Three New Methods for Construction of Extremal Type II $\mathbb{Z}_4$-Codes

BY

KIN HANG CHAN
B.Sc. (Chinese University of Hong Kong) 2004
M.Phil. (Chinese University of Hong Kong) 2006

THESIS

Submitted as partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Mathematics
in the Graduate College of the
University of Illinois at Chicago, 2012

Chicago, Illinois

Defense Committee:

Lawrence Ein, Chair
Vera Pless, Advisor
Janet Bessinger
Robert Sloan, Computer Science
William Cary Huffman, Loyola University Chicago

To my parents, Lai Kam Lilian Tsang and King Yee Johnnny Chan,

my beloved, family and friends,

who support me endlessly to achieve my childhood dream.

ACKNOWLEDGMENTS

First and foremost, my deepest thanks go to my advisor Dr. Vera Pless for her unlimited support throughout my whole period of PhD studies. She has introduced me to a fantastic world of coding theory with many nice ideas. On top of that, she taught me to think and tackle problems in an elegant way, which serves as a key to success. Her effort and assistance have contributed significantly towards my graduation.

I also have to thank Dr. Jon-Lark Kim, a former student of Dr. Pless, for his assistance and discussion. He offered me some intuition to choose a right problem which has eventually evolved into the main idea of my thesis. He also introduced the computer algebra system MAGMA to me. A lot of computation in my work would become impractical without the use of the system.

During my several years in UIC, I have learned a lot from different professors. My mathematical maturity has been greatly raised throughout their courses. These great courses include Graph Theory by Dr. Dhruv Mubayi, Machine Learning Theory by Dr. Gyorgy Turan, Computability and Complexity Theory by Dr. Robert Sloan and Computer Algorithms by Dr. Bhaskar DasGupta. They have taught so well that I could understand abstract concepts with ease.

My department has also helped me a lot in various ways. I wish to acknowledge Department of Mathematics, Statistics and Computer Science, especially Dr. Lawrence Ein and the assistant director of graduate studies Ms. Kari Dueball, for their advice when things did not go smoothly.

ACKNOWLEDGMENTS (Continued)

I want to express my gratitude to several important teachers who offered me golden opportunities to broaden my horizon in mathematics during different times in my life. They include Mr. Chang-Keung Tse, Ms. Mei-Yin Tse and Ms. Man-Wah Chan at S.K.H. Kei Hin Primary School, Mr. Fuk-Ip Leung at Bishop Hall Jubilee School, Dr. Thomas Kwok-Keung Au and Dr. Luen-Fai Tam at Chinese University of Hong Kong and Dr. Kin-Yin Li from Hong Kong University of Science and Technology.

Many friends, both in and not in US, have encouraged and accompanied me during ups and downs. I must mention some names that can never be forgotten: Ming-Yan Leung, Hong-Fu Ko, Po-Lam Yung, Cornelia Yuen, Man-Chuen Cheng, Mo Deng, Kin-Ping Ma, my roommates Lei Song and Rajashankar Kaliappan.

Last but not least, my special and wholehearted thanks go to my beloved family in Hong Kong, particularly my parents. They have provided me with the best environment to study and grow. I would like to dedicate this thesis to my parents and share my pride with them.

KHC

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

CHAPTER **PAGE**

LIST OF TABLES

SUMMARY

Coding theory has been a topic of interest since Claude Shannon published his paper "A Mathematical Theory of Communication" in 1948. Since then, it has become one of the core subjects in information theory. In coding theory, we discuss various methods for data compression and error correction. Data compression needs efficient algorithms to encode messages to shorter ones for the sake of quicker transmission. Error correction considers methods to encode messages which are to be sent through a noisy channel, so that errors below a certain tolerance rate can be detected or even corrected. Both directions have been widely discussed in the past decades.

My thesis is discussing error-correcting codes. In particular, all three chapters of my paper are about linear codes over $\mathbb{Z}_4$, the cyclic ring of four elements. These codes have been a great interest of mathematicians and engineers due to their fascinating properties and close relation with binary codes. For example, we can construct a number of families of famous non-linear binary codes from $\mathbb{Z}_4$-linear codes. This close relation was first published by Hammons, Kumar, Calderbank and Sloane in "The $\mathbb{Z}_4$-linearity of Kerdock, Preparata, Goethals, and related codes" in 1994. I became interested in $\mathbb{Z}_4$-linear codes, in particular self-dual codes, after reading the book "Fundamentals of Error-Correcting Codes" written by W. Cary Huffman and Vera Pless, as well as some papers on classification of self-dual codes over $\mathbb{Z}_4$ published by Huffman (1), Pless (2) and Masaaki Harada (3). From their work, I have learned the importance of linear codes of high weight, so I decided to focus on extremal codes which are the codes to achieve the highest possible weight.

In this thesis, we construct some new extremal Type II linear codes over $\mathbb{Z}_4$ with three different methods. The first method, which I call the *doubling method*, is new while the other two methods are generalizations of methods given by Harada. Hundreds of new extremal codes of length 32 and 40 have been discovered with the three methods.

This thesis consists of 5 chapters.

Chapter 1 is an introduction to basic definitions.

Chapter 2 considers the doubling method. It is an efficient method to construct a large number of Type II codes of type $4^a 2^b$ from a given Type II code of $4^{a+1} 2^{b-2}$.

Chapter 3 discusses the generalized building-up method. The original building-up method was first introduced by Harada on self-dual codes over $GF(2)$. Jon-Lark Kim and Yoonjin Lee generalized it to codes over a finite field of all orders (4) (5). Heisook Lee and Yoonjin Lee also generalized it to codes over $\mathbb{Z}_{p^m}$ where p is an odd prime (6). We generalize the method to Type II codes over $\mathbb{Z}_4$.

In Chapter 4 we construct new codes with a method generalized from a construction method given by Harada (7). His original method only works on codes of type 4^n only. We generalize it to generate codes of type $4^a 2^b$ for $b > 0$.

Chapter 5 is a summary of the work.

CHAPTER 1

INTRODUCTION

1.1 Basic Definitions

Let $\mathbb{Z}_4 = \mathbb{Z}/4\mathbb{Z} = \{0, 1, 2, 3\}$ be the cyclic ring of four elements. A $\mathbb{Z}_4$-linear *code* C of length n is a $\mathbb{Z}_4$-submodule of $\mathbb{Z}_4^n$. Elements of a code are called *codewords*. For convenience, we call coordinate values 0 and 2 *even*, while coordinate values 1 and 3 are called *odd*. A codeword with even coordinates only is said to be *even*. In the following context, all codes are $\mathbb{Z}_4$-linear codes unless otherwise stated. We consider the usual inner product between codewords.

Let $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ and $\mathbf{y} = (y_1, y_2, \ldots, y_n)$ be two codewords. The *inner product* is defined as

$$\langle \mathbf{x}, \mathbf{y} \rangle = x_1 y_1 + x_2 y_2 + \cdots + x_n y_n \in \mathbb{Z}_4.$$

The dual code $C^\perp$ of a code C is defined as

$$C^\perp = \{\mathbf{y} \in \mathbb{Z}_4^n | \langle \mathbf{x}, \mathbf{y} \rangle = 0 \text{ for all } \mathbf{x} \in C\}.$$

We call a code C *self-orthogonal* if $C \subseteq C^\perp$ and *self-dual* if $C = C^\perp$. By considering the size of the solution set of $C\mathbf{x} = \mathbf{0}$, we see that $|C||C^\perp| = 4^n$ if C is a code of length n. Therefore, a self-dual code must have 2^n codewords.

1.2 Euclidean weight and Type II Code

Suppose the number of coordinates i (where $i = 0, 1, 2, 3$) in a codeword $\mathbf{x}$ is $n_i(\mathbf{x})$. We define the *Euclidean weight* of a codeword $\mathbf{x}$ as

$$\mathrm{wt}_E(\mathbf{x}) = n_1(\mathbf{x}) + 4n_2(\mathbf{x}) + n_3(\mathbf{x}).$$

The Euclidean weight of a code C is defined as the minimum weight of all non-zero codewords of C. If the Euclidean weight of all codewords of a self-dual code C are multiples of 8, we say that the code is a *Type II code.*

From (8), we know that a Type II code of length n exists if and only if n is a multiple of 8. Also, the Euclidean weight of a Type II code of length n must be smaller than or equal to

$$8\left\lfloor \frac{n}{24} \right\rfloor + 8.$$

If the Euclidean weight of a code C meets the optimal bound $8\left\lfloor \frac{n}{24} \right\rfloor + 8$, the code is said to be an *extremal* code. If the weight of a code C of length n is not less than that of any other codes of length n, we say the code is an *optimal* code. It is clear that the extremal condition must be at least as strong as the optimal condition. As shown in (9), there are extremal codes of length 8, 16, 24, 32, 40, 48, 56 and 64. The existence of extremal codes at higher lengths remains an open question.

1.3 Generator Matrix

Let M_n be the group (under multiplication) of all $n \times n$ invertible monomial matrices over $\mathbb{Z}_4$. Saying in another way, they are the $n \times n$ matrices with exactly one 1 or one 3 in each row and each column. Two codes C and $\widetilde{C}$ are considered *equivalent* if there exists a matrix $M \in M_n$ such that $\widetilde{C} = MC$. From the properties of a group, we see that the equivalence of codes is an equivalence relation.

For every code C, we let R be a maximal free $\mathbb{Z}_4$-submodule of C. Suppose that R has a basis B and B' is a minimal generating set of C containing B. If $|B| = a$ and $|B'| = a + b$, then we say that C is of *type* $4^a 2^b$. It can be proved easily that the choice of R does not affect the value of a and b, so the type is an intrinsic value of a code. We can also see that elements in B must be non-even while elements in $B' \backslash B$ must be even. By considering the number of codewords, we know that $2a + b = n$ for a self-dual code C of length n.

A *generator matrix* of a code C is a matrix such that its rows are precisely a minimal generating set of C. From (10, Page 469), we know that with permutation of coordinates, there is a generator matrix of a code C of type $4^a 2^b$ in the form

$$G = \begin{bmatrix} I_a & A & B_1 + 2B_2 \\ O & 2I_b & 2D \end{bmatrix}$$

where I_a and I_b are identity matrices of order a and b respectively, while A, B_1, B_2 and D are matrices with all entries as 0 and 1 while O is a zero matrix. We say G is in a *standard form.*

1.4 Weight Enumerator

For every code C, we let $c_{p,q,r,s}$ be the number of codewords with exactly p 0's, q 1's, r 2's and s 3's. The complete weight enumerator of a code C of length n is defined as

$$cwe_C(x_0, x_1, x_2, x_3) = \sum_{p+q+r+s=n} c_{p,q,r,s} x_0^p x_1^q x_2^r x_3^s.$$

The symmetric weight enumerator of a code C is defined as

$$swe_C(x_0, x_1, x_2) = cwe_C(x_0, x_1, x_2, x_1)$$

while the Euclidean weight enumerator of a code C is defined as

$$ewe_C(x) = swe_C(1, x, x^4).$$

It is clear that the coefficients of x^k in $ewe_C(x)$ is the number of codewords of Euclidean weight k in C.

The *residue code* C^1 and *torsion code* C^2 of a code C are defined as

$$C^1 = \{\mathbf{v} \pmod 2 | \mathbf{v} \in C\} \subseteq \mathbb{Z}_2^n \text{ and } C^2 = \{\mathbf{v} \in \mathbb{Z}_2^n | 2\mathbf{v} \in C\} \subseteq \mathbb{Z}_2^n.$$

Similarly, we can define the weight enumerator of a binary code C of length n as

$$we_C(x_0, x_1) = \sum_{p+q=n} c_{p,q} x_0^p x_1^q$$

where $c_{p,q}$ is the number of codewords with p 0's and q 1's.

1.5 An example: The octacode

We illustrate the definitions above with an example. Let C be the *octacode* with a generator matrix

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 2 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 3 & 2 & 1 & 3 \\ 0 & 0 & 1 & 0 & 3 & 3 & 2 & 1 \\ 0 & 0 & 0 & 1 & 3 & 1 & 3 & 2 \end{bmatrix}.$$

Because $HH^T = \mathbf{0}$, we see that H is self-dual. On the other hand, we can check that the octacode has weight enumerators

$$cwe_C(x_0, x_1, x_2, x_3) = x_0^8 + 14x_0^4x_2^4 + 56x_0^3x_1^3x_2x_3 + 56x_0^3x_1x_2x_3^3 + 56x_0x_1^3x_2^3x_3$$
$$+ 56x_0x_1x_2^3x_3^3 + x_1^8 + 14x_1^4x_3^4 + x_2^8 + x_3^8,$$
$$ewe_C(x) = cwe_C(1, x, x^4, x) = 1 + 128x^8 + 126x^{16} + x^{32}.$$

From the above, we see that the Euclidean weight of C is 8. Also, because the Euclidean weight of all codewords of C are multiples of 8 and C is self-dual, we see that C is a Type II code. Since there are 4 non-even rows in H, C is of type 4^4.

CHAPTER 2

NEW EXTREME TYPE II $\mathbb{Z}_4$-LINEAR CODES OF LENGTH 32 AND 40 BY DOUBLING METHOD

As extremal Type II codes have their particular importance in coding theory, we would like to classify them or find as many as we can. All Type II codes of length 8 and 16 are classified (11) (2). Some Type II codes with a larger length have been classified too. For example, all extremal codes of length 24 with an automorphism of prime order $p \geq 5$ are classified by W. Cary Huffman (1). The purpose of this chapter is to introduce a new method, which I call the *doubling method*, to find new extremal Type II codes. This method is based on modifying known extremal Type II codes of type 4^n to generate new codes of type $4^{n-1}2^2$ with some exhaustive search. With the known codes of length 32 and 40 given by Gaborit, Huffman and Harada (7) (12) (13), 543 and 830 new inequivalent codes are found respectively. It largely reduces the redundant exhaustive search and the running time on computing the symmetric weight enumerator, which is usually necessary for identifying the weight of a code. We also generalize the method to Type II codes of types 4^a2^b for $b > 0$ to generate Type II codes of type $4^{a-1}2^{b+2}$.

2.1 General Mechanism

Theorem 1. Let C be a Type II code (of length $n = 8k$) of type 4^{4k} and $G = [I|A]$ be a generator matrix of C. Here I denotes an identity matrix of order $4k$ and A is a $4k \times 4k$

matrix. Let $\mathbf{u} \in \mathbb{Z}_4^{8k}$ such that $2\mathbf{u} \notin C$ and have an even number of 2's in its coordinates. Let $C_0 = \{\mathbf{v} \in C | \langle \mathbf{u}, \mathbf{v} \rangle = 0\}$. Then $\widetilde{C} = C_0 \oplus \langle 2\mathbf{u} \rangle$ is a Type II code.

We quote two well-known properties (2) about Euclidean weight before proving the theorem.

Fact 2. $\langle \mathbf{x}, \mathbf{x} \rangle \equiv \mathrm{wt}_E(\mathbf{x}) \pmod 8$. Here we identify the identities of $\mathbb{Z}_4$ and $\mathbb{Z}$.

Proof. It follows directly from checking $\langle i, i \rangle = i^2 \equiv \mathrm{wt}_E(i) \pmod 8$ for all $i = 0, 1, 2$ and 3. □

Fact 3. If $\mathbf{x}$ and $\mathbf{y}$ are two orthogonal codewords while $\mathrm{wt}_E(\mathbf{x})$ and $\mathrm{wt}_E(\mathbf{y})$ are multiples of 8, then any linear combination of $\mathbf{x}$ and $\mathbf{y}$ are self-orthogonal and have a Euclidean weight divisible by 8.

Proof. It suffices to prove the statement for $\mathbf{x} + \mathbf{y}$. First of all, $\langle \mathbf{x}, \mathbf{x} \rangle \equiv \mathrm{wt}_E(\mathbf{x}) \equiv \langle \mathbf{y}, \mathbf{y} \rangle \equiv \mathrm{wt}_E(\mathbf{y}) \equiv 0 \pmod 8$. We also note that $2\langle \mathbf{x}, \mathbf{y} \rangle$ is a multiple of 8 because $\langle \mathbf{x}, \mathbf{y} \rangle \equiv 0 \pmod 4$. The results follow from linearity of the inner product as

$$\mathrm{wt}_E(x + y) \equiv \langle \mathbf{x} + \mathbf{y}, \mathbf{x} + \mathbf{y} \rangle = \langle \mathbf{x}, \mathbf{x} \rangle + 2\langle \mathbf{x}, \mathbf{y} \rangle + \langle \mathbf{y}, \mathbf{y} \rangle \equiv 0 + 0 + 0 = 0 \pmod 8.$$

□

Here comes the proof of Theorem 1.

Proof. Since C is self-dual and $2\mathbf{u} \notin C$, there must be an element $\mathbf{w} \in C$ such that $\langle \mathbf{u}, \mathbf{w} \rangle \neq 0$. Since $2\mathbf{u}$ is an even codeword, it forces $\langle \mathbf{u}, \mathbf{w} \rangle = 2$ and hence $\mathbf{w}$ is not even, for otherwise $\langle \mathbf{u}, \mathbf{w} \rangle = 0$. Suppose $\widetilde{\mathbf{w}} \in C \backslash C_0$. Again, the parity of $2\mathbf{u}$ implies that $\langle \mathbf{u}, \widetilde{\mathbf{w}} \rangle = 2$. Therefore,

$\widetilde{\mathbf{w}} - \mathbf{w} \in C_0$ and hence $\widetilde{\mathbf{w}} \in C_0 + \mathbf{w}$, the coset of C_0 containing $\mathbf{w}$. As $\widetilde{\mathbf{w}}$ is arbitrary in $C \backslash C_0$, we see that C_0 and $C_0 + \mathbf{w}$ are the only two cosets of C_0 in C and so C_0 has 2^{8k-1} codewords. This implies that $\widetilde{C} = C_0 \oplus \langle 2\mathbf{u} \rangle$ is a code with 2^{8k} codewords. As $\mathrm{wt}_E(2\mathbf{u})$ is a multiple of 8 and $\langle 2\mathbf{u}, \mathbf{v} \rangle = 0$ for all $\mathbf{v} \in \widetilde{C}$, $\widetilde{C}$ is self-orthogonal while all codewords in $\widetilde{C}$ have a Euclidean weight divisible by 8 from Lemma 3. The number of codewords in $\widetilde{C}$ forces that $\widetilde{C}$ is self-dual and hence a Type II code. □

Theorem 4. The choice of $2\mathbf{u}$ can be limited to codewords with 0's on the first $4k$ coordinates. Each coset will have only one such choice of $2\mathbf{u}$. On the other hand, each such choice leads to a different Type II code (when equivalence is not considered).

Proof. We note that choosing two codewords $2\mathbf{u}$ and $2\mathbf{u}'$ in the same coset of C in $\mathbb{Z}_4^{8k}$ yields the same code $\widetilde{C}$. The submatrix of the $4k$ leftmost columns of G is an identity matrix, so for any $4k$-tuple of $\{0, 2\}$, we can always find a unique codeword $2\mathbf{v}$ such that $2\mathbf{u}$ coincides with the $4k$-tuples on the first $4k$ entries. Therefore, given any arbitrary choice $2\mathbf{u}$, we can always find a corresponding $2\mathbf{v}$ where $2\mathbf{u}' = 2\mathbf{u} - 2\mathbf{v}$ has all zeros on its first $4k$ entries and $2\mathbf{u} + C = 2\mathbf{u}' + C$. The uniqueness of such $2\mathbf{u}'$ in $2\mathbf{u} + C$ follows from the uniqueness of the choice of $2\mathbf{v}$ immediately. □

2.2 From The Point of View Using Generator Matrix

The process can be seen more clearly from the generator matrix.

Theorem 5. Let $B = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_{4k}\}$ be a basis of a Type II code C of type 4^{4k}. Let $2\mathbf{u} \in \mathbb{Z}_4^{8k} \backslash C$ such that the first $4k$ coordinates of $2\mathbf{u}$ are 0 and $\mathrm{wt}_E(2\mathbf{u})$ is a multiple of 8. The following process will yield a Type II code of type $4^{4k-1}2^2$.

Step 1: Let $B_E = \{\mathbf{v}_i \in B | \langle \mathbf{v}_i, \mathbf{u} \rangle = 0\}$ and $B_O = B \backslash B_E$.

Step 2: Pick $\mathbf{v}_i \in B_O$ arbitrarily. Consider $B'_O = \{\mathbf{v}_i + \mathbf{v}_j | \mathbf{v}_j \in B_O\}$.

Step 3: The set $\tilde{B} = B'_O \cup B_E \cup \{2\mathbf{u}\}$ is a basis of a Type II code $\tilde{C}$ of type $4^{4k-1}2^2$.

The resultant code is independent of the choice of $\mathbf{v}_i$ in Step 2.

Proof. First of all, we have to prove that the process can be done. The only concern is whether B_O is empty. From the proof of Theorem 1, there must be a codeword $\mathbf{w}$ such that $\langle \mathbf{w}, 2\mathbf{u} \rangle = 2$. For the existence of such a codeword, there must be an element $\mathbf{v}$ in any basis such that $\langle \mathbf{v}, 2\mathbf{u} \rangle \neq 0$ due to linearity of the inner product. It shows that B_O is not empty and hence the process can be done. Next, since $\langle \mathbf{v}_i + \mathbf{v}_j, 2\mathbf{u} \rangle = \langle \mathbf{v}_i, 2\mathbf{u} \rangle + \langle \mathbf{v}_j, 2\mathbf{u} \rangle = 2 + 2 = 0$ for all $\mathbf{v}_j \in B_O$, all codewords in B'_O are orthogonal to $2\mathbf{u}$. Because $\mathbf{v}_i + \mathbf{v}_j$ is an even codeword if and only if $i = j$, there are $4k - 1$ non-even codewords and one even codeword in $B'_O \cup B_E$. Note that $\langle B'_O \cup B_E \rangle = \{\mathbf{v} \in C | \langle \mathbf{v}, 2\mathbf{u} \rangle = 0\}$. Therefore, $\tilde{C}$ is a Type II code by Theorem 1. The independence follows from the fact that $\mathbf{v}_j + \mathbf{v}_k = (\mathbf{v}_i + \mathbf{v}_j) + (\mathbf{v}_i + \mathbf{v}_k) + (\mathbf{v}_i + \mathbf{v}_i)$. Therefore, the basis B'_O when $\mathbf{v}_j$ is chosen is a linear combination of another basis when $\mathbf{v}_i$ is chosen. It implies that they span the same set. □

We can illustrate the theorem on the octacode, a Type II code of length 8.

Let C be the octacode with a generator matrix

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 & 2 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 3 & 2 & 1 & 3 \\ 0 & 0 & 1 & 0 & 3 & 3 & 2 & 1 \\ 0 & 0 & 0 & 1 & 3 & 1 & 3 & 2 \end{bmatrix}.$$

As seen in Chapter 1, the octacode is a Type II code. Let $2\mathbf{u} = (0,0,0,0,2,2,0,0)$. It is easy to check that $2\mathbf{u} \notin C$ and $\mathrm{wt}_E(2\mathbf{u}) = 8$. Let $\mathbf{v}_i$ be the i-th row of G. With some simple computation, we can see that $\langle 2\mathbf{u}, \mathbf{v}_1\rangle = \langle 2\mathbf{u}, \mathbf{v}_2\rangle = 2$ and $\langle 2\mathbf{u}, \mathbf{v}_3\rangle = \langle 2\mathbf{u}, \mathbf{v}_4\rangle = 0$, so $B_O = \{\mathbf{v}_1, \mathbf{v}_2\}$ and $B_E = \{\mathbf{v}_3, \mathbf{v}_4\}$. We pick $\mathbf{v}_i = \mathbf{v}_1$ as in Theorem 5. Then $C_0 = \{\mathbf{v} \in C | \langle 2\mathbf{u}, \mathbf{v}\rangle = 0\}$ has a generator matrix

$$G_0 = \begin{bmatrix} 2 & 0 & 0 & 0 & 0 & 2 & 2 & 2 \\ 1 & 1 & 0 & 0 & 1 & 3 & 2 & 0 \\ 0 & 0 & 1 & 0 & 3 & 3 & 2 & 1 \\ 0 & 0 & 0 & 1 & 3 & 1 & 3 & 2 \end{bmatrix}.$$

Finally, we obtain the code $\tilde{C}$ with a generator matrix

$$\tilde{G} = \begin{bmatrix} 2 & 0 & 0 & 0 & 0 & 2 & 2 & 2 \\ 1 & 1 & 0 & 0 & 1 & 3 & 2 & 0 \\ 0 & 0 & 1 & 0 & 3 & 3 & 2 & 1 \\ 0 & 0 & 0 & 1 & 3 & 1 & 3 & 2 \\ 0 & 0 & 0 & 0 & 2 & 2 & 0 & 0 \end{bmatrix}$$

by adding the row $2\mathbf{u}$ to G_0. It is easy to verify that $\tilde{C}$ is a Type II code of type $4^3 2^2$.

2.3 The Condition Of An Extremal Code

We can obtain a new Type II code easily with Theorem 5. However, a lot of newly generated codes are not extremal. It is extremely time consuming to generate all candidates and evaluate their symmetric weight enumerators to select appropriate codes. The following section introduces a method to look for candidates of $2\mathbf{u}$ as needed in Theorem 5 which avoids lengthy computation when $n = 24, 32$ and 40.

Theorem 6. Let $n = 8k = 24, 32$ or 40. We denote $S_i(\mathbf{w})$ as the set of coordinate positions with i's in a codeword $\mathbf{w}$ where $i = 0, 1, 2$ or 3. Let C be an extremal Type II code of length $8k$ and type 4^{4k} where no codeword has exactly four odd coordinates. Suppose $2\mathbf{u}$ is an element of $\mathbb{Z}_4^{8k}$ such that $S_2(2\mathbf{u}) \subseteq \{4k+1, 4k+2, \ldots, 8k\}$ where $|S_2(2\mathbf{u})|$ is even and at least 4. If there is no codeword $\mathbf{v}$ of C satisfying any of the following conditions:

1. $S_2(\mathbf{v}) \subseteq S_2(2\mathbf{u}) \subseteq S_1(\mathbf{v}) \cup S_2(\mathbf{v}) \cup S_3(\mathbf{v})$ and $|S_1(\mathbf{v}) \cup S_3(\mathbf{v})| = 8$.

2. $|S_2(2\mathbf{u}) \backslash (S_1(\mathbf{v}) \cup S_3(\mathbf{v}))| = |(S_1(\mathbf{v}) \cup S_3(\mathbf{v})) \backslash S_2(2\mathbf{u})| = 1$

3. $S_2(2\mathbf{u}) \subseteq (S_1(\mathbf{v}) \cup S_3(\mathbf{v}))$ and $|S_1(\mathbf{v}) \cup S_3(\mathbf{v})| - |S_2(2\mathbf{u})| = 2$

then the Type II code generated by $2\mathbf{u}$ and C as in Theorem 5 is extremal. These choices of $2\mathbf{u}$ are also the only candidates for the code C in the method given by Theorem 5.

Proof. Note that when $8k = 24, 32$ or 40, an extremal Type II code must have weight $8\left\lfloor \frac{8k}{24} \right\rfloor + 8 = 16$. Recall that the Euclidean weight of all codewords in a Type II code must be a multiple of 8. This means that a Type II code of these lengths is not extremal if and only if there exists a codeword with Euclidean weight 8 in the code. Since all codewords of C have a Euclidean weight 16 or above, a codeword with Euclidean weight 8 must be of the form $\mathbf{v} + 2\mathbf{u}$ where

$\mathbf{v} \in C$ such that $\langle \mathbf{v}, 2\mathbf{u} \rangle = 0$. We are going to prove that such a codeword $\mathbf{v}$ does not exist. Suppose there is such a codeword $\mathbf{v}$. We write $\mathbf{w} = \mathbf{v} + 2\mathbf{u}$ for convenience. Then we have

$$S_0(\mathbf{w}) = (S_0(\mathbf{v}) \cap S_0(2\mathbf{u})) \cup (S_2(\mathbf{v}) \cap S_2(2\mathbf{u}))$$

$$S_1(\mathbf{w}) = (S_1(\mathbf{v}) \cap S_0(2\mathbf{u})) \cup (S_3(\mathbf{v}) \cap S_2(2\mathbf{u}))$$

$$S_2(\mathbf{w}) = (S_2(\mathbf{v}) \cap S_0(2\mathbf{u})) \cup (S_0(\mathbf{v}) \cap S_2(2\mathbf{u}))$$

$$S_3(\mathbf{w}) = (S_3(\mathbf{v}) \cap S_0(2\mathbf{u})) \cup (S_1(\mathbf{v}) \cap S_2(2\mathbf{u}))$$

We have made use of the fact that $S_1(2\mathbf{u}) = S_3(2\mathbf{u}) = \emptyset$ as $2\mathbf{u}$ is even. We also note that $S_1(\mathbf{w}) \cup S_3(\mathbf{w}) = S_1(\mathbf{v}) \cup S_3(\mathbf{v})$. Therefore,

$$\mathrm{wt}_E(\mathbf{w}) = |S_1(\mathbf{v}) \cup S_3(\mathbf{v})| + 4\left\{|(S_2(\mathbf{v}) \cap S_0(2\mathbf{u}))| + |(S_0(\mathbf{v}) \cap S_2(2\mathbf{u}))|\right\}. \qquad (2.1)$$

Here are three different cases:

Case 1: $|S_1(\mathbf{v}) \cup S_3(\mathbf{v})| > 8$

It is obvious that $\mathrm{wt}_E(\mathbf{w}) > 8$. As $\mathbf{w} = \mathbf{v} + 2\mathbf{u}$, $\mathrm{wt}_E(\mathbf{w})$ is divisible by 8 by Lemma 3. We have $\mathrm{wt}_E(\mathbf{w}) \geq 16$ consequently.

Case 2: $|S_1(\mathbf{v}) \cup S_3(\mathbf{v})| = 8$

If $\mathrm{wt}_E(\mathbf{w}) = 8$, we need the second summand of (Equation 2.1) to be 0, in particular $S_2(\mathbf{v}) \cap S_0(2\mathbf{u})$ and $S_0(\mathbf{v}) \cap S_2(2\mathbf{u})$ are empty sets. They are equivalent to

$$S_2(\mathbf{v}) \subseteq S_2(2\mathbf{u}) \text{ and } S_2(2\mathbf{u}) \subseteq S_1(\mathbf{v}) \cup S_2(\mathbf{v}) \cup S_3(\mathbf{v}).$$

However, this condition is given not to occur as required.

Case 3: $|S_1(\mathbf{v}) \cup S_3(\mathbf{v})| < 8$

Since the number of odd coordinates of $\mathbf{v}$ must be a multiple of 4 and it cannot be 4 as required, $\mathbf{v}$ must have no odd coordinate. In other words, $\mathbf{v}$ is even, so (Equation 2.1) can be simplified as

$$\mathrm{wt}_E(\mathbf{w}) = 4\left\{|(S_2(\mathbf{v}) \cap S_0(2\mathbf{u}))| + |(S_0(\mathbf{v}) \cap S_2(2\mathbf{u}))|\right\}. \tag{2.2}$$

By considering the $4k$ leftmost coordinates, we observe that $\mathbf{v} = 2\sum_{i \in S_2(\mathbf{v}) \cap \{1,2,\ldots,4k\}} G_i$. Therefore, there is a codeword $\widetilde{\mathbf{v}} \in C$ such that $2\widetilde{\mathbf{v}} = \mathbf{v}$. It is obvious that $S_1(\widetilde{\mathbf{v}}) \cup S_3(\widetilde{\mathbf{v}}) = S_2(\mathbf{v})$ and $S_0(\widetilde{\mathbf{v}}) \cup S_2(\widetilde{\mathbf{v}}) = S_0(\mathbf{v})$. For $\mathrm{wt}_E(\mathbf{w}) < 16$ in (Equation 2.2) to hold, we must have

$$\left(|(S_1(\widetilde{\mathbf{v}}) \cup S_3(\widetilde{\mathbf{v}})) \cap S_0(2\mathbf{u})|, |(S_0(\widetilde{\mathbf{v}}) \cup S_2(\widetilde{\mathbf{v}})) \cap S_2(2\mathbf{u})|\right) = (0,0), (0,2), (1,1) \text{ or } (2,0)$$

as $\mathrm{wt}_E(\mathbf{w})$ is a multiple of 8.

We note that the first two candidates $(0,0)$ and $(0,2)$ cannot be reached unless $\widetilde{\mathbf{v}} = \mathbf{0}$. It is because they lead to a fact that $(S_1(\widetilde{\mathbf{v}}) \cup S_3(\widetilde{\mathbf{v}})) \subseteq S_2(2\mathbf{u})$, but it is absurd as all of the first $4k$ coordinates of $\mathbf{u}$ are 0 while some of the first $4k$ coordinates of $\widetilde{\mathbf{v}}$ are odd unless $\widetilde{\mathbf{v}} = \mathbf{0}$. The third and fourth candidates $(1,1)$ and $(2,0)$ can be satisfied if and only if the second and third conditions given not to occur respectively when we replace $\mathbf{v}$ by $\widetilde{\mathbf{v}}$ in the conditions.

□

2.4 Searching Procedures

Although Theorem 6 has outlined the exact conditions on the element $2\mathbf{u}$ for the generation of an extremal Type II code, the search takes too much time if we test all codewords $\mathbf{v}$ against $\mathbf{u}$ for the conditions mentioned in the theorem. We introduce a quick procedure to find all candidates $\mathbf{u}$ for a given Type II code C of type 4^{8k}. We begin with some preparation work.

Theorem 7. Let $C^1 = \{\mathbf{v} \pmod 2 | \mathbf{v} \in C\} \subseteq \mathbb{Z}_2^{8k}$ be the binary residue code of a Type II code C. Then $S_1(\mathbf{v}_1) = S_1(\mathbf{v}) \cup S_3(\mathbf{v})$ where $\mathbf{v}_1$ is the image of $\mathbf{v}$ in C^1.

Proof. The results are immediate as only coordinates 1 and 3 in $\mathbb{Z}_4$ of any codeword $\mathbf{v}$ can be mapped to 1 in $\mathbb{Z}_2$ of $\mathbf{v}_1$. □

By searching on the binary residue code C^1 instead of C, the number of codewords to check is reduced from $O(4^{8k})$ to $O(2^{8k})$, thus it is much more efficient and practical when $8k = 24, 32$ and 40. The following algorithm is adopted.

Theorem 8. Implementing the following steps will find all possible candidates $2\mathbf{u}$ for an extremal Type II code C to generate a new extremal Type II code $\widetilde{C}$ as described in Theorem 6.

Step 1: Let C^1 be the binary residue code of C. Let G^1 be a generator matrix of C^1 in the form of $[I|A]$. Coordinates are permuted in C if necessary.

Step 2: For every subset $F \subseteq \{1, 2, \ldots, 8k\}$ with $1 \leq |F| \leq 8$, we do the following:

Step 2.1: Find the sum of rows in C^1 with row indices in F. Denote the sum as $\mathbf{s}_F^1$.

Step 2.2: If $S_1(\mathbf{s}_F^1) \leq 8$, we repeat the following steps on all subsets E of F:

Step 2.2.1: Evaluate $\mathbf{v} = \mathbf{s}_F + 2\mathbf{s}_E$ where $\mathbf{s}_F$ (resp. $\mathbf{s}_E$) is the sum of rows in C with row indices F (resp. E).

Step 2.2.2: Let $T_\mathbf{v} = S_2(\mathbf{v}) \cap \{4k+1, 4k+2, \ldots, 8k\}$ and $O_\mathbf{v} = (S_1(\mathbf{v}) \cup S_3(\mathbf{v})) \cap \{4k+1, 4k+2, \ldots, 8k\}$.

Step 2.2.3: Consider all subsets B such that $B = T_\mathbf{v} \cup O$ where $O \subseteq O_\mathbf{v}$ and $|B|$ is even. Include all such sets B in a set $\mathcal{B}$. (The set $\mathcal{B}$ will contain the support of all unsuitable choices of $2\mathbf{u}$.)

Step 3: For every element $i \in \{1, 2, \ldots, 4k\}$, we do the following:

Step 3.1: Let $O_i = (S_1(G_i) \cup S_3(G_i)) \cap \{4k+1, 4k+2, \ldots, 8k\}$ where G_i is the i-th row of G.

Step 3.2: Include all subsets $B \subseteq O_i$ such that $|O_i| - |B| = 1$ into $\mathcal{B}$.

Step 3.3: Include all subsets $B = O_i \cup \{k\}$ such that $k \in \{4k+1, 4k+2, \ldots, 8k\} \setminus O_i$ into $\mathcal{B}$.

Step 4: For every two-element subset $\{i, j\} \subseteq \{1, 2, \ldots, 4k\}$, we do the following:

Step 4.1: Let $\mathbf{v}_{i,j} = G_i + G_j$ and $O_{i,j} = (S_1(\mathbf{v}_{i,j}) \cup S_3(\mathbf{v}_{i,j})) \cap \{4k+1, 4k+2, \ldots, 8k\}$.

Step 4.2: Include $O_{i,j}$ into $\mathcal{B}$.

Step 5: Let $\mathcal{S}$ be the collection of all subsets S of $\{4k+1, 4k+2, \ldots, 8k\}$ such that $|S|$ is even and $|S| \geq 4$. Then $\mathcal{G} = \mathcal{S} \backslash \mathcal{B}$ is the set of support of all even elements $2\mathbf{u}$ which can be used in Theorem 6 for the code C.

Here we explain why this algorithm works.

Proof. The first condition in Theorem 6 is checked in Step 2. As the condition requires that $S_2(\mathbf{v}) \subseteq S_2(2\mathbf{u})$, the coefficients of rows of C in the linear combination of $\mathbf{v}$ can only be 0, 1 or 3. Step 2.2.1 generates all such codewords $\mathbf{v}$. All subsets of coordinates fulfilling the first condition are included in the set $\mathcal{B}$.

Because $|(S_1(\mathbf{v}) \cup S_3(\mathbf{v})) \backslash S_2(2\mathbf{u})| = 1$ or 2 in the second and third conditions, $\mathbf{v}$ can only be a linear combination of two rows of C to satisfy at least one of them. The coefficients of the linear combinations can be $\{0,1\}, \{0,3\}, \{1,1\}, \{1,3\}$ and $\{3,3\}$. Since $3\mathbf{v}+\mathbf{w}$ and $\mathbf{v}+\mathbf{w}$ shares the same positions of odd coordinates, we can skip checking the coefficients $\{0,3\}$, $\{1,3\}$ and $\{3,3\}$. These two conditions with $|(S_1(\mathbf{v}) \cup S_3(\mathbf{v})) \backslash S_2(2\mathbf{u})| = 1$ are checked in Step 3 while the case for $|(S_1(\mathbf{v}) \cup S_3(\mathbf{v})) \backslash S_2(2\mathbf{u})| = 2$ are checked in Step 4. Step 5 is merely an exclusion of all sets satisfying at least one of the three conditions. $\square$

2.5 New Codes For $n = 32$

We have taken the code $C(W_{16,11})$ in (3) as our seed C. The generator matrix G_W of $C(W_{16,11})$ is given explicitly in (12) as

$$G_W = \begin{bmatrix}
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&2&1&0&0&0&1&1&1&0&1&1&1&1&1&1&1\\
0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&3&2&0&0&3&0&3&1&1&0&3&1&1&3&1&3\\
0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&2&1&1&3&0&1&3&3&0&1&3&3&1&1\\
0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&3&2&1&1&3&0&3&1&3&0&3&1&1&3\\
0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&1&3&3&2&3&0&0&1&3&3&3&0&1&1&1\\
0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&3&0&1&3&1&2&0&0&1&1&3&1&3&0&3&1\\
0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&3&1&0&1&0&0&2&3&1&3&1&1&3&1&0&3\\
0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&3&3&3&0&0&0&1&2&1&1&1&3&3&3&1&0\\
0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&3&1&1&3&3&3&3&2&1&0&0&0&1&1&1\\
0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&3&0&1&3&1&3&1&3&3&2&0&0&1&0&1&3\\
0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&3&1&0&1&1&1&3&3&0&0&2&3&1&3&0&1\\
0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&3&3&3&0&1&3&3&1&0&0&1&2&1&1&3&0\\
0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&3&3&1&1&0&1&1&1&0&3&3&3&2&1&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&3&1&1&3&3&0&3&1&3&0&1&3&3&2&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&3&3&3&3&3&1&0&3&3&3&0&1&0&0&2&1\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&3&1&3&1&3&3&1&0&3&1&3&0&0&0&3&2
\end{bmatrix}.$$

With the use of the algorithm run on MAGMA, we have found at least 338 new inequivalent Type II codes of type $4^{15}2^2$. Their inequivalence is shown by the weight enumerator of their binary residue code and the number of codewords of Euclidean weight 16. There are seven different weight enumerators of the binary residue codes, namely

$$\begin{aligned} P_i =& x_0^{32} + (292+8i)x_0^{24}x_1^8 + (7008-32i)x_0^{20}x_1^{12} + (18166+48i)x_0^{16}x_1^{16} \\ &+ (7008-32i)x_0^{12}x_1^{20} + (292+8i)x_0^8x_1^{24} + x_1^{32} \end{aligned}$$

for integers $1 \le i \le 7$.

In order to save space, only 30 codes C_i are listed with the right half of the element $2\mathbf{u}_i$ used. (The left half of $2\mathbf{u}_i$ must be sixteen 0's.) All 338 codes are available from the author. All these codes are of type $4^{15}2^2$. These codes are proved inequivalent by their binary residue codes and $E_{16}(C_i)$, the number of codewords with Euclidean weight 16. $B_{m,n}$ denotes a binary code with $we(B_{m,n}) = P_m$ where $B_{m,n}$ and $B_{m,n'}$ are equivalent if and only if $n = n'$. Note that inequivalent binary codes can have identical weight enumerators.

TABLE I

30 EXTREMAL CODES FROM DOUBLING METHOD FOR $N = 32$

i	right half of $2\mathbf{u}_i$	$B_{m,n}$	$E_{16}(C_i)$
1	2220220222202000	$B_{1,1}$	108392
2	2220020022002020	$B_{1,1}$	108384
3	2222002000022222	$B_{1,2}$	108544
4	2200002000200022	$B_{1,2}$	108480
5	2220000200020002	$B_{1,3}$	108400
6	2022002000200200	$B_{1,3}$	108352
7	2002022000220022	$B_{1,4}$	108464
8	2222220000220000	$B_{1,4}$	108336
9	2222202020002202	$B_{1,5}$	108440
10	2222200020020222	$B_{1,5}$	108400
11	2022200220000220	$B_{1,6}$	108528
12	2200220020222000	$B_{1,6}$	108480
13	2222202000220220	$B_{1,7}$	108392
14	2220222020000222	$B_{1,7}$	108352
15	2022200020002000	$B_{1,8}$	108352
16	2022200020000200	$B_{1,9}$	108352
17	2222002200202202	$B_{2,1}$	108473
18	2222000200200022	$B_{2,2}$	108473
19	2220220020002200	$B_{2,3}$	108497
20	2200220020202020	$B_{2,4}$	108481
21	2222202220002020	$B_{3,1}$	108466
22	2222222000200000	$B_{3,2}$	108506
23	2222220000202220	$B_{3,3}$	108474
24	2220222022200002	$B_{3,4}$	108514
25	2222222220222022	$B_{3,5}$	108482
26	2222002200222222	$B_{3,6}$	108594
27	2222222020200002	$B_{4,1}$	108523
28	2022022020002002	$B_{5,1}$	108644
29	2220002020222000	$B_{6,1}$	108621
30	0022000000000022	$B_{7,1}$	108662

2.6 New Codes For $n = 40$

We have taken the code $C_{40,11}$ in (12) as our seed C. The generator matrix $G_{40,11}$ of $C_{40,11}$ is given explicitly in (12) as

$$G_W = \left[\begin{array}{cccccccccccccccccccccccccccccccccccccccc}
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1&3&1&3&3&1&3&2&0&2&0&2&0&0&0&0&1\\
0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1&1&2&2&2&2&2&1&1&1&1&1&2&2&2&2&1\\
0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&3&3&1&0&0&1&3&2&0&2&1&1&2&2&0&1&1&3&2&1\\
0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&3&3&3&2&2&0&0&1&1&0&2&0&1&3&2&1&1&2&1&1\\
0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&3&1&2&1&1&2&0&0&2&1&0&0&2&2&1&1&1&3&1&1\\
0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&3&2&3&2&0&3&1&2&2&1&0&0&3&3&1&2&0&3&1&1\\
0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&1&2&2&1&0&1&2&3&3&0&3&2&1&2&3&3&0&3&0&1\\
0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&3&0&1&2&3&0&3&3&1&2&3&0&2&3&0&1&1\\
0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&1&2&0&2&1&0&3&3&1&2&0&1&2&3&1&2&3&1&0&1\\
0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&3&0&2&0&3&2&3&0&1&3&1&3&1&2&2&1&0&2&3&1\\
0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&2&3&3&0&0&2&0&1&3&3&3&1&2&0&1&2&2&1&1&3\\
0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&2&3&0&1&2&1&3&0&1&0&0&3&1&2&3&2&3&2&3&1\\
0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&3&0&3&0&0&3&3&0&3&2&3&3&1&2&3&2&1&0&1\\
0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&2&3&0&2&1&3&1&1&2&2&3&2&2&3&3&1&0&2&3&1\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&2&1&0&0&1&1&0&2&1&3&3&0&3&1&2&2&3&1&2&1\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&2&0&1&3&1&1&0&0&1&2&2&1&0&1&0&3&0&1&3&3\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&3&1&3&0&1&1&0&0&1&2&3&0&0&0&3&3&3&1\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&2&2&3&3&2&2&3&2&3&1&3&0&0&3&3&1&3&2&2&1\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&2&2&3&2&1&3&0&1&2&3&2&3&1&0&3&1&1&2&2&3\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&3&3&3&1&1&1&1&1&1&3&3&3&3&1&1&3&1&1&2
\end{array}\right].$$

With the use of the algorithm run on MAGMA, we have found at least 329 new inequivalent Type II codes of type $4^{19}2^2$. Their inequivalence is shown by the weight enumerator of their binary residue code and the number of codewords of Euclidean weight 16. There are seven different weight enumerators of the binary residue codes, namely

$$
\begin{aligned}
P_{i,j} =& x_0^{40} + (117+8j)x_0^{32}x_1^8 + (10768+128i-48j)x_0^{32}x_1^8 + (119690-512i+120j)x_0^{28}x_1^{12} \\
&+ (18166+48i)x_0^{24}x_1^{16} + (263136+768i-160j)x_0^{20}x_1^{20} + (18166+48i)x_0^{16}x_1^{24} \\
&+ (119690-512i+120j)x_0^{12}x_1^{28} + (10768+128i-48j)x_0^8x_1^{32} + (117+8j)x_0^8x_1^{32} + x_1^{40}
\end{aligned}
$$

for $(i,j) = (0,1),(0,2),(0,3),(0,4),(0,5),(1,4),(1,6),(2,5),(2,7),(2,9)$.

In order to save space, only 30 codes C_i are listed with the right half of the element $2\mathbf{u}_i$ used. (The left half of $2\mathbf{u}_i$ must be sixteen 0's.) All 329 codes are available from the author. All these codes are of type $4^{19}2^2$. These codes are proved inequivalent by their binary residue codes and $E_{16}(C_i)$, the number of codewords with Euclidean weight 16.

TABLE II

30 EXTREMAL CODES FROM DOUBLING METHOD FOR $N = 40$

i	right half of $2\mathbf{u}_i$	$P_{i,j}$	$E_{16}(C_i)$
1	02200020222000022220 0	$B_{0,1}$	34866
2	20202222000200002022 2	$B_{0,1}$	34982
3	22202200220200000220 2	$B_{0,2}$	34472
4	22022200222220022022 2	$B_{0,2}$	34508
5	22020220220000200220 2	$B_{0,2}$	34510
6	20202000222202202022 2	$B_{0,2}$	34554
7	22000002220202002220 2	$B_{0,3}$	34508
8	00202200200220202220 0	$B_{0,3}$	34530
9	22220002222002222000 2	$B_{0,3}$	34538
10	20002220002020220022 2	$B_{0,3}$	34556
11	02022220222022022000 0	$B_{0,4}$	34478
12	00222022200022000220 0	$B_{0,4}$	34566
13	02000200002002202202 0	$B_{0,4}$	34578
14	02022222000022220202 0	$B_{0,4}$	34618
15	22222020002222202000 2	$B_{1,4}$	34457
16	22200220200022022022 2	$B_{1,4}$	34497
17	22220222200222222220 2	$B_{1,4}$	34503
18	20000202222200200022 2	$B_{0,5}$	34632
19	20002002220022020022 2	$B_{0,5}$	34644
20	00000222222222002022 0	$B_{0,5}$	34658
21	00222200000202202202 0	$B_{0,5}$	34662
22	00000222200000000000 0	$B_{2,5}$	34450
23	00002202220002220220 0	$B_{2,5}$	34642
24	00020002022022202220 0	$B_{2,5}$	34874
25	22020220202002000220 2	$B_{1,6}$	34561
26	00002022220220020000 0	$B_{1,6}$	34643
27	00200222020000222220 0	$B_{1,6}$	34663
28	22000220020022200202 2	$B_{2,7}$	34616
29	02020022020202200202 0	$B_{2,7}$	34620
30	02220200022020020000 0	$B_{2,9}$	34836

2.7 Generalization

The method can be generalized to find extremal Type II codes of type $4^{a-1}2^{b+2}$ from a given extremal Type II code C of type 4^a2^b when $b > 0$. In the modified version, we have to look for elements $2\mathbf{u} \in \mathbb{Z}_4^n$ where $S_2(2\mathbf{u}) \geq 4$ such that $\mathbf{v} \in C$ and $\langle \mathbf{v}, \mathbf{u} \rangle = 0$ implies $\mathrm{wt}_E(\mathbf{v} + \mathbf{u}) > 8$.

The modified version is as follows:

Theorem 9. Let $n = 24, 32$ or 40 and adopt the notation of $S_i(\mathbf{w})$ as in Theorem 6. Let C be an extremal Type II code of length n and type 4^a2^b where no codeword has exactly four odd coordinates. Suppose $2\mathbf{u}$ is an element of $\mathbb{Z}_4^n$ such that $S_2(2\mathbf{u}) \subseteq \{a + b + 1, a + b + 2, \ldots, n\}$ where $|S_2(2\mathbf{u})|$ is even and at least 4.

If there is no codeword $\mathbf{v}$ of C that satisfies any of the following conditions:

1. $S_2(\mathbf{v}) \subseteq S_2(2\mathbf{u}) \subseteq S_1(\mathbf{v}) \cup S_2(\mathbf{v}) \cup S_3(\mathbf{v})$ and $|S_1(\mathbf{v}) \cup S_3(\mathbf{v})| = 8$.

2. $|S_2(2\mathbf{u})\backslash S_2(\mathbf{v})| + |S_2(\mathbf{v})\backslash S_2(2\mathbf{u})| = 2$.

then the Type II code generated by $2\mathbf{u}$ and C as in Theorem 5 is extremal. These choices of $2\mathbf{u}$ are also the only candidates for the code C in the method given by Theorem 5.

Proof. As in the proof of Theorem 6, the first condition is satisfied if and only if $\mathrm{wt}_E(2\mathbf{u}+\mathbf{v}) = 8$ and $2\mathbf{u}+\mathbf{v}$ is non-even. Suppose there is an even codeword $2\mathbf{v} \in C$ such that $\mathrm{wt}_E(2\mathbf{u}+2\mathbf{v}) = 8$. Then the number of coordinates where $2\mathbf{u}$ and $2\mathbf{v}$ do not agree must be 2. The converse is also true. Therefore, the second condition holds if and only if there is an even codeword of Euclidean weight 8 in the new code. □

Here is a modified algorithm for the generalization.

Theorem 10. Let C be a extremal Type II code of type 4^a2^b with a generator matrix G in standard form $\begin{bmatrix} I_a & A & B_1+2B_2 \\ 0 & 2I_b & 2D \end{bmatrix}$.

Step 1: Let C^1 be the binary residue code of C. Let G^1 be a generator matrix of C^1 in the form of $[I|A]$. Coordinates are permuted in C if necessary. Let H be the set of even rows in G.

Step 2: For every subset $F \subseteq \{1, 2, \ldots, a\}$ with $1 \leq |F| \leq 8$, we do the following:

Step 2.1: Find the sum of rows in C^1 with row indices in F. Denote the sum as $\mathbf{s}_F^1$.

Step 2.2: If $S_1(\mathbf{s}_F^1) \leq 8$, we repeat the following steps on all subsets E of F and all subsets H' of H:

Step 2.2.1: Evaluate $\mathbf{v} = \mathbf{s}_F + 2\mathbf{s}_E + 2\mathbf{s}_{H'}$ where $\mathbf{s}_F$ (resp. $\mathbf{s}_E$ and $\mathbf{s}_{H'}$) is the sum of rows in C with row indices F (resp. E and H').

Step 2.2.2: Let $T_{\mathbf{v}} = S_2(\mathbf{v}) \cap \{a+b+1, a+b+2, \ldots, n\}$ and $O_{\mathbf{v}} = (S_1(\mathbf{v}) \cup S_3(\mathbf{v})) \cap \{a+b+1, a+b+2, \ldots, n\}$.

Step 2.2.3: Consider all subsets B such that $B = T_{\mathbf{v}} \cup O$ where $O \subseteq O_{\mathbf{v}}$ and $|B|$ is even. Include all such sets B in a set $\mathcal{B}$.

Step 3: For every subset $S \subseteq \{2G_1, 2G_2, \ldots, 2G_{a-1}, 2G_a, G_{a+1}, \ldots, G_{a+b}\}$ where $1 \leq |S| \leq 2$, we do the following:

Step 3.1: Let $\mathbf{s}$ be the sums of elements in S.

Step 3.2: Let $O_i' = S_2(\mathbf{s}) \cap \{1, 2, \ldots, a+b\}$ and $O_i = S_2(\mathbf{s}) \cap \{a+b+1, a+b+2, \ldots, n\}$.

Step 3.3: If $|O_i'| = 1$, then

Step 3.3.1: Include all subsets B of O_i such that $|O_i| - |B| = 1$ into $\mathcal{B}$.

Step 3.3.2: Include all subsets $B = O_i \cup \{k\}$ such that $k \in \{a+b+1, a+b+2, \ldots, n\} \backslash O_i$ into $\mathcal{B}$.

Step 3.4: If $|O_i'| = 2$, then include O_i into $\mathcal{B}$.

Step 4: Let $\mathcal{G}$ be collection of all subsets of $\{a+b+1, a+b+2, \ldots, n\} \backslash \mathcal{B}$ such that $|G|$ is even and $|G| \geq 4$. Then $\mathcal{G}$ is the set of coordinate subsets G where an even element $2\mathbf{u}$ of $\mathbb{Z}_4^n$ with $S_2(2\mathbf{u}) = G$ can be used in Theorem 6 for the code C.

Proof. Step 2 in Theorem 10 is similar to Step 2 in Theorem 8. This step removes all choices of $2\mathbf{u}$ such that $\text{wt}_E(2\mathbf{u} + \mathbf{v}) = 8$ where $\mathbf{w} = 2\mathbf{u} + \mathbf{v}$ is non-even. We consider the second condition in Theorem 9 in Step 3. In Steps 3.3, we consider the situation when $2\mathbf{v}$ has exactly one 2 in the first $a+b$ coordinates. It requires $2\mathbf{u}$ to differ from $2\mathbf{v}$ by at least 2 coordinates in the positions $\{a+b+1, \ldots, n\}$ so that $S_2(2\mathbf{u} + 2\mathbf{v}) > 2$. In Step 3.4, we consider the situation when $2\mathbf{v}$ has exactly two 2's in the first $a+b$ coordinates. We have to remove all $2\mathbf{u}$ such that $2\mathbf{u}$ and $2\mathbf{v}$ coincide in the positions $\{a+b+1, \ldots, n\}$. Since no non-zero codeword has all zeros in the first $a+b$ coordinates, the search is complete. $\square$

2.8 Results From Generalized Algorithm

We let

$$G_1 = \begin{bmatrix}
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&2&3&2&3&1&3&2&3&3&1&3&1&0&1&1 \\
0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&2&0&1&2&3&3&3&2&1&1&3&3&0&3&1 \\
0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&1&2&0&1&3&0&1&3&3&0&1&3&3&0&1&1 \\
0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&1&1&2&1&1&0&2&3&3&2&0&1&1&0&3&3&1 \\
0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&1&3&1&0&3&2&0&3&3&3&1&2&3&2&1&1 \\
0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&1&3&1&1&3&3&2&0&0&3&1&1&1&3 \\
0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&3&2&0&3&1&2&1&0&2&2&1&0&3&3&0 \\
0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&1&2&0&2&3&2&1&3&2&2&2&3&0&1&0&1 \\
0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&1&1&3&3&3&3&3&2&1&0&0&0&1&0&1&1 \\
0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&1&3&2&2&2&1&3&1&3&1&1&1&3&1 \\
0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&1&1&2&3&3&3&3&1&2&2&0&3&3&3&0&1&0 \\
0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&1&0&1&1&3&1&3&3&2&2&3&2&3&1&0&0&3 \\
0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&1&1&3&3&3&2&1&3&3&2&3&1&3&0&2&0&0 \\
0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&1&1&1&1&3&3&2&1&3&1&2&3&0&2&2&1&2 \\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1&3&1&1&1&2&1&3&1&0&2&0&0&2&3 \\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&2&0&2&2&2&2&0&2&2&2&2&0&2&0&0&0&0 \\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&2&0&0&2&0&2&0&2&0&0&2&2&2&2&0&0
\end{bmatrix}.$$

The code C_1 generated by G_1 is a Type II code of type $4^{15}2^2$ obtained by using Theorem 7 with $2\mathbf{u} = 2220220222202000$ after suitable permutation of coordinates. It is the first code in Table I. When we apply the generalized algorithm on C_1, 205 inequivalent extremal Type II codes of type $4^{14}2^4$ are found. There are seven different weight enumerators of the binary residue codes, namely

$$\begin{aligned} P_i = & x_0^{32} + (136+4i)x_0^{24}x_1^8 + (3536-16i)x_0^{20}x_1^{12} + (9038+24i)x_0^{16}x_1^{16} \\ & + (3536-16i)x_0^{12}x_1^{20} + (136+4i)x_0^8x_1^{24} + x_1^{32} \end{aligned}$$

for integers $1 \le i \le 9$. In order to save some space, only 30 new codes are listed. All codes are available from the author.

TABLE III

30 EXTREMAL CODES FROM GENERALIZED DOUBLING METHOD FOR $N = 32$

i	15 rightmost entries of $2\mathbf{u}_i$	P_i	$E_{16}(C_i)$
1	200202222222200	P_1	108176
2	022020002202022	P_1	108192
3	002220000200220	P_1	108200
4	020220022220200	P_1	108208
5	000022202022000	P_2	108241
6	002220200020002	P_2	108249
7	020002022002020	P_2	108257
8	020002200020000	P_3	108258
9	020002200000002	P_2	108265
10	200000002220202	P_3	108274
11	220200022200220	P_3	108298
12	000020222222020	P_3	108314
13	002022002200200	P_4	108323
14	020200022202220	P_4	108339
15	222000000020202	P_4	108355
16	000000002002202	P_4	108363
17	002222000200200	P_5	108372
18	200200022220202	P_5	108388
19	200220202002022	P_5	108404
20	222200222022020	P_6	108413
21	020020020222202	P_5	108420
22	200200020000002	P_6	108437
23	222220222020002	P_6	108453
24	020000202000002	P_6	108477
25	022000022020222	P_7	108510
26	222000222020222	P_7	108534
27	002000022202222	P_7	108542
28	202220020202002	P_7	108558
29	022000220022202	P_8	108559
30	202202002020000	P_9	108752

Similarly, we let

$$G_2 = \left[\begin{array}{cccccccccccccccccccccccccccccccccccccccc}
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1&3&1&3&3&1&3&2&0&2&0&2&0&0&0&0&1\\
0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&2&0&2&3&3&3&3&3&0&0&0&0&2&3&1&3&3&3\\
0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&2&0&3&1&2&0&3&1&3&0&0&1&1&1&2&0&0&3&3\\
0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&3&3&2&2&0&0&1&1&0&2&2&3&3&0&1&1&2&3&3\\
0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&0&0&0&2&3&1&1&3&2&3&3&1&1&2&2&0&0&2&3\\
0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&1&0&1&1&0&2&3&3&2&3&1&0&2&0&3&3&0&0&1\\
0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&2&0&1&2&3&0&0&1&2&1&0&1&0&0&3&0&1&3\\
0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&1&1&3&2&2&1&2&3&0&1&0&2&2&3&2&3&3&2&1&0&1\\
0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&1&1&1&2&1&2&1&0&0&2&3&3&2&3&2&0&3&2&2&3&1\\
0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&3&0&3&2&3&0&1&3&1&1&3&2&0&1&0&2&1&3\\
0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&1&0&2&3&3&1&3&1&2&0&0&2&2&3&3&0&3&1&2&0&3\\
0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&1&1&2&3&0&3&2&0&1&2&1&3&0&2&1&2&3&2&3&2&1\\
0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&1&1&2&1&2&1&1&0&0&1&0&1&0&0&0&1&0&1&2&3&1\\
0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&1&1&2&3&1&2&0&2&2&3&3&2&3&3&2&2&2&3&3&2&1\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&1&2&0&1&1&0&2&1&3&3&0&3&1&2&2&3&1&2&1\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&1&0&2&3&1&1&0&0&1&2&2&1&0&1&0&3&0&1&3&3\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&1&0&0&1&3&0&1&1&0&0&1&0&1&0&2&0&3&3&1&3\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&1&0&1&3&2&3&3&0&3&0&2&2&1&1&2&2&2&2&3&1&1\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&0&1&3&1&2&0&1&2&3&0&1&0&2&3&2&2&0&3&1&3\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&2&0&2&2&2&2&2&2&2&2&2&2&0&0&2&0&2&2&2&0&2\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&2&0&0&0&0&0&0&0&0&0&0&2&2&0&2&0&0&0&2&2
\end{array}\right].$$

The code C_2 is an extremal Type II code. When we apply the generalized algorithm on C_2, 501 inequivalent extremal Type II codes of type $4^{13}2^6$ are found. There are 21 different weight enumerators of the binary residue codes, namely

$$\begin{aligned}
P_{i,j} =& x_0^{40} + (57+4i)x_0^{32}x_1^{8} + (5432-24i+64j)x_0^{28}x_1^{12} + (59654+60i-256j)x_0^{24}x_1^{16} \\
& + (131856-80i+384j)x_0^{20}x_1^{20} + (59654+60i-256j)x_0^{16}x_1^{24} + (5432-24i+64j)x_0^{12}x_1^{28} \\
& + (57+4i)x_0^{8}x_1^{32} + x_1^{40}
\end{aligned}$$

for $(i,j) = (1,0),(2,0),(3,0),(4,0),(5,0),(6,0),(7,0),(3,1),(4,1),(5,1),(6,1),(7,1),(8,1),$ $(5,2),(6,2),(7,2),(8,2),(9,2),(12,2),(9,3),(12,4)$. In order to save some space, only 30 new codes are listed. All codes are available from the author.

TABLE IV

30 EXTREMAL CODES FROM GENERALIZED DOUBLING METHOD FOR $N = 40$

i	19 rightmost entries of $2\mathbf{u}_i$	P_i	$E_{16}(C_i)$
1	2022000222202220202	$P_{1,0}$	34561
2	2220022222200222022	$P_{1,0}$	34635
3	0202200200000000202	$P_{2,0}$	34469
4	0200220202002022000	$P_{2,0}$	34511
5	2020022002022222022	$P_{3,0}$	34401
6	0222020022020222220	$P_{3,0}$	34413
7	0020222000022000022	$P_{3,1}$	34738
8	2200202022222222220	$P_{4,0}$	34309
9	2220022200020222022	$P_{4,0}$	34391
10	2222002220200202202	$P_{4,1}$	34376
11	0202022000002020000	$P_{4,1}$	34452
12	2200200222002020022	$P_{5,0}$	34457
13	0200220202220222000	$P_{5,0}$	34525
14	2202220202002202220	$P_{5,1}$	34338
15	0220002220022222022	$P_{5,1}$	34384
16	0200222002220222220	$P_{5,2}$	34551
17	2222220002200000220	$P_{5,2}$	34621
18	0222202222020222220	$P_{6,0}$	34473
19	0222022200200222000	$P_{6,0}$	34513
20	0002200020000222022	$P_{6,1}$	34436
21	2002220000002222022	$P_{6,2}$	34455
22	0220000002202200022	$P_{7,0}$	34757
23	0202022000222200202	$P_{7,1}$	34388
24	0002202220002220220	$P_{7,2}$	34485
25	0202022000220202220	$P_{8,1}$	34772
26	0000000000222222220	$P_{8,2}$	34503
27	2022220222020002202	$P_{9,2}$	34381
28	2222002202022200220	$P_{9,3}$	34528
29	2022220222200200220	$P_{12,2}$	34669
30	0202022000002002220	$P_{12,4}$	34523

2.9 Conclusion

The results above can be summarized as follows.

Corollary 11. There are at least 543 extremal Type II codes of length 32 found with the doubling method, of which 338 of them are of type $4^{15}2^2$ and 205 of them are of type $4^{14}2^4$. There are at least 830 extremal Type II codes of length 40 found with the doubling method, of which 329 of them are of type $4^{19}2^2$ and 501 of them are of type $4^{18}2^4$.

CHAPTER 3

BUILDING-UP METHOD FOR EXTREMAL TYPE II $\mathbb{Z}_4$-LINEAR CODES OF LENGTHS 32 AND 40

In this chapter, we discuss a generalization of the building-up method developed by M. Harada for self-dual binary codes to self-dual codes over $\mathbb{Z}_4$. Harada introduced a method of construction called the building-up method on binary self-dual codes. The method requires a known binary self-dual code of length n to construct a new binary self-dual code of length $n+2$. The general idea of the method is to add a new row to the generator matrix of the code of length n. Two new columns are then appended to the matrix depending on the inner product of the generator matrix so that the new code is self-dual. There are a lot of generalizations of this method. For example, Jon-Lark Kim and Yoonjin Lee have generalized the method to self-dual codes over finite fields of all orders (4) (5). Heisook Lee and Yoonjin Lee also have a generalization to self-dual codes over $\mathbb{Z}_{p^m}$ for odd primes p (6). However, self-dual codes over $\mathbb{Z}_{2^m}$ where $m \geq 2$ remain untouched.

The purpose of this chapter is to discover new extremal Type II codes of lengths 32 and 40. We have found 219 new extremal Type II codes of length 32 and 133 new extremal Type II codes of length 40.

3.1 Building-Up Method

We prove the following theorem.

Theorem 12 (Building-up Method)**.** Let n be a multiple of 8 and C be a Type II $\mathbb{Z}_4$-code of length n. Let G be a generator matrix of C. Suppose X is a 4×8 matrix and Y is a $4 \times n$ matrix where the matrix $[X|Y]$ generates a self-orthogonal code and Euclidean weight of rows of $[X|Y]$ are multiples of 8. Let

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 2 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 3 & 2 & 1 & 3 \\ 0 & 0 & 1 & 0 & 3 & 3 & 2 & 1 \\ 0 & 0 & 0 & 1 & 3 & 1 & 3 & 2 \end{bmatrix}$$

be a generator matrix of the octacode. Suppose the rows of H and X span $\mathbb{Z}_4^8$; then the matrix

$$G' = \begin{bmatrix} X & Y \\ -GY^T(HX^T)^{-1}H & G \end{bmatrix}$$

is a generator matrix of a Type II code C' of length $n+8$.

Proof. First of all, we check the Euclidean weight of each row of G'. The Euclidean weight of the first four rows are multiples of 8 by construction. Next, we see that each of the remaining rows is a concatenation of a codeword of the octacode and a codeword of C. We recall that the octacode is a Type II code and C is also a Type II code by assumption. Therefore, every row has its Euclidean weight divisible by 8.

Next, we show that every two rows are orthogonal. Since each of $[X|Y]$, H and G has its own rows pairwise orthogonal, it suffices to show that each row in $[X|Y]$ is orthogonal to all rows in $[-GY^T(HX^T)^{-1}H|G]$. It can be proved by

$$\begin{aligned}\langle[-GY^T(HX^T)^{-1}H|G]_i,[X|Y]_j\rangle &= [-GY^T(HX^T)^{-1}H|G]_i[X^T|Y^T]_j\\ &= [-GY^T(HX^T)^{-1}HX^T+GY^T]_{i,j}\\ &= [-GY^TI_4+GY^T]_{i,j}\\ &= [-GY^T+GY^T]_{i,j}\\ &= 0.\end{aligned}$$

Since the Euclidean weight of every row is divisible by 8, the Euclidean weight of all codewords of C' are divisible by 8 by Fact 3 in Chapter 2. It remains to show C' is self-dual. As the code is self-orthogonal, it suffices to show that C' has $\sqrt{|\mathbb{Z}_4^{n+8}|}=2^{n+8}$ codewords. Note that C has 2^n codewords as C is a self-dual code of length n. There are 4^4 linear combinations of rows in $[X|Y]$. As C' is a linear space over $\mathbb{Z}_4$, C' has $2^n\times 4^4=2^{n+8}$ codewords if there is no non-trivial linear combination in C' equal to $\mathbf{0}$.

Suppose the contrary that there is such a non-trivial linear combination $\mathbf{v}$. We consider the first eight coordinates. As rows of H and X span $\mathbb{Z}_4^8$, the linear combination cannot involve any row of X since each row of X appears only once among all rows. In other words, the linear combination can only involve the rows in $[-GY^T(HX^T)^{-1}H|G]$. However, it is absurd as G is a minimum generating set of C by assumption. Therefore, the code C' is a self-orthogonal code with 2^{n+8} codewords, implying that its dual space cannot contain C' properly, which forces C' to be self-dual. □

We can illustrate the construction above with an example. Let C be a Type II code with a generator matrix

$$G = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 2 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 2 & 3 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 3 & 2 \\ 0 & 0 & 0 & 2 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 2 & 2 & 2 & 2 \end{bmatrix}.$$

Let

$$X = \begin{bmatrix} 0 & 1 & 0 & 1 & 2 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 2 & 0 & 1 & 1 & 2 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 2 \end{bmatrix} \text{ and } Y = \begin{bmatrix} 1 & 0 & 0 & 0 & 2 & 2 & 0 & 0 \\ 2 & 1 & 0 & 2 & 0 & 0 & 2 & 0 \\ 2 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 3 & 2 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

We can see that the matrix $[X|Y]$ is self-orthogonal and all its rows have an Euclidean weight divisible by 8. Also, the rows of X and H span $\mathbb{Z}_4^8$. We compute that

$$-GY^T(HX^T)^{-1}H = \begin{bmatrix} 2 & 3 & 2 & 0 & 3 & 2 & 1 & 1 \\ 2 & 1 & 3 & 0 & 0 & 1 & 1 & 0 \\ 1 & 3 & 2 & 3 & 2 & 0 & 1 & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

Hence the following matrix generates a Type II code from our theorem.

$$\begin{bmatrix} 0 & 1 & 0 & 1 & 2 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 2 & 2 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 2 & 1 & 0 & 2 & 0 & 0 & 2 & 0 \\ 2 & 0 & 1 & 1 & 2 & 1 & 0 & 0 & 2 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 2 & 3 & 2 & 0 & 1 & 1 & 1 & 1 & 1 \\ 2 & 3 & 2 & 0 & 3 & 2 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 2 & 1 & 1 \\ 2 & 1 & 3 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 2 & 3 & 0 \\ 1 & 3 & 2 & 3 & 2 & 0 & 1 & 2 & 0 & 0 & 1 & 1 & 0 & 1 & 3 & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 2 & 2 & 2 \end{bmatrix}$$

Here are some new codes that we have found from this building-up method.

3.2 New Codes For $n = 32$

We let C be the code C_8 in (1). After permuting the columns of C by the permutation (11 8 5 10 7 12 16 13 17 14 18 15 9 6), the generator matrix of the code is

$$G_8 = \begin{bmatrix}
1\,0\,0\,0\,0\,0\,0\,0\,0\,1\,1\,0\,1\,1\,1\,2\,0\,2\,0\,0\,1\,1\,2\,2 \\
0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,1\,1\,1\,0\,0\,2\,0\,0\,0\,1\,1\,3\,2\,3 \\
0\,0\,1\,0\,0\,0\,0\,0\,0\,1\,1\,1\,0\,1\,0\,0\,0\,2\,1\,1\,1\,0\,2\,0 \\
0\,0\,0\,1\,0\,0\,0\,0\,0\,1\,0\,1\,0\,0\,1\,2\,2\,2\,1\,0\,1\,3\,2\,3 \\
0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,1\,1\,1\,3\,1\,2\,2\,0\,3\,2\,1\,2 \\
0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0\,1\,0\,0\,2\,1\,1\,2\,1\,1\,0\,1\,1 \\
0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0\,1\,0\,1\,1\,3\,1\,1\,3\,0\,2\,2 \\
0\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0\,1\,3\,2\,3\,3\,2\,3\,0\,1\,1 \\
0\,0\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,1\,1\,1\,0\,0\,0\,1\,1\,1\,2\,2\,1 \\
0\,0\,0\,0\,0\,0\,0\,0\,0\,2\,0\,0\,0\,0\,0\,2\,0\,0\,2\,0\,0\,2\,2\,2 \\
0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,2\,0\,0\,0\,0\,0\,2\,0\,0\,2\,0\,2\,2\,2 \\
0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,2\,0\,0\,0\,0\,0\,2\,0\,0\,2\,2\,2\,2 \\
0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,2\,0\,0\,0\,0\,0\,0\,2\,2\,0\,0\,2 \\
0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,2\,0\,0\,0\,0\,2\,2\,2\,0\,0\,0 \\
0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,2\,0\,0\,0\,2\,0\,2\,0\,0\,2
\end{bmatrix}.$$

A pseudo-random search was performed on the matrix $[X|Y]$. At least 219 extremal inequivalent codes are found. The number of codewords with Euclidean weight 16 of these codes E_{16} are:

108251, 108301, 108307, 108325, 108331, 108333, 108356, 108357, 108372, 108374, 108382, 108387, 108388, 108389, 108405, 108406, 108408, 108414, 108415, 108422, 108423, 108429, 108430, 108432, 108438, 108446, 108447, 108452, 108454, 108455, 108462, 108463, 108464, 108465, 108469, 108470, 108471, 108477, 108479, 108480, 108481, 108483, 108486, 108487, 108488, 108489, 108493, 108494, 108495, 108499, 108503, 108504, 108505, 108510, 108511, 108512, 108518, 108519, 108521, 108526, 108527, 108534, 108535, 108536, 108537, 108542, 108543, 108544, 108545, 108546, 108550, 108551, 108553, 108559, 108560, 108561, 108562, 108566, 108568, 108569, 108570, 108576, 108577, 108578, 108582, 108583, 108584, 108585, 108586, 108587, 108593, 108599, 108600, 108601, 108602, 108607, 108608, 108609, 108610, 108611, 108617, 108618, 108619, 108623, 108624, 108625, 108626, 108632, 108633, 108634, 108635, 108640, 108641, 108642, 108645, 108648, 108649, 108650, 108657, 108658, 108659, 108664, 108666, 108667, 108668, 108673, 108674, 108676, 108679, 108680, 108681, 108684, 108689, 108690, 108691, 108692, 108698, 108700, 108705, 108707, 108708, 108714, 108715, 108718, 108719, 108724, 108725, 108728, 108730, 108736, 108738, 108739, 108740, 108741, 108746, 108749, 108750, 108756, 108757, 108763, 108764, 108770, 108771, 108772, 108779, 108780, 108781, 108787, 108788, 108791, 108795, 108796, 108797, 108803, 108804, 108805, 108813, 108820, 108821, 108830, 108834, 108835, 108836, 108838, 108846, 108850, 108855, 108861, 108862, 108869, 108870, 108880, 108889, 108894, 108910, 108911, 108912, 108925, 108933, 108934, 108937, 108943, 108946, 108958, 108959, 108963, 108966, 108968, 108978, 108991, 109002, 109004, 109018, 109082, 109091, 109160, 109187, 109271, 109467.

All the 219 codes are of type $4^{13}2^6$. In order to save some space, only the matrix $[X|Y]$ for the first twenty codes are listed. The remaining codes are available from the author.

TABLE V: 20 EXTREMAL CODES FROM BUILDING-UP

METHOD FOR $N = 32$

Index	$[X\|Y]$	E_{16}
1	$\begin{bmatrix} 2\ 0\ 0\ 0\ 1\ 0\ 1\ 2\ 0\ 1\ 0\ 2\ 1\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 2\ 0\ 2\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ 0 \\ 0\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 2\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 2\ 0\ 1 \\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ 2\ 0\ 0\ 2\ 0\ 1\ 2\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 2\ 1\ 0\ 0\ 2\ 1\ 0\ 0\ 2 \\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 2\ 0\ 1\ 2\ 0\ 0\ 2\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 2\ 0 \end{bmatrix}$	108251
2	$\begin{bmatrix} 0\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ 1\ 2\ 0\ 2\ 2\ 0\ 1\ 1\ 2\ 0\ 0\ 2\ 1\ 1\ 1\ 0 \\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 2\ 1\ 0\ 1\ 0\ 2\ 0\ 0\ 0\ 0\ 0\ 1 \\ 0\ 1\ 0\ 1\ 1\ 2\ 0\ 0\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 2\ 2\ 0\ 2\ 2\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 2 \\ 1\ 1\ 2\ 1\ 0\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 0\ 2\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 2\ 0\ 0\ 0\ 0\ 2\ 1\ 0 \end{bmatrix}$	108301
3	$\begin{bmatrix} 0\ 1\ 1\ 2\ 1\ 2\ 0\ 2\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 2\ 0\ 1\ 1\ 1\ 1\ 0\ 1\ 1\ 0\ 0\ 2\ 0\ 1\ 0 \\ 0\ 0\ 1\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 2\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 1\ 1\ 2\ 0\ 1\ 0\ 0 \\ 1\ 2\ 0\ 0\ 1\ 0\ 0\ 2\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 2\ 1\ 1\ 1\ 0\ 1\ 2\ 0\ 2\ 0\ 0\ 2\ 0\ 0 \\ 1\ 0\ 1\ 2\ 1\ 0\ 0\ 0\ 0\ 1\ 2\ 0\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 2\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 2 \end{bmatrix}$	108307
4	$\begin{bmatrix} 0\ 2\ 1\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 1\ 0\ 1\ 2\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 2\ 2\ 1\ 2\ 0\ 1\ 1 \\ 1\ 1\ 1\ 1\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 2\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 2\ 0\ 1\ 1\ 0\ 0 \\ 0\ 2\ 0\ 2\ 0\ 0\ 1\ 0\ 0\ 1\ 1\ 2\ 0\ 0\ 0\ 0\ 1\ 0\ 2\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 2\ 0\ 1\ 1\ 2 \\ 0\ 0\ 1\ 0\ 2\ 0\ 0\ 2\ 1\ 1\ 1\ 0\ 2\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 2\ 0\ 0\ 0\ 1\ 0\ 0 \end{bmatrix}$	108325
5	$\begin{bmatrix} 0\ 0\ 1\ 2\ 1\ 0\ 2\ 0\ 0\ 0\ 1\ 2\ 0\ 1\ 1\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 2\ 1\ 2\ 0\ 1\ 0\ 0 \\ 1\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 2\ 1\ 0\ 2\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 0\ 1\ 1 \\ 0\ 2\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 2\ 1\ 0\ 2\ 2\ 1\ 2\ 0\ 0\ 2\ 1\ 0\ 1 \\ 0\ 0\ 2\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 2\ 0\ 1\ 0\ 0\ 0\ 1\ 2\ 0\ 0\ 1\ 0\ 0\ 2\ 0\ 1\ 0\ 1 \end{bmatrix}$	108331
6	$\begin{bmatrix} 0\ 2\ 1\ 0\ 1\ 0\ 1\ 0\ 2\ 1\ 1\ 0\ 1\ 0\ 1\ 2\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 2\ 2\ 1\ 0\ 0\ 0\ 1 \\ 1\ 1\ 1\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 2\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 2\ 0\ 0\ 1\ 0\ 0 \\ 0\ 2\ 0\ 2\ 0\ 0\ 1\ 0\ 0\ 1\ 1\ 2\ 0\ 0\ 2\ 0\ 1\ 0\ 2\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 1\ 2 \\ 0\ 0\ 1\ 0\ 2\ 0\ 0\ 2\ 0\ 1\ 1\ 0\ 2\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 2\ 0\ 1\ 1\ 1\ 0\ 0 \end{bmatrix}$	108333
7	$\begin{bmatrix} 1\ 0\ 1\ 0\ 1\ 1\ 0\ 2\ 1\ 2\ 2\ 0\ 1\ 0\ 0\ 0\ 0\ 2\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 1\ 1\ 1\ 2\ 0\ 1 \\ 0\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 1\ 2\ 1\ 0\ 1\ 1\ 1\ 2\ 1\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0 \\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 2\ 0\ 2\ 0\ 0\ 2\ 0\ 0\ 0\ 1\ 1\ 0\ 1\ 1\ 0\ 2\ 0\ 2\ 2\ 0\ 0\ 1\ 1 \\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 2\ 2\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 2\ 0\ 0\ 2\ 0\ 0\ 0 \end{bmatrix}$	108356
8	$\begin{bmatrix} 0\ 2\ 0\ 1\ 1\ 0\ 2\ 0\ 1\ 1\ 2\ 0\ 1\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 2\ 1\ 0\ 2\ 0\ 0\ 0\ 0\ 1\ 1 \\ 1\ 0\ 1\ 0\ 0\ 1\ 2\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 1\ 1\ 2\ 1\ 0\ 1\ 1 \\ 2\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 2\ 0\ 0\ 2\ 1\ 0\ 2\ 0\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 2\ 2\ 0\ 0\ 0\ 1\ 0\ 1 \\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 2\ 0\ 2\ 2\ 0\ 1\ 2\ 0\ 1\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 1 \end{bmatrix}$	108357
9	$\begin{bmatrix} 0\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 2\ 1\ 0\ 2\ 2\ 1\ 0\ 1\ 2\ 1\ 0\ 2\ 0\ 0\ 1\ 1\ 0\ 0\ 1\ 0 \\ 1\ 0\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 1\ 1\ 2\ 1\ 0\ 1\ 0\ 2\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 1\ 0 \\ 0\ 1\ 0\ 1\ 2\ 2\ 2\ 1\ 1\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 2\ 0\ 0\ 0\ 2\ 1\ 1\ 0\ 2\ 0\ 0\ 0 \\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 2\ 0\ 0\ 0\ 2\ 0\ 1\ 0\ 0\ 0\ 1\ 1\ 2\ 0\ 2\ 1\ 1 \end{bmatrix}$	108372
10	$\begin{bmatrix} 0\ 1\ 0\ 1\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ 1\ 2\ 0\ 2\ 2\ 0\ 1\ 1\ 2\ 0\ 0\ 2\ 1\ 1\ 0\ 0 \\ 1\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 2\ 1\ 0\ 1\ 0\ 2\ 0\ 0\ 0\ 0\ 1\ 1 \\ 0\ 0\ 0\ 1\ 1\ 2\ 0\ 0\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 2\ 2\ 0\ 2\ 2\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 2 \\ 1\ 1\ 2\ 1\ 0\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 0\ 2\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 2\ 0\ 0\ 0\ 0\ 2\ 1\ 0 \end{bmatrix}$	108374

TABLE V: 20 EXTREMAL CODES FROM BUILDING-UP

METHOD FOR $N = 32$ (con't)

Index	$[X\|Y]$	E_{16}
11	1 1 0 0 0 0 1 1 2 0 0 2 0 0 0 1 0 1 0 1 0 1 2 2 1 0 1 1 0 1 2 0 0 0 1 1 2 1 0 0 0 0 0 2 1 1 1 1 1 0 0 0 1 0 1 1 1 1 0 1 0 1 0 1 0 0 1 0 0 0 2 0 0 0 1 0 1 0 2 1 2 2 1 1 0 1 2 0 1 2 0 0 0 0 0 0 2 0 1 1 0 0 0 2 0 2 0 0 0 0 0 1 0 0 1 1 0 0 0 0 0 1 0 1 1 2 0	108382
12	1 0 1 1 1 1 2 1 0 0 0 1 0 0 1 0 1 1 2 0 2 0 0 0 2 0 0 1 1 2 0 0 0 1 1 0 1 0 0 1 0 0 0 0 1 1 0 1 1 0 1 1 1 2 1 0 0 1 1 0 0 2 1 1 0 0 0 2 1 1 0 0 0 0 1 1 1 2 0 2 1 2 0 1 2 0 2 1 0 0 0 0 0 0 0 0 2 0 0 0 0 0 2 1 1 2 1 1 1 0 0 0 1 0 0 0 0 0 0 0 0 1 0 1 2 0 0 0	108387
13	1 1 1 2 1 1 1 0 0 0 0 1 1 0 0 2 0 2 0 1 0 0 1 0 2 0 1 0 0 2 0 1 0 0 0 1 0 1 0 2 1 1 1 0 1 0 1 0 1 1 0 1 1 1 1 1 0 1 0 1 0 2 0 0 0 0 1 2 2 1 1 0 0 0 2 0 1 1 2 0 0 0 1 0 0 0 0 1 0 2 0 1 0 0 0 2 2 0 1 0 0 1 0 0 1 0 0 2 0 0 0 0 0 0 1 0 0 0 1 1 2 0 1 0 2 0 1 0	108388
14	0 1 0 0 1 1 0 1 1 0 2 1 1 0 1 0 2 2 0 2 1 2 1 0 0 1 0 0 0 0 1 0 1 1 0 0 0 0 0 0 0 1 1 0 1 1 0 1 2 0 1 0 0 1 0 2 1 1 0 1 1 1 1 1 0 1 0 0 0 2 1 0 1 1 2 0 0 0 1 2 0 0 2 0 2 0 0 0 0 1 1 0 0 2 0 1 0 0 1 2 2 0 1 1 1 1 0 0 1 0 0 0 0 0 0 2 0 0 2 0 0 1 0 0 1 0 0 0	108389
15	0 1 0 1 0 0 0 0 1 1 2 2 1 2 1 1 1 1 0 1 2 1 0 0 0 0 0 0 0 0 1 2 1 0 1 1 1 0 1 1 1 0 1 2 1 0 0 0 0 0 1 1 0 0 1 0 1 0 0 2 1 1 0 1 0 0 2 1 1 1 2 0 0 2 2 0 1 0 0 0 1 2 2 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 1 2 2 1 1 0 0 1 0 0 1 0 0 2 1 0 0 1 2 0	108405
16	0 1 0 1 1 0 2 0 1 0 0 1 1 2 1 1 1 0 1 2 2 0 0 0 0 0 2 0 1 1 0 0 1 0 1 1 0 0 0 0 0 1 1 1 0 1 1 0 0 0 0 1 0 1 2 1 1 1 2 1 1 0 0 1 0 1 0 1 2 0 0 0 2 1 0 0 0 0 0 0 1 1 0 2 0 2 0 2 1 0 0 2 1 0 1 0 0 1 1 0 0 1 2 2 0 1 0 0 1 0 1 2 0 1 2 0 0 0 0 0 0 0 0 0 1 0 0 0	108406
17	1 0 0 0 1 0 1 1 1 1 0 0 0 0 2 2 1 2 1 0 1 0 2 0 1 1 1 2 0 0 0 0 1 1 0 0 0 1 0 1 0 0 1 2 1 0 1 1 1 0 1 1 0 1 2 1 0 0 0 0 1 0 1 1 1 0 1 0 1 2 2 0 0 0 2 0 0 0 2 0 0 0 1 0 0 1 0 1 2 0 1 0 2 1 0 0 0 0 0 2 1 0 0 0 2 2 0 0 1 1 0 0 1 0 1 0 0 1 0 0 0 1 0 2 0 1 0 0	108408
18	0 2 1 0 1 1 0 0 1 0 1 0 2 0 1 0 0 2 2 0 0 1 1 0 2 0 1 1 1 0 1 0 1 1 1 1 0 0 1 1 0 0 0 1 0 1 1 1 1 2 1 1 2 0 1 1 0 0 0 0 1 0 0 0 1 0 1 2 2 0 0 0 1 1 0 0 0 2 0 0 2 0 2 1 0 0 1 0 0 0 1 0 0 0 2 1 0 0 1 0 0 1 0 1 0 1 0 0 2 0 0 0 0 0 0 1 0 2 0 0 0 1 1 2 1 2 0 0	108414
19	1 0 0 1 2 0 0 1 0 1 0 1 0 1 1 0 2 2 0 1 0 1 0 0 1 2 1 0 1 0 0 2 0 1 0 0 2 2 0 0 1 0 1 0 1 0 1 1 0 1 0 0 1 1 0 1 1 1 0 1 1 1 1 0 0 1 0 1 0 0 1 0 0 0 0 2 2 1 1 0 0 0 0 2 0 0 1 2 1 2 0 2 0 0 1 0 0 0 1 1 0 0 1 2 0 1 0 0 0 0 1 0 0 0 2 0 1 1 0 0 0 0 2 0 0 0 1 2	108415
20	0 1 0 1 2 1 0 0 1 0 0 1 1 1 1 0 0 2 0 1 0 1 2 0 2 1 0 0 0 1 2 0 1 0 0 0 1 0 1 0 1 0 1 0 1 1 1 1 2 1 1 0 1 0 0 1 2 0 1 0 1 0 0 1 2 0 1 1 2 1 0 0 0 0 2 0 1 1 0 0 0 0 0 0 2 2 0 1 0 2 0 1 0 0 0 1 0 0 1 1 0 0 0 1 0 2 0 2 0 1 1 1 0 0 0 2 0 0 2 1 0 0 0 0 0 1 0 0	108422

3.3 New Codes For $n = 40$

We let C be the extremal Type II code of length 32 and type $4^{13}2^{6}$ with the following generator matrix

$$G = \begin{bmatrix}
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&3&3&1&1&3&1&3&3&3&0&0&2&1\\
0&1&0&0&0&0&0&0&0&0&0&0&0&1&0&0&1&0&1&1&2&0&1&1&3&1&1&1&2&2&0&3\\
0&0&1&0&0&0&0&0&0&0&0&0&0&1&0&1&0&1&0&0&2&2&0&3&2&2&3&3&2&1&2&0\\
0&0&0&1&0&0&0&0&0&0&0&0&0&1&1&1&1&1&0&3&0&3&0&3&3&1&0&3&1&1&3&1\\
0&0&0&0&1&0&0&0&0&0&0&0&0&0&1&1&0&1&1&0&3&1&3&1&0&1&0&1&2&2&2&1\\
0&0&0&0&0&1&0&0&0&0&0&0&0&1&1&0&0&0&1&0&0&2&2&3&0&3&0&0&3&0&3&0\\
0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&1&0&0&1&1&3&2&2&2&0&0&1&0&0&2&3&1\\
0&0&0&0&0&0&0&1&0&0&0&0&0&0&1&0&0&0&1&2&3&2&2&3&3&0&1&2&2&0&1&2\\
0&0&0&0&0&0&0&0&1&0&0&0&0&0&1&1&1&1&0&3&3&3&0&1&3&0&2&3&2&2&0&1\\
0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&1&1&1&0&3&2&0&1&1&0&1&2&2&3&1&1&3\\
0&0&0&0&0&0&0&0&0&0&1&0&0&1&0&1&1&1&0&1&1&2&1&3&3&2&3&3&0&2&2&2\\
0&0&0&0&0&0&0&0&0&0&0&1&0&1&1&1&1&1&1&3&2&0&1&2&1&3&3&2&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1&1&1&1&3&1&1&3&3&2&0&0&0&2&0&2&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&2&0&0&0&0&0&2&2&0&0&2&0&2&2&2&0&0&2&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&2&0&0&0&0&0&0&0&2&0&0&2&2&0&0&2&0&2\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&2&0&0&0&2&0&0&2&2&2&2&2&0&2&2&2&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&2&0&0&2&0&2&0&2&2&2&0&0&0&2&0&2\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&2&0&2&0&2&2&0&0&0&2&0&2&0&2&2\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&2&0&0&2&2&2&2&2&0&2&0&0&2&0
\end{bmatrix}.$$

A pseudo-random search was performed on the matrix $[X|Y]$. At least 133 extremal inequivalent codes are found. The number of codewords with Euclidean weight 16 of these codes E_{16} are:

32569, 32628, 34206, 34327, 34349, 34357, 34365, 34391, 34395, 34399, 34416, 34419, 34425, 34429, 34441, 34467, 34469, 34476, 34490, 34497, 34498, 34501, 34503, 34508, 34509, 34514, 34516, 34522, 34531, 34538, 34539, 34542, 34547, 34548, 34549, 34553, 34558, 34559, 34561, 34569, 34570, 34573, 34579, 34580, 34581, 34583, 34584, 34585, 34589, 34595, 34597, 34606, 34607, 34608, 34609, 34615, 34616, 34617, 34620, 34621, 34624, 34630, 34631, 34641, 34642, 34644, 34647, 34648, 34650, 34653, 34654, 34660, 34662, 34663, 34666, 34669, 34670, 34673, 34674, 34675, 34679, 34681, 34684, 34685, 34696, 34698, 34700, 34702, 34703, 34714, 34718, 34723, 34724, 34730, 34732, 34739, 34740, 34742, 34751, 34755, 34757, 34758, 34759, 34762, 34766, 34771, 34779, 34791, 34795, 34796, 34797, 34803, 34807, 34809, 34814, 34822, 34824, 34832, 34833, 34838, 34843, 34850, 34856, 34860, 34872, 34883, 34923, 34928, 34934, 34938, 34941, 34950, 34952.

All the 133 codes are of type $4^{17}2^6$. In order to save some space, only the matrix $[X|Y]$ for the first twenty codes are listed. The remaining codes are available from the author.

TABLE VI: 20 EXTREMAL CODES FROM BUILDING-UP METHOD FOR $N = 40$

Index	$[X\|Y]$	E_{16}
1	2 1 0 0 2 0 0 1 1 1 0 1 1 0 2 2 0 1 1 1 1 1 0 2 0 1 0 0 1 0 1 0 0 1 0 1 0 0 2 0 0 0 1 1 1 0 1 0 1 0 1 1 1 0 0 1 0 0 0 0 3 1 1 3 0 0 2 1 2 1 0 1 1 1 1 0 1 0 2 0 0 0 0 0 0 1 0 2 0 2 0 0 1 1 0 2 0 1 0 0 1 1 0 3 1 0 0 2 3 2 0 1 1 2 0 1 2 0 0 0 0 1 0 1 0 0 0 0 0 0 0 1 1 0 2 0 1 1 2 2 2 0 0 3 1 0 0 0 1 0 0 0 1 3 0 0 0 2 0 1	32569
2	0 0 0 1 1 1 0 2 1 1 1 1 1 0 2 2 0 2 0 0 1 1 1 2 0 1 0 0 1 1 1 0 0 0 0 1 0 0 2 0 1 0 1 1 0 0 0 0 0 0 0 1 1 1 0 1 0 1 1 1 3 1 1 3 1 0 1 1 2 0 0 2 1 1 1 0 0 0 2 0 0 1 0 0 1 0 1 0 0 2 2 0 1 0 0 2 0 0 0 2 1 1 2 3 1 0 2 2 3 0 0 0 1 0 0 1 1 0 0 0 0 0 0 0 1 1 1 0 2 0 0 1 1 0 2 0 1 0 1 0 2 0 3 3 1 0 0 0 1 2 0 0 0 0 0 0 0 2 0 1	32628
3	0 1 0 1 1 0 0 2 0 0 0 0 1 1 0 1 2 1 2 1 1 0 0 1 1 1 0 0 0 1 0 2 2 0 1 0 2 1 1 0 0 0 1 0 1 1 0 0 0 1 1 1 0 0 0 0 2 0 0 1 1 1 1 0 1 0 1 1 0 1 2 1 1 1 3 0 3 0 2 1 1 0 0 0 0 0 0 0 0 0 0 0 2 0 0 1 0 0 0 2 0 1 2 0 1 2 2 2 1 1 0 2 0 1 1 1 3 1 3 0 0 2 0 2 1 0 2 2 1 0 1 0 0 0 1 1 0 1 0 3 0 0 0 0 0 0 0 0 1 0 0 0 1 2 1 3 0 1 0	34206
4	2 1 0 0 2 0 0 1 1 1 0 2 1 0 1 2 0 1 1 1 1 1 0 2 0 1 0 0 1 0 1 0 0 1 0 1 0 0 2 0 0 0 1 1 1 0 1 0 1 0 1 0 1 0 1 1 0 0 0 0 3 1 1 3 0 0 2 1 2 1 0 1 1 1 1 0 1 0 2 0 0 0 0 0 0 1 0 2 0 2 0 0 1 1 0 2 0 1 0 0 1 1 0 3 1 0 0 2 3 2 0 1 1 2 0 1 2 0 0 0 0 1 0 1 0 0 0 0 0 0 0 2 1 0 1 0 1 1 2 2 2 0 0 3 1 0 0 0 1 0 0 0 1 3 0 0 0 2 0 1	34327
5	0 2 0 2 1 2 0 0 1 1 1 2 0 1 1 1 0 0 0 0 0 0 0 2 1 0 0 1 1 0 1 0 2 1 1 1 1 1 0 0 1 2 0 0 0 1 1 0 1 0 0 3 1 0 0 2 1 2 0 1 1 1 1 0 0 0 1 1 0 1 3 1 1 1 1 0 1 0 0 0 0 0 1 0 1 2 2 1 0 0 0 3 0 2 0 3 0 0 0 2 0 1 2 0 0 0 0 1 2 0 1 1 0 2 1 1 0 0 0 1 1 0 0 2 0 0 0 0 1 2 2 3 0 0 1 1 0 0 1 0 0 1 0 0 0 1 0 1 0 0 2 0 0 3 0 1 0 0 2 1	34349
6	0 1 0 1 1 0 0 2 0 0 0 0 1 1 0 1 2 0 2 1 1 1 0 1 1 1 0 0 0 1 0 2 2 0 1 0 2 1 1 0 0 0 1 0 1 1 0 0 0 1 1 1 0 0 0 0 2 1 0 1 1 0 1 0 1 0 1 1 0 1 2 1 1 1 3 0 3 0 2 1 1 0 0 0 0 0 0 0 0 0 0 0 2 0 0 1 0 1 0 2 0 0 2 0 1 2 2 2 1 1 0 2 0 1 1 1 3 1 3 0 0 2 0 2 1 0 2 2 1 0 1 0 0 0 1 1 0 0 0 3 0 1 0 0 0 0 0 0 0 1 0 0 0 1 2 1 3 0 1 0	34357
7	1 0 1 1 1 0 0 0 2 1 1 1 0 0 0 0 2 2 0 0 1 1 0 1 0 1 0 2 1 1 1 0 2 2 1 0 0 0 1 0 0 1 0 0 0 0 0 0 3 0 1 1 1 1 1 0 1 2 1 1 1 0 1 2 1 3 1 0 1 0 0 0 1 0 1 1 2 0 0 1 0 2 0 0 0 1 0 0 3 1 1 2 0 0 0 1 2 0 0 1 0 1 0 3 0 1 2 0 1 2 0 1 0 0 0 2 0 0 2 1 2 0 1 0 2 1 1 1 3 1 0 3 0 0 0 0 0 0 0 1 0 0 0 1 1 2 0 0 1 0 0 0 0 2 1 0 0 2 0 0	34365
8	1 0 1 1 1 0 2 1 2 0 0 0 0 0 1 2 1 2 0 0 2 0 0 0 0 1 1 1 0 1 2 0 0 1 0 0 1 1 1 1 0 0 1 0 1 1 3 0 1 2 1 0 1 0 0 2 1 0 1 1 1 1 1 1 1 0 0 0 1 0 0 0 1 2 0 0 1 0 1 3 1 1 0 2 1 0 3 2 0 0 0 1 1 1 0 0 1 0 1 2 2 0 2 0 2 0 0 1 0 0 0 0 0 3 0 0 0 0 2 1 0 1 1 0 0 0 3 0 0 0 1 0 0 0 2 0 1 2 1 0 0 0 0 0 0 1 2 1 0 0 0 2 0 1 1 1 0 0 3 2	34391
9	1 1 1 1 2 1 1 0 0 0 1 1 0 2 0 0 1 0 0 0 1 0 0 1 1 0 1 2 2 0 2 1 0 1 0 0 1 0 2 0 1 0 0 1 3 1 1 0 1 0 1 0 1 1 0 1 3 1 0 0 0 1 0 2 0 1 0 0 1 1 0 0 1 0 1 2 0 1 2 1 1 0 0 0 3 2 1 1 0 0 0 2 2 0 1 0 1 0 0 0 1 2 1 3 0 1 0 0 2 0 0 1 1 2 0 0 0 2 0 0 1 0 1 1 3 3 0 0 0 1 0 0 0 0 0 1 2 0 2 1 0 0 1 1 2 1 0 2 0 0 0 1 0 0 0 0 2 0 0 0	34395
10	2 1 2 1 1 2 0 1 0 1 0 1 1 0 0 1 0 0 0 1 1 1 0 0 0 0 0 0 2 1 2 0 0 1 1 1 1 2 0 0 2 1 0 0 2 3 0 1 1 1 0 0 3 1 1 1 2 1 1 0 0 0 0 1 0 0 1 1 0 0 1 1 1 0 1 0 0 1 0 1 0 2 0 2 3 3 0 1 2 1 0 0 1 0 0 0 0 2 2 0 1 0 0 0 1 1 0 0 0 2 2 0 1 1 0 0 0 0 1 1 0 3 0 0 1 3 1 1 0 0 2 2 2 0 0 1 0 0 0 1 1 0 1 1 1 0 0 0 2 0 0 0 0 0 0 0 2 0 0 1	34399

TABLE VI: 20 EXTREMAL CODES FROM BUILDING-UP METHOD FOR $N = 40$ (con't)

Index	$[X\|Y]$	E_{16}
11	1 0 1 0 1 1 0 0 1 0 1 2 0 1 0 0 0 2 0 0 1 0 2 0 2 1 2 1 1 0 0 0 0 1 0 1 2 1 1 1 1 1 0 1 1 0 1 0 0 1 1 3 1 0 0 0 1 0 0 1 0 1 2 0 1 1 0 0 0 1 2 1 0 0 1 0 1 1 2 3 2 0 0 0 1 0 0 1 0 1 0 3 2 2 1 0 1 0 0 2 1 0 0 1 0 0 0 0 0 0 0 2 0 2 0 1 2 1 3 1 3 0 1 0 0 2 0 0 0 0 0 3 0 0 0 1 1 0 2 0 0 0 0 1 0 1 2 0 2 0 0 0 1 0 1 1 0 1 1 2	34416
12	1 1 1 1 1 0 0 0 2 0 0 1 0 0 0 1 2 2 0 0 1 1 0 1 0 1 0 2 1 1 1 0 2 2 1 0 0 0 1 0 0 1 0 0 0 0 0 0 3 0 1 1 1 1 1 0 1 2 1 1 1 0 1 2 1 3 1 0 1 0 0 0 1 0 1 1 2 0 0 1 0 1 0 0 0 1 0 0 3 1 2 2 0 0 0 1 2 0 0 1 0 1 0 3 0 1 2 0 1 2 0 1 0 0 0 2 0 0 2 1 2 0 1 0 2 1 1 1 3 0 0 3 0 0 0 1 0 0 0 1 0 0 0 1 1 2 0 0 1 0 0 0 0 2 1 0 0 2 0 0	34419
13	1 1 1 1 1 0 0 0 2 1 0 1 0 0 0 0 2 2 0 0 1 1 0 1 0 1 0 2 1 1 1 0 2 2 1 0 0 0 1 0 0 1 0 0 0 0 0 0 3 0 1 1 1 1 1 0 1 2 1 1 1 0 1 2 1 3 1 0 1 0 0 0 1 0 1 1 2 0 0 1 0 1 0 0 0 1 0 0 3 1 2 2 0 0 0 1 2 0 0 1 0 1 0 3 0 1 2 0 1 2 0 1 0 0 0 2 0 0 2 1 2 0 1 0 2 1 1 1 3 1 0 3 0 0 0 0 0 0 0 1 0 0 0 1 1 2 0 0 1 0 0 0 0 2 1 0 0 2 0 0	34425
14	1 2 0 1 1 0 0 2 1 2 2 1 1 0 1 0 0 2 0 1 1 0 1 1 1 1 0 0 0 1 0 0 1 0 0 0 1 0 2 0 0 0 0 1 0 1 0 0 1 3 1 0 1 1 0 1 1 1 1 2 1 0 3 0 0 0 1 1 1 0 1 2 1 0 1 1 0 0 2 0 0 0 0 0 1 1 1 0 0 3 2 0 1 0 0 0 0 0 3 2 1 1 2 2 1 2 2 2 0 0 0 1 1 0 1 0 0 0 0 0 0 2 1 0 1 0 1 0 1 3 0 2 1 0 0 0 0 0 1 1 3 0 2 0 0 0 0 0 0 2 0 0 0 0 0 1 1 2 0 1	34429
15	1 1 0 0 0 1 2 1 1 2 1 0 0 0 2 0 1 1 0 1 1 0 0 0 1 2 1 0 2 1 2 0 0 0 0 1 1 0 0 1 0 0 0 1 1 0 3 0 0 0 1 0 0 2 2 1 0 1 1 0 1 1 1 1 0 1 3 1 0 0 1 1 0 0 1 1 1 1 0 2 0 2 1 0 0 0 3 2 0 0 1 1 0 0 0 0 1 1 1 0 0 2 0 1 1 0 1 2 0 0 2 0 0 0 2 2 0 0 1 3 0 0 0 1 0 2 3 0 2 0 1 0 1 0 0 0 1 0 1 0 1 0 0 0 0 0 2 0 2 1 0 0 1 2 0 3 0 0 1 1	34441
16	0 0 0 1 1 1 0 2 1 1 1 1 1 0 2 1 0 2 0 0 1 0 1 2 0 1 2 0 1 1 1 0 0 0 0 1 0 0 2 0 1 0 1 1 0 0 0 0 1 0 0 0 1 1 0 1 0 1 1 1 3 1 1 3 1 0 1 1 2 0 0 2 1 1 1 0 0 0 2 0 0 1 0 0 1 0 1 0 0 2 2 0 1 0 0 1 0 0 0 2 1 2 2 3 1 1 2 2 3 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 1 1 1 0 1 0 0 2 1 0 2 0 1 0 1 0 2 0 3 3 1 0 0 0 1 2 0 0 0 0 0 0 0 2 0 1	34467
17	0 0 1 1 2 2 1 0 0 0 0 1 2 2 0 1 2 1 1 1 1 0 0 1 0 0 0 2 1 1 0 0 0 0 1 1 0 1 1 0 1 1 3 0 0 1 1 1 1 0 1 0 0 3 0 1 2 1 1 0 0 2 0 0 1 0 0 1 0 1 1 1 1 1 0 0 0 0 2 1 0 1 1 0 0 2 1 2 0 0 1 0 0 3 1 2 0 0 1 0 2 0 0 0 0 1 0 0 0 0 0 0 2 2 2 1 1 1 3 0 1 1 2 2 2 0 1 0 0 1 0 0 0 3 0 3 0 1 0 2 0 0 1 0 0 1 2 0 1 0 0 0 0 0 0 1 0 0 1 0	34469
18	1 1 1 1 2 1 1 0 0 0 0 1 0 2 0 0 0 0 0 0 1 0 0 1 1 0 1 2 2 0 2 1 0 1 0 0 1 1 2 1 1 0 0 1 3 1 1 0 1 0 1 0 1 1 0 1 1 1 0 0 0 1 0 2 0 1 0 0 1 1 0 0 1 0 1 2 0 1 2 3 1 0 0 0 3 2 1 1 0 0 2 2 2 0 1 0 0 0 0 0 1 2 1 3 0 1 0 0 2 0 0 1 1 2 0 0 0 0 0 1 1 0 1 1 3 3 0 0 0 1 0 0 0 0 0 1 0 0 2 1 0 0 1 1 2 1 0 2 0 0 0 1 0 0 0 0 2 0 0 2	34476
19	0 0 1 0 0 1 0 0 2 0 0 1 1 0 0 1 2 1 0 1 0 1 1 1 0 1 0 2 1 1 1 0 2 2 0 2 1 0 1 0 1 0 0 0 1 0 0 0 3 1 1 3 0 1 1 1 1 0 1 0 2 0 1 2 1 1 1 0 1 0 0 0 1 0 1 2 1 0 0 1 2 1 0 1 1 0 0 0 3 0 2 1 0 0 0 0 2 2 0 0 0 1 1 3 0 2 2 0 1 1 0 1 0 0 0 0 0 0 2 1 0 0 1 1 1 0 1 1 3 0 0 2 2 0 0 1 0 0 0 2 0 0 0 1 1 3 0 0 1 1 0 0 0 2 0 0 0 2 0 0	34490
20	0 1 0 1 1 0 0 2 0 0 0 1 1 0 0 0 2 1 2 1 1 0 0 1 1 1 1 0 0 1 0 2 2 0 1 0 2 1 1 0 0 0 1 0 1 1 0 0 0 1 1 0 0 1 0 1 2 0 0 1 1 1 1 0 1 0 0 1 0 1 2 1 1 1 3 0 3 0 2 1 1 0 0 0 0 0 0 0 0 0 0 0 2 0 0 2 0 0 0 2 0 1 2 0 1 2 1 2 1 1 0 2 0 1 1 1 3 1 3 0 0 2 0 2 1 0 2 2 1 0 1 0 0 0 1 0 0 1 0 3 0 0 0 0 0 0 1 0 0 1 0 0 0 1 2 1 3 0 1 0	34497

3.4 Conclusion

The results above can be summarized as follows.

Corollary 13. 219 extremal Type II codes of length 32 and type $4^{13}2^{6}$, and 133 extremal Type II codes of length 40 and type $4^{17}2^{6}$ are found with the generalized building-up method.

CHAPTER 4

GENERALIZATION ON HARADA'S CONSTRUCTION METHOD OF EXTREMAL TYPE II $\mathbb{Z}_4$-LINEAR CODES OF LENGTHS 32 AND 40

M. Harada has developed a construction method of Type II codes from a given code in (3). With the use of Harada's method, P. Gaborit and M. Harada have found 50 and 20 inequivalent extremal Type II codes of lengths 32 and 40 over $\mathbb{Z}_4$ respectively in (12). The method given by Harada is only applicable on codes with a generator matrix of the form $[I|A]$. In this chapter, we are going to generalize the method. The generalized method can be applied on all Type II codes. 17 and 102 new extremal Type II codes of length 32 and 40 are constructed with this generalized method.

4.1 Harada's Original Method

Theorem 14. Let C be a Type II code of length $n = 8k$. Suppose $G = [I|Y]$ is a generator matrix of C. Let $\mathbf{y}_i$ be the i-th row of Y, which is of length $4k$. Suppose that S is a subset of $\{1, 2, 3, \ldots, 4k\}$ where $\mathbf{t}$ is a $(0, 1)$-vectors of length $4k$ such that the j-th coordinate of $\mathbf{t}$ is 1 if and only if $j \in S$. Let $\mathbf{1}$ be the all-one vector of length $4k$. Let

$$\mathbf{y}_i' = \begin{cases} \mathbf{y}_i + 2\mathbf{t} & \text{if } \mathrm{wt}_E(\mathbf{y}_i + 2\mathbf{t}) \equiv 7 \pmod 8 \\ \mathbf{y}_i + 2\mathbf{t} + 2 \cdot \mathbf{1} & \text{otherwise} \end{cases}$$

Then $G' = [I|Y']$ is a generator matrix of a Type II code C'.

4.2 Our Generalized Method

A Type II code C' is generated from the following constructions:

Theorem 15. Let C be a Type II code of length n. Suppose $G = \begin{bmatrix} I & Y \\ O & 2B \end{bmatrix}$ is a generator matrix of C such that B is a $(0,1)$-matrix and the order of the identity matrix I is a. Let $\mathbf{y}_i$ be the i-th row of Y, which is of length $n-a$. Suppose that S is a subset of $\{1,2,3,\ldots,n-a\}$ where $\mathbf{t}$ is a $(0,1)$-vector of length $n-a$ such that the j-th coordinate of $\mathbf{t}$ is 1 if and only if $j \in S$. Let $\mathbf{1}$ be the all-one vector of length $n-a$. A Type II code C' is generated from the following constructions:

Case 1: $n-a$ is even.

Let

$$\mathbf{y}'_i = \begin{cases} \mathbf{y}_i + 2\mathbf{t} + 2\cdot\mathbf{1} & \text{if } \mathrm{wt}_E(\mathbf{y}_i + 2\mathbf{t}) \equiv 7 \pmod 8 \\ \mathbf{y}_i + 2\mathbf{t} + 2 & \text{otherwise} \end{cases}$$

be the i-th row of a matrix Y'. Then $G' = \begin{bmatrix} I & Y' \\ O & 2B \end{bmatrix}$ generates a Type II code C'.

Case 2: $n-a$ is odd.

Let s_j be the number of even coordinates in the j-th coordinate position among the rows of Y. Suppose $\sum_{j\in S} s_j$ is even. Let K be the set of indices i such that $\mathrm{wt}_E(\mathbf{y}_i + 2\mathbf{t}) \equiv 3$

(mod 8) and f be an arbitrary pairing of elements in K, i.e. $f : K \to K$ is a bijective function such that $f(f(i)) = i$ and $f(i) \neq i$ for all $i \in K$. Let

$$(\mathbf{h}_i, \mathbf{y}'_i) = \begin{cases} (0, \mathbf{y}_i + 2\mathbf{t}) & \text{if } \mathrm{wt}_E(\mathbf{y}_i + 2\mathbf{t}) \equiv 7 \pmod 8 \\ (2\mathbf{e}_{f(i)}, \mathbf{y}_i + 2\mathbf{t} + 2 \cdot \mathbf{1}) & \text{otherwise} \end{cases}$$

where $\mathbf{e}_i$ denotes the vector in the canonical basis of $\mathbb{Z}_4^a$ with the i-th coordinate being 1. Then $G' = \begin{bmatrix} I+H & Y' \\ O & 2B \end{bmatrix}$ generates a Type II code C'.

Proof. For the sake of simpler notion, we write $\mathrm{wt}_E(\mathbf{x})$ as $\|\mathbf{x}\|$ in the proof. We prove the two parts separately. When $n - a$ is even, the proof resembles that in Harada's work (3).

We first prove the self duality of C'. It suffices to show that C' is self-orthogonal. Since C is a Type II code, we have $Y'(2B)^T = [Y + \mathbf{1}^T(2\mathbf{t})](2B)^T = Y(2B)^T + \mathbf{1}^T \cdot 0 = 0$. Therefore, it is enough to check the orthogonality of $[I|Y']$.

It is clear that $\|\mathbf{y}_i + 2\mathbf{t}\| = 3$ or $7 \pmod 8$. Therefore, we can write

$$\mathbf{y}'_i = \mathbf{y}_i + 2\mathbf{t} + \frac{\|\mathbf{y}_i + 2\mathbf{t}\| + 1}{4}(2 \cdot \mathbf{1}).$$

Also, $\|\mathbf{y}_i + 2\mathbf{t}\| \equiv \langle \mathbf{y}_i + 2\mathbf{t}, \mathbf{y}_i + 2\mathbf{t} \rangle \equiv \|\mathbf{y}_i\| + 2\langle \mathbf{y}_i, 2\mathbf{t} \rangle + \|4\mathbf{t}\| \equiv 7 + 4\langle \mathbf{y}_i, \mathbf{t} \rangle + 4\|\mathbf{t}\| \pmod 8$, so we have

$$\mathbf{y}'_i = \mathbf{y}_i + 2\mathbf{t} + (\langle \mathbf{y}_i, \mathbf{t} \rangle + \|\mathbf{t}\|)(2 \cdot \mathbf{1}) \pmod 4.$$

Hence

$$\begin{aligned}\langle \mathbf{y}_i', \mathbf{y}_j' \rangle &= \langle \mathbf{y}_i + 2\mathbf{t} + (\langle \mathbf{y}_i, \mathbf{t}\rangle + \|\mathbf{t}\|)(2\cdot\mathbf{1}), \mathbf{y}_j + 2\mathbf{t} + (\langle \mathbf{y}_j, \mathbf{t}\rangle + \|\mathbf{t}\|)(2\cdot\mathbf{1})\rangle \\ &\equiv \langle \mathbf{y}_i, \mathbf{y}_j\rangle + 2\langle \mathbf{y}_i, \mathbf{t}\rangle + 2\left(\langle \mathbf{y}_j, \mathbf{t}\rangle + \|\mathbf{t}\|\right)\left(\langle \mathbf{y}_i, \mathbf{1}\rangle\right) + 2\left(\langle \mathbf{y}_i, \mathbf{t}\rangle + \|\mathbf{t}\|\right)\left(\langle \mathbf{y}_j, \mathbf{1}\rangle\right) \\ &\equiv -\delta_{i,j} + 2\langle \mathbf{y}_i, \mathbf{t}\rangle + 2\langle \mathbf{y}_j, \mathbf{t}\rangle + 2\|\mathbf{t}\| + 2\langle \mathbf{y}_i, \mathbf{t}\rangle + 2\langle \mathbf{y}_j, \mathbf{t}\rangle + 2\|\mathbf{t}\| \\ &\equiv -\delta_{i,j} \pmod 4\end{aligned} \tag{4.1}$$

where $\delta_{a,b} = 1$ if $a = b$ and $\delta_{a,b} = 0$ if $a \neq b$. Therefore,

$$\langle (\mathbf{e}_i, \mathbf{y}_i'), (\mathbf{e}_j, \mathbf{y}_j')\rangle = \langle \mathbf{e}_i, \mathbf{e}_j\rangle + \langle \mathbf{y}_i', \mathbf{y}_j'\rangle \equiv \delta_{i,j} - \delta_{i,j} \equiv 0 \pmod 4.$$

Finally, we have to check if the Euclidean weight of $\mathbf{y}_i'$ is congruent to $7 \pmod 8$. It clearly holds when $\|\mathbf{y}_i + 2\mathbf{t}\| \equiv 7 \pmod 8$. When $\|\mathbf{y}_i + 2\mathbf{t}\| \equiv 3 \pmod 8$, we have

$$\begin{aligned}\|\mathbf{y}_i + 2\mathbf{t} + 2\cdot\mathbf{1}\| &= \|\mathbf{y}_i + 2\mathbf{t}\| + 2\langle \mathbf{y}_i + 2\mathbf{t}, 2\cdot\mathbf{1}\rangle + \|2\cdot\mathbf{1}\| \\ &\equiv 3 + 4\langle \mathbf{y}_i, \mathbf{1}\rangle + 4(n-a) \\ &\equiv 3 + 4 + 0 \\ &= 7 \pmod 8.\end{aligned}$$

Hence G' generates a Type II code C'.

When n is odd, we have to show that K has an even number of elements before proving any other arguments. For a given $\mathbf{y}_i$, let $n_{i,l}$ be the number of l's among its coordinates and $m_{i,l}$

be the number of l's among its coordinates in the coordinate positions in S. We note that the j-th coordinate in $\mathbf{y}_i + 2\mathbf{t}$ is 1 if and only if the j-th coordinate of $\mathbf{y}_i$ is 1 and $j \notin S$, or the j-th coordinate of $\mathbf{y}_i$ is 3 and $j \in S$. Therefore, $(n_{i,1} - m_{i,1}) + m_{i,3}$ coordinates of $\mathbf{y}_i + 2\mathbf{t}$ is 1. Similarly, the numbers of 2's and 3's in $\mathbf{y}_i + 2\mathbf{t}$ are $(n_{i,2} - m_{i,2}) + m_{i,0}$ and $(n_{i,3} - m_{i,3}) + m_{i,1}$.

$$\begin{aligned}
\|\mathbf{y}_i + 2\mathbf{t}\| &= (n_{i,1} - m_{i,1} + m_{i,3}) + 4(n_{i,2} - m_{i,2} + m_{i,0}) + (n_{i,3} - m_{i,3} + m_{i,1}) \\
&= n_{i,1} + 4n_{i,2} + n_{i,3} + 4(m_{i,0} + m_{i,2} - 2m_{i,2}) \\
&\equiv 7 + 4(m_{i,0} + m_{i,2}) \pmod 8.
\end{aligned}$$

The last line is due to the fact that $\|\mathbf{y}_i\| \equiv 7 \pmod 8$. In other words, $\|\mathbf{y}_i + 2\mathbf{t}\| \equiv 3 \pmod 8$ if and only if $m_{i,0} + m_{i,2}$ is odd. On the other hand, we see that $\sum_{i \in S} m_{i,0} + m_{i,2}$ is even as it is equal to the total number of even coordinates of $\mathbf{y}_i$'s among coordinate positions in S. Therefore, there are an even number of indices i such that $m_{i,0} + m_{i,2}$ is even, so K has an even number of elements and hence the pairing function f exists. We first show the self-orthogonality of $[I + J|Y']$. From (Equation 4.1), we see that $\langle \mathbf{y}'_i, \mathbf{y}'_j \rangle \equiv -\delta_{i,j} \pmod 4$. It suffices to show that $\langle \mathbf{e}_i + \mathbf{h}_i, \mathbf{e}_j + \mathbf{h}_j \rangle \equiv -\langle \mathbf{y}'_i, \mathbf{y}'_j \rangle \equiv \delta_{i,j} \pmod 4$.

If $i \notin K$ or $j \notin K$, we have

$$\begin{aligned}
\langle \mathbf{e}_i + \mathbf{h}_i, \mathbf{e}_j + \mathbf{h}_j \rangle &= \delta_{i,j} + \langle \mathbf{h}_i, \mathbf{e}_j \rangle + \langle \mathbf{e}_i, \mathbf{h}_j \rangle + \langle \mathbf{h}_i, \mathbf{h}_j \rangle \\
&\equiv \delta_{i,j} + 0 + 0 + 0 \\
&\equiv \delta_{i,j} \pmod 4.
\end{aligned}$$

The arguments above make use of the fact that $\langle \mathbf{h}_i, \mathbf{e}_j \rangle = \langle \mathbf{h}_i, \mathbf{h}_j \rangle = 0$ if $i \notin K$ or $j \notin K$. Here remains the situation when $i, j \in K$. Note that

$$\begin{aligned}
\langle \mathbf{e}_i + \mathbf{h}_i, \mathbf{e}_j + \mathbf{h}_j \rangle &= \langle \mathbf{e}_i + 2\mathbf{e}_{f(i)}, \mathbf{e}_j + 2\mathbf{e}_{f(j)} \rangle \\
&= \delta_{i,j} + 2\delta_{i,f(j)} + 2\delta_{j,f(i)} + 4\delta_{f(i),f(j)} \\
&\equiv \delta_{i,j} + 4\delta_{i,f(j)} + 4\delta_{f(i),f(j)} \\
&\equiv \delta_{i,j} \pmod 4.
\end{aligned}$$

Finally, we have to check the Euclidean weight of vectors. If $i \notin K$, it is clear that $\|(\mathbf{e}_i + \mathbf{h}_i, \mathbf{y}'_i)\| = \|\mathbf{e}_i\| + \|\mathbf{y}'_i\| \equiv 1 + 7 \equiv 0 \pmod 8$.

If $i \in K$, we have

$$\begin{aligned}
\|(\mathbf{e}_i + \mathbf{h}_i, \mathbf{y}_i + 2\mathbf{t} + 2 \cdot \mathbf{1})\| &= \|\mathbf{e}_i\| + \|\mathbf{h}_i\| + \|\mathbf{y}_i + 2\mathbf{t}\| + 2\langle \mathbf{y}_i + 2\mathbf{t}, 2 \cdot \mathbf{1} \rangle + \|2 \cdot \mathbf{1}\| \\
&\equiv 1 + 4 + 3 + 4\langle \mathbf{y}_i, \mathbf{1} \rangle + 4(n - a) \\
&\equiv 1 + 4 + 3 + 4 + 4 \qquad (4.2) \\
&\equiv 0 \pmod 8.
\end{aligned}$$

The step (Equation 4.2) makes use of the facts that $\langle \mathbf{y}_i, \mathbf{1} \rangle$ and $n - a$ are odd. Since the Euclidean weight of each row of G is a multiple of 8 and every two rows are orthogonal, G' generates a Type II code by Fact 3 in Chapter 1. □

We can prove that we may consider only the choices of $2\mathbf{t}$ limited to a smaller set.

Theorem 16. Let C be of type 4^a2^b in standard form. By choosing all choices of $\mathbf{2t}$ with non-zero coordinates in the last a positions only, we can generate all possible codes C' with this generalized method.

Proof. Recall that the length of $\mathbf{2t}$ is $a+b$ and B is in the form of $[2I_b|2D]$. If $\mathbf{2t}$ has some non-zero values in its first b coordinates, there is a linear combination $\mathbf{2d}$ of rows of B such that $\mathbf{2t}' = \mathbf{2t} + \mathbf{2d}$ has no non-zero values in its first b coordinates. Let $\mathbf{y}_i$ be a row of Y. We have

$$\begin{aligned} \|\mathbf{y}_i + \mathbf{2t}'\| &\equiv \langle \mathbf{y}_i + \mathbf{2t} + \mathbf{2d}, \mathbf{y}_i + \mathbf{2t} + \mathbf{2d}\rangle - \langle \mathbf{y}_i + \mathbf{2t}, \mathbf{y}_i + \mathbf{2t}\rangle \pmod 8 \\ &\equiv \langle \mathbf{y}_i + \mathbf{2t}, \mathbf{y}_i + \mathbf{2t}\rangle + 2\langle \mathbf{y}_i + \mathbf{2t}, \mathbf{2d}\rangle + \langle \mathbf{2d}, \mathbf{2d}\rangle \pmod 8 \\ &\equiv \|\mathbf{y}_i + \mathbf{2t}\| + 2\langle \mathbf{y}_i, \mathbf{2d}\rangle + 2\langle \mathbf{2t}, \mathbf{2d}\rangle \pmod 8 \\ &\equiv \|\mathbf{y}_i + \mathbf{2t}\| \pmod 8. \end{aligned} \tag{4.3}$$

The last line makes use of the fact that $\langle \mathbf{y}_i, \mathbf{2d}\rangle \equiv 0 \pmod 4$ because $(\mathbf{e}_i, \mathbf{y}_i)$ and $(\mathbf{0}, \mathbf{d})$ are codewords of C while $\langle \mathbf{e}_i, \mathbf{0}\rangle = 0$.

From Equation 4.3, we see that $\mathbf{2t}$ and $\mathbf{2t}'$ give the same set of indices K. Therefore, they generate the same new code. □

When new codes are generated, they are related to the original code with their binary residue codes.

Theorem 17. New codes generated by the generalized Harada's method have the same binary residue code as that of the code from which the new codes are generated.

Proof. We note that when rows of the generator matrix of any new code C' only differ from those of the original code C by an even element. Therefore, C' and C have the same image under mod 2 and hence they share the same binary residue code. $\square$

Here is an example. Let C be a Type II code with a generator matrix

$$G = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 2 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 2 & 3 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 3 & 2 \\ 0 & 0 & 0 & 2 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 2 & 2 & 2 & 2 \end{bmatrix}.$$

We can check that C is a Type II code. Let $2\mathbf{t}$ be $(0,0,2,2,0)$. We see that $\langle \mathbf{y}_1, 2\mathbf{t} \rangle = \langle \mathbf{y}_2, 2\mathbf{t} \rangle = 2$ while $\langle \mathbf{y}_3, 2\mathbf{t} \rangle = 0$. Therefore,

$$\begin{cases} \mathbf{y}_1' & = \mathbf{y}_1 + 2\mathbf{t} + 2 \cdot \mathbf{1} = (3,2,2,1,3) \\ \mathbf{y}_2' & = \mathbf{y}_2 + 2\mathbf{t} + 2 \cdot \mathbf{1} = (3,3,2,3,2) \\ \mathbf{y}_3' & = \mathbf{y}_3 + 2\mathbf{t} = (1,0,3,1,2) \end{cases}$$

and f is a pairing function on $\{1,2\}$. The only choice is $f(1) = 2$ and $f(2) = 1$. Therefore, the second entry in the first row and the first entry in the second of row of G becomes 2. The resultant matrix is

$$G' = \begin{bmatrix} 1 & 2 & 0 & 3 & 2 & 2 & 1 & 3 \\ 2 & 1 & 0 & 3 & 3 & 2 & 3 & 2 \\ 0 & 0 & 1 & 1 & 0 & 3 & 1 & 2 \\ 0 & 0 & 0 & 2 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 2 & 2 & 2 & 2 \end{bmatrix}.$$

We can check that G' generates a Type II code.

4.3 New Codes For $n = 32$

We generate codes with a Type II code C_1 with a generator matrix

$$G_1 = \begin{bmatrix}
1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,2\,3\,2\,3\,1\,3\,2\,3\,3\,1\,3\,1\,0\,1\,1 \\
0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1\,0\,2\,0\,1\,2\,3\,3\,3\,2\,1\,1\,3\,3\,0\,3\,1 \\
0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1\,2\,0\,1\,3\,0\,1\,3\,3\,0\,1\,3\,3\,0\,1\,1 \\
0\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1\,1\,2\,1\,1\,0\,2\,3\,3\,2\,0\,1\,1\,0\,3\,3\,1 \\
0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1\,3\,1\,0\,3\,2\,0\,3\,3\,3\,1\,2\,3\,2\,1\,1 \\
0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1\,3\,1\,1\,3\,3\,2\,0\,0\,3\,1\,1\,1\,3 \\
0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,3\,2\,0\,3\,1\,2\,1\,0\,2\,2\,1\,0\,3\,3\,0 \\
0\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,1\,2\,0\,2\,3\,2\,1\,3\,2\,2\,2\,3\,0\,1\,0\,1 \\
0\,0\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,1\,1\,3\,3\,3\,3\,3\,2\,1\,0\,0\,0\,1\,0\,1\,1 \\
0\,0\,0\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,1\,3\,2\,2\,2\,1\,3\,1\,3\,1\,1\,1\,3\,1 \\
0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,1\,1\,2\,3\,3\,3\,3\,1\,2\,2\,0\,3\,3\,3\,0\,1\,0 \\
0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,1\,0\,1\,1\,3\,1\,3\,3\,2\,2\,3\,2\,3\,1\,0\,0\,3 \\
0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1\,0\,0\,1\,1\,3\,3\,3\,2\,1\,3\,3\,2\,3\,1\,3\,0\,2\,0\,0 \\
0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1\,0\,1\,1\,1\,1\,3\,3\,2\,1\,3\,1\,2\,3\,0\,2\,2\,1\,2 \\
0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1\,1\,1\,1\,3\,1\,1\,1\,2\,1\,3\,1\,0\,2\,0\,0\,2\,3 \\
0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,2\,0\,2\,2\,2\,2\,0\,2\,2\,2\,2\,0\,2\,0\,0\,0\,0 \\
0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,2\,0\,0\,2\,0\,2\,0\,2\,0\,0\,2\,2\,2\,2\,0\,0
\end{bmatrix}.$$

This code has been used once in Section 2.8. At least 29 codes are found with this method. Among which, at least 17 inequivalent codes are not covered in Chapters 2 and 3. The codes are listed as follows. The pairing function is listed as a list of 2-element sets with a row index and its image under f.

TABLE VII

17 EXTREMAL CODES FROM GENERALIZED HARADA'S METHOD FOR $N = 32$

i	$2\mathbf{t}$	pairing	$E_{16}(C_i)$
1	00002000002022200	{2, 6}, {5, 8}, {7, 12}, {9, 15}	108400
2	00002200002220000	{1, 12}, {2, 9}, {3, 11}, {7, 8}, {10, 13}	108408
3	00022200202202202	{1, 10}, {2, 4}, {3, 8}, {12, 15}	108416
4	00002200002220000	{1, 3}, {2, 9}, {7, 11}, {8, 13}, {10, 12}	108424
5	00002200002220000	{1, 2}, {3, 13}, {7, 8}, {9, 11}, {10, 12}	108432
6	00022200202202202	{1, 4}, {2, 10}, {3, 12}, {8, 15}	108440
7	00022200202202202	{1, 15}, {2, 4}, {3, 8}, {10, 12}	108448
8	00002200002220000	{1, 11}, {2, 10}, {3, 13}, {7, 9}, {8, 12}	108456
9	00002200002220000	{1, 11}, {2, 13}, {3, 8}, {7, 9}, {10, 12}	108464
10	00022200202202202	{1, 3}, {2, 4}, {8, 15}, {10, 12}	108472
11	00022200202202202	{1, 15}, {2, 10}, {3, 12}, {4, 8}	108480
12	00002000002022200	{2, 8}, {5, 12}, {6, 9}, {7, 15}	108488
13	00022200202202202	{1, 12}, {2, 3}, {4, 10}, {8, 15}	108496
14	00002200002220000	{1, 7}, {2, 10}, {3, 9}, {8, 11}, {12, 13}	108512
15	00020200000220000	{3, 14}, {4, 6}, {5, 12}, {10, 13}	108520
16	00002200002220000	{1, 10}, {2, 13}, {3, 11}, {7, 8}, {9, 12}	108528
17	00202002220220222	{2, 6}, {7, 14}, {8, 11}, {12, 15}	108552

4.4 New Codes For $n = 40$

We generate codes with a Type II code C_2 with a generator matrix

$$G_2 = \begin{bmatrix}
1\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;1\;1\;1\;1\;3\;1\;3\;3\;1\;3\;2\;0\;2\;0\;2\;0\;0\;0\;0\;1 \\
0\;1\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;1\;0\;0\;2\;0\;2\;3\;3\;3\;3\;3\;0\;0\;0\;0\;2\;3\;1\;3\;3\;3 \\
0\;0\;1\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;1\;0\;2\;0\;3\;1\;2\;0\;3\;1\;3\;0\;0\;1\;1\;1\;2\;0\;0\;3\;3 \\
0\;0\;0\;1\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;1\;3\;3\;2\;2\;0\;0\;1\;1\;0\;2\;2\;3\;3\;0\;1\;1\;2\;3\;3 \\
0\;0\;0\;0\;1\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;1\;1\;0\;0\;0\;2\;3\;1\;1\;3\;2\;3\;3\;1\;1\;2\;2\;0\;0\;2\;3 \\
0\;0\;0\;0\;0\;1\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;1\;0\;1\;0\;1\;1\;0\;2\;3\;3\;2\;3\;1\;0\;2\;0\;3\;3\;0\;0\;1 \\
0\;0\;0\;0\;0\;0\;1\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;1\;1\;1\;2\;0\;1\;2\;3\;0\;0\;1\;2\;1\;0\;1\;0\;0\;3\;0\;1\;3 \\
0\;0\;0\;0\;0\;0\;0\;1\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;1\;1\;3\;2\;2\;1\;2\;3\;0\;1\;0\;2\;2\;3\;2\;3\;3\;2\;1\;0\;1 \\
0\;0\;0\;0\;0\;0\;0\;0\;1\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;1\;1\;1\;2\;1\;2\;1\;0\;0\;2\;3\;3\;2\;3\;2\;0\;3\;2\;2\;3\;1 \\
0\;0\;0\;0\;0\;0\;0\;0\;0\;1\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;3\;0\;3\;2\;3\;0\;1\;3\;1\;1\;3\;2\;0\;1\;0\;2\;1\;3 \\
0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;1\;0\;0\;0\;0\;0\;0\;0\;0\;1\;0\;2\;3\;3\;1\;3\;1\;2\;0\;0\;2\;2\;3\;3\;0\;3\;1\;2\;0\;3 \\
0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;1\;0\;0\;0\;0\;0\;0\;0\;1\;1\;2\;3\;0\;3\;2\;0\;1\;2\;1\;3\;0\;2\;1\;2\;3\;2\;3\;2\;1 \\
0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;1\;0\;0\;0\;0\;0\;0\;1\;1\;2\;1\;2\;1\;1\;0\;0\;1\;0\;1\;0\;0\;0\;1\;0\;1\;2\;3\;1 \\
0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;1\;0\;0\;0\;0\;0\;1\;1\;2\;3\;1\;2\;0\;2\;2\;3\;3\;2\;3\;3\;2\;2\;2\;3\;3\;2\;1 \\
0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;1\;0\;0\;0\;0\;0\;0\;1\;2\;0\;1\;1\;0\;2\;1\;3\;3\;0\;3\;1\;2\;2\;3\;1\;2\;1 \\
0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;1\;0\;0\;0\;0\;1\;0\;2\;3\;1\;1\;0\;0\;1\;2\;2\;1\;0\;1\;0\;3\;0\;1\;3\;3 \\
0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;1\;0\;0\;0\;1\;0\;0\;1\;3\;0\;1\;1\;0\;0\;1\;0\;1\;0\;2\;0\;3\;3\;1\;3 \\
0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;1\;0\;1\;0\;1\;3\;2\;3\;3\;0\;3\;0\;2\;2\;1\;1\;2\;2\;2\;2\;3\;1\;1 \\
0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;1\;1\;0\;1\;3\;1\;2\;0\;1\;2\;3\;0\;1\;0\;2\;3\;2\;2\;0\;3\;1\;3 \\
0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;2\;0\;2\;2\;2\;2\;2\;2\;2\;2\;2\;2\;0\;0\;2\;0\;2\;2\;2\;0\;2 \\
0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;2\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;2\;2\;0\;2\;0\;0\;0\;2\;2
\end{bmatrix}.$$

This code has been used once in Section 2.8. At least 201 codes are found with this method. Among which, at least 102 inequivalent codes are not covered in Chapters 2 and 3. 30 of these 102 codes are listed as follows. All codes are available from the author. The pairing function is listed as a list of 2-element sets with a row index and its image under f.

TABLE VIII

30 EXTREMAL CODES FROM GENERALIZED HARADA'S METHOD FOR $N = 40$

i	$2\mathbf{t}$	pairing	$E_{16}(C_i)$
1	00000000220000000020	{ 1, 7 }, { 4, 15 }, { 9, 16 }, { 11, 14 }, { 13, 19 }	34329
2	00000000220000000020	{ 1, 9 }, { 4, 11 }, { 7, 19 }, { 13, 14 }, { 15, 16 }	34371
3	00000000220000000020	{ 1, 11 }, { 4, 16 }, { 7, 15 }, { 9, 19 }, { 13, 14 }	34393
4	00000000220000000020	{ 1, 11 }, { 4, 19 }, { 7, 9 }, { 13, 16 }, { 14, 15 }	34395
5	00000202202202202202	{ 1, 7 }, { 4, 19 }, { 9, 16 }, { 11, 15 }, { 13, 14 }	34399
6	00000000220000000020	{ 1, 9 }, { 4, 14 }, { 7, 15 }, { 11, 13 }, { 16, 19 }	34425
7	00000000220000000020	{ 1, 9 }, { 4, 14 }, { 7, 15 }, { 11, 16 }, { 13, 19 }	34427
8	00000202202202202202	{ 1, 15 }, { 4, 11 }, { 7, 19 }, { 9, 16 }, { 13, 14 }	34439
9	00000202202202202202	{ 1, 14 }, { 4, 11 }, { 7, 9 }, { 13, 15 }, { 16, 19 }	34473
10	00000000220000000020	{ 1, 11 }, { 4, 16 }, { 7, 9 }, { 13, 15 }, { 14, 19 }	34503
11	00000000220000000020	{ 1, 15 }, { 4, 9 }, { 7, 13 }, { 11, 16 }, { 14, 19 }	34505
12	00222020220220002022 0	{ 1, 15 }, { 4, 11 }, { 7, 19 }, { 9, 16 }, { 13, 14 }	34511
13	00000000220000000020	{ 1, 4 }, { 7, 9 }, { 11, 15 }, { 13, 14 }, { 16, 19 }	34513
14	00222020220220002022 0	{ 1, 11 }, { 4, 9 }, { 7, 14 }, { 13, 16 }, { 15, 19 }	34519
15	00000000220000000020	{ 1, 11 }, { 4, 15 }, { 7, 19 }, { 9, 14 }, { 13, 16 }	34523
16	00000000220000000020	{ 1, 4 }, { 7, 9 }, { 11, 19 }, { 13, 14 }, { 15, 16 }	34525
17	00000000220000000020	{ 1, 13 }, { 4, 19 }, { 7, 14 }, { 9, 15 }, { 11, 16 }	34527
18	00222020220220002022 0	{ 1, 15 }, { 4, 19 }, { 7, 16 }, { 9, 11 }, { 13, 14 }	34531
19	00000000220000000020	{ 1, 11 }, { 4, 16 }, { 7, 13 }, { 9, 19 }, { 14, 15 }	34533
20	00000000220000000020	{ 1, 19 }, { 4, 16 }, { 7, 14 }, { 9, 15 }, { 11, 13 }	34535
21	00000000220000000020	{ 1, 11 }, { 4, 15 }, { 7, 16 }, { 9, 19 }, { 13, 14 }	34537
22	00000000220000000020	{ 1, 11 }, { 4, 9 }, { 7, 15 }, { 13, 14 }, { 16, 19 }	34539
23	00000000220000000020	{ 1, 15 }, { 4, 14 }, { 7, 13 }, { 9, 11 }, { 16, 19 }	34549
24	00000202202202202202	{ 1, 15 }, { 4, 14 }, { 7, 19 }, { 9, 16 }, { 11, 13 }	34555
25	00000000220000000020	{ 1, 7 }, { 4, 9 }, { 11, 19 }, { 13, 15 }, { 14, 16 }	34561
26	00000000220000000020	{ 1, 9 }, { 4, 19 }, { 7, 11 }, { 13, 14 }, { 15, 16 }	34567
27	00222020220220002022 0	{ 1, 4 }, { 7, 13 }, { 9, 15 }, { 11, 16 }, { 14, 19 }	34575
28	00000000220000000020	{ 1, 4 }, { 7, 9 }, { 11, 16 }, { 13, 19 }, { 14, 15 }	34581
29	00000000220000000020	{ 1, 4 }, { 7, 13 }, { 9, 11 }, { 14, 15 }, { 16, 19 }	34583
30	00000202202202202202	{ 1, 16 }, { 4, 7 }, { 9, 13 }, { 11, 15 }, { 14, 19 }	34585

4.5 Conclusion

The results above can be summarized as follows.

Corollary 18. 17 extremal Type II codes of length 32 and type $4^{15}2^2$, and 102 extremal Type II codes of length 40 and type $4^{19}2^2$ are found with the generalized Harada's method.

CHAPTER 5

CONCLUSION

We have developed and generalized three methods to construct extremal Type II codes. There are altogether 779 extremal Type II codes of length 32 and 1065 extremal Type II codes of length 40 are found. These numbers are higher than the number of codes found by M. Harada in (7), which was 146 extremal Type II codes of length 32 and 37 extremal Type II codes of length 40.

Doubling method in Chapter 2 is a new method to construct Type II codes of type 4^a2^b from known extremal Type II codes of type $4^{a+1}2^{b-2}$. I believe that a huge number of inequivalent codes is yet to be discovered by this method. It is because hundreds of new codes have been discovered with the use of only one known Type II code. I will try constructing more codes and discover the relation between the new codes, as well as any better way from the nature of the method to distinguish inequivalent codes.

The generalized building-up method in Chapter 3 partially fills the gap left by J.-L. Kim, H. Lee and Y. Lee (4) (5) (6). With this generalized method, we can build up Type II codes in $\mathbb{Z}_4$ from known Type II codes of smaller length. I believe that similar but slightly more complicated construction methods can be applied on codes on $\mathbb{Z}_{2^k}$ where $k \geq 3$.

In Chapter 4, we have generalized Harada's construction method. Again, I believe that a huge number of codes can be found with this method with the growing number of known codes with the use of the other two methods.

CITED LITERATURE

1. Huffman, W. C.: Decompositions and extremal Type II codes over $\mathbb{Z}_4$. IEEE Trans. Inform. Theory, 44:800–809, 1997.

2. Pless, V., Leon, J., and Fields, J.: All $\mathbb{Z}_4$ codes of Type II and length 16 are known. J. Combin. Theory Ser. A, 78:32–50, 1997.

3. Harada, M.: New extremal Type II codes over $\mathbb{Z}_4$. Des. Codes Cryptogr., 13:271–284, 1998.

4. Kim, J.-L. and Lee, Y.: Euclidean and hermitian self-dual mds codes over large finite fields. J. Combin. Theory Ser. A, 105:79–95, 2004.

5. Kim, J.-L. and Lee, Y.: Self-dual codes using the building-up construction. IEEE International Symposium on Information Theory, pages 2400–2402, 2009.

6. Lee, H. and Lee, Y.: Construction of self-dual codes over finite rings $\mathbb{Z}_{p^m}$. J. Comb. Theory, Ser. A, 115, 2008.

7. Harada, M.: On the residue codes of extremal Type II $\mathbb{Z}_4$-codes of lengths 32 and 40. Discrete Mathematics, 311:2148–2157, 2011.

8. Bachoc, C., Bonnecaze, A., Mourrain, B., and Solé, P.: Type II codes over $\mathbb{Z}_4$. IEEE Trans. Inform. Theory, 43:969–976, 1997.

9. Harada, M.: Extremal Type II $\mathbb{Z}_4$-codes of lengths 56 and 64. J. Combin. Theory Ser. A, 117:1285–1288, 2010.

10. Huffman, W. C. and Pless, V.: Fundamentals of Error-Correcting Codes. Cambridge University Press, 3rd edition, 2010.

11. Conway, J. H. and Sloane, N. J. A.: Self-dual codes over the integers modulo 4. J. Combin. Theory Ser. A, 62:30–45, 1993.

12. Gaborit, P. and Harada, M.: Construction of extremal Type II codes over $\mathbb{Z}_4$. Des. Codes Cryptogr., 16:257–269, 1999.

13. Huffman, W. C.: On the classification and enumeration of self-dual codes. Finite Fields Appl., 11:451–490, 2005.

VITA

Kin Hang Chan

851 S Morgan St Chicago IL 60607

Department of Mathematics, Statistics and Computer Science,

322 SEO (M/C 249)

kchan25@uic.edu

Research Interest

Algebraic Coding Theory, especially error correcting codes

Education

PhD in Mathematics, University of Illinois at Chicago (expected in January 2012)

PhD candidate in Mathematics, University of Illinois at Chicago (August 2006 - January 2012)

MPhil in Mathematics, Chinese University of Hong Kong (August 2006)

BSc in Mathematics, Chinese University of Hong Kong (July 2004)

Employment

August 2006 - May 2011: Teaching Assistant at University of Illinois at Chicago

September 2004 - June 2006: Teaching Assistant at Chinese University of Hong Kong

Awards

International Mathematical Olympiad: Bronze Medals at 1999 and 2001, Honorable Mention at 1998

Chinese Mathematical Olympiad: Bronze Awards at 1999, 2000 and 2001

Asian Pacific Mathematics Olympiad: Silver Award at 1998, Bronze Award at 1999 and 2001