023415363475 \\ 00000000000000022204404604264606 \\ 000000000000000400400000000000 \end{pmatrix}, \\
G_9 &= \begin{pmatrix} 10000000000000003076052565220712 \\ 01000000000000003743645266522071 \\ 001000000000000010566173551245033 \\ 000100000000000003637436417665220 \\ 00001000000000000723343601766522 \\ 000001000000000002436374360176652 \\ 000000100000000013075046441402411 \\ 00000001000000000164763736601766 \\ 000000001000000010041231556227000 \\ 00000000010000003221702247363425 \\ 00000000001000002762570234736342 \\ 00000000000100002350263252546347 \\ 00000000000010000663221746347363 \\ 00000000000001001066322154634736 \\ 00000000000000103546232225463473 \\ 00000000000000020024026046066460 \\ 000000000000000444040000000000 \end{pmatrix}, \quad G_{10} = \begin{pmatrix} 10000000000000003436052125220712 \\ 010000000000000011653420430260330 \\ 001000000000000010204547300310546 \\ 000100000000000003637436417665220 \\ 000010000000000012273526053424061 \\ 000001000000000002436374360176652 \\ 000000100000000013153412630555124 \\ 000000010000000012434146100347225 \\ 00000000100000002217022573634254 \\ 00000000010000003221702247363425 \\ 000000000001000002322570674736342 \\ 00000000000100002632617423473634 \\ 00000000000010012573404550005622 \\ 00000000000001001066322154634736 \\ 00000000000000111016015037121732 \\ 00000000000000020620046064204606 \\ 000000000000000400040044000000 \end{pmatrix}.
\end{aligned}$$

So, we obtained at least 10 new inequivalent extremal Type II \mathbb{Z}_8 -codes of length 32. \square

Proposition 4.3, together with results from [13] and [15], yields the following theorem.

Theorem 4.4. *There are at least 16 inequivalent extremal Type II \mathbb{Z}_8 -codes of length 32.*

Remark 4.5. According to [12], binary $[32, 15, 8]$ codes are the optimal binary $[32, 15]$ codes. Therefore, all constructed extremal Type II \mathbb{Z}_8 -codes of length 32 have optimal binary residue codes.

Acknowledgement

This work has been supported by Croatian Science Foundation under the project 4571 and by the University of Rijeka under the Project uniri-iskusni-prirod-23-62-3007.

References

- [1] M. ARZAKI, D. SUPRIJANTO, *Note on Elementary Constructions of Self-Dual Codes over \mathbb{Z}_8* , Int. Math. Forum **6**(16), 785–794 (2011).
- [2] S. BAN, D. CRNKOVIĆ, M. MRAVIĆ, S. RUKAVINA, *New extremal Type II \mathbb{Z}_4 -codes of length 32 obtained from Hadamard matrices*, Discrete Math. Algorithms Appl., **11** 1950057 (18 pp.) (2019).
- [3] S. BAN, S. RUKAVINA, *On some new extremal Type II \mathbb{Z}_4 -codes of length 40*, Math. Commun. **25**(2), 253–268 (2020).
- [4] E. BANNAI, S. T. DOUGHERTY, M. HARADA, M. OURA, *Type II Codes, Even Unimodular Lattices and Invariant Rings*, IEEE Trans. Inf. Theory, **45**(4), 1194–1205 (1999).
- [5] A. BETTEN, M. BRAUN, H. FRIPERTINGER, A. KERBER, A. KOHNERT, A. WASSERMANN, *Error-Correcting Linear Codes: Classification by Isometry and Applications*, Springer, Berlin (2006).
- [6] W. BOSMA, J. CANNON, C. PLAYOUST, *The Magma algebra system I, The user language*, Symb. Comput., 24, 235–265 (1997).
- [7] K. H. CHAN, *Three New Methods for Construction of Extremal Type II \mathbb{Z}_4 -Codes*, PhD Thesis, University of Illinois at Chicago (2012).
- [8] R. CHAPMAN, P. SOLÉ, *Universal codes and unimodular lattices*, J. de th. des nombres de Bordeaux, **8**, 369–376 (1996).
- [9] J. H. CONWAY, N. J. A. SLOANE, *Sphere Packing, Lattices and Groups (3rd ed.)*, Springer-Verlag, New York (1999).
- [10] S. T. DOUGHERTY, T. A. GULLIVER, M. HARADA, *Type II self-dual codes over finite rings and even unimodular lattices*, J. Alg. Combin., **9**, 233–250 (1999).

- [11] S. GEORGIU, M. HARADA, C. KOUKOUVINOS, *Orthogonal designs and Type II codes over \mathbb{Z}_{2k}* , Des. Codes Cryptogr. **25**, 163–174 (2002).
- [12] M. GRASSL, *Code Tables: Bounds on the parameters of various types of codes*: <http://www.codetables.de/> (accessed February 12, 2024)
- [13] T. A. GULLIVER, M. HARADA, *Extremal Self-Dual Codes over $\mathbb{Z}_6, \mathbb{Z}_8$ and \mathbb{Z}_{10}* , AKCE J. Graphs. Combin. **2**(1), 11–24 (2005).
- [14] A. R. HAMMONS, P. V. KUMAR, A. R. CALDERBANK, N. J. A. SLOANE, P. SOLÉ, *The \mathbb{Z}_4 -linearity of Kerdock, Preparata, Goethals and related codes*, IEEE Trans. Inform. Theory, **40**, 301–319 (1994).
- [15] M. HARADA, *Construction of extremal Type II \mathbb{Z}_{2k} -codes*, Finite Fields Appl. **87**, 102154 (2023).
- [16] W. C. HUFFMAN, V. PLESS, *Fundamentals of Error-Correcting Codes*, Cambridge University Press, Cambridge (2003).
- [17] The GAP Group, GAP - Groups, Algorithms and Programming, Version 4.11.0, available at www.gap-system.org
- [18] V. D. TONCHEV, *Codes*, in: *Handbook of Combinatorial Designs*, 2nd ed. (C. J. Colbourn and J. H. Dinitz, Eds.), Chapman and Hall/CRC, Boca Raton, 667-702 (2007).