STUDY ON CONSTRUCTIONS OF QUANTUM ERROR CORRECTION CODES

DISSERTATION

for the Degree of

DOCTOR OF PHILOSOPHY
(Electrical Engineering)

DUC MANH NGUYEN

JANUARY 2020

Study on constructions of quantum error correction codes

DISSERTATION

Submitted in Partial Fulfillment
of the Requirements for the
Degree of

DOCTOR OF PHILOSOPHY
(Electrical Engineering)

at the

UNIVERSITY OF ULSAN

by

Duc Manh Nguyen
Advisor: Prof. Sunghwan Kim

January 2020

Publication No.______________

Study on constructions of quantum error correction codes

Approved by Supervisory Committee:

Prof. Sungoh Kwon, *chair*

Prof. Kyoung-Young Song

Prof. Ji-Woong Jang

Prof. Jiho Song

Prof. Sunghwan Kim, *adviser*

January, 2020

VITA

Duc Manh Nguyen was born in Hai-Duong province, Viet-Nam on November 18, 1989. He received the B.E. degree (2012) from school of electronics and telecommunications from Ha-Noi University of Science and Technology (HUST), Ha-Noi, Viet-Nam. In March 2015, he began working full time towards his Ph.D. at University of Ulsan, Korea under the guidance of professor Sunghwan Kim. Since then, he has conducted researches in quantum information theory, quantum error correction codes, and quantum algorithms.

Dedicated to my grateful family

for their love and support

ACKNOWLEDGMENTS

This thesis would not have been possible without support of many people. I would like to thank many people who have contributed to this thesis.

First and foremost, I would like to express my sincere gratitude to my advisor, Prof. Sunghwan Kim, for giving me the opportunity to be a member of Coding and Information Theory (CIT) lab and for his guidance in weekly meeting which helped me to complete my work. I am greatly indebted to him for his full support, constant encouragement, and advices both in research and living matters.

I would also like to thank my committee members for many useful interactions and for contributing their broad perspective in redefining the ideas in this dissertation. I am grateful to my lab mates (in CIT lab) and my friends in University of Ulsan for their friendship, enthusiastic help, and cheerfulness during my stay in Ulsan, Korea.

I am very much grateful to University of Ulsan for giving me such a wonderful research environment and financial support (through BK21+ programs). UOU gave me the chance to conduct a number of useful courses with excellent instructors that will give me a way to think and work properly in the rest of my life.

I am deeply indebted to my parents and my small family for their love and unyielding supports. In addition, I would like to share my thesis to my unlucky friends (Bui-Thi-Kim-Lam, Cao-Dai-Duong, Nguyen-Tung-Lam) and I hope they will be better and luckier in their next life.

Last, thank you God for giving me all the strength, “IN GOD I TRUST”.

ABSTRACT

Study on constructions of quantum error correction codes

by: **Duc Manh Nguyen**

Advisor: **Prof. Sunghwan Kim**

Submitted in Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy (Electrical Engineering)

January 2020

Quantum computation is proven to give us effective solutions for difficult problems such as factoring large integer numbers in polynomial time, searching in un-ordered database, increasing the security of cryptography protocol; these tasks are difficult or less effective in classical computation. However, the effect of noise and imperfect environment in a quantum channel can affect the performance of quantum computation. Therefore, quantum error correction codes (QECC) is proposed to achieve the fault-tolerant quantum computation.

In this thesis, I study the design of QECCs and provide several contributions to quantum stabilizer code construction. I use stabilizer formalism to explain the quantum error correction codes as quantum stabilizer codes. The quantum stabilizer codes allow to remove and detect the errors by the group of quantum operators. In addition, quantum stabilizer codes can be constructed from binary or qua-ternary codes. So, our methods are using the combinatoric such as circulant, different sets, self-orthogonal, self-dual with Hermitian inner product, trace inner product to construct suitable classical codes; then, I investigate outstanding quantum stabilizer codes.

Contents

Chapter 1

Introduction

1.1 Motivation

Quantum mechanisms utilize two degrees of freedom of photons, allowing for various probabilities of possible measurement outcomes of a physical system [1]. Quantum processing devices are based on quantum mechanisms, which give us the ability to deal with the various tasks such as factoring a large integer number in polynomial time [2], searching from un-ordered sets [3], and improving the security of cryptography [4] [5]. However, the effects of the noisy and imperfect environments of a quantum channel can reduce these performance advantages. Therefore, quantum error correcting codes (QECCs) have been proposed to protect quantum information from noisy environments. The first QECCs were proposed in the 1990s by Shor [6] and Steane [7], and the general theory for QECCs, i.e., the stabilizer formalism, was introduced in 1997 by Daniel Gottesman [8]. In 1996, two independent research groups, Calderbank and Shor [9] and Stean [7] adopted the relationship between quantum codes and self-orthogonal codes. Therefore, quantum codes can be constructed using two classical linear error correction codes, i.e., the Calderbank-Shor-Stean (CSS) structure. The advantage of the CSS structure is that we can obtain the parameters of quantum codes directly from the parameters of two classical codes. Therefore, the CSS structure has been used by numerous researchers to construct quantum binary codes, such as BCH codes [10], Reed Solomon codes [11], quasi-cyclic LDPC codes [12], and SPC codes [13]. Quantum computations are performed through two techniques, the first technique was provided by Deutsch, DeutschJozsa, Grover, P. Shor and so on [2] [3] [14] [15]. In this technique, a set of unitary transformations are applied on a quantum system and then the problem is solved according to the state of one qubit, or more, after measurement process to solves the problem at hands. The second is called Zidan's technique that solves the problem at hands by applying some unitary transformation(s) on a system of size n qubits, then measures the degree of entanglement (by concurrence measure) between two ancillary qubits. Hence, the solution of the problem at hands is obtained based on the concurrence value [16]. Recently, with the development of quantum information theory, more applications for quantum error correction codes in quantum information have been demonstrated, such as quantum algorithms [16] [17], quantum simulations [18], and quantum network coding [19].

Quantum stabilizer code is a kind of QECC constructed based on the stabilizer formalism. The most important advantage of quantum stabilizer codes is that quantum errors that affect an encoded quantum state can be diagnosed and removed by a group of quantum operators, thereby stabilizing this encoded quantum state. In addition, the stabilizer formalism allows quantum codes to be presented by classical error correction codes. Therefore, quantum stabilizer codes can be constructed from binary error correction codes if they satisfy a symplectic inner product (SIP). Many quantum stabilizer codes have been constructed based on the binary formalism with combinatorial design, such as quantum codes based on difference sets [71], based on group association schemes [72], based on circulant matrices [73] [74], or based on CSS structure over Finite field [75]. In paper [76], a quantum stabilizer code was proven to correspond to an additive code over Galois field 4 (*GF(4)*), which is self-orthogonal with respect to the trace-inner product. So far, many papers have focused on (1) the design of classical additive codes over *GF(4)* to achieve corresponding quantum stabilizer codes, such as self-dual codes over *GF(4)*, which have dimension "0" and can be represented by graphs [77]; (2) QECCs based on self-dual codes over *GF(4)* with the highest known minimum weights [78]; and (3) QECCs based on Hermitian self-orthogonal codes with extension design [79]. However, self-orthogonal codes with high error-correcting capacity are restricted, and therefore, further investigation was required to generate good stabilizer codes. Introduction of entanglement-assisted quantum error correction code (EAQECC) by Brun et al. [90] is one answer to this problem. More precisely, it enables us to construct the quantum error-correction codes not only from self-orthogonal classical codes, but also from arbitrary classical codes with the help of copies of maximally entangled quantum states shared between encoder and decoder. To design efficient EAQECC, however, it is desirable to use the fewest entangled states possible, because the cost to prepare those states is relatively high. Hence, the construction of EAQECC with small amounts of entangled states is a much more attractive issue [90, 91]. Therefore, several constructions of EAQECC have been proposed. Since the stabilizer code that is useful for fault-tolerant computation [98], in this thesis, we consider several construction methods for quantum stabilizer codes, entanglement-assisted quantum error correction codes with the aim of design quantum codes with high error correction capacity and large information length.

1.2 A brief background on quantum error correction code

In this section, we introduce the concepts of quantum information theory, quantum error correction codes (QECC), quantum stabilizer codes, binary formalism of QECC, QECC over Galois Field 4 (*GF(4)*), and entanglement-assisted quantum error correction codes (EA-QECC).

1.2.1 Quantum information

Bits or binary digits are the basic units of information that are used in classical computing and digital communication. The basic unit of quantum information is called a quantum bit (*qubit*). If a bit has two basic states of zero or one, a qubit uses the superposition principle of the two basic states. Hence, we use the two-dimensional Hilbert space (H_2) of complex number to model the quantum information. The Hilbert space is spanned by two basic states $H_2 = span\{|0\rangle, |1\rangle\}$ where the mathematical expressions are $|1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ and $|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$. Therefore, the superposition state of a quantum system is denoted as, $|\psi\rangle = \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = a_1 |0\rangle + a_2 |1\rangle$, where a_1 and a_2 are complex numbers that satisfy the equation: $|a_1|^2 + |a_2|^2 = 1$. In a quantum system, if quantum states are used n times in a single qubit (n-qubits) physical system, the system consists of the n times tensor product of two-dimensional Hilbert space (${H_2}^{\otimes n}$).

In classical computation, Boolean functions f: $\{0, 1\} \rightarrow \{0, 1\}$ are performed over a single bit. In the case of quantum computation, reversible operation represented by unitary matrices are performed over a qubit. Representative quantum operations are Pauli operators. Four Pauli operators (matrices) $\mathbf{I}$, $\sigma_{\mathbf{X}}$, $\sigma_{\mathbf{Y}}$, and $\sigma_{\mathbf{Z}}$ are

$$\mathbf{I} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \sigma_{\mathbf{X}} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \sigma_{\mathbf{Y}} = j \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}, \sigma_{\mathbf{Z}} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix},$$

where $j = \sqrt{-1}$. The transformations of quantum states by Pauli operators is as

$$\begin{aligned}
\mathbf{I}\,|\psi\rangle &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \times \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \alpha\,|0\rangle + \beta\,|1\rangle\,, \\
\sigma_{\mathbf{X}}\,|\psi\rangle &= \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \times \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \beta \\ \alpha \end{bmatrix} = \beta\,|0\rangle + \alpha\,|1\rangle\,, \\
\sigma_{\mathbf{Y}}\,|\psi\rangle &= \begin{bmatrix} 0 & -j \\ j & 0 \end{bmatrix} \times \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} -j\beta \\ j\alpha \end{bmatrix} = j(-\beta\,|0\rangle + \alpha\,|1\rangle), \\
\sigma_{\mathbf{Z}}\,|\psi\rangle &= \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \times \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \alpha \\ -\beta \end{bmatrix} = \alpha\,|0\rangle - \beta\,|1\rangle\,.
\end{aligned}$$

Therefore, operators $\sigma_{\mathbf{X}}$, $\sigma_{\mathbf{Z}}$, and $\sigma_{\mathbf{Y}}$ are regarded as a bit flip, a phase flip, and a combination of bit and phase flips, respectively. Multiplications between Pauli operators are defined as

$$\begin{aligned}
&{\sigma_{\mathbf{X}}}^2 = {\sigma_{\mathbf{Y}}}^2 = {\sigma_{\mathbf{Z}}}^2 = \mathbf{I};\\
&\sigma_{\mathbf{X}} \times \sigma_{\mathbf{Y}} = j\sigma_{\mathbf{Z}}\,\text{and}\,\sigma_{\mathbf{Y}} \times \sigma_{\mathbf{X}} = -j\sigma_{\mathbf{Z}} \to \sigma_{\mathbf{X}} \times \sigma_{\mathbf{Y}} = -\sigma_{\mathbf{Y}} \times \sigma_{\mathbf{X}};\\
&\sigma_{\mathbf{Y}} \times \sigma_{\mathbf{Z}} = j\sigma_{\mathbf{X}}\,\text{and}\,\sigma_{\mathbf{Z}} \times \sigma_{\mathbf{Y}} = -j\sigma_{\mathbf{X}} \to \sigma_{\mathbf{Y}} \times \sigma_{\mathbf{Z}} = -\sigma_{\mathbf{Z}} \times \sigma_{\mathbf{Y}};\\
&\sigma_{\mathbf{Z}} \times \sigma_{\mathbf{X}} = j\sigma_{\mathbf{Y}}\,\text{and}\,\sigma_{\mathbf{X}} \times \sigma_{\mathbf{Z}} = -j\sigma_{\mathbf{Y}} \to \sigma_{\mathbf{Z}} \times \sigma_{\mathbf{X}} = -\sigma_{\mathbf{X}} \times \sigma_{\mathbf{Z}}.
\end{aligned}$$

The Pauli group P_1 on a qubit is a group composed of Pauli operators and their multiplications with the factor ± 1, $\pm j$. Then, $P_1 = \pm\{\mathbf{I},\ \sigma_{\mathbf{X}},\ j\sigma_{\mathbf{X}},\ \sigma_{\mathbf{Y}},\ j\sigma_{\mathbf{Y}},\ \sigma_{\mathbf{Z}},\ j\sigma_{\mathbf{Z}}\}$. The Pauli group on n qubits P_n is defined as n tensor product of the Pauli operators. Then, the elements of P_n are either commutative or anti-commutative. The commutative operator $\circ$ for two operators $\mathbf{A}$ and $\mathbf{B}$ is defined as

$$\mathbf{A} \circ \mathbf{B} = \prod_{i=1}^{n} A_i \bullet B_i \text{where} \mathbf{A}_i \bullet \mathbf{B}_i = \begin{cases} +1, \text{if} \mathbf{A}_i \times \mathbf{B}_i = \mathbf{B}_i \times \mathbf{A}_i \\ -1, \text{if}\, \mathbf{A}_i \times \mathbf{B}_i = -\mathbf{B}_i \times \mathbf{A}_i \end{cases}.$$

Quotient group P_n/C where C = $\{\pm\mathbf{I},\ \pm j\mathbf{I}\}$ is defined as the center of P_n [36]. Therefore, the notation $\mathbf{X} \leftrightarrow \sigma_{\mathbf{X}}$, $\mathbf{Y} \leftrightarrow -j\sigma_{\mathbf{Y}}$, $\mathbf{Z} \leftrightarrow \sigma_{\mathbf{Z}}$ [37] are used in the rest of the paper.

The proposals for the first generation of quantum systems make use of two-level systems as the basis elements. However, recent innovations in quantum error correction code, quantum cryptography, and quantum algorithms demonstrate that there are advantages to use high-level quantum systems over qubit analogues. To describe the high-level quantum system, we use the Galois field for the basic elements. Let p be a prime number and the Galois field GF_p be the finite field of p elements $\{0, 1,\ 2,\ ...,\ p-1\}$ that is closed under addition and multiplication modulo p. Additionally, assume a qupit (p-level quantum bit) whose Hilbert space (H_p) is represented by orthogonal bases $H_p = span\{|0\rangle,\ |1\rangle,\ ...,\ |p-1\rangle\}$, where the mathematical expression is

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}_{p\times 1},\ |1\rangle = \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix}_{p\times 1},\ ...,\ |p-1\rangle = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}_{p\times 1}.$$

Let ω be the p-th root of unity, $\omega = e^{\frac{2\pi i}{p}}$. Additionally, we define the generalized Pauli matrix such that the generalization of the bit-flip matrix is

$$\mathbf{X}(p) = \sum_{j=0}^{p-1} |j+1\rangle \langle j| = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \\ 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & 0 \end{bmatrix}.$$

Then, we have $\mathbf{X}(p)|r\rangle = |r+1\rangle$ and $\mathbf{X}(p)^p = \mathbf{X}(p)^0 = \mathbf{I}(p)$. The generalization of the phase-flip matrix is

$$\mathbf{Z}(p) = \sum_{j=0}^{p-1} \omega^j |j\rangle \langle j| = \begin{bmatrix} \omega^0 & 0 & \cdots & 0 \\ 0 & \omega^1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & \omega^{p-1} \end{bmatrix}.$$

Then, we have $\mathbf{Z}(p)|r\rangle = \omega^r |r\rangle$ and $\mathbf{Z}(p)^p = \mathbf{Z}(p)^0 = \mathbf{I}(p)$. We note that here $\mathbf{X}(p)$, $\mathbf{Z}(p)$, $\mathbf{Y}(p)$, and $\mathbf{I}(p)$ are the generalized Pauli matrices over $GF(p)$ with the size $p \times p$.

1.2.2 QECCs and stabilizer formalism

QECCs are used in quantum computing to protect quantum information from errors due to decoherence and other quantum noises. QECCs are essential to achieve fault-tolerant quantum computation [20]. In classical error correcting code, it is easy to make the copy of information. In contrast, it is impossible to make the copy of quantum information due to the non-cloning theorem [21]. Therefore, quantum information can be extended to highly entangled quantum state with the help of ancillary qubits and Unitary transforms. Classical error correcting codes use a syndrome measurement to diagnose errors which corrupt an encoded state. QECC also employs the syndrome detection with the help of quantum stabilizers operators. A block diagram of the QECC process is shown in Figure 1.1. The quantum information can be protected from noisy quantum channel with the help of ancillary qubits, the quantum stabilizer operators, and syndrome measurement.

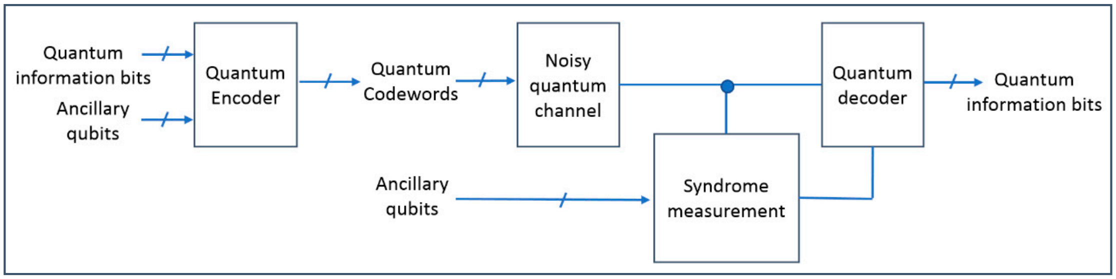


Figure 1.1: Quantum error correction operating process.

The quantum code for the binary case with parameter $[[n, k, d_{min}]]$ encodes k information qubits into the system of n qubits, and it can correct the $\lfloor \frac{d_{min}-1}{2} \rfloor$ error. The first quantum code is the Shor code with parameters [[9,1,3]]; this is based on repetition codes. The second quantum code is [[7,1,3]], which is based on the classical hamming code [7,1,3] with a CSS structure. Next, the stabilizer formalism is used to express the quantum codes. With the stabilizer formalism, quantum codes are viewed via group theory of the quantum stabilizer operator; thus, we are working with quantum operators rather than with quantum states. Let $H^{\otimes n}$ be the state space of n-qubits. The quantum stabilizer group S is an Abelian subgroup of P_n and is closed under multiplication. Further, there is no trivial subspace $C_S \subset H^{\otimes n}$ that is fixed (or stabilized) by S. The stabilized C_S defines a codeword such that $C_S = \{ \ket{\psi} \in H^{\otimes n} : \quad \mathbf{g}\ket{\psi} = \ket{\psi}, \quad \forall \mathbf{g} \in S\}$. The quantum code with parameter $[[n, k, d_{min}]]$ corresponds to the group S with its generators, $g = \{\mathbf{g}_1, \mathbf{g}_2, .., \mathbf{g}_{n-k}\}$. The constraints for the generators in g are such that any two elements in g must commute with each other. Since the number of generators $(n-k)$ is less than or equal to n, $(n-k)$ is a well-defined quantity ; it is called the rank of the stabilizer. If the stabilizer has the rank n $(k = 0)$, the stabilizer will be referred to as the full rank stabilizer and the corresponding quantum code is denoted as $[[n, 0, d_{min}]]$. The codewords, or the stabilizer state of the full rank stabilizers group, will be called the graph state; this has many applications in one-way quantum computers, secure state distribution, secret sharing, etc. It is known that, for a quantum code $[[n, k, d_{\min}]]_p$, the quantum singleton bound or Knill-Laflamme bound is $n-k \geq 2(d_{\min}-1)$. Then, the quantum codes whose parameters satisfy $d_{min} = \left\lfloor \frac{n+2}{2} \right\rfloor$ will be maximum distance separable codes, which are referred to as the optimal quantum codes.

Considering a set of error operators $\{\mathrm{E}\} \subset P_n$, the collection of Pauli operators takes a state $\ket{\psi}$ to the corrupted state $\mathbf{E}\ket{\psi}$. A given operator $\mathbf{E}$ either is commutative or anti-commutative with each stabilizer operator $\mathbf{S}_i$. Then, the corrupted state $\mathbf{E}\left|\psi\right\rangle$ is diagnosed by elements $\mathbf{S}_i$ of the set S. The outcome of the diagnostic procedure is a vector of $\{+1,-1\}$ indicating whether or not $\mathbf{E}$ can be detected. The indication for the error detection is expressed as follows.

$$\mathbf{S}_i \times \mathbf{E}\left|\psi\right\rangle = \begin{cases} \quad\mathbf{E}\times\mathbf{S}_i\left|\psi\right\rangle = \quad\mathbf{E}\left|\psi\right\rangle, & \text{Error undetected.} \\ -\mathbf{E}\times\mathbf{S}_i\left|\psi\right\rangle = -\mathbf{E}\left|\psi\right\rangle, & \text{Error detected.} \end{cases} \tag{1.1}$$

The condition for quantum error correction is that $\mathbf{E}$ is a set of correctable error operators for C_S if

$$\mathbf{E}_i{}^{\dagger}\mathbf{E}_j \notin N(S)\backslash S,\ \forall \mathbf{E}_i, \mathbf{E}_j \in \mathbf{E},$$

where $\mathbf{E}_i{}^{\dagger}$ is conjugate transpose of $\mathbf{E}_i$ and $N(S)$ is the normalize of S in P_n such as

$$N(S) = \{A \in P_n \mid A^{\dagger}EA \in S,\ \forall E \in S\}.$$

Note that $N(S)$ is the collection of all operators in P_n that commutes with S. Then, the minimum distance of stabilizer code is determined by $d_{\min} = \min(\mathrm{W}(\mathbf{E}))$ s.t. $\mathbf{E} \in N(S)\backslash S$, where the weight of an operator, $\mathrm{W}(*)$, is the numbers of positions not equal to Pauli operator $\mathbf{I}$.

Since we can express the generator of the quantum stabilizer code as a binary field, due to the fact that any n-qubit Pauli operator can be expressed as a multiplication of an $\mathbf{X}$-containing operator and an $\mathbf{Z}$-containing operator, we define the mapping between Pauli operators and binary vectors as $\mathbf{I} \leftrightarrow (0,0)$, $\mathbf{X} \leftrightarrow (1,0)$, $\mathbf{Z} \leftrightarrow (0,1)$, and $\mathbf{Y} \leftrightarrow (1,1)$. As a consequence, $(n-k)$ generators of an $[[n,k]]$ code are formed in a binary field as $\mathbf{H} = [\mathbf{H}_{\mathbf{X}}|\mathbf{H}_{\mathbf{Z}}]$ where $\mathbf{H}_{\mathbf{X}}, \mathbf{H}_{\mathbf{Z}}$ are $(n-k)\times n$ binary matrices and "|" denotes the row concatenation. Hence, $\mathbf{H}$ represents binary matrices with the size

$(n-k) \times 2n$. The commutative constraint between generators must change to the symplectic product constraint as

$$\mathbf{H_Z} \times \mathbf{H_X}^T \;+\; \mathbf{H_X} \times \mathbf{H_Z}^T \;=\; \mathbf{0}_m \;, \tag{1.2}$$

where $\mathbf{0}_m$ is the matrix of all zero elements with size $m \times m$. From the binary form of a stabilizer code, by using Gaussian elimination, the parity-check matrix $\mathbf{H}$ can be uniquely determined in standard form as follows,

$$\left[\begin{array}{cccccc} \overbrace{\mathbf{I}}^{r} & \overbrace{\mathbf{A}_1}^{n-k-r} & \overbrace{\mathbf{A}_2}^{k} & \overbrace{\mathbf{B}}^{r} & \overbrace{\mathbf{C}_1}^{n-k-r} & \overbrace{\mathbf{C}_2}^{k} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{D} & \mathbf{I} & \mathbf{E} \end{array}\right] \begin{array}{ll} \} & \qquad r \\ \} & n-k-r \end{array} \tag{1.3}$$

The linear combinations among rows of parity-check matrix $\mathbf{H}$ generate the stabilizer group S in binary modulo-2 addition. Since the dual-space of $\mathbf{H}$ has the dimension of $2n - m = m + 2k$, the normalize group *N(S)* that commutes with S can be considered as the dual-space of S generated by a $(m + 2k) \times 2n$ binary matrix. The last $2k$ rows are called the logical operators $\overline{\mathbf{X}}$ and $\overline{\mathbf{Z}}$ which satisfy the following conditions

$$\begin{cases} \overline{\mathbf{X}_i} \circ \overline{\mathbf{X}_j} = +1 \\ \overline{\mathbf{Z}_i} \circ \overline{\mathbf{Z}_j} = +1 \\ \overline{\mathbf{X}_i} \circ \overline{\mathbf{Z}_j} = +1 \text{ for } i \neq j \\ \overline{\mathbf{X}_i} \circ \overline{\mathbf{Z}_j} = -1 \text{ for } i = j \end{cases} \tag{1.4}$$

The standard form of the logical operators is given as follows:

$$\begin{cases} \overline{\mathbf{X}} = \left[\begin{array}{cccccc} \mathbf{0} & \mathbf{E}^T & \mathbf{I} & (\mathbf{E}^T\mathbf{C}_1 + \mathbf{C}_2{}^T) & \mathbf{0} & \mathbf{0} \end{array}\right] \\ \overline{\mathbf{Z}} = \left[\begin{array}{cccccc} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{A}_2{}^T & \mathbf{0} & \mathbf{I} \end{array}\right] \end{cases} \tag{1.5}$$

The operation of encoding a general stabilizer code can be described as

$$|\overline{\sigma_1\sigma_2...\sigma_k}\rangle = \frac{1}{\sqrt{2^{n-k}}} \times \left(\prod_{i=1}^{n-k} (\mathbf{I} + \mathbf{g}_i)\right) \times \overline{\mathbf{X}_1}^{\sigma_1} \times \overline{\mathbf{X}_2}^{\sigma_2} \times ... \times \overline{\mathbf{X}_k}^{\sigma_k} |00...0\rangle_n, \tag{1.6}$$

where $|00...0\rangle$ is the n-qubits state and $c_i \in \{0,1\}$.

Above, we considered quantum stabilizer codes with a binary form over $\{0,1\}^{2n}$. Since these codes are defined over $\{0,1\}^{2n}$, we call them quantum binary codes. Generally, we denote quantum non-binary codes with parameters $[[n,k,d_{min}]]_p$, which are defined over ${GF_p}^n$(the *qupits* case). The quantum code for this case corresponds to the commutative group of generalized Pauli operators. Based on the generalized Pauli matrices $\mathbf{X}(p)$ and $\mathbf{Z}(p)$ in Section 2.1, we have $\mathbf{Z}(p)^{\mathbf{b}}\mathbf{X}(p)^{\mathbf{a}} = \omega^{\mathbf{a}\bullet\mathbf{b}}\mathbf{X}(p)^{\mathbf{a}}\mathbf{Z}(p)^{\mathbf{b}}$ with the following notations:

1. $\mathbf{a}\bullet\mathbf{b} = \sum_{i=1}^{n} a_i \cdot b_i$ where $\mathbf{a} = (a_1 a_2 \cdots a_n)$ and $\mathbf{b} = (b_1 b_2 \cdots b_n)$,
2. $\mathbf{X}(p)^{\mathbf{a}} = \mathbf{X}(p)^{a_1} \otimes \mathbf{X}(p)^{a_2} \otimes \cdots \otimes \mathbf{X}(p)^{a_n}$,
3. $\mathbf{Z}(p)^{\mathbf{b}} = \mathbf{Z}(p)^{b_1} \otimes \mathbf{Z}(p)^{b_2} \otimes \cdots \otimes \mathbf{Z}(p)^{b_n}$.

We consider the commutative property between $\mathbf{X}(p)^{\mathbf{u}_1}\mathbf{Z}(p)^{\mathbf{v}_1}$ and $\mathbf{X}(p)^{\mathbf{u}_2}\mathbf{Z}(p)^{\mathbf{v}_2}$. Since,

1. $\mathbf{X}(p)^{\mathbf{u}_1}\mathbf{Z}(p)^{\mathbf{v}_1})(\mathbf{X}(p)^{\mathbf{u}_2}\mathbf{Z}(p)^{\mathbf{v}_2}) = (\mathbf{X}(p)^{\mathbf{u}_1}\omega^{\mathbf{u}_2\cdot\mathbf{v}_1}\mathbf{X}(p)^{\mathbf{u}_2})(\mathbf{Z}(p)^{\mathbf{v}_1}\mathbf{Z}(p)^{\mathbf{v}_2}) = \omega^{\mathbf{u}_2\cdot\mathbf{v}_1}(\mathbf{X}(p)^{\mathbf{u}_1}\mathbf{X}(p)^{\mathbf{u}_2})(\mathbf{Z}(p)^{\mathbf{v}_1}\mathbf{Z}(p)^{\mathbf{v}_2}$,
2. $\mathbf{X}(p)^{\mathbf{u}_2}\mathbf{Z}(p)^{\mathbf{v}_2})(\mathbf{X}(p)^{\mathbf{u}_1}\mathbf{Z}(p)^{\mathbf{v}_1}) = (\mathbf{X}(p)^{\mathbf{u}_2}\omega^{\mathbf{u}_1\cdot\mathbf{v}_2}\mathbf{X}(p)^{\mathbf{u}_1})(\mathbf{Z}(p)^{\mathbf{v}_2}\mathbf{Z}(p)^{\mathbf{v}_1}) = \omega^{\mathbf{u}_1\cdot\mathbf{v}_2}(\mathbf{X}(p)^{\mathbf{u}_2}\mathbf{X}(p)^{\mathbf{u}_1})(\mathbf{Z}(p)^{\mathbf{v}_2}\mathbf{Z}(p)^{\mathbf{v}_1}$,

$\mathbf{X}(p)^{\mathbf{u}_1}\mathbf{Z}(p)^{\mathbf{v}_1}$ and $\mathbf{X}(p)^{\mathbf{v}_2}\mathbf{Z}(p)^{\mathbf{v}_2}$ are commutative if and only if $\omega^{\mathbf{u}_1\bullet\mathbf{v}_2} = \omega^{\mathbf{u}_2\bullet\mathbf{v}_1}$ hence $\mathbf{u}_1\bullet\mathbf{v}_2 = \mathbf{u}_2\bullet\mathbf{v}_1$. Then, the representation of the quantum stabilizer code for the qupits case is $\mathbf{H} = \begin{bmatrix} \mathbf{H}_{\mathbf{X}(p)} & \mathbf{H}_{\mathbf{Z}(p)} \end{bmatrix}$, where $\mathbf{H}_{\mathbf{X}(p)}$, $\mathbf{H}_{\mathbf{Z}(p)}$ are the matrices over the Galois field and the symplectic inner product (SIP), which is satisfies

$$\mathbf{H}_{\mathbf{X}(p)} \otimes {\mathbf{H}_{\mathbf{Z}(p)}}^T = \ \mathbf{H}_{\mathbf{Z}(p)} \otimes {\mathbf{H}_{\mathbf{X}(p)}}^T, \tag{1.7}$$

Here, the multiplication ($\otimes$) and summation ($\oplus$) operators are over the $GF(p)$.

The CSS structure is an advantageous construction for quantum codes since the quantum codes can be investigated using the best classical codes based on the CSS structure. In addition, both quantum binary codes and quantum non-binary codes can be constructed by the CSS structure. Hence, we summarize the generalized CSS structure for the construction of binary and non-binary quantum codes in the following Lemma.

Lemma 1.1 *(Quantum CSS structure) Let C_1 and C_2 be two linear codes with parameters $[[n, k_1, d_1]]_p$ and $[[n, k_2, d_2]]_p$, respectively. If ${C_2}^{\perp} \subseteq C_1$, then there exists a quantum code with the parameter $[[n, k_1 + k_2 - n, d \geq min\{d_1, d_2\}]]_p$.*

As with the stabilizer formalism, the parity-check matrix of quantum codes based on the CSS structure can be expressed as follows:

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}(C_2) & \mathbf{0} \\ \mathbf{0} & \mathbf{H}(C_1) \end{bmatrix}.$$

The SIP for the above matrix is given as $\mathbf{H}(C_2) \times \mathbf{H}(C_1)^T = \mathbf{0}$.

1.2.3 QECCs over Galois field 4

The Galois field with 2 elements (*GF(2)*) is defined over the set of 0 and 1, i.e., *GF(2)* = {0, 1}, under addition and multiplication forms that satisfy pre-defined axioms. Galois field *GF(4)* is an extension of *GF(2)*, where its primitive element is ω,

Table 1.1: Mapping between Pauli operators and *GF(4)* elements.

I	0
X	1
Y	$\omega^2 = \omega + 1$
Z	ω
Multiplication operator	Addition
Commutative	Trace-inner product

where $\omega^2 = \omega+1$. Hence, *GF(4)* is the set of four elements $\{0,\ 1,\ \omega,\ \omega+1\}$ for additive form or $\{0,\ 1,\ \omega,\ \omega^2\}$ for multiplicative form. We define some basic functions over *GF(4)* as follows:

1. Conjugate function: For any $x \in GF(4)$: $\overline{x} = x^2$.

2. Trace function: For any $x \in GF(4)$, Tr : $GF(4) \rightarrow GF(2)$, $Tr(x) = x + \overline{x} = x + x^2$.

3. Trace-inner product: For two vectors over $GF(4)$: $\mathbf{u} = (u_1, u_2, ..., u_n)$ and $\mathbf{v} = (v_1, v_2, ..., v_n)$, Trace-inner product: ” $\bullet$ ” : $GF(4)^n \rightarrow GF(2)$, we have: $\mathbf{u} \bullet \mathbf{v} = Tr(\mathbf{u} \times \overline{\mathbf{v}}) = \sum\limits_{i=1}^{n} Tr(u_i \times {v_i}^2)$.

A quantum stabilizer code can be considered as an additive code over the finite field *GF(4)* by identifying the four Pauli operators with the elements of *GF(4)*. The mapping between the Pauli operators and elements of *GF(4)* is shown in Table 1.1.

Proposition: We can consider a quantum stabilizer code as an additive code over the finite field *GF(4)* by identifying the four Pauli operators with the elements of *GF(4)*. We denote *GF(4)* $= \{0, 1, \omega, \omega^2\}$ where $\omega^2 = \omega + 1$. The mapping is in Table 1.1.

Given four elements in *GF(4)*, some products have been defined as follows:

Definition: Conjugation in *GF(4)* is defined by $\overline{x} := x^2$. The trace function, tr : *GF(4)* $\rightarrow$ *GF(2)*, is defined by $tr(x) := x + \overline{x}$. The trace inner product of two vector lengths, n, over *GF(4)*, $\mathbf{u}$ and $\mathbf{v}$, is given by $\mathbf{u}*\mathbf{v} := \sum\limits_i tr(u_i\overline{v_i})$. The Hermitian product (.) of two vector lengths, n, over *GF(4)*, $\mathbf{u}$ and $\mathbf{v}$, is defined as $\mathbf{u}.\mathbf{v} := \sum\limits_i u_i\overline{v_i}$.

In particular, we have: $tr(0) = tr(1) = 0$, $tr(\omega) = tr(\omega^2) = 1$ and $\overline{0} = 0$, $\overline{1} = 1$, $\overline{\omega} = \omega^2$. Hence, as in the mapping in Table 1.1, adding two vectors over *GF(4)* corresponds to multiplying two mapping Pauli operators. For single Pauli operators, they commute when, in the first case, one of them is $\mathbf{I}$, or in the second case, when they are equal to each other. Hence, their trace product is always 0, due to $tr(0) = 0$ and when $x \neq 0, x^3 = 1$, then, $tr(x^3) = tr(1) = 0$. Otherwise, the trace product of single Pauli operators is 0. From that, Pauli operators arising from vector $\mathbf{x}$, $\mathbf{y}$ commute when the trace product (components-wise) is even; or stated another way, $\mathbf{x} * \mathbf{y} = \sum tr(x_i\overline{y_i}) = 0$. To state the relationship between QECC and additive code over *GF(4)*, the two following lemmas have been studied:

Lemma 1: An additive code $(n, 2^{n-k})$ code C such that there are no vectors of weight $< d$ in $C^{\perp}\backslash C$, where $C^{\perp}$ is the Hermitian dual of C, that yields a quantum code with parameters $[[n, k, d]]$.

Lemma 2: Linear code C is self-orthogonal with respect to trace if and only if it is self-orthogonal with respect to Hermitian product.

Proof: We notice C is linear, which means if $\mathbf{u}, \mathbf{v} \in C$, we have $\omega\mathbf{u}$, $\omega\mathbf{v} \in C$.

From $\mathbf{u}.\overline{\mathbf{v}} = \alpha + \omega\beta$, we have: $0 = tr(\mathbf{u}.\overline{\mathbf{v}}) = tr(\alpha + \beta\omega) = \beta$. From $\omega^2\mathbf{u}.\overline{\mathbf{v}} = \omega^2\alpha + \omega^3\beta = \omega^2\alpha + \beta$, we have $0 = tr(\omega^2\mathbf{u}.\overline{\mathbf{v}}) = tr(\omega^2\alpha + \beta) = \alpha$. Since the linear code C is an additive code, its parameters are $(n, 2^{2k}) = (n, 2^{n-(n-2k)})$. Therefore, as Lemma 1.2.3, the stabilizer code $[[n, n-2k, d]]$ is obtained.

Combining Lemma 1.2.3 and Lemma 1.2.3, we have the following corollary:

Corollary: Let C be a Hermitian self-orthogonal linear $[n, k]$ code over *GF(4)* such that there are no vectors of weight $< d$ in $C^{\perp} \backslash C$, where $C^{\perp}$ is the Hermitian dual of C. Then, there is a quantum stabilizer code $[[n, n-2k, d]]$.

From Corollary 1.2.3, we will change the problem of building the stabilizer code into finding a Hermitian self-orthogonal linear code. First, it is an additive code over *GF(4)*; then, it actually requires the two following conditions for each row vector, (for example, $\mathbf{u}_1$, $\mathbf{u}_2$):

1. to be orthogonal to each other; that is, $\mathbf{u}_i.\overline{\mathbf{u}_j} = 0$ for any $i, j \in \{1, 2\}$; and
2. to be orthogonal to itself; that is, the weight of $\mathbf{u}_1$, $\mathbf{u}_2$ (the number of elements with difference 0) has to be even.

1.2.4 Entanglement-assisted QECC

An EAQECC is an extension of quantum stabilizer codes with parameter $[[n,\ k,\ d_{\min};\ c]]$. Like classical coding theory, it also encodes k logical qubits into n physical qubits but with the help of c copies of maximally entangled Bell states. It has been shown that EAQECCs have considerable advantages over standard quantum stabilizer codes from pre-existing entanglement between transmitter and receiver to improve the reliability of transmission. In addition, while quantum stabilizer codes based on CSS can use dual-containing classical linear binary or quaternary code, non-self-orthogonal codes can be transformed into an EAQECCs.

Let a size of a group be the number of elements in the group. If $\boldsymbol{S}$ is the non-Abelian in Pauli group P_n of size m, then there exists a set of generators for $\boldsymbol{S}$ of the form $\{\mathbf{Z}_1,\ \mathbf{Z}_2,\ \dots\ ,\ \mathbf{Z}_{s+c},\ \mathbf{X}_{s+1},\ \mathbf{X}_{s+2},\ \dots\ ,\ \mathbf{X}_{s+c}\}$ (where $s\ +\ c\ =\ m$) with the following commutation properties:

$$\begin{cases} [\mathbf{Z}_i, \mathbf{Z}_j] = 0 \text{ and } [\mathbf{X}_i, \mathbf{X}_j] = 0 \text{ for all } i, j; \\ [\mathbf{X}_i, \mathbf{Z}_j] = 0 \text{ for all } i \neq j; \\ \{\mathbf{X}_i, \mathbf{Z}_i\} = 0 \text{ for all } i, \end{cases} \tag{1.8}$$

where $[\mathbf{A},\ \mathbf{B}]$ and $\{\mathbf{A},\ \mathbf{B}\}$ are a commutator and a anti-commutator of generator $\mathbf{A}$ and $\mathbf{B}$, respectively. The $[\mathbf{A},\ \mathbf{B}]$ and $\{\mathbf{A},\ \mathbf{B}\}$ of generator $\mathbf{A}$ and $\mathbf{B}$ can be expressed as $\mathbf{A}\ \times\ \mathbf{B}\ -\ \mathbf{B}\ \times\ \mathbf{A}$ and $\mathbf{A}\ \times\ \mathbf{B}\ +\ \mathbf{B}\ \times\ \mathbf{A}$, respectively. Then, the non-Abelian group can be partitioned into:

1. A commuting subgroup, the isotropic group $\boldsymbol{S}_{\mathrm{I}} = \{\mathbf{Z}_{c+1},\ \mathbf{Z}_{c+2},\ \dots\ ,\ \mathbf{Z}_{c+s}\}$.

2. Entanglement subgroup pairs $\boldsymbol{S}_{\mathrm{E}} = \{\mathbf{Z}_1,\ \mathbf{Z}_2,\ \dots\ ,\ \mathbf{Z}_c,\ \mathbf{X}_1,\ \mathbf{X}_2,\ \dots\ ,\ \mathbf{X}_c\}$ with anti-commuting pairs; the anti-commuting pairs $(\mathbf{Z}_i,\ \mathbf{X}_i)$ being shared between source and receiver.

The Gram–Schmidt procedure to drop the non-Abelian group into the partitions of operators with the above properties (called the isotropic and entanglement subgroup) was introduced and discussed [90]. From the isotropic and entanglement subgroup, EAQECC code C_{EA} are defined as $[[n,\ k;\ c]]$ that encodes $k = n\ -\ s\ -\ c$ qubits into n physical qubits with the help of $s = n\ -\ k\ -\ c$ ancillas qubits and c ebits shared between the sender and receiver, that can correct any error from the following set of errors, N:

$$N = \left\{ \mathbf{E}_m \;\middle|\; \forall \mathbf{E}_1,\ \mathbf{E}_2 \ \Rightarrow\ {\mathbf{E}_2}^{\dagger}\mathbf{E}_1 \ \in\ \boldsymbol{S}_{\mathbf{I}} \ \cup\ (\mathrm{P}_n \ -\ N(\boldsymbol{S})) \right\}.$$

Code space C_{EA} is a simultaneous eigenspace of the Abelian extension of $\boldsymbol{S}$ [96]. The Abelian extension is Galois extension by using ancilla operators. From N, we can determine $d_{\min}$, and it tells us the detectable and correctable EAQECCs.

From the isotropic and entanglement subgroup, the operation of EAQECC can be considered. For example, the following state shared between A (Alice) and B (Bob) is an entanglement state:

$$|\Phi\rangle = \frac{1}{\sqrt{2}} \left(|0\rangle \otimes |0\rangle + |1\rangle \otimes |1\rangle \right)$$

The first half of the entanglement pair belongs to Alice and the second half to Bob. Then, the operating principle is illustrated in Figure 1.2. The Sender A encodes the quantum information state $|\psi\rangle$ with the help of local ancillary qubits $|0\rangle$ and her half of shared ebits, $|\Phi\rangle$, and then shares the encoded qubits over a noisy quantum channel. The Receiver B performs multi-qubit measurement on all qubits to diagnose the channel error and perform recovery unitary operation R to reserve the action of the channel.

The most important relationship between EAQECCs and classical codes is given in the following theorem [90, 92]:

Theorem 1. *Let C be a binary classical code [n, k, d] with parity check matrix* $\mathbf{H}$. *We can obtain a corresponding [[n, 2k − n + c, d; c]]* EAQECC, *where $c = rank \left\{ \mathbf{H}\mathbf{H}^T \right\}$*

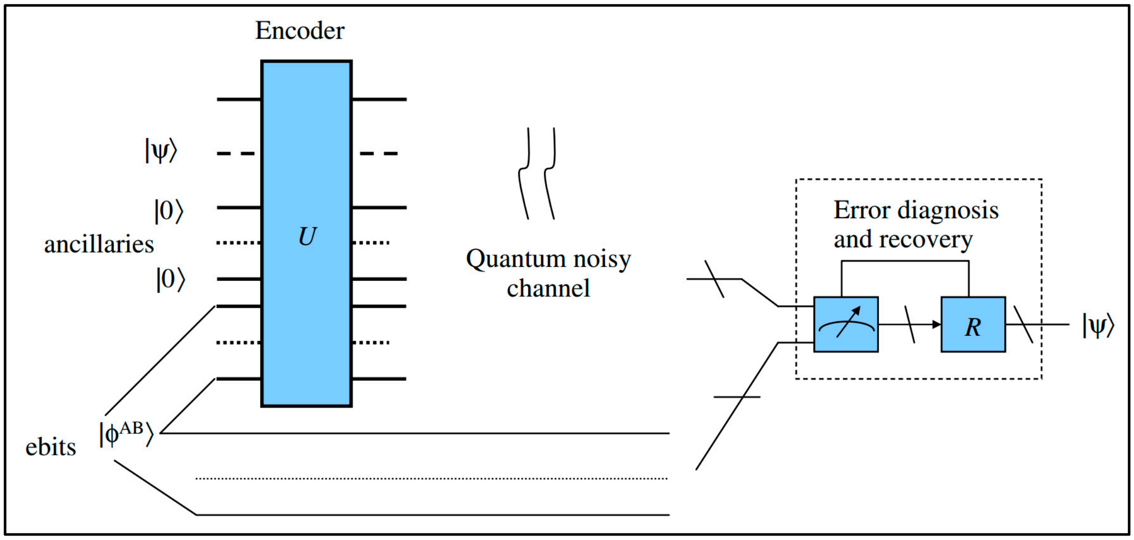


Figure 1.2: Entanglement-assisted (EA) quantum error correction operating principle.

is the number of ebits needed.

As a consequence, there are lots of papers using this theorem for EAQECC construction. From binary code C with parameter [n, k, d] and the generator matrix $\mathbf{G} = [\mathbf{I}_k \mid \mathbf{A}_{k\text{x}(n\text{-}k)}]$, the EAQECC with [[$2n - k$, k, d'; c]] can be made [92], where $c = 2n - 2k$ and $d' \geq d$. Tomas [93] used the generalized quadrangle GQ(s,1) for the construction of classical binary code and proved the number of ebits is 2. The circulant matrix $\mathbf{P}_m = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ 1 & 0 & 0 & \dots & 0 \end{bmatrix}$, and $\mathbf{A}_m{}^{(\text{i})}$ (i = 0, 1, 2, … , $m - 1$) is an $m \times m$ binary matrix where the (i + 1) row is **1** and other rows are **0**, and defines classical binary codes [94]. Then, the number of ebits is proven to be 1 for some conditions. Qian and Zhang [95] used shortened Hamming codes with parameter $\left[\frac{m(m-1)}{2}, \frac{(m-1)(m-2)}{2}, 3\right]$, and proved EAQECCs with parameter $\left[\left[\frac{m(m-1)}{2}, \frac{(m-1)(m-4)}{2} + 1, 3; 1\right]\right]$ exist if m is even.

1.3 Contribution and layout of the thesis

The dissertation consists of seven chapters structured as follows:

In Chapter 1, we sum up all the background knowledge relevant to my studies of quantum information theory, quantum error correction codes, and related works of construction of quantum error correction codes. Then, we present the outline of the dissertation.

In Chapter 2, we study the construction of quantum stabilizer codes based on difference sets. From the suitable DSs, the circulant matrices are designed and used to construct the parity-check matrix. Then, the generators of the stabilizer should first be chosen to make independent rows of parity-check matrix. Finally, the codewords and minimum distance are determined.

In Chapter 3, we study the construction of quantum codes from symmetric matrices that are based on the CSS structure. The parity-check matrices are first generated from two constructions and proven to satisfy the symplectic inner product for the construction of binary and non-binary quantum stabilizer codes. Then, the parameters of these codes are calculated and explained in detail. Some quantum codes are proven to achieve equality of the quantum singleton bound.

In Chapter 4, we study the approach to the construction of additive codes over *GF(4)*, which are self-orthogonal with respect to Hermitian product. The minimum distance of this classical linear code was proved to be 4 in all cases. The corresponding quantum stabilizer code can be transformed from this classical code; we prove all the optimal codes that can be accomplished from this construction with lengths 5, 6, 7, 8, 9, and 10.

In Chapter 5, we study the construction of self-orthogonal trace-inner product codes over *GF(4)*. From two binary vectors, we generate the circulant and modified circulant matrices, and the generator matrix for quaternary linear codes is proposed. Then, the quantum stabilizer codes are derived from the linear codes. The advantage of the proposed construction is that our proposed codes give various dimensions of QECCs, and these minimum distances have good values.

In Chapter 6, we study the novel approaches to construction of EAQECC. First, we propose a new method for the construction of the isotropic subgroup based on circulant matrices. Then, the entanglement subgroup can be determined from a method of transforming the isotropic group into standard form; hence, the parameters of codes are found, and for effective preparation of the entangled state, the number of ebits should be as few as possible. To explain the practical construction of the quantum codes, design of the proposed EAQECC with lengths up to 12 are shown. In addition, the minimum distance is calculated and explained to show that the proposed construction has good correctable capability, in comparison with recent EAQECC.

Finally, Chapter 7 concludes the dissertation with a summary and discussion of future research directions.

Chapter 2

New Constructions of Quantum Stabilizer Codes Based on Difference Sets

2.1 Introduction

Low-density parity-check (LDPC) codes were first introduced by Gallager [22]. Then, an excellent performance close to Shannon channel capacity was obtained according to large block size of binary parity-check matrix in classical communication [23]. Innovative designs of the parity-check matrix have been proposed for LDPC codes with better performances or with easy implementation. The application of combinatoric design on LDPC codes was proposed to increase the girth of the parity-check matrices [24]. Adaptive selection of quasi-cyclic LDPC (QC-LDPC) codes suitable for visible light communication had been studied to adjust the dimming control [25]. New quantum codes have been proposed based on LDPC codes with the Calderbank-Shor-Steane (CSS) form in [26,27] and quantum LDPC with the non-CSS form in [28, 29].

A difference set (DS) in combinatorics [30–32] is defined as a subset in which each difference of two elements occurs in the group. Perfect DSs have been used to build up cyclic codes which have remarkable performance in classical channels. Hence, the new trial using DSs on quantum code was first studied in [33] where DSs are used to construct dual-containing sparse-graph codes for QECCs. Further, one-time DSs were used to construct entanglement-assisted quantum LDPC codes in [34] and these quantum codes have shown a significant improvement in the error probability performance. The quantum QC-LDPC codes based on the DSs in [35], where the set of DSs is easily generated by only a single parameter; however, a lot of the DSs cannot be defined except for prime numbers of the form $n = 4k - 1$, where k is even number.

In this chapter, new constructions of quantum stabilizer codes based on DSs are proposed. From the suitable DSs, the circulant matrices are designed and used to construct the parity-check matrix. Then, the generators of the stabilizer should first be chosen to make independent rows of parity-check matrix. Finally, the codeword and minimum distance are determined. Two quantum stabilizer codes with lengths of seven and 15 from the proposed design are shown to express the practical application.

2.2 Proposed construction

2.2.1 Difference sets and shifted difference sets

A $(n,\, k, \lambda)$ difference set (DS) $D = \{d_1,\, d_2,\, \ldots,\, d_k\}$ is defined as a collection of k residues ($\in \{0,\, 1,\, 2,\, \ldots,\, n-1\}$). Then, for any residue $\alpha \neq 0$, the congruence $d_i - d_j = \alpha$ (modulo n) has exactly λ solution pairs (d_i, d_j) with $d_i, d_j \in D$. The necessary condition of the parameters (n, k, λ) is $k(k-1) = \lambda(n-1)$ [30]. Assume that the (n, k, λ) DS $D = \{d_1,\, d_2,\, \ldots,\, d_k\}$ is given, then the shifted set $D(s) = \{d_1 + s,\, d_2 + s,\, \ldots,\, d_k + s\}$ is also a new DS with the same parameters (n, k, λ). A DS with three elements and its shifted DS are shown in Example 1.

Example 1. *A perfect DS is (7, 3, 1) with* $D = \{1,\, 2,\, 4\}$*,*

$$\begin{cases} 1-2\equiv 6 \quad 2-1\equiv 1 \quad 4-1\equiv 3 \\ 1-4\equiv 4 \quad 2-4\equiv 5 \quad 4-2\equiv 2 \end{cases} \text{modulo}\, 7.$$

The shifted (7, 3, 1) DS with offset 6 is $D\ (6) = \{0,\, 1,\, 3\}$*,*

$$\begin{cases} 0-1\equiv 6 \quad 1-0\equiv 1 \quad 3-0\equiv 3 \\ 0-3\equiv 4 \quad 1-3\equiv 5 \quad 3-1\equiv 2 \end{cases} \text{modulo}\, 7.$$

The notation $D(s)$ stands for the shifted DS from D with the offset s.

2.2.2 Circulant permutation matrices

Let $\mathbf{I}_n$ be the identity matrix of size $n \times n$. Then, $\mathbf{I}_n(x)$ is the shift of $\mathbf{I}_n$ where the rows of $\mathbf{I}_n$ are circularly shifted to the right by x positions $(0 \leq x \leq n-1)$.

Generally, we notice that $\mathbf{I}_n(0) = \mathbf{I}_n$ and $\mathbf{I}_n(x \pm kn) = \mathbf{I}_n(x)$ for any integer k. Let $\mathbf{I}_n(1)^c$ be the c times of multiplying $\mathbf{I}_n(1)$, we have $\mathbf{I}_n(1)^c = \mathbf{I}_n(c)$ $(0 \leq c \leq n-1)$.

Example 2. *With $n = 4$, we have:*

$$\mathbf{I}_4(0) = \mathbf{I}_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \mathbf{I}_4(2) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \text{ and}$$

$$\mathbf{I}_4(2) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} = \mathbf{I}_4(1)^2.$$

A $n \times n$ circulant permutation binary matrix $\mathbf{P}_n$ is defined as

$$\mathbf{P}_n = \begin{bmatrix} i_0 & i_1 & i_2 & \cdots & i_{n-1} \\ i_{n-1} & i_0 & i_1 & \cdots & i_{n-2} \\ i_{n-2} & i_{n-1} & i_0 & \cdots & i_{n-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ i_1 & i_2 & i_3 & \cdots & i_0 \end{bmatrix},$$

where i_k is the binary value. $\mathbf{P}_n$ can be given as the linear combination of identity matrix and its shifted matrices.

$$\mathbf{P}_n = i_0 \times \mathbf{I}_n(0) + i_1 \times \mathbf{I}_n(1) + i_2 \times \mathbf{I}_n(2) + \ldots + i_{n-1} \times \mathbf{I}_n(n-1). \tag{2.1}$$

It is assumed that $i_0 + i_1 + \ldots + i_{n-1} = k$. Let $t_0¡t_1 < \ldots < t_{k-1}$ be the position index of nonzero elements in the sequence set $\{i_0, i_1, \ldots, i_{n-1}\}$. For example, if the sequence set $\{i_0, i_1, \ldots, i_{n-1}\}$ is $\{1, 1, 0, 0, 1, 0, 1\}$, then $t_0 = 0$, $t_1 = 1$, $t_2 = 4$, and $t_3 = 6$. The matrix $\mathbf{P}_n$ can also be expressed by using the Hall-polynomial form $p_n(x)$ [30] as

$$p_n(x) = x^{t_0} + x^{t_1} + \ldots + x^{t_{k-1}} \tag{2.2}$$

Let T be the transpose operator. Then, the transpose matrix of $\mathbf{P}_n$ is denoted as $\mathbf{P}_n{}^T$. Let $p_n(x)^T$ be the Hall-polynomial form of $\mathbf{P}_n{}^T$. Then, the polynomial $p_n(x)^T$ is expressed as

$$p_n(x)^T = x^{-t_0} + x^{-t_1} + \ldots + x^{-t_{k-1}}, \tag{2.3}$$

where $t_0, t_1, \ldots, t_{k-1}$ are the values in (8). For a (n, k, λ) DS $D = \{d_1, d_2, \ldots, d_k\}$, the circulant permutation matrix $\mathbf{P}_n$ in (7) is made where the element i_j is 1 if $j \in D$ and is 0 otherwise. Then, the Hall-polynomial form $p_n(x)^D$ for the DS D is expressed as

$$p_n(x)^D = x^{d_1} + x^{d_2} + \ldots + x^{d_k} \tag{2.4}$$

2.2.3 Construction of quantum stabilizer code based on DS

With difference sets (n, k, λ) D, the product of the two circulant permutation matrices can be expressed as a function of parameter of DS and the shift values in the following theorem.

Theorem 1. *Let $h_1(x)$ and $h_2(x)$ be the Hall-polynomials of $D(s_1)$ and $D(s_2)$, which are defined as $h_1(x) = p_n{}^{D(s_1)}$ and $h_2(x) = p_n{}^{D(s_2)}$, respectively. Let the circulant permutation matrices $\mathbf{H}_1$ and $\mathbf{H}_2$ correspond to $h_1(x)$ and $h_2(x)$, respectively. Then, the product of the two polynomials $h_1(x)$, $h_2(x)^T$ and the product of the two matrices*

$\mathbf{H}_1$ *and* ${\mathbf{H}_2}^T$ *are given as*

$$h_1(x)\times h_2(x)^T = (k-\lambda)\times x^{s_1-s_2}+\lambda\times\sum_{l=0}^{n-1} x^l \text{ and } \mathbf{H}_1\times{\mathbf{H}_2}^T = (k-\lambda)\times\mathbf{I}_n(s_1-s_2)+\lambda\times\mathbf{J}_n,$$

where the size of matrix $\boldsymbol{J}_n$ *is* $n \times n$ and whose entries are all one.

Proof. From the definition of the Hall-polynomial, $h_1(x)$ and $h_2(x)$ can be expressed as

$$h_1(x) = x^{d_1+s_1} + x^{d_2+s_1} + \ldots + x^{d_k+s_1} \text{ and } h_2(x) = x^{d_1+s_2} + x^{d_2+s_2} + \ldots + x^{d_k+s_2}.$$

Then, the Hall-polynomial $h_2(x)^T$ for (9) is given as $h_2(x)^T = x^{-d_1-s_2} + x^{-d_2-s_2} + \ldots + x^{-d_k-s_2}$. Therefore, the product of the two polynomials $h_1(x)$ and $h_2(x)^T$ is given as

$$\begin{aligned} h_1(x)\times h_2(x)^T &= (x^{d_1+s_1} + x^{d_2+s_1} + \ldots + x^{d_k+s_1}) \times (x^{-d_1-s_2} + x^{-d_2-s_2} + \ldots + x^{-d_k-s_2}) \\ &= \sum_{i=1}^{k} \left[x^{(d_i+s_1)-(d_1+s_2)} + x^{(d_i+s_1)-(d_2+s_2)} + \ldots + x^{(d_i+s_1)-(d_k+s_2)}\right] \\ &= \sum_{i=1}^{k} x^{s_1-s_2} \times \left[x^{d_i-d_1} + x^{d_i-d_2} + \ldots + x^{d_i-d_k}\right] \\ &= x^{s_1-s_2} \times \sum_{u=1}^{k}\sum_{v=1}^{k} x^{d_u-d_v} = x^{s_1-s_2} \times \left[k \times x^0 + \sum_{u=1}^{k}\sum_{v=1,\, v\neq u}^{k} x^{d_u-d_v}\right]. \end{aligned} \tag{2.5}$$

$\sum_{u=1}^{k}\sum_{v=1,\, v\neq u}^{k} x^{d_u-d_v}$ in 2.5 can be expressed as

$$\sum_{u=1}^{k}\sum_{v=1,\, v\neq u}^{k} x^{d_u-d_v} = \lambda\times\sum_{l=1}^{n-1} x^l = \lambda\times\sum_{l=0}^{n-1} x^l - \lambda\times x^0.$$

Hence, Equation 2.5 is expressed as

$$\begin{aligned} &x^{s_1-s_2}\times\left[k\times x^0 + \sum_{u=1}^{k}\sum_{v=1,\, v\neq u}^{k} x^{d_u-d_v}\right] = x^{s_1-s_2}\times\left[k\times x^0 + \lambda\times\sum_{l=0}^{n-1} x^l - \lambda\times x^0\right] \\ &= (k-\lambda)\times x^{s_1-s_2} + \lambda\times x^{s_1-s_2}\times\sum_{l=0}^{n-1} x^l = (k-\lambda)\times x^{s_1-s_2} + \lambda\times\sum_{l=0}^{n-1} x^l. \end{aligned} \tag{2.6}$$

Since the circulant permutation matrices corresponding to the polynomials $x^{s_1-s_2}$ and $\sum_{l=0}^{n-1} x^l$ are $\mathbf{I}_n(s_1 - s_2)$ and $\mathbf{J}_n$, respectively, the product of $\mathbf{H}_1$ and ${\mathbf{H}_2}^T$ is expressed as

$$\mathbf{H}_1 \times {\mathbf{H}_2}^T = (k - \lambda) \times \mathbf{I}_n(s_1 - s_2) + \lambda \times \mathbf{J}_n \tag{2.7}$$

Therefore, the expressions in 2.6 and 2.7 prove Theorem 1.□

Since the product of $\mathbf{H}_1$ and ${\mathbf{H}_2}^T$ in Theorem 1 is expressed as the function of k, λ, s_1, and s_2, the constraint on parameter of DSs to satisfy the SIP condition of parity-check matrix is explained in the following theorem.

Theorem 2. *For any (n, k, λ) DS D where $k \equiv \lambda$ modulo 2 and any integers $s_1 \neq s_2$ where $s_1, s_2 \in \{0, 1, \ldots, n-1\}$, parity-check matrix $\mathbf{H} = [\mathbf{H}_1|\mathbf{H}_2]$ where $\mathbf{H}_1$ and $\mathbf{H}_2$ corresponding to $h_1(x) = {p_n}^{D(s_1)}$ and $h_2(x) = {p_n}^{D(s_2)}$, respectively, satisfies the SIP condition (2).*

Proof. From Theorem 1, we have:

$$\mathbf{H}_1 \times {\mathbf{H}_2}^T = (k - \lambda) \times \mathbf{I}_n(s_1 - s_2) + \lambda \times \mathbf{J}_n \tag{2.8}$$

$$\mathbf{H}_2 \times {\mathbf{H}_1}^T = (k - \lambda) \times \mathbf{I}_n(s_2 - s_1) + \lambda \times \mathbf{J}_n \tag{2.9}$$

The summation of 2.8 and 2.9 is

$$\begin{aligned} \mathbf{H}_1 \times {\mathbf{H}_2}^T + \mathbf{H}_2 \times {\mathbf{H}_1}^T &= (k - \lambda) \times \mathbf{I}_n(s_1 - s_2) + \lambda \times \mathbf{J}_n + (k - \lambda) \times \mathbf{I}_n(s_2 - s_1) + \lambda \times \mathbf{J}_n \\ &= (k - \lambda) \times [\mathbf{I}_n(s_1 - s_2) + \mathbf{I}_n(s_2 - s_1)] + 2\lambda \times \mathbf{J}_n. \end{aligned} \tag{2.10}$$

If $k - \lambda$ is even, all elements of the matrix $(k - \lambda) \times [\mathbf{I}_n(s_1 - s_2) + \mathbf{I}_n(s_2 - s_1)]$ in 2.10 are even. Moreover, all elements of the matrix $2\lambda \times \mathbf{J}_n$ in 2.10 are also even.

Table 2.1: Difference sets (DSs) with parameters $k \equiv \lambda$ modulo 2.

No	$\boldsymbol{n,k,\lambda}$	**Difference Set**
1	7, 3, 1	1 2 4.
2	7, 4, 2	0 3 5 6.
3	15, 7, 3	0 1 2 4 5 8 10.
4	21, 5, 1	3 6 7 12 14.
5	23, 11, 5	1 2 3 4 6 8 9 12 13 16 18.
6	31, 15, 7	1 2 3 4 6 8 12 15 16 17 23 24 27 29 30.
7	47, 23, 11	1 2 3 4 6 7 8 9 12 14 16 17 18 21 24 25 27. 28 32 34 36 37 42.
8	199, 99, 49	1 2 4 5 7 8 9 10 13 14 16 18 20 23 25 26 28 29 31 32 33 35 36 40 43 45 46 47 49 50 51 52 53 56 57 58 61 62 63 64 65 66 70 72 79 80 81 86 89 90 91 92 94 98 100 102 103 104 106 111 112 114 115 116 117 121 122 123 124 125 126 128 130 131 132 139 140 144 145 151 155 157 158 160 161 162 165 169 172 175 177 178 180 182 184 187 188 193 196.

Then, all elements of the matrix $(k-\lambda)\times[\mathbf{I}_n(s_1-s_2)+\mathbf{I}_n(s_2-s_1)]+2\lambda\times\mathbf{J}_n$ in 2.10 are even. Therefore, if $k \equiv \lambda$ modulo 2, the equation $\mathbf{H}_1 \times {\mathbf{H}_2}^T + \mathbf{H}_2 \times {\mathbf{H}_1}^T = 0_n$ is always true. Therefore, the parity-check matrix $\mathbf{H}$ of $\mathbf{H}_1$ and $\mathbf{H}_2$ which is made from the parameter of DS with the constraint $k \equiv \lambda$ modulo 2 satisfies the SIP condition.□

In Table 2.1, eight DSs with the constraint $k \equiv \lambda$ modulo 2 are listed among the DSs in. For the practical applications of proposed construction, two DSs with parameters (7, 4, 2) and (15, 7, 3) are considered in Examples 3 and 4.

Example 3. *For the DS $D = \{0,\ 3,\ 5,\ 6\}$ with parameter (7, 4, 2), two shifted DSs are considered as $D(1) = \{0+1,\ 3+1,\ 5+1,\ 6+1\} = \{0,\ 1,\ 4,\ 6\}$, $D(4) = \{0+4,\ 3+4,\ 5+4,\ 6+4\} = \{0,\ 2,\ 3,\ 4\}$. Then, the Hall-polynomials for*

D(1) and D(4) are $h_1(x) = {p_7}^{D(1)}$ *and* $h_2(x) = {p_7}^{D(4)}$*, respectively. Therefore, the corresponding binary matrices for the Hall-polynomials are given as*

$$\mathbf{H}_1 = \begin{bmatrix} 1\,1\,0\,0\,1\,0\,1 \\ 1\,1\,1\,0\,0\,1\,0 \\ 0\,1\,1\,1\,0\,0\,1 \\ 1\,0\,1\,1\,1\,0\,0 \\ 0\,1\,0\,1\,1\,1\,0 \\ 0\,0\,1\,0\,1\,1\,1 \\ 1\,0\,0\,1\,0\,1\,1 \end{bmatrix}, \mathbf{H}_2 = \begin{bmatrix} 1\,0\,1\,1\,1\,0\,0 \\ 0\,1\,0\,1\,1\,1\,0 \\ 0\,0\,1\,0\,1\,1\,1 \\ 1\,0\,0\,1\,0\,1\,1 \\ 1\,1\,0\,0\,1\,0\,1 \\ 1\,1\,1\,0\,0\,1\,0 \\ 0\,1\,1\,1\,0\,0\,1 \end{bmatrix}, \mathbf{H} = [\mathbf{H}_1|\mathbf{H}_2]. \tag{2.11}$$

It follows that two products $\mathbf{H}_1 \times {\mathbf{H}_2}^T$ *and* $\mathbf{H}_2 \times {\mathbf{H}_1}^T$ *are given by:*

$$\mathbf{H}_1 \times {\mathbf{H}_2}^T = \begin{bmatrix} 2\,2\,2\,2\,4\,2\,2 \\ 2\,2\,2\,2\,2\,4\,2 \\ 2\,2\,2\,2\,2\,2\,4 \\ 4\,2\,2\,2\,2\,2\,2 \\ 2\,4\,2\,2\,2\,2\,2 \\ 2\,2\,4\,2\,2\,2\,2 \\ 2\,2\,2\,4\,2\,2\,2 \end{bmatrix} = (4-2) \times \mathbf{I}_7(1-4) + 2 \times \mathbf{J}_7,$$

$$\mathbf{H}_2 \times \mathbf{H}_1{}^T = \begin{bmatrix} 2&2&2&4&2&2&2 \\ 2&2&2&2&4&2&2 \\ 2&2&2&2&2&4&2 \\ 2&2&2&2&2&2&4 \\ 4&2&2&2&2&2&2 \\ 2&4&2&2&2&2&2 \\ 2&2&4&2&2&2&2 \end{bmatrix} = (4-2) \times \mathbf{I}_7(4-1) + 2 \times \mathbf{J}_7.$$

Then, the SIP product is $\mathbf{H}_1 \times \mathbf{H}_2{}^T + \mathbf{H}_2 \times \mathbf{H}_1{}^T = \begin{bmatrix} 2&2&2&6&6&2&2 \\ 2&2&2&2&6&6&2 \\ 2&2&2&2&2&6&6 \\ 6&2&2&2&2&2&6 \\ 6&6&2&2&2&2&2 \\ 2&6&6&2&2&2&2 \\ 2&2&6&6&2&2&2 \end{bmatrix} = 0_7$ *modulo 2.*

The seven quantum stabilizer operators corresponding to the seven rows in ***H*** *2.11 are given as*

$\mathbf{g}_1 = \mathbf{YXZZYIX}$; $\mathbf{g}_2 = \mathbf{XYXZZYI}$; $\mathbf{g}_3 = \mathbf{I\,XYXZZY}$;

$\mathbf{g}_4 = \mathbf{YIXYXZZ}$; $\mathbf{g}_5 = \mathbf{ZYIXYXZ}$; $\mathbf{g}_6 = \mathbf{ZZY\ IXYX}$; $\mathbf{g}_7 = \mathbf{XZZYIXY}$.

Among the seven operators, there are a maximum of three linearly independent operators. If $\mathbf{g}_1$, $\mathbf{g}_2$ *and* $\mathbf{g}_3$ *are chosen as the maximum of three linearly independent operators, the other operators are expressed as* $\mathbf{g}_4 = \mathbf{g}_1 \times \mathbf{g}_3$; $\mathbf{g}_5 = \mathbf{g}_1 \times \mathbf{g}_2 \times \mathbf{g}_3$; $\mathbf{g}_6 = \mathbf{g}_1 \times \mathbf{g}_2$; $\mathbf{g}_7 = \mathbf{g}_2 \times \mathbf{g}_3$. *With* $S = \langle \mathbf{g}_1, \mathbf{g}_2, \mathbf{g}_3 \rangle$, *a stabilizer subgroup is composed as*

$S = \{\mathbf{YXZZYIX}, \mathbf{XYXZZYI}, \mathbf{IXYXZZY}, \mathbf{YIXYXZZ}, \mathbf{ZYIXYXZ}, \mathbf{ZZYIXYX},$

$\mathbf{XZZYIXY}, \mathbf{IIIIII}\}.$

Using Equation 1.3, we transform the **H** *matrix in 2.11 into its standard form as*

$$\begin{bmatrix} 1\,0\,0\,1\,0\,1\,1\,0\,1\,1\,1\,0\,0\,1 \\ 0\,1\,0\,1\,1\,1\,0\,1\,1\,0\,0\,1\,0\,1 \\ 0\,0\,1\,0\,1\,1\,1\,1\,1\,1\,0\,0\,1\,0 \end{bmatrix}.$$

Then, as Equation 1.5, the logical operators $\overline{\mathbf{X}}$ *and* $\overline{\mathbf{Z}}$ *are calculated as*

$$\overline{\mathbf{X}} = \begin{bmatrix} 0\,0\,0\,1\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,1\,0\,0\,0\,1\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0\,1\,0\,0\,0\,1\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0\,0\,1\,1\,1\,0\,0\,0\,0\,0 \end{bmatrix} \text{and}\,\overline{\mathbf{Z}} = \begin{bmatrix} 0\,0\,0\,0\,0\,0\,0\,1\,1\,0\,1\,0\,0\,0 \\ 0\,0\,0\,0\,0\,0\,0\,0\,1\,1\,0\,1\,0\,0 \\ 0\,0\,0\,0\,0\,0\,0\,1\,1\,1\,0\,0\,1\,0 \\ 0\,0\,0\,0\,0\,0\,0\,1\,0\,1\,0\,0\,0\,1 \end{bmatrix}.$$

$$\Leftrightarrow \begin{cases} \overline{\mathbf{X}_1} = \mathbf{ZIIXIII} \\ \overline{\mathbf{X}_2} = \mathbf{IZIIXII} \\ \overline{\mathbf{X}_3} = \mathbf{IIZIIXI} \\ \overline{\mathbf{X}_4} = \mathbf{ZZIIIIX} \end{cases} \text{and} \begin{cases} \overline{\mathbf{Z}_1} = \mathbf{ZZIZIII} \\ \overline{\mathbf{Z}_2} = \mathbf{IZZIZII} \\ \overline{\mathbf{Z}_3} = \mathbf{ZZZIIZI} \\ \overline{\mathbf{Z}_4} = \mathbf{ZIZIIIZ} \end{cases}$$

The codewords of the quantum stabilizer code [[7,4]] are expressed as

$$\begin{aligned} |c_1c_2c_3c_4\rangle &= \tfrac{1}{\sqrt{2^3}} \times \left(\prod_{i=1}^{3} (I + g_i)\right) \times \overline{\mathbf{X}_1}^{c_1} \times \overline{\mathbf{X}_2}^{c_2} \times \overline{\mathbf{X}_3}^{c_3} \times \overline{\mathbf{X}_4}^{c_4} |0000000\rangle \\ &= \tfrac{1}{\sqrt{2^3}} \times \overline{\mathbf{X}_1}^{c_1} \times \overline{\mathbf{X}_2}^{c_2} \times \overline{\mathbf{X}_3}^{c_3} \times \overline{\mathbf{X}_4}^{c_4} \left(\sum_{s\in S} s\,|0000000\rangle\right), \end{aligned}$$

where $\prod_{i=1}^{3} (I + g_i) = \sum_{s\in S} s \,\text{and}\, c_i \in \{0, 1\}.$

The minimum distance d_{min} *of the [[7,4]] code is determined by the smallest weight of N(**S**)\**S**. One of the smallest weights is* $\overline{\mathbf{X}_1} \times \mathbf{IIIIIII}$. *Since W(* $\overline{\mathbf{X}_1} \times \mathbf{IIIIIII}$*) = 2, the minimum distance* d_{min} *is 2. Therefore, the quantum stabilizer code from the DS with parameter (7, 4, 2) is [[7,4,2]].*

Example 4. *A DS $D = \{0\ 1\ 2\ 4\ 5\ 8\ 10\}$ with parameters (15, 7, 3) is considered to construct a quantum stabilizer code with length 15. The parity-check matrix is given as $\mathbf{H} = [\mathbf{H}_1\mathbf{H}_2]$ where*

$$\mathbf{H}_1 = \begin{bmatrix} 0&1&1&1&0&1&1&0&0&1&0&1&0&0&0 \\ 0&0&1&1&1&0&1&1&0&0&1&0&1&0&0 \\ 0&0&0&1&1&1&0&1&1&0&0&1&0&1&0 \\ 0&0&0&0&1&1&1&0&1&1&0&0&1&0&1 \\ 1&0&0&0&0&1&1&1&0&1&1&0&0&1&0 \end{bmatrix}, \mathbf{H}_2 = \begin{bmatrix} 0&1&1&0&0&1&0&1&0&0&0&0&1&1&1 \\ 1&0&1&1&0&0&1&0&1&0&0&0&0&1&1 \\ 1&1&0&1&1&0&0&1&0&1&0&0&0&0&1 \\ 1&1&1&0&1&1&0&0&1&0&1&0&0&0&0 \\ 0&1&1&1&0&1&1&0&0&1&0&1&0&0&0 \end{bmatrix}.$$

Five independent generators can be chosen as

$$\begin{cases} \mathbf{g}_1 = \mathbf{IYYX\ IYXZIX\ IXZZZ} \\ \mathbf{g}_2 = \mathbf{Z\ IYYXIYXZIX\ IXZZ} \\ \mathbf{g}_3 = \mathbf{ZZIYYXIYXZIXI\ XZ} \\ \mathbf{g}_4 = \mathbf{ZZZIYYXI\ YXZIXIX} \\ \mathbf{g}_5 = \mathbf{XZZZIYYXIYXZIX\ I} \end{cases}.$$

By using Gaussian elimination, the logical operators $\overline{\mathbf{X}}$ and $\overline{\mathbf{Z}}$ can be written as

$$\overline{\mathrm{X}_1} = \mathbf{Z\ I\ IZZX\ I\ I\ I\ I\ I\ I\ I\ I\ I}, \qquad \overline{\mathrm{Z}_1} = \mathbf{Z\ I\ Z\ I\ ZZ\ I\ I\ I\ I\ I\ I\ I\ I\ I}$$

$$\overline{\mathrm{X}_2} = \mathbf{ZZZI\ I\ I\ X\ I\ I\ I\ I\ I\ I\ I\ I}, \qquad \overline{\mathrm{Z}_2} = \mathbf{ZZZZZ\ I\ Z\ I\ I\ I\ I\ I\ I\ I\ I}$$

$$\overline{\mathrm{X}_3} = \mathbf{I\ ZZZ\ I\ I\ I\ X\ I\ I\ I\ I\ I\ I\ I}, \qquad \overline{\mathrm{Z}_3} = \mathbf{ZZ\ I\ Z\ I\ I\ I\ Z\ I\ I\ I\ I\ I\ I\ I}$$

$$\overline{\mathrm{X}_4} = \mathbf{I\ IZZZ\ I\ I\ IX\ I\ I\ I\ I\ I\ I}, \qquad \overline{\mathrm{Z}_4} = \mathbf{I\ Z\ Z\ IZI\ I\ I\ Z\ I\ I\ I\ I\ I\ I}$$

$$\overline{\mathrm{X}_5} = \mathbf{ZIZZ\ I\ I\ I\ I\ IX\ I\ I\ I\ I\ I}, \qquad \overline{\mathrm{Z}_5} = \mathbf{Z\ I\ I\ Z\ ZI\ I\ I\ I\ Z\ I\ I\ I\ I\ I}$$

$$\overline{\mathrm{X}_6} = \mathbf{I\ ZIZZ\ I\ I\ I\ I\ IX\ I\ I\ I\ I}, \qquad \overline{\mathrm{Z}_6} = \mathbf{ZZZ\ I\ I\ I\ I\ I\ I\ I\ Z\ I\ I\ I\ I}$$

$$\overline{\mathrm{X}_7} = \mathbf{Z\ I\ I\ I\ I\ I\ I\ I\ I\ I\ IX\ I\ I\ I}, \qquad \overline{\mathrm{Z}_7} = \mathbf{I\ ZZ\ Z\ I\ I\ I\ I\ I\ I\ I\ Z\ I\ I\ I}$$

$$\overline{\mathrm{X}_8} = \mathbf{I\ ZI\ I\ I\ I\ I\ I\ I\ I\ I\ XI\ I}, \qquad \overline{\mathrm{Z}_8} = \mathbf{I\ \ I\ Z\ ZZ\ I\ I\ I\ I\ I\ I\ I\ Z\ I\ I}$$

$$\overline{\mathrm{X}_9} = \mathbf{I\ IZ\ I\ I\ I\ I\ I\ I\ I\ I\ I\ IX\ I}, \qquad \overline{\mathrm{Z}_9} = \mathbf{Z\ I\ Z\ Z\ I\ I\ I\ I\ I\ I\ I\ I\ I\ Z\ I}$$

$$\overline{\mathrm{X}_{10}} = \mathbf{I\ I\ I\ Z\ I\ I\ I\ I\ I\ I\ I\ I\ I\ I\ X}, \qquad \overline{\mathrm{Z}_{10}} = \mathbf{IZ\ I\ Z\ Z\ I\ I\ I\ I\ I\ I\ I\ I\ I\ Z}$$

Therefore, the codewords for the [[15,10,2]] stabilizer code can be expressed as

$$\begin{aligned} |c_1c_2\ldots c_{10}\rangle_L &= \tfrac{1}{\sqrt{2^5}} \times \left(\prod_{i=1}^{5}(I+g_i)\right) \times \overline{\mathbf{X}_1}^{c_1} \times \overline{\mathbf{X}_2}^{c_2} \times \ldots \times \overline{\mathbf{X}_{10}}^{c_{10}} \,|0_1 0_2 \ldots 0_{15}\rangle \\ &= \tfrac{1}{\sqrt{2^5}} \times \overline{\mathbf{X}_1}^{c_1} \times \overline{\mathbf{X}_2}^{c_2} \times \ldots \times \overline{\mathbf{X}_{10}}^{c_{10}} \left(\sum_{s\in S} s\,|0_1 0_2 \ldots 0_{15}\rangle\right). \end{aligned}$$

As shown in Table 2.2, the parameter constraints for difference sets in proposed construction are different from the ones in [35]. Since $2p-1 \equiv p-1$ modulo 2 where p is an even number, DSs which are used in [35] can be also used in the proposed construction. In contrast, DSs in the proposed construction are not always used in [35] because $4p-1$ must be a prime number. As a result, the proposed construction is more general than [35]s construction and the proposed construction enlarges the results of using DSs for quantum stabilizer code construction. In addition, in comparison to the proposed codes with existing quantum codes, quantum codes with length 7 and 15 are discussed. It is known that existing quantum stabilizer

Table 2.2: Comparison of our proposed method and [35]s method.

Paper [35]'s Construction	**Proposed Construction**
Focus on the difference set with parameters: $(n, k, \lambda) = (4p - 1, 2p - 1, p - 1)$ where p is even number and $4p - 1$ is a prime number.	Focus on the difference set with parameters: (n, k, λ) where $k \equiv \lambda(\text{modulo } 2)$

codes with length 7 have code parameters [[7,3,2]] from quadratic residue sets in [38], or [[7,3,2]] and [[7,4,2]] constructed over the quaternary alphabet, listed in [39]. To compare to the proposed codes and codes in [38], the number of information bits of the proposed codes is 1 bit larger than the referenced code. As referenced in the list in [39], a stabilizer with length 15 and the same parameters of [[15,10,2]] that were constructed over quaternary alphabet are found.

Chapter 3

New construction of binary and nonbinary quantum stabilizer codes based on symmetric matrices

3.1 Introduction

The high-dimensional degrees of freedom of photons can encode more quantum information than their two-dimensional counterparts, and this increased information capacity has advantages in quantum applications, such as quantum communication, quantum cryptography, and quantum algorithm. However, controlling and manipulating these systems has many challenging; QECCs are one solution that aims to solve these issues [40]. Additionally, QECCs for high-dimensional quantum systems must be considered in non-binary cases, by using non-binary quantum stabilizer codes. The CSS structure has been considered for non-binary quantum codes for qudits, where the classical codes are over the Galois field [41]. Since the self-orthogonal codes over a finite field that satisfy the conditions of the CSS structure-based quantum code and self-orthogonal codes can be constructed effectively by combinatoric design, cyclic codes, and constacyclic codes, many quantum codes have been constructed in recent years; these are based on the CSS structure via a finite field [42] [43] [44]. In addition, the stabilizer formalism allows quantum codes to be presented by binary matrices, i.e., parity-check matrices with symplectic inner product (SIP) constraints [45] [46] [47]. Hence, we can consider the construction of quantum codes based on the CSS structure to be in the form of matrices. For example, in [13] the authors used a permutation-based technique for this construction, and in [48] the authors searched for a suitable monomial matrix for the construction. The papers [13] [48] provided some good quantum non-binary codes with the singleton bound; however, many constructions remain to be discovered.

In this chapter, we propose the new construction of quantum codes from symmetric matrices that are based on the CSS structure. The parity-check matrices are first generated from two constructions and proven to satisfy the symplectic inner product for the construction of binary and non-binary quantum stabilizer codes. Then, the parameters of these codes are calculated and explained in detail. Some quantum codes are proven to achieve equality of the quantum singleton bound.

3.2 Proposed construction

3.2.1 Quantum stabilizer codes for the binary case

In this subsection, we propose the construction of quantum stabilizer codes for the binary case. Construction 1a considers the construction of a parity-check matrix based on identity and symmetric matrices. In another case, the parity-check matrix in construction 2a is based on the CSS structure.

Construction 1a: Let $\mathbf{I}$ be the identity binary matrix with size $n \times n$. Let $\mathbf{A}$ be the symmetric matrix ($\mathbf{A}^T = \mathbf{A}$) over binary with size $n \times n$. The proposed parity-check matrix has the following form:

$$\mathbf{H} = \begin{bmatrix} \mathbf{I} & \mathbf{A} \end{bmatrix} \tag{3.1}$$

This corresponds to the quantum stabilizer code with parameter $[[n, 0]]$.

Proof: Based on the properties of symmetric matrices, we have:

$$\mathbf{I}\mathbf{A}^T + \mathbf{A}\mathbf{I}^T = \mathbf{A}^T + \mathbf{A} = \mathbf{0}.$$

The SIP equation 1.2 for the parity-check matrix in equation 3.1 is satisfied. Since $\mathbf{H}$ has the size $n \times 2n$, we get the quantum stabilizer code with parameter $[[n, 0]]$.

Construction 2a: Let $\mathbf{I}$ be the identity binary matrix with size $n \times n$. Let $\mathbf{A}$ be the symmetric matrix ($\mathbf{A}^T = \mathbf{A}$) over binary with size $n \times n$. The proposed parity-check matrix has the following form:

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_2 \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{A} \end{bmatrix} & \mathbf{0} \\ \mathbf{0} & \begin{bmatrix} \mathbf{A} & \mathbf{I} \end{bmatrix} \end{bmatrix} \tag{3.2}$$

This corresponds to the quantum stabilizer code with parameter $[[2n, 0, d]]$.

Proof: Based on the definition in equation 3.2, we have following formulation:

$$\mathbf{H}_1 \times \mathbf{G}_2{}^T = \begin{bmatrix} \mathbf{I} & \mathbf{A} \end{bmatrix} \times \begin{bmatrix} \mathbf{A} & \mathbf{I} \end{bmatrix}^T = \begin{bmatrix} \mathbf{I} & \mathbf{A} \end{bmatrix} \times \begin{bmatrix} \mathbf{A}^T \\ \mathbf{I}^T \end{bmatrix}$$

$$= \mathbf{I} \times \mathbf{A}^T + \mathbf{A} \times \mathbf{I}^T = \mathbf{A}^T + \mathbf{A} = \mathbf{0}.$$

In addition, we also have:

$$\mathbf{G}_2 \times \mathbf{H}_1{}^T = \begin{bmatrix} \mathbf{A} & \mathbf{I} \end{bmatrix} \times \begin{bmatrix} \mathbf{I} & \mathbf{A} \end{bmatrix}^T = \begin{bmatrix} \mathbf{A} & \mathbf{I} \end{bmatrix} \times \begin{bmatrix} \mathbf{I}^T \\ \mathbf{A}^T \end{bmatrix}$$

$$= \mathbf{A} \times \mathbf{I}^T + \mathbf{I} \times \mathbf{A}^T = \mathbf{A} + \mathbf{A}^T = \mathbf{0}.$$

Then, the SIP for the parity-check matrix in (4) is:

$$\mathbf{H_X} \times \mathbf{H_Z}{}^T + \mathbf{H_Z} \times \mathbf{H_X}{}^T = \begin{bmatrix} \mathbf{H}_1 \\ \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{0} \\ \mathbf{G}_2 \end{bmatrix}^T + \begin{bmatrix} \mathbf{0} \\ \mathbf{G}_2 \end{bmatrix} \begin{bmatrix} \mathbf{H}_1 \\ \mathbf{0} \end{bmatrix}^T$$

$$= \begin{bmatrix} \mathbf{H}_1 \\ \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{0}^T & \mathbf{G}_2{}^T \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{G}_2 \end{bmatrix} \begin{bmatrix} \mathbf{H}_1{}^T & \mathbf{0} \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{H}_1 \times \mathbf{0}^T & \mathbf{H}_1 \times \mathbf{G}_2{}^T \\ \mathbf{0} \times \mathbf{0}^T & \mathbf{0} \times \mathbf{G}_2{}^T \end{bmatrix} + \begin{bmatrix} \mathbf{0} \times \mathbf{H}_1{}^T & \mathbf{0} \times \mathbf{0}^T \\ \mathbf{G}_2 \times \mathbf{H}_1{}^T & \mathbf{G}_2 \times \mathbf{0}^T \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{0} & \mathbf{H}_1 \times \mathbf{G}_2{}^T \\ \mathbf{G}_2 \times \mathbf{H}_1{}^T & \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}.$$

Since $\mathbf{H}$ has the size $2n \times 4n$, we get the quantum stabilizer code with parameter $[[2n, 0]]$.

In the following examples, the quantum stabilizer codes for the qubits case are given. In example 1 through example 3, we consider the quantum stabilizer codes for the qubits case based on construction 1a where the code lengths are 5, 6, and 7. In the examples, we first give the construction of the parity-check matrices, and then the parameter of the code is calculated by the Magma tools function *QuantumCode*, *MinimumWeight* [52]. Since any symmetric matrix satisfies our construction, there are many candidates for the parity-check matrices. Therefore, for each code length, we give the quantum stabilizer code with a largest minimum distance, which we calculated by the Magma tools function.

Example 1: We consider the following parity-check matrix with the size 5 × 10:

$$\mathbf{H} = \begin{bmatrix} 1,0,0,0,0 & 1,1,0,1,1 \\ 0,1,0,0,0 & 1,0,1,1,0 \\ 0,0,1,0,0 & 0,1,1,0,1 \\ 0,0,0,1,0 & 1,1,0,1,0 \\ 0,0,0,0,1 & 1,0,1,0,1 \end{bmatrix}.$$

It corresponds to the quantum stabilizer code with parameter $[[5,0,3]]$.

Example 2: We consider the following parity-check matrix with the size 6 × 12:

$$\mathbf{H} = \begin{bmatrix} 1,0,0,0,0,0 & 1,0,1,1,0,1 \\ 0,1,0,0,0,0 & 0,0,1,0,1,1 \\ 0,0,1,0,0,0 & 1,1,0,0,0,1 \\ 0,0,0,1,0,0 & 1,0,0,1,1,1 \\ 0,0,0,0,1,0 & 0,1,0,1,0,1 \\ 0,0,0,0,0,1 & 1,1,1,1,1,0 \end{bmatrix}.$$

This corresponds to the quantum stabilizer code with parameter $[[6,0,4]]$.

Example 3: We consider the following parity-check matrix with the size 7 × 14:

$$\mathbf{H} = \begin{bmatrix} 1,0,0,0,0,0,0 & 1,0,1,0,1,0,1 \\ 0,1,0,0,0,0,0 & 0,1,0,0,1,0,1 \\ 0,0,1,0,0,0,0 & 1,0,1,1,0,1,1 \\ 0,0,0,1,0,0,0 & 0,0,1,0,1,1,0 \\ 0,0,0,0,1,0,0 & 1,1,0,1,1,0,1 \\ 0,0,0,0,0,1,0 & 0,0,1,1,0,1,0 \\ 0,0,0,0,0,0,1 & 1,1,1,0,1,0,1 \end{bmatrix}.$$

This corresponds to the quantum stabilizer code with parameter $[[7,0,3]]$.

In the following examples (from example 4 to example 8), we consider the quantum stabilizer codes for the qubits case based on the construction 2a with the code lengths up to 12. As construction 2a requires the symmetric matrices to construct the parity-check matrix, there are many candidates for the symmetric matrices. By using the CSS structure, we can determine the parameter of the corresponding quantum stabilizer code via the parameters of two classical codes. Then, we choose the symmetric matrices with large maximum distances to get quantum stabilizer codes with large maximum distances.

Example 4: We consider two classical codes with the parity-check matrices and the generators matrices as follows,

$$\mathbf{H}_1 = \begin{bmatrix} 1,0,1,0 \\ 0,1,0,1 \end{bmatrix}$$

(classical [4,2,2] code) and

$$\mathbf{G}_2 = \begin{bmatrix} 1,0,1,0 \\ 0,1,0,1 \end{bmatrix}$$

(classical [4,2,2] code).

This corresponds to the quantum stabilizer code with parameter [[4,0,2]].

Example 5: We consider two classical codes with the parity-check matrices and the generators matrices as follows:

$$\mathbf{H}_1 = \begin{bmatrix} 1,0,0,1,1,1 \\ 0,1,0,1,1,0 \\ 0,0,1,1,0,1 \end{bmatrix}$$

(classical [6,3,3] code) and

$$\mathbf{G}_2 = \begin{bmatrix} 1,1,1,1,0,0 \\ 1,1,0,0,1,0 \\ 1,0,1,0,0,1 \end{bmatrix}$$

(classical [6,3,3] code).

This corresponds to the quantum stabilizer code with parameter [[6,0,3]].

Example 6: We consider two classical codes with the parity-check matrices and the generators matrices as follows:

$$\mathbf{H}_1 = \begin{bmatrix} 1,0,0,0,0,0,1,1,1,1 \\ 0,1,0,0,0,1,1,0,1,1 \\ 0,0,1,0,0,1,0,1,1,1 \\ 0,0,0,1,0,1,1,1,1,0 \\ 0,0,0,0,1,1,1,1,0,1 \end{bmatrix}$$

([8,4,4] code) and

$$\mathbf{G}_2 = \begin{bmatrix} 0,1,1,1,1,1,0,0,0,0 \\ 1,1,0,1,1,0,1,0,0,0 \\ 1,0,1,1,1,0,0,1,0,0 \\ 1,1,1,1,0,0,0,0,1,0 \\ 1,1,1,0,1,0,0,0,0,1 \end{bmatrix}$$

([8,4,4] code).

This corresponds to the quantum stabilizer code with parameter [[8,0,4]].

Example 7: We consider two classical codes with the parity-check matrices and the generators matrices as follows:

$$\mathbf{H}_1 = \begin{bmatrix} 1,0,0,0,0,0,1,1,1,1 \\ 0,1,0,0,0,1,1,0,1,1 \\ 0,0,1,0,0,1,0,1,1,1 \\ 0,0,0,1,0,1,1,1,1,0 \\ 0,0,0,0,1,1,1,1,0,1 \end{bmatrix}$$

([10,5,4] code) and

$$\mathbf{G}_2 = \begin{bmatrix} 0,1,1,1,1,1,0,0,0,0 \\ 1,1,0,1,1,0,1,0,0,0 \\ 1,0,1,1,1,0,0,1,0,0 \\ 1,1,1,1,0,0,0,0,1,0 \\ 1,1,1,0,1,0,0,0,0,1 \end{bmatrix}$$

([10,5,4] code).

This corresponds to the quantum stabilizer code with parameter [[10,0,4]].

Example 8: We consider two classical codes with the parity-check matrices and the generators matrices as follows:

$$\mathbf{H}_1 = \begin{bmatrix} 1,0,0,0,0,0,1,0,1,0,1,0 \\ 0,1,0,0,0,0,0,0,1,1,1,1 \\ 0,0,1,0,0,0,1,1,1,0,1,1 \\ 0,0,0,1,0,0,0,1,0,1,1,1 \\ 0,0,0,0,1,0,1,1,1,1,1,0 \\ 0,0,0,0,0,1,0,1,1,1,0,1 \end{bmatrix}$$

([12,6,4] code) and

$$\mathbf{G}_2 = \begin{bmatrix} 1,0,1,0,1,0,1,0,0,0,0,0 \\ 0,0,1,1,1,1,0,1,0,0,0,0 \\ 1,1,1,0,1,1,0,0,1,0,0,0 \\ 0,1,0,1,1,1,0,0,0,1,0,0 \\ 1,1,1,1,1,0,0,0,0,0,1,0 \\ 0,1,1,1,0,1,0,0,0,0,0,1 \end{bmatrix}$$

([12,6,4] code).

This corresponds to the quantum stabilizer code with parameter [[12,0,4]].

3.2.2 Quantum stabilizer codes for the nonbinary case

In this subsection, we propose the construction of quantum stabilizer codes for the non-binary case. Construction 1b considers the construction of the parity-check matrix over GF_p, which is based on identity and symmetric matrices over GF_p. In construction 2b, the parity-check matrices over GF_p are based on the CSS structure.

Construction 1b: Let $\mathbf{I}$ be a matrix with size $n \times n$ over GF_p, where all the elements are zeros except for those on the main diagonal, which are one. Additionally, the matrix $\mathbf{A}$ is a symmetric matrix with size $n \times n$ over GF_p, $\mathbf{A}^T = \mathbf{A}$. The proposed parity-check matrix has the following form:

$$\mathbf{H} = \begin{bmatrix} \mathbf{I} & \mathbf{A} \end{bmatrix}, \tag{3.3}$$

which corresponds to the quantum stabilizer code with parameter $[[n, 0]]_p$.

Proof: Based on the properties of $\mathbf{A}$ and $\mathbf{I}$, we have:

$$\mathbf{I} \times \mathbf{A}^T = \mathbf{A}^T = \mathbf{A} = \mathbf{A} \times \mathbf{I}^T.$$

The SIP equation 1.7 for the parity-check matrix in equation 3.3 is satisfied. Since $\mathbf{H}$ has the size $n \times 2n$, we get the quantum stabilizer code with parameter $[[n, 0]]_p$.

Construction 2b: Let $\mathbf{I}$ be the matrix with size $n \times n$ over GF_p, where all the elements are zeros except for those on the main diagonal, which are one. Additionally, the matrix $\mathbf{A}$ is the symmetric matrix ($\mathbf{A}^T = \mathbf{A}$) with size $n \times n$ over GF_p, and $-\mathbf{A}$ denotes the matrix where its elements are the minus modulo p for corresponding elements of $\mathbf{A}$. The proposed parity-check matrix has the following form,

$$\mathbf{H} = \begin{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{A} \end{bmatrix} & \mathbf{0} \\ \mathbf{0} & \begin{bmatrix} -\mathbf{A} & \mathbf{I} \end{bmatrix} \end{bmatrix}, \tag{3.4}$$

which corresponds to the quantum stabilizer code with parameter $[[2n, 0]]_p$.

Proof: Based on the definition in equation 3.4, we have the following formula:

$$\mathbf{H}_1 \times \mathbf{G}_2{}^T = \begin{bmatrix} \mathbf{I} & \mathbf{A} \end{bmatrix} \times \begin{bmatrix} -\mathbf{A} & \mathbf{I} \end{bmatrix}^T = \begin{bmatrix} \mathbf{I} & \mathbf{A} \end{bmatrix} \times \begin{bmatrix} -\mathbf{A}^T \\ \mathbf{I}^T \end{bmatrix}$$

$$= \mathbf{I} \times (-\mathbf{A}^T) + \mathbf{A} \times \mathbf{I}^T = -\mathbf{A} + \mathbf{A} = \mathbf{0}.$$

Then, we have:

$$\mathbf{H_X} \times \mathbf{H_Z}^T = \begin{bmatrix} \mathbf{H}_1 \\ \mathbf{0} \end{bmatrix} \times \begin{bmatrix} \mathbf{0} \\ \mathbf{G}_2 \end{bmatrix}^T = \begin{bmatrix} \mathbf{H}_1 \\ \mathbf{0} \end{bmatrix} \times \begin{bmatrix} \mathbf{0}^T & \mathbf{G}_2{}^T \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{H}_1 \times \mathbf{0}^T & \mathbf{H}_1 \times \mathbf{G}_2{}^T \\ \mathbf{0} \times \mathbf{0}^T & \mathbf{0} \times \mathbf{G}_2{}^T \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{H}_1 \times \mathbf{G}_2{}^T \\ \mathbf{0} & \mathbf{0} \end{bmatrix}.$$

In addition, we also have the following formula:

$$\mathbf{H_Z} \times \mathbf{H_X}^T = \begin{bmatrix} \mathbf{0} \\ \mathbf{G}_2 \end{bmatrix} \times \begin{bmatrix} \mathbf{H}_1 \\ \mathbf{0} \end{bmatrix}^T = \begin{bmatrix} \mathbf{0} \\ \mathbf{G}_2 \end{bmatrix} \times \begin{bmatrix} \mathbf{H}_1{}^T & \mathbf{0} \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{0} \times \mathbf{H}_1{}^T & \mathbf{0} \times \mathbf{0}^T \\ \mathbf{G}_2 \times \mathbf{H}_1{}^T & \mathbf{G}_2 \times \mathbf{0}^T \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{G}_2 \times \mathbf{H}_1{}^T & \mathbf{0} \end{bmatrix}.$$

Thus, $\mathbf{H_X} \times \mathbf{H_Z}^T = \mathbf{H_Z} \times \mathbf{H_X}^T$ $(= [\mathbf{0}])$ and the parity-check matrix in equation 3.4 is satisfied by the SIP equation 1.7. Since $\mathbf{H}$ has the size $2n \times 4n$, we get the quantum stabilizer code with parameter $[[2n, 0]]_p$.

In the following examples, we consider the quantum stabilizer codes over *GF(3)* and *GF(5)* that are based on construction 1b, where the code lengths are 4, 5, and 6. Using the same process as section 3.1, we choose the candidates and then calculate the parameters of quantum stabilizer codes by using the Magma tools function *QuantumCode*, *MinimumWeight*. Quantum stabilizer codes with large minimum distances are given in the examples.

Example 9: We consider the following parity-check matrix with the size 4 × 8:

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

This corresponds to the quantum stabilizer code with parameter $[[4,0,2]]_3$.

Example 10: We consider the following parity-check matrix with the size 5 × 10:

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

This corresponds to quantum stabilizer code with parameter $[[5,0,3]]_3$.

Example 11: We consider the following parity-check matrix with the size 6 × 12:

$$\mathbf{H} = \begin{bmatrix} 1,0,0,0,0,0 & 4,0,2,1,0,1 \\ 0,1,0,0,0,0 & 0,0,1,0,1,1 \\ 0,0,1,0,0,0 & 2,1,0,0,0,1 \\ 0,0,0,1,0,0 & 1,0,0,3,1,1 \\ 0,0,0,0,1,0 & 0,1,0,1,0,1 \\ 0,0,0,0,0,1 & 1,1,1,1,1,0 \end{bmatrix}.$$

This corresponds to quantum stabilizer code with parameter $[[6,0,4]]_5$.

In the following examples (from example 12 to example 15), we consider the quantum stabilizer codes over *GF(3)* and *GF(7)* based on construction 2b with code length up to 12. By using the CSS structure, we can determine the parameter of the corresponding quantum stabilizer code via the parameters of two classical codes. Then, we choose the symmetric matrices with large maximum distances to get quantum stabilizer codes with large maximum distances.

Example 12: We consider two classical codes with parity-check matrices and generators matrices as follows: $[4, 2, 2]_3$ code, where

$$\mathbf{H}_1 = \begin{bmatrix} 1, 0, 2, 0 \\ 0, 1, 0, 2 \end{bmatrix}$$

and $[4, 2, 2]_3$ code, where

$$\mathbf{G}_2 = \begin{bmatrix} 1, 0, 1, 0 \\ 0, 1, 0, 1 \end{bmatrix}.$$

This corresponds to the quantum stabilizer code with parameter $[[4, 0, 2]]_3$.

Example 13: We consider two classical codes with parity-check matrices and generators matrices as follows: $[6, 3, 3]_7$ code where

$$\mathbf{H}_1 = \begin{bmatrix} 1, 0, 0, 0, 1, 1 \\ 0, 1, 0, 1, 2, 0 \\ 0, 0, 1, 1, 0, 3 \end{bmatrix}$$

and $[6, 3, 3]_7$ code where

$$\mathbf{G}_2 = \begin{bmatrix} 0, 6, 6, 1, 0, 0 \\ 6, 5, 0, 0, 1, 0 \\ 6, 0, 4, 0, 0, 1 \end{bmatrix}.$$

This corresponds to the quantum stabilizer code with parameter $[[6, 0, 3]]_7$.

Example 14: We consider two classical codes with parity-check matrices and generators matrices as follows: $[8, 4, 4]_3$ code, where

$$\mathbf{H}_1 = \begin{bmatrix} 1, 0, 0, 0, 2, 1, 1, 1 \\ 0, 1, 0, 0, 1, 0, 1, 1 \\ 0, 0, 1, 0, 1, 1, 2, 0 \\ 0, 0, 0, 1, 1, 1, 0, 2 \end{bmatrix}$$

and $[8, 4, 4]_3$ code, where

$$\mathbf{G}_2 = \begin{bmatrix} 1, 2, 2, 2, 1, 0, 0, 0 \\ 2, 0, 2, 2, 0, 1, 0, 0 \\ 2, 2, 1, 0, 0, 0, 1, 0 \\ 2, 2, 0, 1, 0, 0, 0, 1 \end{bmatrix}.$$

This corresponds to the quantum stabilizer code with parameter $[[8, 0, 4]]_3$.

Example 15: We consider two classical codes with parity-check matrices and generators matrices as follows: $[10, 5, 4]_3$ code, where

$$\mathbf{H}_1 = \begin{bmatrix} 1, 0, 0, 0, 0, 2, 1, 0, 0, 2 \\ 0, 1, 0, 0, 0, 1, 2, 0, 2, 2 \\ 0, 0, 1, 0, 0, 0, 0, 1, 1, 1 \\ 0, 0, 0, 1, 0, 0, 2, 1, 1, 0 \\ 0, 0, 0, 0, 1, 2, 2, 1, 0, 1 \end{bmatrix}$$

and $[10, 5, 4]_3$ code, where

$$\mathbf{G}_2 = \begin{bmatrix} 1,2,0,0,1,1,1,0,0,0,0 \\ 2,1,0,1,1,0,1,0,0,0,0 \\ 0,0,2,2,2,0,0,1,0,0 \\ 0,1,2,2,0,0,0,0,1,0 \\ 1,1,2,0,2,0,0,0,0,1 \end{bmatrix}.$$

This corresponds to the quantum stabilizer code with parameter $[[10, 0, 4]]_3$.

In Table 3.1, we summarize some quantum binary and non-binary codes with lengths ranging from four to 12 over two proposed constructions. The optimal quantum stabilizer codes are defined as the codes where the parameters equalize the equation of the Knill-Laflamme bound; detail discussion in section 2. The proposed construction aims to provide quantum stabilizer codes with the full rank quantum stabilizer group ($[[n, 0, d]]_p$). As was previously discussed, the full rank quantum stabilizer codes can provide a perfect graph state, which has many applications in one-way quantum computers, secure state distribution, secret sharing, and quantum algorithm [54] [55].

Table 3.1: Binary and non-binary quantum stabilizer codes from proposed construction.

Construction	**Code Length**	**Code Parameters**	**Note**
2a	4	[[4,0,2]]	Optimal quantum stabilizer code
1b	4	$[[4,0,2]]_3$	Optimal quantum stabilizer code
2b	4	$[[4,0,2]]_3$	Optimal quantum stabilizer code
1a	5	[[5,0,3]]	Optimal quantum stabilizer code
1b	5	$[[5,0,3]]_3$	Optimal quantum stabilizer code
1a	6	[[6,0,4]]	Optimal quantum stabilizer code
2a	6	[[6,0,3]]	
1b	6	$[[6,0,4]]_5$	Optimal quantum stabilizer code
2b	6	$[[6,0,3]]_7$	
1a	7	[[7,0,3]]	Optimal quantum stabilizer code
2a	8	[[8,0,4]]	Optimal quantum stabilizer code
2b	8	$[[8,0,4]]_3$	Optimal quantum stabilizer code
2a	10	[[10,0,4]]	Optimal quantum stabilizer code
2b	10	$[[10,0,4]]_3$	Optimal quantum stabilizer code
2a	12	[[12,0,4]]	

Chapter 4

Quantum Stabilizer Codes Construction from Hermitian Self-orthogonal Codes over *GF(4)*

4.1 Introduction

Stabilizer codes, first introduced by Gottesman [56], have become an important class of QECC. These codes are useful for building quantum fault-tolerant circuits [57]. Stabilizer codes append ancilla qubits to qubits to be protected, and the most important advantage of stabilizer codes is that errors can be detected and removed by stabilizer operators, rather than from the quantum state itself. In addition, the stabilizer formalism allows us to construct quantum stabilizer code from binary formalism as the classical parity-check matrix over binary in the constraint referred to as the symplectic inner product (SIP) [56]. Therefore, several stabilizer codes have been proposed where constructions are analogous to classical linear codes, such as quantum BCH codes [58], entanglement-assisted quantum code based on LDPC [59], quantum Reed-Solomon codes [60], quantum code based on classical cyclic and modified cyclic [61] and analogous to combinatorial design, such as cyclic difference sets [62], quadratic residue sets [63], and group association scheme [64]. It also turns out that another useful construction can be found by considering classical error correction codes, but instead of using binary vectors, we use vectors over the Galois field (GF) [65]. Since additive codes over GF can be defined as additive subgroups, the additive codes have been popularly used in construction of quantum codes. Hence, the problem of finding QECC is transformed into a problem of finding additive self-orthogonal code under a certain inner product over *GF(4)*. So our proposal is to construct good Hermitian self-orthogonal code in order to construct good QECCs using the idea of Calderbank et al. [65].

The key result of this chapter is to propose a new approach to the construction of additive codes over *GF(4)*, which are self-orthogonal with respect to Hermitian product. The minimum distance of this classical linear code was proved to be 4 in all cases. The corresponding quantum stabilizer code can be transformed from this classical code; we prove all the optimal codes that can be accomplished from this construction with lengths 5, 6, 7, 8, 9, and 10.

4.2 Proposed construction

In this section, the construction is first proposed and proven to satisfy the conditions of Hermitian self-orthogonal codes. We prove the codes have a good minimum distance. Then, six optimal quantum stabilizer codes are showed as transformation from the Hermitian self-orthogonal codes. In addition, the explanations for general length are mentioned.

4.2.1 Extension for generator matrix

Theorem 4.1 *Let $\boldsymbol{G}$ be the generator matrix over GF(4) in following form: $\boldsymbol{G} = [\boldsymbol{I}|\boldsymbol{G}^0]$ where*

$$\boldsymbol{I} = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{bmatrix}, \boldsymbol{G}^0 = \begin{bmatrix} \boldsymbol{g}_1 \\ \boldsymbol{g}_2 \\ \vdots \\ \boldsymbol{g}_m \end{bmatrix}, \tag{4.1}$$

and all the elements of matrix $\boldsymbol{I}$ are in GF(4), and $\mathbf{g}_i(i = 1, 2, \cdots, m)$ are the vectors with finite length over GF(4) that satisfy two conditions:

1. *the number of nonzero elements in $\boldsymbol{g}_i$ is odd, and*

2. *the Hermitian product of any pair $(\boldsymbol{g}_i, \boldsymbol{g}_j)$ (where $i, j = 1, 2, \cdots, m$) is zero.*

Then, $\boldsymbol{G}$ is the generator matrix of a Hermitian self-orthogonal code over GF(4).

Proof: From Condition 1, each vector $\mathbf{g}_i(i = 1, 2, \cdots, m)$ has odd weight, and then, each row in $\mathbf{G}$ has even weight. Hence, the Hermitian product of each row in $\mathbf{G}$ with itself is zero (the summation of even numbers of 1 is 0). In addition, the Hermitian product of each of the two rows in $\mathbf{G}$ is 0 due to Condition 2; the Hermitian product of each pair $(\mathbf{g}_i, \mathbf{g}_j)$(where $i, j = 1, 2, \cdots, m$) is zero.

We call $\mathbf{G}^0$ the matrix created from $\mathbf{g}_i(i = 1, 2, \cdots, m)$, or the right-part of generator matrix $\mathbf{G}$. Then, we have the extension from $\mathbf{G}^0$ as the following theorem to get larger matrices, $\mathbf{G}^1$, $\mathbf{G}^2$, that protect the two conditions in Theorem 4.1. $\mathbf{G}^1$, $\mathbf{G}^2$ can also be the right part of $\mathbf{G}$.

Theorem 4.2 *From generator matrix $\boldsymbol{G}^0$ with length l and dimension m, Let's extend new matrices $\boldsymbol{G}^1$ and $\boldsymbol{G}^2$ with length $l+2$, and the dimension to $m+1$ or $m+2$, as in the following form:*

$$\boldsymbol{G}^1 = \begin{bmatrix} \boldsymbol{A}^1 & \boldsymbol{X}^1 \\ \boldsymbol{Y} & \boldsymbol{G}^0 \end{bmatrix}, \boldsymbol{G}^2 = \begin{bmatrix} \boldsymbol{A}^2 & \boldsymbol{X}^2 \\ \boldsymbol{Y} & \boldsymbol{G}^0 \end{bmatrix}, \tag{4.2}$$

where $\boldsymbol{A}^1 = [0 \;\; 1]$, $\boldsymbol{A}^2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, $\boldsymbol{X}^1 = [x_1 \;\; x_2 \;\; ... \;\; x_n]$ is an even weight vector with the elements over GF(4), $\boldsymbol{X}^2 = \begin{bmatrix} \boldsymbol{X}^1 \\ \boldsymbol{X}^1 \end{bmatrix}$, $\boldsymbol{Y} = \begin{bmatrix} y_1 & y_1 \\ y_2 & y_2 \\ \cdots & \\ y_m & y_m \end{bmatrix}$ with elements $\overline{y_i} := \boldsymbol{X}^1.\boldsymbol{g}_i$ (where $\overline{y_i}$ denotes the conjugate of y_i and $(.)$ denotes the Hermitian product; $\boldsymbol{g}_i$ is ith row of $\boldsymbol{G}^0$. Then, $\boldsymbol{G}^1$ and $\boldsymbol{G}^2$ satisfy the two conditions in Theorem 4.1 and can be used as the right part in generator matrix $\boldsymbol{G}$ of a Hermitian self-orthogonal code over GF(4).

Proof: The weight of $[x_1 \;\; x_2 \;\; ... \;\; x_n]$ is even, and the weight of each $\mathbf{g}_i(i = 1, 2, , m)$ is odd, so it is clear that the weight of each row in $\mathbf{G}^1$ and $\mathbf{G}^2$ is odd. So, the first condition in Theorem 4.1 is satisfied. In addition, the extension elements help us to save the Hermitian product since it comes from the definition of y_i:

$$\overline{y_i} := [x_1 \ \ x_2 \ \ ... \ \ x_n] \, .\mathbf{g}_i$$

$$\Rightarrow 1\overline{y_i} + [x_1 \ \ x_2 \ \ ... \ \ x_n] \, .\mathbf{g}_i = 1\overline{y_i} + 1\overline{y_i} = 0.$$

So, the second condition in Theorem 4.1 is satisfied.

Since $\mathbf{G}^1$ and $\mathbf{G}^2$ satisfy the two conditions in Theorem 4.1 and can be used as the right part in generator matrix $\mathbf{G}$ of a Hermitian self-orthogonal code over *GF(4)*, Theorem 4.2 is proven.

4.2.2 Optimal quantum stabilizer code results

In this part, the results from our proposal have showed. We divide the results into two cases. We first consider the odd lengths with 5, 7, and 9 qubits. Then, we consider the qubits with even lengths 6, 8, and 10. The relations between the outcomes are explained in each example.

Example 1: In the case of five qubits, the optimal code we expected is QECC $[[5, 1, 3]]$ code. With the form in Eq. 4.1, it reduces to find that $\mathbf{G}^0$ with two vectors has size 3. Because elements of $\mathbf{G}^0$ are from GF(4) $= \{0, 1, \omega, \omega^2\}$, it is trivial to check out the conditions, and we get three candidates for $\mathbf{G}^0$ as follows:

$$\mathbf{G}^0{}_1 = \begin{bmatrix} 1 & \omega & \omega^2 \\ \omega^2 & \omega & 1 \end{bmatrix} \tag{4.3}$$

$$\mathbf{G}^0{}_2 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & \omega & \omega^2 \end{bmatrix} \tag{4.4}$$

$$\mathbf{G}^0{}_3 = \begin{bmatrix} 1 & \omega & \omega \\ \\ \omega & \omega & 1 \end{bmatrix}. \tag{4.5}$$

As mapping between elements in Table 1.1, we can get the standard form (the theorem about binary form with standard form can be found at Chapter 1 of generators for each stabilizer code as follows

Case 1: From $\mathbf{G}^0{}_1$ in Eq. 4.3, we have Hermitian self-orthogonal code $[5, 2, 4]$ with the generators $\mathbf{G} = \begin{bmatrix} 1 & 0 & 1 & \omega & \omega^2 \\ \\ 0 & 1 & \omega^2 & \omega & 1 \end{bmatrix}$, and it corresponds to stabilizer code in binary standard form:

$$\begin{cases} \mathbf{g}_1 = [1000011110] \\ \\ \mathbf{g}_2 = [0100011011] \\ \\ \mathbf{g}_3 = [0010111101] \\ \\ \mathbf{g}_4 = [0001110110] \end{cases}.$$

From standard form, the logical operators and minimum distance are calculated exactly and we have QECC $[[5, 1, 3]]$ code.

Case 2: From $\mathbf{G}^0{}_2$ in Eq. 4.4, we have Hermitian self-orthogonal code $[5, 2, 4]$ with the generators $\mathbf{G} = \begin{bmatrix} 1 & 0 & 1 & 1 & 1 \\ \\ 0 & 1 & 1 & \omega & \omega^2 \end{bmatrix}$, and it corresponds to stabilizer code in binary standard form:

$$\begin{cases} \mathbf{g}_1 = [1001100101] \\ \\ \mathbf{g}_2 = [0101011001] \\ \\ \mathbf{g}_3 = [0010110110] \\ \\ \mathbf{g}_4 = [0000001111] \end{cases}.$$

This code was already reported in a database [68].

Case 3: From $\mathbf{G}^0{}_3$ in Eq. 4.5, we have Hermitian self-orthogonal code $[5, 2, 4]$ with the generators $\mathbf{G} = \begin{bmatrix} 1 & 0 & 1 & \omega & \omega \\ 0 & 1 & \omega & \omega & 1 \end{bmatrix}$, and it corresponds to stabilizer code in binary standard form:

$$\begin{cases} \mathbf{g}_1 = [1000111011] \\ \mathbf{g}_2 = [0100100110] \\ \mathbf{g}_3 = [0010111000] \\ \mathbf{g}_4 = [0001110111] \end{cases}.$$

This code was already reported [56] as binary cyclic construction.

Example 2: With quantum stabilizer code length 7, the existing good code is $[[7,1,3]]$ QECC, which was reported as Steane code with CSS construction. Here, the construction is based on Theorem 4.1 from following $\mathbf{G}^0$, and it is easy to verify that $\mathbf{G}^0$ satisfies the two conditions of Theorem 4.1 due to its trivial length:

$$\mathbf{G}^0 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Applying Theorem 4.2, we get extension $\mathbf{G}^1$ from $\mathbf{G}^0$:

$$\mathbf{G}^1 = \begin{bmatrix} 1 & 0 & \omega^2 & \omega^2 \\ \omega & \omega & 1 & 0 \\ \omega & \omega & 0 & 1 \end{bmatrix}.$$

Then, the generator matrix for Hermitian self-orthogonal code is:

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & \omega^2 & \omega^2 \\ 0 & 1 & 0 & \omega & \omega & 1 & 0 \\ 0 & 0 & 1 & \omega & \omega & 0 & 1 \end{bmatrix}. \tag{4.6}$$

The generators in Eq. 4.6 correspond to additive quaternary code $[7, 3, 4]$. We transform them to quantum stabilizer code $[[7, 1, 3]]$ with binary standard form:

$$\begin{cases} \mathbf{g}_1 = [10001001010110] \\ \mathbf{g}_2 = [01000011000100] \\ \mathbf{g}_3 = [00100010001100] \\ \mathbf{g}_4 = [00011000011101] \\ \mathbf{g}_5 = [00000111001000] \\ \mathbf{g}_6 = [00000000110011] \end{cases}.$$

Example 3: With quantum stabilizer code length 9, the existing good code is Shor code [2]. Here, we construct the new $[[9, 1, 3]]$ quantum code starting from the following $\mathbf{G}^0$ that satisfies the two conditions of Theorem 4.1

$$\mathbf{G}^0 = \begin{bmatrix} 1 & \omega & \omega \\ \omega & 1 & \omega \\ \omega & \omega & 1 \end{bmatrix}.$$

Applying Theorem 4.2, we get the extension from $\mathbf{G}^0$:

$$\mathbf{G}^1 = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & \omega & \omega \\ \omega^2 & \omega^2 & \omega & 1 & \omega \\ \omega^2 & \omega^2 & \omega & \omega & 1 \end{bmatrix}.$$

Then, the generator matrix for Hermitian self-orthogonal code is:

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & \omega & \omega \\ 0 & 0 & 1 & 0 & \omega^2 & \omega^2 & \omega & 1 & \omega \\ 0 & 0 & 0 & 1 & \omega^2 & \omega^2 & \omega & \omega & 1 \end{bmatrix}. \tag{4.7}$$

Generators $\mathbf{G}$ in Eq. 4.7 correspond to additive linear code $[9, 4, 4]$, and it transforms to quantum stabilizer code $[[9, 1, 3]]$ in binary standard form:

$$\begin{cases} \mathbf{g}_1 = [100001101000100111] \\ \mathbf{g}_2 = [010000100000000011] \\ \mathbf{g}_3 = [001000100000111010] \\ \mathbf{g}_4 = [000100100001011001] \\ \mathbf{g}_5 = [000011101001000111] \\ \mathbf{g}_6 = [000000011001100000] \\ \mathbf{g}_7 = [000000000100010011] \\ \mathbf{g}_8 = [000000000011100111] \end{cases}.$$

In the following examples, we consider the codes when encoding lengths are $k = 0$. It is a special case of quantum code when stabilizer codes are $[[n, 0, d]]$; it means quantum code has a one-dimensional code subspace, and there is only one encoded state. It is useful for studies of the correlations in decoherence, and code state is maximally entangled [69].

Example 4: We consider $\mathbf{G}^0$ with the size 3×3; it is a trivial case, and we get candidates as follows:

$$\mathbf{G^0}_1 = \begin{bmatrix} 1 & \omega & \omega \\ \omega & 1 & \omega \\ \omega & \omega & 1 \end{bmatrix} \tag{4.8}$$

$$\mathbf{G^0}_2 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & \omega & \omega^2 \\ 1 & \omega^2 & \omega \end{bmatrix} \tag{4.9}$$

The corresponding quantum stabilizer code $[[6,0,4]]$ over *GF(4)* from Hermitian self-orthogonal code has the following form:

Case 1: From $\mathbf{G^0}_1$ in Eq. 4.8, we have $[6,3,4]$ Hermitian self-orthogonal code with generator $\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 1 & \omega & \omega \\ 0 & 1 & 0 & \omega & 1 & \omega \\ 0 & 0 & 1 & \omega & \omega & 1 \end{bmatrix}$, which corresponds to stabilizer code in binary standard form:

$$\begin{cases} \mathbf{g}_1 = [100001010100] \\ \mathbf{g}_2 = [010001011101] \\ \mathbf{g}_3 = [001001000110] \\ \mathbf{g}_4 = [000101010111] \\ \mathbf{g}_5 = [000011011000] \\ \mathbf{g}_6 = [000000111111] \end{cases}.$$

Case 2: From $\mathbf{G^0}_2$ in Eq. 4.9, we have $[6,3,4]$ Hermitian self-orthogonal code with generator $\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & \omega & \omega^2 \\ 0 & 0 & 1 & 1 & \omega^2 & \omega \end{bmatrix}$, which corresponds to stabilizer code in binary standard form:

$$\begin{cases} \mathbf{g}_1 = [100100001101] \\ \mathbf{g}_2 = [010101000011] \\ \mathbf{g}_3 = [001101001110] \\ \mathbf{g}_4 = [000011001101] \\ \mathbf{g}_5 = [000000100111] \\ \mathbf{g}_6 = [000000011011] \end{cases}.$$

Example 5: With construction base on Theorem 4.1 from $\mathbf{G}^0$, QECCs with length eight satisfy the two conditions of Theorem 4.1 due to the simple form of $\mathbf{G}^0$:

$$\mathbf{G}^0 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Applying Theorem 4.2, we get extension $\mathbf{G}^2$ from $\mathbf{G}^0$:

$$\mathbf{G}^2 = \begin{bmatrix} 1 & 0 & \omega^2 & \omega^2 \\ 0 & 1 & \omega^2 & \omega^2 \\ \omega & \omega & 1 & 0 \\ \omega & \omega & 0 & 1 \end{bmatrix}.$$

Then, the generator matrix for Hermitian self-orthogonal code is:

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & \omega^2 & \omega^2 \\ 0 & 1 & 0 & 0 & 0 & 1 & \omega^2 & \omega^2 \\ 0 & 0 & 1 & 0 & \omega & \omega & 1 & 0 \\ 0 & 0 & 0 & 1 & \omega & \omega & 0 & 1 \end{bmatrix}. \tag{4.10}$$

We have corresponding quantum stabilizer code $[[8, 0, 4]]$, interpreted from Hermitian self-orthogonal code $[8, 0, 4]$ with the generators in [63].

Example 6: We construct the new $[[10, 0, 4]]$ quantum code starting from $\mathbf{G}^0$. It is easy to verify the two conditions of Theorem 4.1 due to the simple form of $\mathbf{G}^0$:

$$\mathbf{G}^0 = \begin{bmatrix} 1 & \omega & \omega \\ \omega & 1 & \omega \\ \omega & \omega & 1 \end{bmatrix}.$$

Applying Theorem 4.2, we get extension $\mathbf{G}^2$ from $\mathbf{G}^0$ in the following form:

$$\mathbf{G}^2 = \begin{bmatrix} 0 & 1 & 0 & \omega^2 & \omega^2 \\ 1 & 0 & 0 & \omega^2 & \omega^2 \\ 0 & 0 & 1 & \omega & \omega \\ 1 & 1 & \omega & 1 & \omega \\ 1 & 1 & \omega & \omega & 1 \end{bmatrix}.$$

Then, the generator matrix for Hermitian self-orthogonal code is:

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & \omega^2 & \omega^2 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & \omega^2 & \omega^2 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & \omega & \omega \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & \omega & 1 & \omega \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & \omega & \omega & 1 \end{bmatrix}. \qquad (4.11)$$

We have corresponding QECC $[[10, 0, 4]]$ that interprets Hermitian self-orthogonal code $[10, 5, 4]$ with the generators in Eq. 4.11.

4.2.3 Extension to get a longer length

We have already studied the proposed construction, and the optimal results of quantum stabilizer codes with odd lengths are $[[5, 1, 3]], [[7, 1, 3]], [[9, 1, 3]]$ and in the case of even lengths, we consider the optimal codes to be $[[6, 0, 4]], [[8, 0, 4]], [[10, 0, 4]]$. To analyze longer lengths, we considered the two following lemmas for $n = 2m$ and $n = 2m - 1$.

Lemma 4.1 *Let* $\boldsymbol{G}$ *be the generator matrix of* $[2(m-2), m-2, 4]$ *Hermitian self-orthogonal code, where* $\boldsymbol{G} = [\boldsymbol{I}|\boldsymbol{G}^0]$*. Then, the generator matrix* $\boldsymbol{G}$ *of the Hermitian self-orthogonal* $[2m, m, 4]$ *code* $(m > 5)$ *will be* $\boldsymbol{G} = [\boldsymbol{I}|\boldsymbol{G}^2]$ *where* $\boldsymbol{G}^2$ *is achieved by extending it from* $\boldsymbol{G}^0$ *by Theorem 4.2. It is interpreted to be* $[[2m, 0, 4]]$ *QECC.*

Proof: From Theorem 4.2, the construction of the generator matrix is

$$\mathbf{G}^2 = \begin{bmatrix} \mathbf{A}^2 & \mathbf{X}^2 \\ \mathbf{Y} & \mathbf{G}^0 \end{bmatrix},$$

and we have the following comments.

1. The distance between the two first rows of $\mathbf{G}$ is 4. They are calculated directly.

2. The linear code comes from the last $m - 2$ rows of $\mathbf{G}$ where the distance at least equals the distance of $[\mathbf{I}|\mathbf{G}^0]$, which is already known to have the distance 4.

3. We consider one row in the two first rows of $\mathbf{G}$ and one from the last $n - 2$. In the first half, the distance is 2. In the second half, the first two elements have minimum distance at least 1, and the remaining have a minimum distance of at least 1 (because their weights are even and odd). Then, we have a minimum distance for two rows of at least 4.

From three comments above, minimum distance for linear code is 4. Then, the new code with proposed construction with $\mathbf{G}^2$ be the right part is $[2m, m, 4]$.

Example 7: We construct the new [[12,0]] quantum code starting from $\mathbf{G}^0$ with the right part of the generator matrix in Eq. 4.10:

$$\mathbf{G}^0 = \begin{bmatrix} 1 & 0 & \omega^2 & \omega^2 \\ 0 & 1 & \omega^2 & \omega^2 \\ \omega & \omega & 1 & 0 \\ \omega & \omega & 0 & 1 \end{bmatrix}.$$

Then, from $\mathbf{G}^0$, for even length, we extend $\mathbf{G}^0$ to $\mathbf{G}^2$ under Theorem 4.2, with $x = [1\ \ \omega\ \ 1\ \ \omega]$. The generator matrix is as follows:

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & \omega & 1 & \omega \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & \omega & 1 & \omega \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & \omega^2 & \omega^2 \\ 0 & 0 & 0 & 1 & 0 & 0 & \omega & \omega & 0 & 1 & \omega^2 & \omega^2 \\ 0 & 0 & 0 & 0 & 1 & 0 & \omega & \omega & \omega & \omega & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & \omega & \omega & 0 & 1 \end{bmatrix}. \tag{4.12}$$

Generators in Eq. 4.12 correspond to $[12, 6, 4]$ Hermitian linear code. Then, $[[12, 0, 4]]$ quantum stabilizer code is transformed.

When we consider the extension by $\mathbf{G}^1 = \begin{bmatrix} \mathbf{A}^1 & \mathbf{X}^1 \\ \mathbf{Y} & \mathbf{G}^0 \end{bmatrix}$, we get the following lemma:

Lemma 4.2 *Let $\boldsymbol{G}$ be the generator matrix of $[2(m-2), m-2, 4]$ Hermitian self-orthogonal code, where $\boldsymbol{G} = [\boldsymbol{I}|\boldsymbol{G}^0]$. Then, generator matrix $\boldsymbol{G}$ of Hermitian self-orthogonal $[2m-1, m-1, 4]$ code $(m > 5)$ will be $\boldsymbol{G} = [\boldsymbol{I}|\boldsymbol{G}^1]$ where $\boldsymbol{G}^1$ is achieved by extending it from $\boldsymbol{G}^0$ with Theorem 4.2. It can be interpreted to be $[[2m-1, 1, 3]]$ QECC.*

Example 8: We construct the new $[[11, 1]]$ QECC starting from $\mathbf{G}^0$ with the right part of $\mathbf{G}$ in Eq. 4.10:

$$\mathbf{G}^0 = \begin{bmatrix} 1 & 0 & \omega^2 & \omega^2 \\ 0 & 1 & \omega^2 & \omega^2 \\ \omega & \omega & 1 & 0 \\ \omega & \omega & 0 & 1 \end{bmatrix}.$$

Then, from $\mathbf{G}^0$, an even length, we extend $\mathbf{G}^0$ to $\mathbf{G}^1$ with Theorem 4.2, for example with $x = [1 \;\; \omega \;\; 1 \;\; \omega]$. We have the following generator matrix:

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & \omega & 1 & \omega \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & \omega^2 & \omega^2 \\ 0 & 0 & 1 & 0 & 0 & \omega & \omega & 0 & 1 & \omega^2 & \omega^2 \\ 0 & 0 & 0 & 1 & 0 & \omega & \omega & \omega & \omega & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & \omega & \omega & 0 & 1 \end{bmatrix}. \tag{4.13}$$

The generator in Eq. 4.13 corresponds to $[11, 5, 4]$ Hermitian linear code and $[[11, 1, 3]]$ quantum stabilizer code.

Chapter 5

Quantum Stabilizer Codes Based on a New Construction of Self-orthogonal Trace-inner Product Codes over *GF(4)*

5.1 Introduction

Quantum stabilizer code is a kind of QECC constructed based on the stabilizer formalism. The most important advantage of quantum stabilizer codes is that quantum errors that affect an encoded quantum state can be diagnosed and removed by a group of quantum operators, thereby stabilizing this encoded quantum state. In addition, the stabilizer formalism allows quantum codes to be presented by classical error correction codes. Therefore, quantum stabilizer codes can be constructed from binary error correction codes if they satisfy a symplectic inner product (SIP). Many quantum stabilizer codes have been constructed based on the binary formalism with combinatorial design, such as quantum codes based on difference sets [71], based on group association schemes [72], based on circulant matrices [73] [74], or based on CSS structure over Finite field [75]. In paper [76], a quantum stabilizer code was proven to correspond to an additive code over Galois field 4 (*GF(4)*), which is self-orthogonal with respect to the trace-inner product. So far, many papers have focused on (1) the design of classical additive codes over *GF(4)* to achieve corresponding quantum stabilizer codes, such as self-dual codes over *GF(4)*, which have dimension "0" and can be represented by graphs [77]; (2) QECCs based on self-dual codes over *GF(4)* with the highest known minimum weights [78]; and (3) QECCs based on Hermitian self-orthogonal codes with extension design [79]. Most of these designed algorithms have focused on the self-dual trace-inner product codes or Hermitian self-orthogonal code over *GF(4)*. Hence, many constructions remain to be discovered by using the design of self-orthogonal trace-inner product codes.

The key result of this chapter is to propose a new construction of self-orthogonal trace-inner product codes over *GF(4)*. From two binary vectors, we generate the circulant and modified circulant matrices, and the generator matrix for quaternary linear codes is proposed. Then, the quantum stabilizer codes are derived from the linear codes. The advantage of the proposed construction is that our proposed codes give various dimensions of QECCs, and these minimum distances have good values.

5.2 Proposed construction

5.2.1 Proposed self-orthogonal trace-inner product codes over *GF(4)*

In this subsection, we first propose the construction of self-orthogonal, trace-inner product codes over *GF(4)*. The proposed construction if given as follows:

Construction 1: Let **A** and **B** be binary matrices. Specifically, **A** is the circulant matrix generated from the vector $[a_0\ a_1\ \cdots\ a_{n-1}]$ and its circulants to the right, and **B** is the circulant matrix generated from the vector $[b_0\ b_1\ \cdots\ b_{n-1}]$ and its circulants to left. Matrices **A** and **B** are size $n \times n$ and can be represented in the following form:

$$\mathbf{A} = \begin{bmatrix} a_0 & a_1 & \cdots & a_{n-1} \\ a_{n-1} & a_0 & \cdots & a_{n-2} \\ \vdots & \vdots & \ddots & \vdots \\ a_1 & a_2 & \cdots & a_0 \end{bmatrix} \text{ and } \mathbf{B} = \begin{bmatrix} b_0 & b_1 & \dots & b_{n-1} \\ b_1 & b_2 & \dots & b_0 \\ \vdots & \vdots & \ddots & \vdots \\ b_{n-1} & b_0 & \dots & b_{n-2} \end{bmatrix} \tag{5.1}$$

Next, the generator matrices of the additive code over *GF(4)* can be constructed as follows:

$$\mathbf{F} = \mathbf{A} + \omega\mathbf{B} = \begin{bmatrix} a_0 & a_1 & \cdots & a_{n-1} \\ a_{n-1} & a_0 & \cdots & a_{n-2} \\ \vdots & \vdots & \ddots & \vdots \\ a_1 & a_2 & \cdots & a_0 \end{bmatrix} + \omega \begin{bmatrix} b_0 & b_1 & \dots & b_{n-1} \\ b_1 & b_2 & \dots & b_0 \\ \vdots & \vdots & \ddots & \vdots \\ b_{n-1} & b_0 & \dots & b_{n-2} \end{bmatrix}$$

$$= \begin{bmatrix} a_0+\omega b_0 & a_1+\omega b_1 & \cdots & a_{n-1}+\omega b_{n-1} \\ a_{n-1}+\omega b_1 & a_0+\omega b_2 & \cdots & a_{n-2}+\omega b_0 \\ \vdots & \vdots & \ddots & \vdots \\ a_1+\omega b_{n-1} & a_2+\omega b_0 & \cdots & a_0+\omega b_{n-2} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_0 \\ \mathbf{f}_1 \\ \vdots \\ \mathbf{f}_{n-1} \end{bmatrix}. \tag{5.2}$$

Then, the matrix **F** with the above construction satisfies the conditions necessary to be the generator of a self-orthogonal trace-inner product code.

Proof:

For any binary values a_k, b_k, and a *GF(4)* element ω, we first consider some basic equations over *GF(4)*:

1. ${a_k}^2 = {a_k}^3 = a_k$

2. $(a_k + \omega \times b_k)^2 = a_k + \omega^2 \times b_k$

3. $\omega^2 = \omega + 1$ and $\omega^3 = 1$

4. $Tr(a_k \times b_k) = 0$

5. $Tr(\omega \times a_k + \omega^2 \times b_k) = \omega \times a_k + \omega^2 \times b_k + (\omega \times a_k + \omega^2 \times b_k)^2 = \omega \times a_k + \omega^2 \times b_k + \omega^2 \times a_k + \omega \times b_k = (\omega + \omega^2) \times (a_k + b_k) = a_k + b_k$.

First, we consider the trace-inner product of two elements $\mathbf{f}_0$ and $\mathbf{f}_1$ as following.

$$\begin{aligned}
&\mathbf{f}_0 \bullet \mathbf{f}_1 \\
&= Tr\left[(a_0+\omega b_0)\times(a_{n-1}+\omega b_1)^2\right] + Tr\left[(a_1+\omega b_1)\times(a_0+\omega b_2)^2\right] + ... \\
&+ Tr\left[(a_{n-1}+\omega b_{n-1})\times(a_{n-2}+\omega b_0)^2\right] \\
&= Tr\left[(a_0+\omega b_0)\times(a_{n-1}+\omega^2 b_1)\right] + Tr\left[(a_1+\omega b_1)\times(a_0+\omega^2 b_2)\right] + ... \\
&+ Tr\left[(a_{n-1}+\omega b_{n-1})\times(a_{n-2}+\omega^2 b_0)\right] \\
&= Tr\left[a_0a_{n-1}+b_0b_1+\omega b_0a_{n-1}+\omega^2 a_0b_1\right] + Tr\left[a_1a_0+b_1b_2+\omega b_1a_0+\omega^2 a_1b_2\right] + ... \\
&+ Tr\left[a_{n-1}a_{n-2}+b_{n-1}b_0+\omega b_{n-1}a_{n-2}+\omega^2 a_{n-1}b_0\right] \\
&= Tr\left[\omega b_0a_{n-1}+\omega^2 a_0b_1\right] + Tr\left[\omega b_1a_0+\omega^2 a_1b_2\right] + ... + Tr\left[\omega b_{n-1}a_{n-2}+\omega^2 a_{n-1}b_0\right] \\
&= (b_0a_{n-1}+a_0b_1) + (b_1a_0+a_1b_2) + (b_2a_1+a_2b_3) + ... + (b_{n-1}a_{n-2}+a_{n-1}b_0) \\
&= b_0a_{n-1} + (a_0b_1+b_1a_0) + (a_1b_2+b_2a_1) + (a_2b_3+...+b_{n-1}a_{n-2}) + a_{n-1}b_0 \\
&= b_0a_{n-1}+a_{n-1}b_0 = 0.
\end{aligned} \tag{5.3}$$

Generally, to prove the self-orthogonal nature of matrix $\mathbf{F}$, we consider the trace-inner product of any two rows, such as the l-th row and the $(l+k)$-th row, of matrix $\mathbf{F}$. We will prove that $\mathbf{f}_l \bullet \mathbf{f}_{l+k} = 0$. Based on the full circulant properties of $\mathbf{F}$, we can also generate the full matrix $\mathbf{F}$ using the vector in the l-th row. Hence, without loss of generality and to reduce the complexity of the proof equation, we just need to consider the trace-inner product of $\mathbf{f}_0$ and $\mathbf{f}_k$. Their trace-inner product is expressed as following.

$$
\begin{aligned}
&\mathbf{f}_0 \bullet \mathbf{f}_k \\
&= Tr\left[(a_0+\omega b_0)\times(a_{n-k}+\omega b_k)^2\right] + Tr\left[(a_1+\omega b_1)\times(a_{n-k+1}+\omega b_{k+1})^2\right] + ... \\
&+ Tr\left[(a_{n-1}+\omega b_{n-1})\times(a_{n-k-1}+\omega b_{k-1})^2\right] \\
&= Tr\left[(a_0+\omega b_0)\times(a_{n-k}+\omega^2 b_k)\right] + Tr\left[(a_1+\omega b_1)\times(a_{n-k+1}+\omega^2 b_{k+1})\right] + ... \\
&+ Tr\left[(a_{n-1}+\omega b_{n-1})\times(a_{n-k-1}+\omega^2 b_{k-1})\right] \\
&= Tr\left[a_0 a_{n-k} + b_0 b_k + \omega b_0 a_{n-k} + \omega^2 a_0 b_k\right] + ... \\
&+ Tr\left[a_{n-1}a_{n-k-1} + b_{n-1}b_{k-1} + \omega b_{n-1}a_{n-k-1} + \omega^2 a_{n-1}b_{k-1}\right] \\
&= Tr\left[\omega b_0 a_{n-k} + \omega^2 a_0 b_k\right] + ... + Tr\left[\omega b_{n-1}a_{n-k-1} + \omega^2 a_{n-1}b_{k-1}\right] \\
&= (b_0 a_{n-k} + a_0 b_k) + (b_1 a_{n-k+1} + a_1 b_{k+1}) + (b_2 a_{n-k+2} + ... + (b_{n-1}a_{n-k-1} + a_{n-1}b_{k-1}) \\
&= \sum_{x=0}^{n-1} b_x a_{n-k+x} + \sum_{y=0}^{n-1} a_y b_{k+y} = \sum_{z+k-n=0}^{n-1} b_{z+k-n} a_z + \sum_{y=0}^{n-1} a_y b_{k+y} \\
&= \sum_{z=0}^{n-1} b_{z+k} a_z + \sum_{y=0}^{n-1} a_y b_{k+y} = 0.
\end{aligned}
\tag{5.4}
$$

As shown by the above explanation, the trace-inner product of any two vectors with generators in **F** is zero, which implies that matrix **F** is the generator matrix of a self-orthogonal, trace-inner product code.

5.2.2 Generator matrix generation of the proposed quantum stabilizer codes with length from 7 to 12

In this subsection, quantum stabilizer codes based on the proposed construction are investigated. For each code length, we first give the construction of the generator matrix of the additive code over *GF(4)* based on Construction 1. Then, using Table I, we obtain the generator matrix of quantum stabilizer codes that corresponding to the self-orthogonal trace-inner product codes. The parameters of the quantum stabilizer codes are calculated from the generator matrix of the linear codes over *GF(4)* by using the Magma calculation tool's *QuantumCode*, *MinimumWeight* functions [82]. Since any two binary vectors are satisfied by our proposed construction, there are many candidates for the generator matrix of additive codes over *GF(4)*. Therefore, for each code length, we consider quantum stabilizer codes with various dimensions and the minimum distances that were determined by the Magma tool's functions.

Example 1: Quantum stabilizer codes with a length of seven.

We explain the code construction of the proposed quantum stabilizer code with length $n = 7$. First, we consider an additive code over *GF(4)*, where its generator is generated from two vectors $\mathbf{u} = [1\ 1\ 0\ 0\ 1\ 0\ 1]$ and $\mathbf{v} = [1\ 0\ 0\ 1\ 0\ 1\ 1]$.

From the two vectors $\mathbf{u}$ and $\mathbf{v}$, as shown in Construction 1, we have the corresponding generator matrix:

$$\mathbf{F} = \begin{bmatrix} 1,1,0,0,1,0,1 \\ 1,1,1,0,0,1,0 \\ 0,1,1,1,0,0,1 \\ 1,0,1,1,1,0,0 \\ 0,1,0,1,1,1,0 \\ 0,0,1,0,1,1,1 \\ 1,0,0,1,0,1,1 \end{bmatrix} + \omega \begin{bmatrix} 1,0,0,1,0,1,1 \\ 0,0,1,0,1,1,1 \\ 0,1,0,1,1,1,0 \\ 1,0,1,1,1,0,0 \\ 0,1,1,1,0,0,1 \\ 1,1,1,0,0,1,0 \\ 1,1,0,0,1,0,1 \end{bmatrix} = \begin{bmatrix} \omega+1, 1, 0,\omega, 1,\omega,\omega+1 \\ 1, 1,\omega+1, 0,\omega,\omega+1,\omega \\ 0,\omega+1, 1,\omega+1,\omega,\omega, 1 \\ \omega^2, 0,\omega^2,\omega^2,\omega+1, 0, 0 \\ 0,\omega+1,\omega,\omega+1, 1, 1,\omega \\ \omega,\omega,\omega+1, 0, 1,\omega+1, 1 \\ \omega+1,\omega, 0, 1,\omega, 1,\omega+1 \end{bmatrix}.$$

Using Magma calculation tool's *QuantumCode* and *MinimumWeight* functions with the matrix $\mathbf{F}$ as the input, we get a quantum stabilizer code with parameter

$[[7,1,3]]$, the standard form for its generators are:

$$\mathbf{G} = \begin{bmatrix} 1,\ 0,\ 0,\ 1, 0, 1, 1 \\ \mathrm{w}, 0, 0, \mathrm{w}, 0, \mathrm{w}, \mathrm{w} \\ 0,\ 1,\ 0,\ 1, 1, 1, 0 \\ 0, \mathrm{w}, 0, \mathrm{w}, \mathrm{w}, \mathrm{w}, 0 \\ 0,\ 0,\ 1,\ 0,\ 1, 1, 1 \\ 0, 0, \mathrm{w}, 0, \mathrm{w}, \mathrm{w}, \mathrm{w} \end{bmatrix} \text{ and } \begin{cases} \mathbf{g}_1 = \mathbf{XIIXIXX} \\ \mathbf{g}_2 = \mathbf{ZIIZIZZ} \\ \mathbf{g}_3 = \mathbf{IXIXXXI} \\ \mathbf{g}_4 = \mathbf{IZIZZZI} \\ \mathbf{g}_5 = \mathbf{IIXIXXX} \\ \mathbf{g}_6 = \mathbf{IIZIZZZ} \end{cases}.$$

Similar to the mapping between *GF(4)* and Pauli matrix in Table 1, we have quantum stabilizer operators for quantum stabilizer code [[7,1,3]] as shown above.

Example 2: Quantum stabilizer codes with lengths from 8 to 10.

For $n = 8$, let us consider an additive code over *GF(4)*, where its generator is generated from two vectors $\mathbf{u} = [0\ 1\ 1\ 1\ 0\ 1\ 0\ 0]$, and $\mathbf{v} = [1\ 1\ 1\ 0\ 1\ 0\ 0\ 0]$. This results in a quantum stabilizer code with parameter $[[8,1,3]]$ and its generators are reduced as follows.

$$\mathbf{G} = \begin{bmatrix} 0,1,1,1,0,1,0,0 \\ 0,0,1,1,1,0,1,0 \\ 0,0,0,1,1,1,0,1 \\ 1,0,0,0,1,1,1,0 \\ 0,1,0,0,0,1,1,1 \\ 1,0,1,0,0,0,1,1 \\ 1,1,0,1,0,0,0,1 \end{bmatrix} + \omega \begin{bmatrix} 1,1,1,0,1,0,0,0 \\ 1,1,0,1,0,0,0,1 \\ 1,0,1,0,0,0,1,1 \\ 0,1,0,0,0,1,1,1 \\ 1,0,0,0,1,1,1,0 \\ 0,0,0,1,1,1,0,1 \\ 0,0,1,1,1,0,1,0 \end{bmatrix}.$$

For $n = 9$, let us consider an additive code over *GF(4)*, where its generator is generated from two vectors $\mathbf{u} = [0\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 1]$, and $\mathbf{v} = [1\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0]$. This results in a quantum stabilizer code with parameter $[[9,1,3]]$, and its generators are reduced as follows.

$$\mathbf{G} = \begin{bmatrix} 0,1,1,0,0,1,0,0,1 \\ 1,0,1,1,0,0,1,0,0 \\ 0,1,0,1,1,0,0,1,0 \\ 0,0,1,0,1,1,0,0,1 \\ 1,0,0,1,0,1,1,0,0 \\ 0,1,0,0,1,0,1,1,0 \\ 0,0,1,0,0,1,0,1,1 \\ 1,0,0,1,0,0,1,0,1 \end{bmatrix} + \omega \begin{bmatrix} 1,1,0,0,1,0,0,1,0 \\ 1,0,0,1,0,0,1,0,1 \\ 0,0,1,0,0,1,0,1,1 \\ 0,1,0,0,1,0,1,1,0 \\ 1,0,0,1,0,1,1,0,0 \\ 0,0,1,0,1,1,0,0,1 \\ 0,1,0,1,1,0,0,1,0 \\ 1,0,1,1,0,0,1,0,0 \end{bmatrix}.$$

For $n = 10$, let us consider an additive code over *GF(4)*, where its generator is generated from two vectors $\mathbf{u} = [0\ 1\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 0]$, and $\mathbf{v} = [1\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 0]$. This results in a quantum stabilizer code with parameter $[[10,1,3]]$, and its generators are reduced as follows.

$$\mathbf{G} = \begin{bmatrix} 0,1,1,1,0,1,1,0,1,0 \\ 0,0,1,1,1,0,1,1,0,1 \\ 1,0,0,1,1,1,0,1,1,0 \\ 0,1,0,0,1,1,1,0,1,1 \\ 1,0,1,0,0,1,1,1,0,1 \\ 1,1,0,1,0,0,1,1,1,0 \\ 0,1,1,0,1,0,0,1,1,1 \\ 1,0,1,1,0,1,0,0,1,1 \\ 1,1,0,1,1,0,1,0,0,1 \end{bmatrix} + \omega \begin{bmatrix} 1,1,1,0,1,1,0,1,0,0 \\ 1,1,0,1,1,0,1,0,0,1 \\ 1,0,1,1,0,1,0,0,1,1 \\ 0,1,1,0,1,0,0,1,1,1 \\ 1,1,0,1,0,0,1,1,1,0 \\ 1,0,1,0,0,1,1,1,0,1 \\ 0,1,0,0,1,1,1,0,1,1 \\ 1,0,0,1,1,1,0,1,1,0 \\ 0,0,1,1,1,0,1,1,0,1 \end{bmatrix}.$$

Example 3: Quantum stabilizer codes with a length of eleven.

For $n = 11$, let us consider an additive code over *GF(4)* where its generator is generated from two vectors $\mathbf{u} = [1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1]$, and $\mathbf{v} = [1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 1]$. This results in quantum stabilizer codes with parameters $[[11, 1, 3]]$ and $[[11, 2, 3]]$, and these generators are reduced as shown below:

For the quantum stabilizer code with parameter $[[11, 1, 3]]$:

$$\mathbf{G} = \begin{bmatrix} 1,1,1,0,0,1,1,0,0,0,1 \\ 1,1,1,1,0,0,1,1,0,0,0 \\ 0,1,1,1,1,0,0,1,1,0,0 \\ 0,0,1,1,1,1,0,0,1,1,0 \\ 0,0,0,1,1,1,1,0,0,1,1 \\ 1,0,0,0,1,1,1,1,0,0,1 \\ 1,1,0,0,0,1,1,1,1,0,0 \\ 0,1,1,0,0,0,1,1,1,1,0 \\ 0,0,1,1,0,0,0,1,1,1,1 \\ 1,0,0,1,1,0,0,0,1,1,1 \end{bmatrix} + \omega \begin{bmatrix} 1,1,0,0,1,1,0,0,0,1,1 \\ 1,0,0,1,1,0,0,0,1,1,1 \\ 0,0,1,1,0,0,0,1,1,1,1 \\ 0,1,1,0,0,0,1,1,1,1,0 \\ 1,1,0,0,0,1,1,1,1,0,0 \\ 1,0,0,0,1,1,1,1,0,0,1 \\ 0,0,0,1,1,1,1,0,0,1,1 \\ 0,0,1,1,1,1,0,0,1,1,0 \\ 0,1,1,1,1,0,0,1,1,0,0 \\ 1,1,1,1,0,0,1,1,0,0,0 \end{bmatrix}.$$

For the quantum stabilizer code with parameter $[[11, 2, 3]]$:

$$\mathbf{G} = \begin{bmatrix} 1,1,1,0,0,1,1,0,0,0,1 \\ 1,1,1,1,0,0,1,1,0,0,0 \\ 0,1,1,1,1,0,0,1,1,0,0 \\ 0,0,1,1,1,1,0,0,1,1,0 \\ 1,0,0,0,1,1,1,1,0,0,1 \\ 1,1,0,0,0,1,1,1,1,0,0 \\ 0,1,1,0,0,0,1,1,1,1,0 \\ 0,0,1,1,0,0,0,1,1,1,1 \\ 1,0,0,1,1,0,0,0,1,1,1 \end{bmatrix} + \omega \begin{bmatrix} 1,1,0,0,1,1,0,0,0,1,1 \\ 1,0,0,1,1,0,0,0,1,1,1 \\ 0,0,1,1,0,0,0,1,1,1,1 \\ 0,1,1,0,0,0,1,1,1,1,0 \\ 1,0,0,0,1,1,1,1,0,0,1 \\ 0,0,0,1,1,1,1,0,0,1,1 \\ 0,0,1,1,1,1,0,0,1,1,0 \\ 0,1,1,1,1,0,0,1,1,0,0 \\ 1,1,1,1,0,0,1,1,0,0,0 \end{bmatrix}.$$

Example 4: Quantum stabilizer codes with a length of twelve.

For $n = 12$, let us consider an additive code over *GF(4)*, where its generator is generated from two vectors $u = [1\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1]$, and $v = [1\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 1]$. This results in quantum stabilizer codes with parameters $[[12, 1, 4]]$, $[[12, 2, 3]]$, $[[12, 3, 3]]$, and $[[12, 4, 3]]$. These generators are reduced as shown below.

For the quantum stabilizer code with parameter $[[12, 1, 4]]$.

$$\mathbf{G} = \begin{bmatrix} 1,1,1,0,0,1,0,1,0,1,0,1 \\ 1,1,1,1,0,0,1,0,1,0,1,0 \\ 0,1,1,1,1,0,0,1,0,1,0,1 \\ 1,0,1,1,1,1,0,0,1,0,1,0 \\ 0,1,0,1,1,1,1,0,0,1,0,1 \\ 1,0,1,0,1,1,1,1,0,0,1,0 \\ 0,1,0,1,0,1,1,1,1,0,0,1 \\ 1,0,1,0,1,0,1,1,1,1,0,0 \\ 0,1,0,1,0,1,0,1,1,1,1,0 \\ 0,0,1,0,1,0,1,0,1,1,1,1 \\ 1,0,0,1,0,1,0,1,0,1,1,1 \end{bmatrix} + \omega \begin{bmatrix} 1,1,0,0,1,0,1,0,1,0,1,1 \\ 1,0,0,1,0,1,0,1,0,1,1,1 \\ 0,0,1,0,1,0,1,0,1,1,1,1 \\ 0,1,0,1,0,1,0,1,1,1,1,0 \\ 1,0,1,0,1,0,1,1,1,1,0,0 \\ 0,1,0,1,0,1,1,1,1,0,0,1 \\ 1,0,1,0,1,1,1,1,0,0,1,0 \\ 0,1,0,1,1,1,1,0,0,1,0,1 \\ 1,0,1,1,1,1,0,0,1,0,1,0 \\ 0,1,1,1,1,0,0,1,0,1,0,1 \\ 1,1,1,1,0,0,1,0,1,0,1,0 \end{bmatrix}.$$

For the quantum stabilizer code with parameter $[[12, 2, 3]]$:

$$\mathbf{G} = \begin{bmatrix} 1,1,1,1,0,0,1,0,1,0,1,0 \\ 0,1,1,1,1,0,0,1,0,1,0,1 \\ 1,0,1,1,1,1,0,0,1,0,1,0 \\ 0,1,0,1,1,1,1,0,0,1,0,1 \\ 1,0,1,0,1,1,1,1,0,0,1,0 \\ 0,1,0,1,0,1,1,1,1,0,0,1 \\ 1,0,1,0,1,0,1,1,1,1,0,0 \\ 0,1,0,1,0,1,0,1,1,1,1,0 \\ 0,0,1,0,1,0,1,0,1,1,1,1 \\ 1,0,0,1,0,1,0,1,0,1,1,1 \end{bmatrix} + \omega \begin{bmatrix} 1,0,0,1,0,1,0,1,0,1,1,1 \\ 0,0,1,0,1,0,1,0,1,1,1,1 \\ 0,1,0,1,0,1,0,1,1,1,1,0 \\ 1,0,1,0,1,0,1,1,1,1,0,0 \\ 0,1,0,1,0,1,1,1,1,0,0,1 \\ 1,0,1,0,1,1,1,1,0,0,1,0 \\ 0,1,0,1,1,1,1,0,0,1,0,1 \\ 1,0,1,1,1,1,0,0,1,0,1,0 \\ 0,1,1,1,1,0,0,1,0,1,0,1 \\ 1,1,1,1,0,0,1,0,1,0,1,0 \end{bmatrix}.$$

For the quantum stabilizer code with parameter $[[12,3,3]]$:

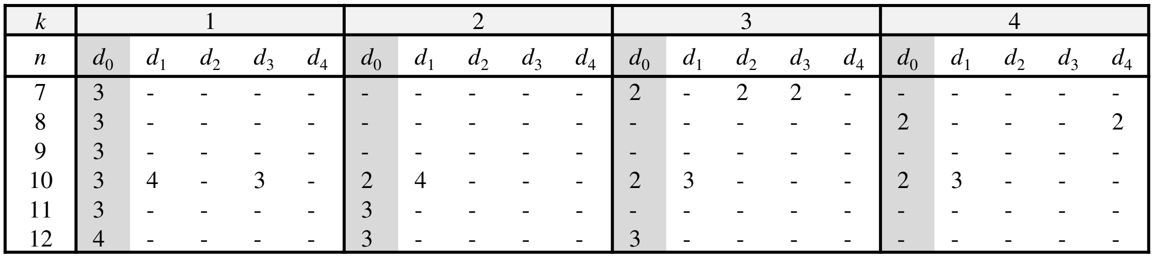
$$\mathbf{G} = \begin{bmatrix} 0,1,1,1,1,0,0,1,0,1,0,1 \\ 1,0,1,1,1,1,0,0,1,0,1,0 \\ 0,1,0,1,1,1,1,0,0,1,0,1 \\ 1,0,1,0,1,1,1,1,0,0,1,0 \\ 0,1,0,1,0,1,1,1,1,0,0,1 \\ 1,0,1,0,1,0,1,1,1,1,0,0 \\ 0,1,0,1,0,1,0,1,1,1,1,0 \\ 0,0,1,0,1,0,1,0,1,1,1,1 \\ 1,0,0,1,0,1,0,1,0,1,1,1 \end{bmatrix} + \omega \begin{bmatrix} 0,0,1,0,1,0,1,0,1,1,1,1 \\ 0,1,0,1,0,1,0,1,1,1,1,0 \\ 1,0,1,0,1,0,1,1,1,1,0,0 \\ 0,1,0,1,0,1,1,1,1,0,0,1 \\ 1,0,1,0,1,1,1,1,0,0,1,0 \\ 0,1,0,1,1,1,1,0,0,1,0,1 \\ 1,0,1,1,1,1,0,0,1,0,1,0 \\ 0,1,1,1,1,0,0,1,0,1,0,1 \\ 1,1,1,1,0,0,1,0,1,0,1,0 \end{bmatrix}.$$

For the quantum stabilizer code with parameter $[[12,4,3]]$:

$$\mathbf{G} = \begin{bmatrix} 0,1,1,1,1,0,0,1,0,1,0,1 \\ 1,0,1,1,1,1,0,0,1,0,1,0 \\ 0,1,0,1,1,1,1,0,0,1,0,1 \\ 1,0,1,0,1,1,1,1,0,0,1,0 \\ 1,0,1,0,1,0,1,1,1,1,0,0 \\ 0,1,0,1,0,1,0,1,1,1,1,0 \\ 0,0,1,0,1,0,1,0,1,1,1,1 \\ 1,0,0,1,0,1,0,1,0,1,1,1 \end{bmatrix} + \omega \begin{bmatrix} 0,0,1,0,1,0,1,0,1,1,1,1 \\ 0,1,0,1,0,1,0,1,1,1,1,0 \\ 1,0,1,0,1,0,1,1,1,1,0,0 \\ 0,1,0,1,0,1,1,1,1,0,0,1 \\ 0,1,0,1,1,1,1,0,0,1,0,1 \\ 1,0,1,1,1,1,0,0,1,0,1,0 \\ 0,1,1,1,1,0,0,1,0,1,0,1 \\ 1,1,1,1,0,0,1,0,1,0,1,0 \end{bmatrix}.$$

5.2.3 Comparison between proposed codes with referenced codes

The minimum distance of quantum stabilizer codes with lengths ranging from 7 to 12 and the dimension k ranging from 1 to 4, which were derived from our proposed



k	1					2					3					4				
n	d_0	d_1	d_2	d_3	d_4	d_0	d_1	d_2	d_3	d_4	d_0	d_1	d_2	d_3	d_4	d_0	d_1	d_2	d_3	d_4
7	3	-	-	-	-	-	-	-	-	-	2	-	2	2	-	-	-	-	-	-
8	3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	2	-	-	-	2
9	3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
10	3	4	-	3	-	2	4	-	-	-	2	3	-	-	-	2	3	-	-	-
11	3	-	-	-	-	3	-	-	-	-	-	-	-	-	-	-	-	-	-	-
12	4	-	-	-	-	3	-	-	-	-	3	-	-	-	-	-	-	-	-	-

Figure 5.1: Minimum distance of proposed $[[n, k, d_{min}]]$ in comparison with four referenced papers.

construction and referenced researches [83] [84] [85] and [86], are listed together in Figure 5.1. For each value in row k and column n, the notation d_0 stands for the minimum distance of the existing quantum stabilizer code $[[n, k, d_0]]$ derived from our proposed construction. Similarly, the notations d_1, d_2, d_3, and d_4 denote the minimum distance of the existing $[[n, k, d_i]]$ $(i = 1, 2, 3, 4)$ in the referenced papers [83] [84] [85] and [86], respectively. The blanks in Figure 5.1 mean that there are no existing quantum stabilizer codes with length n and dimension k.

As can be seen in Figure 5.1, the proposed code construction methods can generate more quantum stabilizer codes than the referenced constructions for code length ranging from 7 to 12. Moreover, the minimum distances of the proposed quantum stabilizer codes are than or equal to the ones of three referenced codes, i.e., [83] [84] and [86]. The minimum distances of quantum stabilizer codes in [85] are larger than the ones of the proposed codes for a code length of 10; the code construction in [85] was specifically designed for only a length of 10. Therefore, quantum stabilizer codes where these lengths are 7, 8, 9, 11, or 12 cannot be generated from the code construction in [85]. The constraint method, limitations of code length, and distinct results of

Table 5.1: Comparison between referenced papers and proposed method.

Paper	**Construction Method**	**Limitation of Code Length**	**Main Results**
[83]	Based on the combination of properties of Legendre symbols and the Pauli block matrix.	Limited due to the condition of Legendre symbols; only applicable for $p = 4m + 1$ and $p = 4m + 3$.	No optimal quantum stabilizer codes in the literature.
[84]	Based on difference sets and cyclic code	Limited because difference sets do not exist for all lengths.	Many results with minimum distance of two: $[[5, 1, 2]]$ and $[[6, 1, 2]]$.
[85]	Based on non-residue sets, extended to block square matrix and cyclic code.	Limited code length since residue sets are just for $p = 4n + 1, 8n - 1$ and $p = 4n - 1$, $4n + 1$ to get codes with lengths equal to pk.	Quantum code with length of 10: $[[10, 1, 4]]$, $[[10, 2, 4]]$, $[[10, 3, 3]]$, and $[[10, 4, 3]]$.
[86]	Based on the Pauli block transformation for codes with even lengths.	Quantum codes limited to even lengths.	No optimal quantum stabilizer codes in the literature.
Proposed method	Based on the circulant matrix.	No limitations; any length has its own parity-check matrix.	Optimal codes with generators in the standard form can be constructed. Codes with a minimum distance of three or four are shown.

the proposed code constructions and the four referenced construction are summarized in Table. 5.1.

Chapter 6

Minimal-Entanglement Entanglement-Assisted Quantum Error Correction Codes from Modified Circulant Matrices

6.1 Introduction

Stabilizer codes, first introduced by Gottesman [87], have become an important class of QECC, since these codes are useful for building quantum fault-tolerant circuits [88]. Stabilizer codes append ancilla qubits to qubits to be protected, and the most important advantage of stabilizer codes is that errors can be detected and removed from stabilizer operators, rather than from the quantum state itself. In addition, the stabilizer formalism allows us to construct quantum stabilizer codes from binary matrices over binary in the constraint referred to as the symplectic inner product (SIP) [87]. With Calderbank-Shor-Steane (CSS)-based construction [89], the problem turned out to be utilizing self-orthogonal classical codes. However, self-orthogonal codes with high error-correcting capacity are restricted, and therefore, further investigation was required to generate good stabilizer codes. Introduction of entanglement-assisted quantum error correction code (EAQECC) by Brun et al. [90] is one answer to this problem. More precisely, it enables us to construct the quantum error-correction codes not only from self-orthogonal classical codes, but also from arbitrary classical codes with the help of copies of maximally entangled quantum states shared between encoder and decoder. To design efficient EAQECC, however, it is desirable to use the fewest entangled states possible, because the cost to prepare those states is relatively high. Hence, the construction of EAQECC with small amounts of entangled states is a much more attractive issue [90, 91]. Therefore, several constructions of EAQECC have been proposed, such as construction from arbitrary matrices where the number of ebits is determined by parameters of classical codes [92], from low-density parity-check (LDPC) codes [91], from generalized quadrangles [93], from circulant permutation matrices [94], and from shortened Hamming codes [95]. Almost all existing constructions consider classical codes to calculate the number of ebits. To do so, the problem of transforming the classical form to basic form of EAQECC was proposed in the Gram–Schmidt procedure. This aims to classify the classical form into isotropic and entanglement subgroups, but the complexity of the Gram–Schmidt procedure also increases in proportion to the length of the codes [96]. Furthermore, the encoding transforms the non-commuting set of generators into its canonical form [97].

Then, quantum circuits composed of $CNOT$, H, and S gates can be derived directly with complexity $\mathrm{O}(n^2)$. The canonical form gives the relationship between EAQECC and quantum stabilizer codes, even though we can use the property of the stabilizer code that is useful for fault-tolerant computation [98].

The key result of this chapter is to propose novel approaches to construction of EAQECC. First, we propose a new method for the construction of the isotropic subgroup based on circulant matrices. Then, the entanglement subgroup can be determined from a method of transforming the isotropic group into standard form; hence, the parameters of codes are found, and for effective preparation of the entangled state, the number of ebits should be as few as possible. To explain the practical construction of the quantum codes, design of the proposed EAQECC with lengths up to 12 are shown. In addition, the minimum distance is calculated and explained to show that the proposed construction has good correctable capability, in comparison with recent EAQECC.

6.2 Proposed construction

In this section, general properties of cyclic matrices and circulant matrices are introduced first. Then, we discuss the construction of the isotropic subgroup from the proposed modified circulant matrix, and then the calculation for an entanglement group is given. As a consequence, the parameters for EAQECC are obtained.

6.2.1 Cyclic matrices

Definition 1. *(**Cyclic Matrix**) Let $\mathbf{I}_n$ be the $n \times n$ identity matrix. A cyclic matrix $\mathbf{I}_n(x)$ is a shifted identity matrix with the rows of $\mathbf{I}_n$ circularly shifted to the right by x positions, where $x \in \{0, 1, ..., n-1\}$ is the offset.*

In general, it is known that $\mathbf{I}_n(0) = \mathbf{I}_n$ and $\mathbf{I}_n(x \pm kn) = \mathbf{I}_n(x)$ for any integer k. The multiplication of $\mathbf{I}_n(1)$ and $\mathbf{I}_n(1)$ is $\mathbf{I}_n(2)$. Therefore, if $\mathbf{I}_n(1)^c$ is denoted as c times the multiplication of $\mathbf{I}_n(1)$, then $\mathbf{I}_n(1)^c = \mathbf{I}_n(c)$.

Example 1. *For $n = 4$, the cyclic matrix and relations are given as follows:*

$$\mathbf{I}_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \ \mathbf{I}_4(1) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}, \tag{6.1}$$

$$\text{and } \mathbf{I}_4(1)^2 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} = \mathbf{I}_4.$$

Definition 2. *(**Left-cyclic Matrix**) Let $\mathbf{J}_n$ be the $n \times n$ binary matrix made from the π-rotation of identity matrix $\mathbf{I}_n$. A cyclic matrix $\mathbf{J}_n(x)$ is a shifted $\mathbf{J}_n$ with the rows of $\mathbf{J}_n$ circularly shifted to the right by x positions, where $x \in \{0, 1, ..., n-1\}$ is the offset.*

In general, it is known that the transpose matrix of any **Left-**cyclic matrix is equal to itself. Therefore, any **Left-**cyclic matrix is a symmetric matrix.

Example 2. *For $n = 4$, the following **Left-**cyclic matrix and relations are given:*

$$\mathbf{J}_4 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}, \ \mathbf{J}_4(1) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \text{ and } \mathbf{J}_4(1)^T = \mathbf{J}_4(1) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}. \tag{6.2}$$

6.2.2 Circulant matrices

Definition 3. ***(Circulant Matrix-CM)*** *An $n \times n$ binary matrix $\mathbf{Q}_1$ is called a CM if it is expressed as*

$$\mathbf{Q}_1 = \begin{bmatrix} i_0 & i_1 & i_2 & \cdots & i_{n-1} \\ i_{n-1} & i_0 & i_1 & \cdots & i_{n-2} \\ i_{n-2} & i_{n-1} & i_0 & \cdots & i_{n-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ i_1 & i_2 & i_3 & \cdots & i_0 \end{bmatrix},$$

where the entries $\{i_0,\ i_1,\ \ldots\ ,\ i_{n-1}\}$ of matrix $\mathbf{Q}_1$ are the binary values.

Circulant matrix $\mathbf{Q}_1$ can be expressed by using a cyclic matrix as follows:

$$\mathbf{Q}_1 = i_0 \times \mathbf{I}_n(0) + i_1 \times \mathbf{I}_n(1) + i_2 \times \mathbf{I}_n(2) + \ldots + i_{n-1} \times \mathbf{I}_n(n-1) = \begin{bmatrix} \mathbf{u} \times \mathbf{I}_n(0) \\ \mathbf{u} \times \mathbf{I}_n(1) \\ \vdots \\ \mathbf{u} \times \mathbf{I}_n(n-1) \end{bmatrix},$$

where $\mathbf{u} = [i_0\ i_1\ \ldots\ i_{n-1}]$. Therefore, we can denote $\mathbf{Q}_1$ as the function of vector $\mathbf{u}$ and variable n. Hereafter, we denote $\mathbf{Q}_1$ as $\mathbf{P}_1(\mathbf{u},\ n)$.

Definition 4. *(**Left-circulant Matrix—Left-CM**) An $n \times n$ binary matrix $\mathbf{Q}_2$ is called a Left-CM if it is expressed as*

$$\mathbf{Q}_2 = \begin{bmatrix} j_0 & j_1 & j_2 & \cdots & j_{n-1} \\ j_1 & j_2 & j_3 & \cdots & j_0 \\ j_2 & j_3 & j_4 & \cdots & j_1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ j_{n-1} & j_0 & j_1 & \cdots & j_{n-2} \end{bmatrix},$$

where the entries $\{j_0, j_1, \ldots, j_{n-1}\}$ of matrix $\mathbf{Q}_2$ are the binary values.

Circulant matrix $\mathbf{Q}_2$ can be expressed by using a left-cyclic matrix, such as

$$\mathbf{Q}_2 = j_0 \times \mathbf{J}_n(1) + j_1 \times \mathbf{J}_n(2) + j_2 \times \mathbf{J}_n(3) + ... + j_{n-2} \times \mathbf{J}_n(n-1) + j_{n-1} \times \mathbf{J}_n(0) = \begin{bmatrix} \mathbf{v} \times \mathbf{I}_n(0) \\ \mathbf{v} \times \mathbf{I}_n(n-1) \\ \vdots \\ \mathbf{v} \times \mathbf{I}_n(1) \end{bmatrix},$$

where $\mathbf{v} = [j_0 \ j_1 \ \ldots \ j_{n\text{-}1}]$. Therefore, we can denote $\mathbf{Q}_2$ as the function of vector $\mathbf{v}$ and variable n. Hereafter, we denote $\mathbf{Q}_2$ as $\mathbf{P}_2(\mathbf{v}, n)$.

6.2.3 Construction of parity-check matrices of EAQECC based on modified circulant matrices

From the combination of circulant matrices and the left-circulant matrices, the parity check matrix that corresponds to isotropic subgroup have been formed in Theorem 2, then the anti-commuting subgroup are determined as logical operators of the parity check matrix as Theorem 3. Finally, the EAQECCs $[[n, k, d_{\min}; 1]]$ are found as the Corollary 1.

Theorem 2. *For any two binary vectors* $\mathbf{u}$, $\mathbf{v}$ *of size* n*, two circulant matrices are* $\mathbf{P}_1(\mathbf{u},\ n)$ *and* $\mathbf{P}_2(\mathbf{v},\ n)$*. Then, the parity-check matrix* $\mathbf{H} = [\mathbf{H_X} \mid \mathbf{H_Z}]$*, where* $\mathbf{H_X}$ *and* $\mathbf{H_Z}$ *correspond to* $\mathbf{P}_1(\mathbf{u},\ n)$ *and* $\mathbf{P}_2(\mathbf{v},\ n)$*, respectively, satisfies the SIP condition in* (2)*.*

Proof: From Definitions 3 and 4, $\mathbf{P}_1(\mathbf{u},\ n)$ and $\mathbf{P}_2(\mathbf{v},\ n)$ can be written as

$$\mathbf{P}_1(\mathbf{u}, n) = \mathbf{I}_n(i_{\mathbf{u}_1}) + \mathbf{I}_n(i_{\mathbf{u}_2}) + ... + \mathbf{I}_n(i_{\mathbf{u}_k}) \Leftrightarrow \mathbf{H_X} = \mathbf{I}_n(i_{\mathbf{u}_1}) + \mathbf{I}_n(i_{\mathbf{u}_2}) + ... + \mathbf{I}_n(i_{\mathbf{u}_k}),$$

$$\mathbf{P}_2(\mathbf{v}, n) = \mathbf{J}_n(i_{\mathbf{v}_1}) + \mathbf{J}_n(i_{\mathbf{v}_2}) + ... + \mathbf{J}_n(i_{\mathbf{v}_h}) \Leftrightarrow \mathbf{H_Z} = \mathbf{J}_n(i_{\mathbf{v}_1}) + \mathbf{J}_n(i_{\mathbf{v}_2}) + ... + \mathbf{J}_n(i_{\mathbf{v}_h}),$$

where $\{u_1,\ u_2,\ \dots,\ u_k\}$ and $\{v_1,\ v_2,\ \dots,\ v_k\}$ are the positions of 1 at vectors $\mathbf{u}$ and $\mathbf{v}$, respectively.

From the properties of circulant matrices, we get following equation for any $0 < m,\ l\ l\ n$:

$$\begin{cases} \mathbf{J}_n(m) = \mathbf{J}_n(0) \times \mathbf{I}_n(m), \\ \mathbf{I}_n(l) \times \mathbf{J}_n(0) = \mathbf{J}_n(0) \times \mathbf{I}_n(n-l), \\ \mathbf{I}_n(l)^T = \mathbf{I}_n(n-l). \end{cases}$$

In addition, any left-cyclic matrix is a symmetric matrix. So, the following equation is always true:

$$\mathbf{I}_n(l) \times \mathbf{J}_n(m) = \mathbf{J}_n(m) \times \mathbf{I}_n(l)^T \Leftrightarrow \mathbf{I}_n(l) \times \mathbf{J}_n(m)^T = \mathbf{J}_n(m) \times \mathbf{I}_n(l)^T. \qquad (6.3)$$

From 6.3, we get:

$$(\mathbf{I}_n(i_{\mathbf{u}_1}) + \mathbf{I}_n(i_{\mathbf{u}_2}) + ... + \mathbf{I}_n(i_{\mathbf{u}_k}))(\mathbf{J}_n(i_{\mathbf{v}_1}) + \mathbf{J}_n(i_{\mathbf{v}_2}) + ... + \mathbf{J}_n(i_{\mathbf{v}_h}))^T$$

$$= (\mathbf{J}_n(i_{\mathbf{v}_1}) + \mathbf{J}_n(i_{\mathbf{v}_2}) + ... + \mathbf{J}_n(i_{\mathbf{v}_h}))(\mathbf{I}_n(i_{\mathbf{u}_1}) + \mathbf{I}_n(i_{\mathbf{u}_2}) + ... + \mathbf{I}_n(i_{\mathbf{u}_k}))^T.$$

$$\Leftrightarrow \mathbf{H_X} \times \mathbf{H_Z}^T = \mathbf{H_Z} \times \mathbf{H_X}^T \Leftrightarrow \mathbf{H_X} \times \mathbf{H_Z}^T + \mathbf{H_Z} \times \mathbf{H_X}^T = \mathbf{0}_n \text{ modulo 2.}$$

Therefore, the matrix $\mathbf{H} = [\mathbf{H_X} \mid \mathbf{H_Z}]$ satisfies the SIP condition in (2), and Theorem 2 is proven.

Since the parity-check matrices constructed from Theorem 2 satisfy the SIP condition in (2), we can choose the independent vectors from $\mathbf{H}$ to create corresponding isotropic subgroup $\mathrm{S_I}$. To have the entanglement subgroup, the following theorem can be considered to satisfy the conditions.

Theorem 3. *Given that parity-check matrix* $\mathbf{H}$ *of size* $(n-k) \times 2n$ *and its vectors are an independent relationship, we can transform* $\mathbf{H}$ *into the standard form* $\mathbf{H}_{\mathrm{st}}$ *in the following form:*

$$\mathbf{H}_{\mathrm{st}} = \left[\begin{array}{cccccc} \overbrace{\mathbf{I}}^{r} & \overbrace{\mathbf{A}_1}^{n-k-r} & \overbrace{\mathbf{A}_2}^{k} & \overbrace{\mathbf{B}}^{r} & \overbrace{\mathbf{C}_1}^{n-k-r} & \overbrace{\mathbf{C}_2}^{k} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{D} & \mathbf{I} & \mathbf{E} \end{array}\right] \begin{array}{l} \} \qquad r \\ \} \; n-k-r \end{array} \tag{6.4}$$

Then, the pairs of anti-commuting can be determined as

$$\begin{cases} \mathbf{X}_{\mathrm{E}} = & [\;\mathbf{0} \;\; \mathbf{E}^T \;\; \mathbf{I} \;\; (\mathbf{E}^T\mathbf{C}_1 + \mathbf{C}_2{}^T) \;\; \mathbf{0} \;\; \mathbf{0}\;] \\ \mathbf{Z}_{\mathrm{E}} = & [\;\mathbf{0} \;\; \mathbf{0} \;\; \mathbf{0} \;\; \mathbf{A}_2{}^T \;\; \mathbf{0} \;\; \mathbf{I}\;] \end{cases} \tag{6.5}$$

where the rank of matrix $\mathbf{X}_{\mathrm{E}}$ and $\mathbf{Z}_{\mathrm{E}}$ are k.

Proof.

(1) To transform the parity-check matrix to standard form, we use the Gauss-Jordan elimination, swap the qubits, and add one row to another. The codewords and stabilizer are invariant to these changes. So, step by step, the standard form can be obtained with 6.4.

(2) From the standard form, $\mathbf{H}_{\text{st}}$, we calculate the encoded Pauli operators $\overline{\mathbf{X}}$ and $\overline{\mathbf{Z}}$ that satisfy the following conditions:

$$\begin{cases} [\overline{\mathbf{X}_i}, \overline{\mathbf{X}_j}] = 0, \\ [\overline{\mathbf{Z}_i}, \overline{\mathbf{Z}_j}] = 0, \\ [\overline{\mathbf{X}_i}, \overline{\mathbf{Z}_j}] = 0 \text{ for } i \neq j, \\ \{\overline{\mathbf{X}_i}, \overline{\mathbf{Z}_j}\} = 0 \text{ for } i = j. \end{cases}$$

Consequently, encoded Pauli operators $\overline{\mathbf{X}}$ and $\overline{\mathbf{Z}}$ can be used as $\mathbf{X}_{\text{E}}$ and $\mathbf{Z}_{\text{E}}$. Theorem 3 is proven. □

EAQECCs use pre-existing entanglement between transmitter and receiver to improve the reliability of transmission. Hence, before transmission we must manufacture the entanglement state between transmitter and receiver. It will be difficult to set up if the number of ebits becomes larger. So, an EAQECC design to minimize the numbers of ebits is an important constraint. In following Corollary, we will consider the result of the Theorem 2 when the number of ebits is 1.

Corollary 1. *From the Theorem 3, firstly we choose one pair of anti-commuting from S_E, it denotes as $\{X_1, Z_1\}$. Then, we choose $n - k - 1$ generators $\{\mathbf{Z}_2, \mathbf{Z}_3, \ldots, \mathbf{Z}_{n\text{-}k}\}$ from parity check matrix that satisfy the commutation property of isotropic sub-group S_I. The minimum distance d_{min} is calculated from the generators of S_E and S_I. The EAQECCs with parameter [[n, k, d_{min};1]] is constructed.*

Proof.

As the definition of EAQECC in , the non-Abelian group can be partitioned into:

1. A commuting subgroup, the isotropic group $\boldsymbol{S}_{\text{I}} = \{\mathbf{Z}_{c+1}, \mathbf{Z}_{c+2}, \ldots, \mathbf{Z}_{c+s}\}$.

2. Entanglement subgroup pairs $\boldsymbol{S}_{\mathrm{E}} = \{\mathbf{Z}_1,\ \mathbf{Z}_2,\ \dots,\ \mathbf{Z}_c,\ \mathbf{X}_1,\ \mathbf{X}_2,\ \dots,\ \mathbf{X}_c\}$ with anti-commuting pairs; the anti-commuting pairs $(\mathbf{Z}_i,\ \mathbf{X}_i)$ being shared between source and receiver.

Then, from the isotropic and entanglement subgroup, EAQECC code C_{EA} are defined as $[[n,\ k;\ c]]$ that encodes $k = n - s - c$ qubits into n physical qubits with the help of $s = n - k - c$ ancillas qubits and c ebits shared between the sender and receiver. As the expectation to get the minimum distance, one entanglement pair is chosen, hence $c = 1$, the operators are chosen above, the minimum distance is determined by the minimum weights of operators in the error set N:

$$N = \left\{\mathbf{E}_m \;\middle|\; \forall \mathbf{E}_1,\ \mathbf{E}_2 \ \Rightarrow \mathbf{E}_2{}^{\dagger}\mathbf{E}_1 \ \in \ \boldsymbol{S}_{\mathbf{I}} \ \cup \ (\mathrm{P}_n \ - \ N(\boldsymbol{S}))\right\}.$$

Finally, the parameter of EAQECC $[[n,\ k,\ d_{\min};1]]$ are determined and the Corollary is proven. □

Following examples show the outputs of Corollary 1 where the minimum distance $d_{\min} \geq 3$, we search vectors which make codes with various minimum distance. Then, among many candidates of the vectors, to achieve largest minimum distance that has the error correctable ability the number of error ≥ 1 when the length of code up to 12.

Example 3. *For $n = 7$, let us consider the EAQECC when* $\mathbf{u} = [1\ 1\ 0\ 0\ 1\ 0\ 1]$ and $\mathbf{v} = [1\ 0\ 0\ 1\ 0\ 1\ 1]$. We have a code with the minimum number of ebits and good minimum distance, as in the following explanations.

Per **Theorem 2**, we have the corresponding parity-check matrix:

$$\mathbf{H} = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 \end{bmatrix} \tag{6.6}$$

The six generators are chosen from the first six rows of matrix **H**, which satisfy the independent condition to generate all elements of an inotropic subgroup. By using Gaussian elimination and interchanges of columns, matrix **H** in 6.6 takes the standard form:

$$\mathbf{H}_{\text{st}} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{6.7}$$

And the corresponding entanglement subgroup pair is calculated as:

$$\begin{aligned} \mathbf{X}_1 &= \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \\ \mathbf{Z}_1 &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}. \end{aligned} \tag{6.8}$$

Then, from vectors in 6.7 and 6.8 we have the generators for EAQECC as follows:

$\boldsymbol{S}_{\mathrm{E}} = \{\mathbf{X}_1,\ \mathbf{Z}_1\}$ and $\boldsymbol{S}_{\mathrm{I}} = \{\mathbf{Z}_2,\ \mathbf{Z}_3,\ \mathbf{Z}_4,\ \mathbf{Z}_5,\ \mathbf{Z}_6\}$ where

$$\mathbf{Z}_2 = [1\ 1\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 1]\,,$$

$$\mathbf{Z}_3 = [1\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 1\ 1\ 1]\,,$$

$$\mathbf{Z}_4 = [0\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 0]\,,$$

$$\mathbf{Z}_5 = [1\ 0\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 1\ 1\ 0\ 0]\,,$$

$$\mathbf{Z}_6 = [0\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 0]\,.$$

The generators $\boldsymbol{S} = <\boldsymbol{S}_{\mathrm{I}},\ \boldsymbol{S}_{\mathrm{E}}>$ correspond to EAQECC [[7,1,3;1]] that encodes **one** information qubit into **seven** physical qubits with the help of $s = 6$ ancilla qubits and only one pair entanglement -assisted ebit, and they can correct one error.

Example 4. *For n = 9, let us consider the EAQECC where* $\mathbf{u} = [0\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 1]$ and $\mathbf{v} = [1\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0]$. We have a code with the minimum number of ebits and good minimum distance, as in the following explanations.

From Theorem 2, we have the corresponding parity-check matrix:

$$\mathbf{H} = \begin{bmatrix} 0\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0 \\ 1\ 0\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 1 \\ 0\ 1\ 0\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 1 \\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 1\ 0 \\ 1\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 0 \\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 0\ 1 \\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 1\ 1\ 0\ 0\ 1\ 0 \\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ 1\ 0\ 0\ 1\ 0\ 0 \\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 1 \end{bmatrix} \tag{6.9}$$

The eight generators are chosen from the first eight rows of matrix **H,** which satisfies the independent condition to generate all elements of an inotropic subgroup. By using Gaussian elimination and interchange of columns, matrix **H** in (10) takes the standard form:

$$\mathbf{H}_{\mathrm{st}} = \begin{bmatrix} 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 1\ 1\ 1\ 0\ 1\ 1\ 0 \\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 1\ 1 \\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 1 \\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 1\ 0\ 1\ 0\ 0 \\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 1\ 1\ 1\ 1\ 1 \\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 0 \\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 0 \\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 0\ 1\ 1 \end{bmatrix} \tag{6.10}$$

And the corresponding entanglement subgroup pair is calculated as:

$$\begin{aligned} \mathbf{X}_1 &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 1\ 0]\,, \\ \mathbf{Z}_1 &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1]\,. \end{aligned} \tag{6.11}$$

Then, from vectors in 6.10 and 6.11, we have the generators for EAQECC as follows: $\boldsymbol{S}_{\mathrm{E}} = \{\mathbf{X}_1, \mathbf{Z}_1\}$ and $\boldsymbol{S}_{\mathrm{I}} = \{\mathbf{Z}_2, \mathbf{Z}_3, \mathbf{Z}_4, \mathbf{Z}_5, \mathbf{Z}_6, \mathbf{Z}_7, \mathbf{Z}_8\}$, where

$$\mathbf{Z}_2 = [0\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0]\,,$$

$$\mathbf{Z}_3 = [1\ 0\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 1]\,,$$

$$\mathbf{Z}_4 = [0\ 1\ 0\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 1]\,,$$

$$\mathbf{Z}_5 = [0\ 0\ 1\ 0\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 1\ 0]\,,$$

$$\mathbf{Z}_6 = [0\ 1\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 0\ 1]\,,$$

$$\mathbf{Z}_7 = [0\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 1\ 1\ 0\ 0\ 1\ 0]\,,$$

$$\mathbf{Z}_8 = [1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ 1\ 0\ 0\ 1\ 0\ 0]\,.$$

The generators $\boldsymbol{S} = < \boldsymbol{S}_{\mathrm{I}},\ \boldsymbol{S}_{\mathrm{E}} >$ correspond to EAQECC [[9,1,3;1]] that encodes **one** information qubit into **nine** physical qubits with the help of $s = 7$ ancilla qubits and one pair entanglement -assisted ebit, and they can also correct one error.

Example 5. *For n = 10, let us consider the EAQECC when* $\mathbf{u}$ = [0 1 1 1 0 1 1 0 1 0] and $\mathbf{v}$ = [1 1 1 0 1 1 0 1 0 0]. We have a code with the minimum number of ebits and good minimum distance as in the following explanations.

Per Theorem 2, we have the corresponding parity-check matrix:

$$\mathbf{H} = \begin{bmatrix}
0&1&1&1&0&1&1&0&1&0&1&1&1&0&1&1&0&1&0&0 \\
0&0&1&1&1&0&1&1&0&1&1&1&0&1&1&0&1&0&0&1 \\
1&0&0&1&1&1&0&1&1&0&1&0&1&1&0&1&0&0&1&1 \\
0&1&0&0&1&1&1&0&1&1&0&1&1&0&1&0&0&1&1&1 \\
1&0&1&0&0&1&1&1&0&1&1&1&0&1&0&0&1&1&1&0 \\
1&1&0&1&0&0&1&1&1&0&1&0&1&0&0&1&1&1&0&1 \\
0&1&1&0&1&0&0&1&1&1&0&1&0&0&1&1&1&0&1&1 \\
1&0&1&1&0&1&0&0&1&1&1&0&0&1&1&1&0&1&1&0 \\
1&1&0&1&1&0&1&0&0&1&0&0&1&1&1&0&1&1&0&1
\end{bmatrix} \tag{6.12}$$

The nine rows of matrix **H** satisfy the independent condition to generate all elements of an inotropic subgroup. By using Gaussian elimination and interchange of columns, matrix **H** in 6.12 takes the standard form:

$$\mathbf{H}_{\text{st}} = \begin{bmatrix}
1&0&0&0&0&0&0&0&0&1&1&1&1&1&0&1&0&0&1&0 \\
0&1&0&0&0&0&0&0&0&1&0&0&0&1&1&1&0&1&1&1 \\
0&0&1&0&0&0&0&0&0&1&1&1&0&0&1&1&1&1&0&0 \\
0&0&0&1&0&0&0&0&0&1&0&1&1&0&1&0&1&0&1&1 \\
0&0&0&0&1&0&0&0&0&1&0&0&1&0&0&0&0&1&0&0 \\
0&0&0&0&0&1&0&0&0&1&1&0&1&1&0&1&1&0&1&0 \\
0&0&0&0&0&0&1&0&0&1&1&0&0&1&1&0&0&1&1&1 \\
0&0&0&0&0&0&0&1&0&1&1&1&0&0&0&1&1&1&0&1 \\
0&0&0&0&0&0&0&0&1&1&0&1&1&1&1&0&1&0&0&1
\end{bmatrix} \tag{6.13}$$

And the corresponding entanglement subgroup pair is calculated as:

$$\begin{aligned} \mathbf{X}_1 &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 0\ 1\ 1\ 1\ 0]\,, \\ \mathbf{Z}_1 &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1]\,. \end{aligned} \tag{6.14}$$

a/ Then, from vectors in (14) and (15) we have the generators for EAQECC as follows: $\boldsymbol{S}_{\mathrm{E}} = \{\mathbf{X}_1, \mathbf{Z}_1\}$ and $\boldsymbol{S}_{\mathrm{I}} = \{\mathbf{Z}_2, \mathbf{Z}_3, \mathbf{Z}_4, \mathbf{Z}_5, \mathbf{Z}_6, \mathbf{Z}_7, \mathbf{Z}_8, \mathbf{Z}_9\}$ where

$$\begin{aligned} \mathbf{Z}_2 &= [0\ 1\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 0]\,, \\ \mathbf{Z}_3 &= [0\ 0\ 1\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 1]\,, \\ \mathbf{Z}_4 &= [1\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 1\ 1]\,, \\ \mathbf{Z}_5 &= [0\ 1\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 1\ 1\ 1]\,, \\ \mathbf{Z}_6 &= [1\ 0\ 1\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 1\ 1\ 0\ 1\ 0\ 0\ 1\ 1\ 1\ 0]\,, \\ \mathbf{Z}_7 &= [1\ 1\ 0\ 1\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 0\ 1\ 0\ 0\ 1\ 1\ 1\ 0\ 1]\,, \\ \mathbf{Z}_8 &= [0\ 1\ 1\ 0\ 1\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 1]\,, \\ \mathbf{Z}_9 &= [1\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 1\ 0]\,. \end{aligned}$$

The generators $\boldsymbol{S} = < \boldsymbol{S}_{\mathrm{I}}, \boldsymbol{S}_{\mathrm{E}} >$ correspond to EAQECC [[10,1,3;1]] that encodes **one** information qubit into **10** physical qubits with the help of $s = 8$ ancilla qubits and only one pair entanglement-assisted ebit, and they can correct one error.

b/ When we calculate with $\boldsymbol{S}_{\mathrm{E}} = \{\mathbf{X}_1, \mathbf{Z}_1\}$ where

$$\begin{aligned} \mathbf{X}_1 &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 0\ 1\ 1\ 1\ 0]\,, \\ \mathbf{Z}_1 &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1]\,. \end{aligned}$$

and $\boldsymbol{S}_{\mathrm{I}} = \{\mathbf{Z}_2, \mathbf{Z}_3, \mathbf{Z}_4, \mathbf{Z}_5, \mathbf{Z}_6, \mathbf{Z}_7, \mathbf{Z}_8\}$ where

$$\mathbf{Z}_2 = [0\ 1\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 0]\,,$$

$$\mathbf{Z}_3 = [0\ 0\ 1\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 1]\,,$$

$$\mathbf{Z}_4 = [0\ 1\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 1\ 1\ 1]\,,$$

$$\mathbf{Z}_5 = [1\ 0\ 1\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 1\ 1\ 0\ 1\ 0\ 0\ 1\ 1\ 1\ 0]\,,$$

$$\mathbf{Z}_6 = [0\ 1\ 1\ 0\ 1\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 1]\,,$$

$$\mathbf{Z}_7 = [1\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 1\ 0]\,,$$

$$\mathbf{Z}_8 = [1\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 1\ 0\ 1]\,.$$

The generator $\boldsymbol{S} = < \boldsymbol{S}_{\mathrm{I}}, \boldsymbol{S}_{\mathrm{E}} >$ correspond to EAQECC [[10,2,3;1]] that encodes **two** information qubits into **10** physical qubits with the help of $s = 7$ ancilla qubits and only one pair entanglement-assisted ebit, and they can correct one error.

Example 6. *For $n = 11$, let us consider the EAQECC when* $\mathbf{u} = [1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1]$, $\mathbf{v} = [1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 1]$. We have a code with the minimum number of ebits and good minimum distance as in the following explanations.

Per Theorem 2, we have the corresponding parity-check matrix:

$$\mathbf{H} = \begin{bmatrix} 1&1&1&0&0&1&1&0&0&0&1&1&1&0&0&1&1&0&0&0&1&1 \\ 1&1&1&1&0&0&1&1&0&0&0&1&0&0&1&1&0&0&0&1&1&1 \\ 0&1&1&1&1&0&0&1&1&0&0&0&0&1&1&0&0&0&1&1&1&1 \\ 0&0&1&1&1&1&0&0&1&1&0&0&1&1&0&0&0&1&1&1&1&0 \\ 0&0&0&1&1&1&1&0&0&1&1&1&1&0&0&0&1&1&1&1&0&0 \\ 1&0&0&0&1&1&1&1&0&0&1&1&0&0&0&1&1&1&1&0&0&1 \\ 1&1&0&0&0&1&1&1&1&0&0&0&0&0&1&1&1&1&0&0&1&1 \\ 0&1&1&0&0&0&1&1&1&1&0&0&0&1&1&1&1&0&0&1&1&0 \\ 0&0&1&1&0&0&0&1&1&1&1&0&1&1&1&1&0&0&1&1&0&0 \\ 1&0&0&1&1&0&0&0&1&1&1&1&1&1&1&0&0&1&1&0&0&0 \end{bmatrix} \tag{6.15}$$

The ten rows of matrix **H** satisfy the independent condition to generate all elements of an inotropic subgroup. By using Gaussian elimination and interchange of columns, matrix **H** in 6.15 takes the standard form:

$$\mathbf{H}_{\mathrm{st}} = \begin{bmatrix} 1&0&0&0&0&0&0&0&0&0&1&0&1&1&0&0&1&0&0&1&1&1 \\ 0&1&0&0&0&0&0&0&0&0&1&1&0&1&0&1&1&0&1&0&0&1 \\ 0&0&1&0&0&0&0&0&0&0&1&0&0&1&1&1&1&1&0&1&0&0 \\ 0&0&0&1&0&0&0&0&0&0&1&0&0&0&1&1&0&0&1&1&1&1 \\ 0&0&0&0&1&0&0&0&0&0&1&0&1&0&1&0&1&1&1&0&0&1 \\ 0&0&0&0&0&1&0&0&0&0&1&1&1&0&0&1&0&1&0&1&0&1 \\ 0&0&0&0&0&0&1&0&0&0&1&1&1&1&1&0&0&0&1&1&0&0 \\ 0&0&0&0&0&0&0&1&0&0&1&1&0&0&0&0&1&1&1&1&1&0 \\ 0&0&0&0&0&0&0&0&1&0&1&0&1&1&0&1&0&1&1&0&1&0 \\ 0&0&0&0&0&0&0&0&0&1&1&1&0&1&1&0&0&1&0&0&1&1 \end{bmatrix} \tag{6.16}$$

And the corresponding entanglement subgroup pair is calculated as:

$$\begin{aligned} \mathbf{X}_1 &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 1\ 1\ 0\ 0\ 0\ 1\ 0]\,, \\ \mathbf{Z}_1 &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1]\,. \end{aligned} \tag{6.17}$$

a/ Then, from vectors in 6.13 and 6.14, we have the generators for EAQECC as follows: $\boldsymbol{S}_{\mathrm{E}} = \{\mathbf{X}_1, \mathbf{Z}_1\}$ and $\boldsymbol{S}_{\mathrm{I}} = \{\mathbf{Z}_2, \mathbf{Z}_3, \mathbf{Z}_4, \mathbf{Z}_5, \mathbf{Z}_6, \mathbf{Z}_7, \mathbf{Z}_8, \mathbf{Z}_9, \mathbf{Z}_{10}\}$ where

$$\mathbf{Z}_2 = [1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 1]\,,$$

$$\mathbf{Z}_3 = [1\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1]\,,$$

$$\mathbf{Z}_4 = [0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0]\,,$$

$$\mathbf{Z}_5 = [0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0]\,,$$

$$\mathbf{Z}_6 = [1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1]\,,$$

$$\mathbf{Z}_7 = [1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 1]\,,$$

$$\mathbf{Z}_8 = [0\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 0]\,,$$

$$\mathbf{Z}_9 = [0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0]\,,$$

$$\mathbf{Z}_{10} = [1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0]\,.$$

The generators $\boldsymbol{S} = < \boldsymbol{S}_{\mathrm{I}},\ \boldsymbol{S}_{\mathrm{E}} >$ correspond to EAQECC [[11,1,4;1]] that encodes **one** information qubit into **11** physical qubits with the help of $s = 9$ ancilla qubits and only one entanglement-assisted ebit, and the minimum distance is four.

b/ When we calculate with $\boldsymbol{S}_{\mathrm{E}} = \{\mathbf{X}_1,\ \mathbf{Z}_1\}$ where

$$\mathbf{X}_1 = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 1\ 1\ 0\ 0\ 0\ 1\ 0]\,,$$

$$\mathbf{Z}_1 = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1]\,.$$

and $\boldsymbol{S}_{\mathrm{I}} = \{\mathbf{Z}_2,\ \mathbf{Z}_3,\ \mathbf{Z}_4,\ \mathbf{Z}_5,\ \mathbf{Z}_6,\ \mathbf{Z}_7,\ \mathbf{Z}_8,\ \mathbf{Z}_9\}$ where

$$\mathbf{Z}_2 = [1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 1]\,,$$

$$\mathbf{Z}_3 = [0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0]\,,$$

$$\mathbf{Z}_4 = [0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0]\,,$$

$$\mathbf{Z}_5 = [1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1]\,,$$

$$\mathbf{Z}_6 = [1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 1]\,,$$

$$\mathbf{Z}_7 = [0\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 0]\,,$$

$$\mathbf{Z}_8 = [0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0]\,,$$

$$\mathbf{Z}_9 = [1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0]\,.$$

The generator $\boldsymbol{S} = <\ \boldsymbol{S}_{\mathrm{I}},\ \boldsymbol{S}_{\mathrm{E}}\ >$ correspond to EAQECC [[11,2,3;1]] that encodes **two** information qubits into **11** physical qubits with the help of $s = 8$ ancilla qubits and only one pair entanglement-assisted ebit, and they can correct one error.

c/ When we calculate with $\boldsymbol{S}_{\mathrm{E}} = \{\mathbf{X}_1,\ \mathbf{Z}_1\}$ where

$$\mathbf{X}_1 = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 1\ 1\ 0\ 0\ 0\ 1\ 0]\,,$$

$$\mathbf{Z}_1 = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1]\,.$$

and $\boldsymbol{S}_{\mathrm{I}} = \{\mathbf{Z}_2,\ \mathbf{Z}_3,\ \mathbf{Z}_4,\ \mathbf{Z}_5,\ \mathbf{Z}_6,\ \mathbf{Z}_7,\ \mathbf{Z}_8\}$ where

$$\mathbf{Z}_2 = [1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 1]\,,$$

$$\mathbf{Z}_3 = [0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0]\,,$$

$$\mathbf{Z}_4 = [1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1]\,,$$

$$\mathbf{Z}_5 = [1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 1]\,,$$

$$\mathbf{Z}_6 = [0\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 0]\,,$$

$$\mathbf{Z}_7 = [0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0]\,,$$

$$\mathbf{Z}_8 = [1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0]\,.$$

The generator $\boldsymbol{S} = <\boldsymbol{S}_{\mathrm{I}},\ \boldsymbol{S}_{\mathrm{E}}\ >$ correspond to EAQECC [[11,3,3;1]] that encodes **three** information qubits into **11** physical qubits with the help of $s = 7$ ancilla qubits and only one pair entanglement-assisted ebit, and they can correct one error.

d/ When we calculate with $\boldsymbol{S}_{\mathrm{E}} = \{\mathbf{X}_1, \mathbf{Z}_1\}$ where

$$\mathbf{X}_1 = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 1\ 1\ 0\ 0\ 0\ 1\ 0]\,,$$

$$\mathbf{Z}_1 = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1]\,.$$

and $\boldsymbol{S}_{\mathrm{I}} = \{\ \mathbf{Z}_2, \mathbf{Z}_3, \mathbf{Z}_4, \mathbf{Z}_5, \mathbf{Z}_6, \mathbf{Z}_7\}$ where

$$\mathbf{Z}_2 = [1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 1]\,,$$

$$\mathbf{Z}_3 = [0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0]\,,$$

$$\mathbf{Z}_4 = [1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1]\,,$$

$$\mathbf{Z}_5 = [1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 1]\,,$$

$$\mathbf{Z}_6 = [0\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 0]\,,$$

$$\mathbf{Z}_7 = [0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0]\,.$$

The generator $\boldsymbol{S} = <\boldsymbol{S}_{\mathrm{I}}, \boldsymbol{S}_{\mathrm{E}}>$ correspond to EAQECC [[11,4,3;1]] that encodes **four** information qubits into **11** physical qubits with the help of $s = 6$ ancilla qubits and only one pair entanglement-assisted ebit, and they can correct one error.

Example 7. *For $n = 12$, let us consider the EAQECC where* $\mathbf{u} = [1\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1]$, $\mathbf{v} = [1\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 1]$. We have a code with the minimum number of ebits and good minimum distance, as in the following explanations.

Per Theorem 2, we have the corresponding parity-check matrix:

$$\mathbf{H} = \begin{bmatrix}
1&1&1&0&0&1&0&1&0&1&0&1&1&1&0&0&1&0&1&0&1&0&1&1 \\
1&1&1&1&0&0&1&0&1&0&1&0&1&0&0&1&0&1&0&1&0&1&1&1 \\
0&1&1&1&1&0&0&1&0&1&0&1&0&0&1&0&1&0&1&0&1&1&1&1 \\
1&0&1&1&1&1&0&0&1&0&1&0&0&1&0&1&0&1&0&1&1&1&1&0 \\
0&1&0&1&1&1&1&0&0&1&0&1&1&0&1&0&1&0&1&1&1&1&0&0 \\
1&0&1&0&1&1&1&1&0&0&1&0&0&1&0&1&0&1&1&1&1&0&0&1 \\
0&1&0&1&0&1&1&1&1&0&0&1&1&0&1&0&1&1&1&1&0&0&1&0 \\
1&0&1&0&1&0&1&1&1&1&0&0&0&1&0&1&1&1&1&0&0&1&0&1 \\
0&1&0&1&0&1&0&1&1&1&1&0&1&0&1&1&1&1&0&0&1&0&1&0 \\
0&0&1&0&1&0&1&0&1&1&1&1&0&1&1&1&1&0&0&1&0&1&0&1 \\
1&0&0&1&0&1&0&1&0&1&1&1&1&1&1&1&0&0&1&0&1&0&1&0
\end{bmatrix} \tag{6.18}$$

The eleven rows of matrix **H** satisfy the independent condition to generate all elements of an inotropic subgroup. By using Gaussian elimination and interchange of columns, matrix **H** in (19) takes the standard form:

$$\mathbf{H}_{\mathrm{st}} = \left[\begin{array}{cccccccccccccccccccccccc}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1
\end{array}\right] \tag{6.19}$$

And the corresponding entanglement subgroup pair is calculated as:

$$\begin{aligned}
\mathbf{X}_1 &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 1\ 1\ 0]\,, \\
\mathbf{Z}_1 &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1]\,.
\end{aligned} \tag{6.20}$$

a/ Then, from vectors in 6.13 and 6.14 we have the generators for EAQECC as follows: $\boldsymbol{S}_{\mathrm{E}} = \{\mathbf{X}_1, \mathbf{Z}_1\}$ and $\boldsymbol{S}_{\mathrm{I}} = \{\mathbf{Z}_2, \mathbf{Z}_3, \mathbf{Z}_4, \mathbf{Z}_5, \mathbf{Z}_6, \mathbf{Z}_7, \mathbf{Z}_8, \mathbf{Z}_9, \mathbf{Z}_{10}, \mathbf{Z}_{11}\}$ where

$$\begin{aligned}
\mathbf{Z}_2 &= [1\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 1]\,, \\
\mathbf{Z}_3 &= [1\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 1]\,, \\
\mathbf{Z}_4 &= [0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 1]\,, \\
\mathbf{Z}_5 &= [1\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 0]\,, \\
\mathbf{Z}_6 &= [0\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 0]\,, \\
\mathbf{Z}_7 &= [1\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1]\,, \\
\mathbf{Z}_8 &= [0\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 0]\,, \\
\mathbf{Z}_9 &= [1\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 1]\,, \\
\mathbf{Z}_{10} &= [0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 0]\,, \\
\mathbf{Z}_{11} &= [0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1]\,.
\end{aligned}$$

The generators $\boldsymbol{S} = <\boldsymbol{S}_{\mathrm{I}},\, \boldsymbol{S}_{\mathrm{E}} >$ correspond to EAQECC [[12,1,4;1]] that encodes **one** information qubit into **12** physical qubits with the help of $s = 10$ ancilla qubits and only one pair entanglement -assisted ebit and the minimum distance is four.

b/ When we calculate with $\boldsymbol{S}_{\mathrm{E}} = \{\mathbf{X}_1,\, \mathbf{Z}_1\}$ where

$$\begin{aligned}
\mathbf{X}_1 &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 1\ 1\ 0]\,, \\
\mathbf{Z}_1 &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1]\,.
\end{aligned}$$

and $\boldsymbol{S}_{\mathrm{I}} = \{\mathbf{Z}_2,\, \mathbf{Z}_3,\, \mathbf{Z}_4,\, \mathbf{Z}_5,\, \mathbf{Z}_6,\, \mathbf{Z}_7,\, \mathbf{Z}_8,\, \mathbf{Z}_9,\, \mathbf{Z}_{10}\}$ where

$$\mathbf{Z}_2 = [1\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 1],$$

$$\mathbf{Z}_3 = [0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 1],$$

$$\mathbf{Z}_4 = [1\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 0],$$

$$\mathbf{Z}_5 = [0\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 0],$$

$$\mathbf{Z}_6 = [1\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1],$$

$$\mathbf{Z}_7 = [0\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 0],$$

$$\mathbf{Z}_8 = [1\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 1],$$

$$\mathbf{Z}_9 = [0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 0],$$

$$\mathbf{Z}_{10} = [0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1].$$

The generator $\boldsymbol{S} = <\boldsymbol{S}_{\mathrm{I}},\ \boldsymbol{S}_{\mathrm{E}}>$ correspond to EAQECC [[12,2,4;1]] that encodes **two** information qubits into **12** physical qubits with the help of $s = 9$ ancilla qubits and only one pair entanglement-assisted ebit, and they can correct one error.

The results of the proposed EAQECC with lengths up to 12 are listed in Table 6.1. The detailed values of the operators are calculated in Examples 3–7.

In comparison with the results of referenced studies [92–95], the proposed EAQECC shows outperform as a smaller number of ebits and larger minimum distance. As the construction of EAQECC by generalized quadrangles in [93] and circulant matrix in [94], the two main things are clearly to conclude the advances of proposed methods that are code lengths and the classification of generators. Firstly, in [93, 94], the code lengths are limited as the conditions to construct the parity check matrix, in contrast the proposed method can find the corresponding EAQECCs for any length. In addition, proposed codes are expressed as the standard form transformation to classify subgroups $\boldsymbol{S}_{\mathbf{I}}$ and $\boldsymbol{S}_{\mathbf{E}}$; hence, the operators of subgroups are clearly deter-

Table 6.1: Entanglement-assisted quantum error correction code (EAQECC) $[[n,\ k,\ d;\ c]]$ from proposed method with $c = 1$ and $d \geq 3$.

$\boldsymbol{n}$	$\boldsymbol{k}$	$\boldsymbol{d}$
7	1	3
8	1	3
9	1	3
10	1	3
10	2	3
11	1	4
11	2	3
11	3	3
11	4	3
12	1	4
12	2	4
12	3	3
12	4	3
12	5	3

mined, instead of knowing the numbers and operators not being determined, as seen in elsewhere, hence the minimum distance of outputs are not calculated in [93, 94], furthermore the determined generators also open the effective way to implement the quantum system in future works. For more details, the comparisons are listed in Table 6.2.

Table 6.2: Comparison of this proposed paper to other research.

Construction	**Ebits**	**To classify subgroups**	**Minimum distance**
Arbitrary binary linear code [92]	$2n - 2k$	Gram-Schmidt procedure	≥3
Generalized quadrangles [93]	2	Gram-Schmidt procedure	Not mentioned
Circulant permutation [94]	1	Gram-Schmidt procedure	Not mentioned
Shortened Hamming code [95]	1	Gram-Schmidt procedure	3
The proposed method	1	Standard form transformation of matrix	3, ≥4

Chapter 7

Summary of Contributions and Further Works

7.1 Thesis conclusion

Quantum computer have proven to have many advantages of quantum mechanical phenomena such as quantum superposition of quantum state and quantum entanglements between qubits, to solve many certain problems efficiently, faster, and confidential than the classical counterparts. Since the affect of noise and unwanted environments, the ability to mitigate the noise resulting from decoherence will determine whether building of quantum computer is feasible. Quantum error correction codes are essential to achieve fault-tolerant quantum computation.

The correspondence between quantum error correction codes and classical error correction codes is a topic has been researched during past three decades. There are some similarities between QECCs and classical codes as well as there are some substantial differences. In this thesis, we consider some types of classical codes which are suitable for construction of quantum error correction codes. The main contributions of this dissertation can be listed as follows.

- First, the conditions of a DS are examined to satisfy the SIP condition and a new construction method of quantum stabilizer codes from the DS is proposed. The condition of a DS to satisfy the SIP constraint is equivalent to determine a DS with $k \equiv \lambda$ modulo 2. Quantum stabilizer codes [[7,4,2]] and [[15,10,2]] are presented from the proposed construction with DS (7, 4, 2) and DS (15, 7, 3), respectively, for practical applications. Moreover, since there are many DSs with parameters that satisfy $k \equiv \lambda$ modulo 2, it is possible to produce new quantum stabilizer codes with greater length. In comparison with the referenced construction, the proposed construction provides more candidates for the quantum stabilizer code based on DSs.

- Second, we studied the quantum stabilizer code constructions based on the symmetric matrices for binary and non-binary cases. The quantum stabilizer codes based on the two proposed constructions whose parameters achieved equality of the quantum singleton bound are explained in detail. These optimal quantum stabilizer codes are candidates for use in quantum applications such as quantum cryptography, quantum communication, and quantum entanglement based on the graph state.

- Third, a new approach to constructing additive codes over *GF(4)*, which are self-orthogonal with respect to Hermitian product, is proposed for even lengths and odd lengths, and the minimum distance is proven to be four. Moreover, the transformation to quantum stabilizer code is also considered. Six optimal quantum stabilizer codes $[[5,1,3]]$, $[[7,1,3]]$, $[[9,1,3]]$, $[[6,0,4]]$, $[[8,0,4]]$, and $[[10,0,4]]$ have been interpreted from corresponding linear codes with the standard form to show their practicality in quantum stabilizer codes. This code construction method can be applied for the code, can correct at least one error in quantum information theory, and can have a good quantum state.

- Fourth, we propose quantum stabilizer codes based on a new construction method by using self-orthogonal linear codes over *GF(4)*, which satisfy the trace-inner product. The proposed quantum stabilizer codes provide various dimensions for any length and demonstrate improved error correction in comparison with referenced quantum codes. The comparison results between our proposed codes and referenced codes show that the proposed codes can support various dimensions and have better correction capabilities.

- Fifth, the construction of EAQECC-based on circulant matrices has been studied. Not using the Gram-Schmidt procedure to classify the subgroups of EAQECC, we first propose the construction and calculation for each subgroup. This work aims to reduce the complexity in the classification and determination of ebits. Some EAQECCs with a minimum number of ebits, and the capability to correct errors were showed clearly, with generators of each subgroup. This promises effective codes in comparison with other results.

7.2 Future research directions

Although many aspects of quantum error correction codes were studied in the literature, there remain interesting research topics for the quantum information processing. In this dissertation, we only studied the construction of one famous type of quantum error correction codes called quantum stabilizer codes. From the design of QECCs from stabilizer formalism, we can get the quantum stabilizer group. And not only we gain the error correction from this group from syndrome calculation but also we can get full entanglement state, or stabilizer frames. These help us on designing of quantum computation circuit model, quantum key distribution protocol, and the design of stabilizer frames which can boost quantum circuits. It promises the further research on applications of construction of quantum stabilizer codes.

In addition, some quantum algorithms such that quantum walks, Deutsch-Jozsa algorithm, and quantum three-stage protocol have been discussed to show the advantages of quantum computation in order of comparison with classical computation. Most quantum algorithms based on the quantum computation which use the unitary transform over Hilbert space to utilize ancilla qubits as input qubits, by Hadarmard transform or Fourier transform, we can get the superposition states from arbitrary state or role back from superposition to arbitrary qubits. The oracle or rotation gates are applied to the superposition state. The advantages of quantum algorithms lie on quantum entanglement and the no-cloning theorem. And they required small times to query from oracle in comparison to classical computation. Then, we give some improvements quantum algorithms to achieve better quantum algorithms on security, better effective protocol, and re-consider the quantum walk on the 1-D lines.

Currently, there are some open source to implement quantum computation and quantum algorithm such as IBMQX. We plan to research on this setups to test, implement out standing algorithms, and our proposed algorithms.

Publications

SCI(E) Journals

[1] **D.M. Nguyen**, S. Kim. *Minimal-Entanglement Entanglement-Assisted Quantum Error Correction Codes from Modified Circulant Matrices*. Symmetry. 9(7) pp(122) (7/2017).

[2] **D.M. Nguyen**, S. Kim. *Quantum stabilizer codes construction from Hermitian self-orthogonal codes over GF(4)*. Journal of Communications and Networks. 20(3) pp(309-315) (6/2018).

[3] **D.M. Nguyen**, S. Kim. *New Constructions of Quantum Stabilizer Codes Based on Difference Sets*. Symmetry. 10(11) pp(655) (11/2018).

[4] **D.M. Nguyen**, S. Kim. *Quantum Key Distribution Protocol Based on Modified Generalization of Deutsch-Jozsa Algorithm in d-level Quantum System*. International Journal of Theoretical Physics, 58(1) (01/2019).

[5] **D.M. Nguyen**, S. Kim. *Multi-Bits Transfer Based on the Quantum Three-Stage Protocol with Quantum Error Correction Codes*. International Journal of Theoretical Physics, 58(6) (04/2019).

[6] **D.M. Nguyen**, S. Kim. *The fog on: Generalized teleportation by means of*

discrete-time quantum walks on N -lines and N -cycles. Modern Physics Letters B. 33(23), pp(1950270), (08/2019).

[7] **D.M. Nguyen**, S. Kim. *New construction of binary and non-binary quantum stabilizer codes based on symmetric matrices.* International Journal of Modern Physics B. 33(24) pp(1950274), (10/2019).

[8] **D.M. Nguyen**, S. Kim. *Quantum Stabilizer Codes Based on a New Construction of Self-orthogonal Trace-inner Product Codes over $GF(4)$.* International Journal of Modern Physics B, (Revision/2019).

International Conferences

[9] **D.M. Nguyen**, S. Kim. *Construction and complement circuit of a quantum stabilizer code with length 7.* in Proceedings of Eighth International Conference on Ubiquitous and Future Networks (ICUFN), Wien, Austria. pp(332-336) (7/2016).

[10] **D.M. Nguyen**, S. Kim. *Application of additive codes over GF(4) on quantum error correction codes.* in Proceedings of the 7th International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA 2018, Da-Nang, Viet-Nam). (10/2019).

[11] **D.M. Nguyen**, S. Kim. *A quantum three pass protocol with phase estimation for many bits transfer.* in Proceedings of International Conference on Advanced Technologies for Communications (ATC 2019, Ha-Noi, Viet-Nam). (10/2019).

References

[1] R. Feynman, *Int. J. Theor. Phys.* **21**, 467 (1982).

[2] P.W. Shor, *Proceedings 35th Annual Symposium on Foundations of Computer Science. IEEE Comput. Soc. Press. 1994*

[3] L.K. Grover, *Phys. Rev. Lett.* **79**, 325 (1997).

[4] D.M. Nguyen, S. Kim, *Int. J. Theor. Phys.* **58(6)**, 2043-2053 (2019).

[5] D.M. Nguyen, S. Kim, *Int. J. Theor. Phys.* **58(1)**, 71-82 (2019).

[6] P. W. Shor, *Phys. Rev. A.* **52(4)**, 2493, (1995).

[7] A.M.Steane, *Phys. Rev. A.* **54(6)**, 4741-4751, (1996).

[8] D. Gottesman, “Stabilizer codes and quantum error correction”, Caltech Ph.D. thesis, 1997, arxiv quant-ph/9705052.

[9] A. R.Calderbank, P. W. Shor, *Phys. Rev. A.* **54**, 1098, (1996).

[10] M. Grassl, T. Beth, *International Symposium on Theoretical Electrical Engineering, Magdeburg, 1999* pp. 207–212.

[11] M. Grassl, W. Geiselmann, T. Beth, *in Proceedings Applied Algebra, Algebraic Algorithms and Error-Correcting Codes (AAECC-13), Springer, 1999.*

[12] M. Hagiwara, H. Imai, *IEEE International Symposium on Information Theory, 2017.*

[13] M. Hivadi, *Quantum Information Processing* **17**, 324, (2018).

[14] D. Deutsch, R. Jozsa, *Proceedings of the Royal Society of London A*, 1992.

[15] K. Nagata. et. Al., *Asian J. Math. Phys.*, **2**, 6–13, (2018).

[16] M. Zidan, et. al. *Appl. Math. Inf. Sci.* 2018, 12(1), 265.

[17] H.J.Garcia, I.L.Markov, *IEEE Transaction on Computers* 2015, 64(8).

[18] M.Steudtner, S.Wehner, *Phys. Rev. A* 2019, 99, 022308.

[19] M. Epping, H. Kampermann, and D. Bruf. *New journal of Phys.* 2016, 18, 103052

[20] Nguyen, D.M.; Kim, S. *Int. J. Theor. Phys.* **2018**.

[21] Calderbank, A.R.; Shor, P.W. *Phys. Rev. A* **1996**, *54*, 1098–1105.

[22] Gallager, R.G. *IRE Trans. Inf. Theory* **1962**, *8*, 21–28.

[23] Chung, S.Y.; Forney, G.D.; Richardson, T.J.; Urbanke, R. *IEEE Comm. Lett.* **2001**, *45*, 58–60.

[24] Kim, S.; No, J.S.; Chung, H.; Shin, D.J. *IEEE Trans. Inf. Theory* **2007**, *53*, 2885–2891.

[25] Kim, S. *IEEE Photonics Technol. Lett.* **2015**, *27*, 967–969.

[26] Postol, M.S. *arXiv*, 2001; arXiv:quant-ph/010813.

[27] Hagiwara, M.; Imai, H. In Proceedings of the IEEE International Symposium on Information Theory (ISIT), Nice, France, 24–29 June 2007.

[28] Hwang, Y.; Chung, Y.; Jeon, M. *Quantum Inf. Process.* **2013**, *12*, 2219–2239.

[29] Tan, P.; Li, J. *IEEE Trans. Inf. Theory* **2010**, *56*, 476–491.

[30] Baumert, L.D. *Cyclic Difference Sets*; Springer: New York, NY, USA, 1971.

[31] Anderson, I. *Combinatorial Designs: Construction Methods*; Ellis Horwood Limited: New York, NY, USA, 1990.

[32] Beth, T.; Jungnickel, D.; Lenz, H. *Design Theory*; Cambridge University Press: New York, NY, USA, 1986.

[33] MacKay, D.; Mitchison, G.; McFadden, P. *IEEE Trans. Inf. Theory* **2004**, *50*, 2315–2330.

[34] Liu, Y.; Wang, Y.; Zhao, S.; Zheng, B. In Proceedings of the IEEE 11th International Conference on Signal Processing (ICSP), Beijing, China, 21–25 October 2012.

[35] Xie, Y.; Yuan, J.; Malaney, R. In Proceedings of the IEEE International Symposium on Information Theory (ISIT), Istanbul, Turkey, 7–12 July 2013.

[36] Vos, A.D.; Baerdemacker, S.D. *Symmetry* **2011**, *3*, 305–324.

[37] Nguyen, D.M.; Kim, S. *J. Commun. Netw.* **2018**, *20*, 209–315.

[38] Xie, Y.; Yuan, J.; Sun, T.Q. *IEEE Trans. Inf. Theory* **2014**, *66*, 3721–3735.

[39] Grassl, M. Available online: http://codetables.de/ (accessed on 7 November 2018).

[40] X. Gao, et. al. *Physical review A* **99**, 023825, (2019).

[41] A. Ketkar, et. al. *IEEE Transasctions on information theory* **52(11)**, 4892-4914, (2006).

[42] G. Xu, et. al. *International journal of modern physicals B* **31(6)**, (2017).

[43] D.M. Nguyen, S. Kim, *Journal of Communications and Networks* 2018, 20(3), 309-315.

[44] F. Li, Q. Yue, *Modern Physics Letters B* 2015, 29(1).

[45] D.M. Nguyen, S. Kim, *Symmetry* 2017, 10(11), 655.

[46] D.M. Nguyen, S. Kim, *Symmetry* 2017, 9(7), 122.

[47] D.M. Nguyen, S. Kim, *in Proceedings Eighth International Conference on Ubiquitous and Future Networks (ICUFN), Austria, 2016.*

[48] J. Gao, Y. Wang, *IEEE Access* **7**, 26418-26421, (2019).

[49] M. Hein, et.al. *in Proceedings of the International School of Physics Enrico Fermi on Quantum Computers, Algorithms and Chaos, Italy, 2005.*

[50] I. Djordjevic, *Quantum Information Processing and Quantum Error Correction: An Engineering Approach* 2012.

[51] E. Knill, R. Laflamme, *Phys. Rev. A* **55**, 900, (1997).

[52] MAGMA. "Computer and Algebra calculation tool", online at http://magma.maths.usyd.edu.au/calc/

[53] S. Y. Looi, et. Al. *Physical Review A* , 2008 **78**, 042303.

[54] A. Keet, et. Al. *Physical Review A* **82**, 062315, (2010).

[55] T-A. Isdraila, *Science Report* **9**, 6337, (2019).

[56] D. Gottesman, *PhD thesis.* California Institute of Technology, 1997.

[57] R. Penrose, *Quantum error correction and fault tolerant quantum computing.* CRC Press. Inc. BocaRaton. FL. USA ,2007.

[58] A. R. Calderbank and P. W. Shor, *IEEE Trans. Info. Theory* vol. 53, pp. 1183–1188, 2007.

[59] Dong. C, Yaoliang. S, *Journal of Comm. Netw.* vol. 15, no. 2, Aprial 2013.

[60] M. Grassl, W. Geiselmann, T. Beth, *App. Alge. Algorm. and Error-Correcting Codes* vol. 1719, pp. 231-244, 1996.

[61] D. M. Nguyen, S. Kim, *Symmetry*, vol. 9(7), pp. 122, 2017.

[62] S. M. Zhao, Y. Xiao, Y. Zhu, X. L. Zhu, M. H. Hsieh, *Int. J. Quantum Inform.* vol. 10, pp. 1250015, 2012.

[63] Y. Xie, J. Yuan, Q. Sun, https://arxiv.org/pdf/1407.8249v1.pdf.

[64] A. Naghipour, M. A. Jafarizadeh, S. Shahmorad, *Int. J. Quantum Inform.* vol. 13(03), pp. 1550021, 2015.

[65] A. R. Calderbank, E. M. Rains, P. W. Shor, N. J. A. Sloane, *IEEE Trans. Inform. Theory* vol. 44, pp. 1369-1387, 1998.

[66] J. L. Kim *et al.*, *IEEE Trans. Inform. Theory* vol. 47, pp. 1575–1580, 2001.

[67] A. Thangaraj, S. W. McLaughlin, *IEEE Trans. Inform. Theory* vol. 47, pp. 1176-1178, 2001.

[68] M. Grassl, *Bounds on the minimum distance of linear codes and quantum codes.* http://www.codetables.de.

[69] L. E. Danielsen, *Master thesis.* University of Bergen, 2005.

[70] R. Penrose, *Quantum computing for computer scientists.* Cambridge University Press New York, NY, USA, 2008.

[71] D.M.Nguyen, S.Kim, *Symmetry* **10**, 11, 655, (2018).

[72] A.Naghipour et. al. *Int. J. Quantum Inform.* **13** 03, 1550021, (2015).

[73] D.M.Nguyen, S.Kim, *Symmetry* **9**, 7, 122, (2017).

[74] D.M.Nguyen, S.Kim, *Proceedings of Eighth International Conference on Ubiquitous and Future Networks (ICUFN)*, (2016).

[75] G. Xu, et. al. *International journal of modern physicals B* **31(6)**, (2017).

[76] A. R. Calderbank, et. al. *IEEE Trans. Inf. Theory* **44**, 4, 13691387, (1998).

[77] J. Gao, Y.Wang, *IEEE Access* **7**, 26418 - 26421, (2019).

[78] J.L. Kim, *IEEE Trans. Inf. Theory* **47**, 4, 15751580, (2001).

[79] D.M. Nguyen, S.Kim, *J. of Comm. and Net.* **20**, 3, 309-315, (2018).

[80] I. Djordjevic, Quantum Information Processing and Quantum Error Correction: An Engineering Approach, Science Direct, 2012.

[81] M.Grassl, Online available at http://www.codetables.de. accessed on 2019-07-12.

[82] MAGMA. Online at http://magma.maths.usyd.edu.au/calc/

[83] Y. G. Zeng, M. H. Lee, *Advances in Mathematical Physics* **2010**, 469124, (2010).

[84] S. M. Zhao, et al. *Int. J. Quantum Inform.* **10**, 1250015, (2012).

[85] Y. Xie, J. Yuan, Q. Sun, *IEEE Trans. on Comm.* **66** 9, (2018).

[86] Y.Guo, M. H. Lee, *Quantum Inf. Proces.* **8**, 361-378, (2009).

[87] Gottesman, D. Stabilizer Codes and Quantum Error Correction. Ph.D. Thesis, California Institute of Technology, Pasadena, CA, USA, 1997.

[88] Gottesman, D. *Phys. Rev. A.* **1998**, *57*, 127.

[89] Calderbank, A.R.; Shor, P.W. *Phys. Rev. A.* **1996**, *54*, 1098–1105.

[90] Brun, T.A.; Devetak, I.; Hsieh, M.H. *Science* **2006**, *314*, 436.

[91] Hsieh, M.H.; Yen, W.T.; Hsu, L.Y. *IEEE Trans. Inf. Theory* **2011**, *57*, 1761–1769.

[92] Quian, J.; Zhang, L. *Des. Codes Cryptogr.* **2015**, *77*, 193–202.

[93] Thomas, W. *Rose-Hulman Undergrad. Math. J.* **2013**, *14*, 2.

[94] Wada, M.; Kodaira, K.; Shibuya, T. *Int. Symp. Inf. Theory Appl.* **2014**, *26–29*, 158–162.

[95] Qian, J.; Zhang, L. In Proceedings of the IEEE International Conference on Signal and Image Processing (ICSIP), Beijing, China, 3–15 August 2016; pp. 347–351.

[96] Djordjevic, I. *Quantum Information Processing and Quantum Error Correction: An Engineering Approach*; Academic Press: Oxford, UK, 2012.

[97] Yun-Jiang, W. *Chin. Phys. B* **2012**, *21*, 2.

[98] Shaw, B.; Wilde, M.M.; Oreshkov, O.; Kremsky, I.; Lidar, D.A. *Phys. Rev. A* **2008**, *78*, 012337.