International Journal of Modern Physics B
Vol. 34, No. 5 (2020) 2050017 (15 pages)

DOI: 10.1142/S0217979220500174

Quantum stabilizer codes based on a new construction of self-orthogonal trace-inner product codes over GF(4)

Duc Manh Nguyen* and Sunghwan Kim†

School of Electrical Engineering, University of Ulsan, Ulsan 44610, Korea
**nguyenmanhduc18@gmail.com*
†*sungkim@ulsan.ac.kr*

Received 7 September 2019
Revised 24 November 2019
Accepted 27 November 2019
Published 14 February 2020

In this paper, we propose quantum stabilizer codes based on a new construction of self-orthogonal trace-inner product codes over the Galois field with 4 elements (GF(4)). First, from any two binary vectors, we construct a generator matrix of linear codes whose components are over GF(4). We prove that the proposed linear codes comply with the self-orthogonal, trace-inner product. Then, we propose mapping tables to construct new quantum stabilizer codes by using linear codes. Comparison results show that our proposed quantum codes have various dimensions for any code length with the capacity for better errors correction relative to the referenced quantum codes.

Keywords: Galois field 4 (GF(4)); quantum stabilizer codes; self-orthogonal codes; trace-inner product.

PACS numbers: 03.67.−a, 03.67.Pp

1. Introduction

Quantum mechanics describe nature at the smallest scales, including physical phenomena such as superposition states, quantum entanglement, quantum measurements and the no-cloning theorem.[1] Quantum computation is one of the most important applications of quantum mechanics and given us effective solutions for difficult problems, such as factoring large integer numbers in polynomial time,[2] searching in unordered sets,[3] and increasing the security of quantum cryptography[4]; these tasks are difficult to do using classical computation. However, the effects of noisy and imperfect environments in a quantum channel can affect the performance of quantum computation. Therefore, quantum error correcting codes (QECCs) have

†Corresponding author.

been used to achieve fault-tolerant quantum computation. In 1994, Shor first constructed a quantum code with parameter [[9,1,3]] from repetition codes.[5] In 1996, two independent groups, i.e., Calderbank and Shor[6] and Stean[7] considered the relationship between quantum codes and classical self-orthogonal codes, deriving quantum codes with parameter [[7,1,3]] from the Hamming code. In 1997, the general theory for QECCs, i.e., the stabilizer formalism was introduced by Gottesman.[8] Recently, with the development of quantum information theory, there are various models for quantum computation which are quantum gate array, one-way quantum computer, adiabatic quantum computer, topological quantum computer and Zidan's model. Zidan's model of quantum computation was proposed in 2018[9] which finds the solution of the quantum problem under scrutiny based on the degree of entanglement between two extra qubits. In other words, this model solves the quantum problem by applying some unitary transformation(s) on a system of size n qubits. Then, by measuring the degree of entanglement, via concurrence measure, between two ancillary qubits, the solution of the problem at hands is obtained based on the degree of entanglement. Recently, Zidan's model was used to solve some of the intractable problems in quantum computing such as solving the extended version of Deutsch–Jozsa problem[10] which was intractable for more than 27 years, and quantum machine learning models.[11] All models of quantum computation have been shown to be equivalent; each can simulate the other with no more than polynomial overhead. More applications for quantum codes in quantum information have been demonstrated such as quantum algorithms,[12] quantum simulations[13] and quantum communication.[14]

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,[15] based on group association schemes,[16] based on circulant matrices,[17,18] or based on CSS (Calderbank–Shor–Steane) structure over Finite field.[19] In[20] 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 the corresponding quantum stabilizer codes, such as self-dual codes over GF(4), which have dimension "0" and can be represented by graphs[21]; (2) QECCs based on self-dual codes over GF(4) with the highest known minimum weights[22] and (3) QECCs based on Hermitian self-orthogonal codes with extension design.[23] In addition, other constructions are based on self-orthogonal trace product codes such as the combining binary linear codes in[24] Plotkin sum of two codes in.[25] 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 paper 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. This paper is organized as follows. In Sec. 2, we present some preliminary quantum information and quantum error correction codes. In Sec. 3, a new design for the construction of self-orthogonal trace-inner product codes over GF(4) is proposed, and we explain the comparison between reference papers and our proposed method to show the practicality of our construction.

2. Preliminary

2.1. *Quantum information and quantum error correction codes*

A classical bit is an elementary unit of information; it must be one of the two states, conventionally "0" or "1". Classical bits and classical gates are the two basic elements that are used in classical computation. The basic unit of quantum information is a quantum bit, and it is denoted as a qubit. Contrary to the classical bit, a qubit can take more than the discrete values of "0" or "1". It can also assume all possible linear combinations of them; this is called superposition, which is an important and fundamental property of quantum mechanics. Since the two basis states of quantum information are two column vectors, i.e., $|0\rangle = \left[\begin{smallmatrix}1\\0\end{smallmatrix}\right]$ and $|1\rangle = \left[\begin{smallmatrix}0\\1\end{smallmatrix}\right]$, a superposition state or quantum state is denoted as $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle = \left[\begin{smallmatrix}\alpha\\\beta\end{smallmatrix}\right]$, where α and β are complex numbers that satisfy the equation: $|\alpha|^2 + |\beta|^2 = 1$. Quantum computation operates on its qubits using quantum gates, which are the unitary transformation of quantum states. Then, Pauli matrices, which include the identity matrix $\mathbf{I}$, and three nonidentity matrices $\mathbf{X}$, $\mathbf{Z}$ and $\mathbf{Y}$, are considered as the basis generators of all unitary transformations. The Pauli group for one qubit, i.e., $P_1 = \{\mathbf{I}, \mathbf{X}, \mathbf{Y}, \mathbf{Z}\}$, is closed under multiplication. Generally, an n-qubit Pauli group P_n is n-times the tensor product of the Pauli group P_1.

In classical computation, errors occur with classical data, and this problem is even more serious for quantum computation due to a new phenomenon known as decoherence. Hence, QECCs are used to achieve fault-tolerant quantum computation. The error model for a quantum channel assumes (1) quantum errors occurs independently and (2) quantum errors can be viewed as $\mathbf{I}$ (no flip), $\mathbf{X}$ (bit flip), $\mathbf{Z}$ (phase flip), or $\mathbf{Y}$ (a combination of bit and phase flips). Since the most important property of P_n is that any two elements of P_n are either commutative or anti-commutative, we can produce syndrome measurements to diagnose which error corrupts the encoded state. Classical error correction employs redundancy, and QECCs also extend the length of information qubits that can be protected. Assume that a QECC with the parameter $[[n, k, d_{\min}]]$, which will encode k information qubits into a system of n qubits, can correct $\lfloor \frac{d_{\min}-1}{2} \rfloor$ error.

Next, we explain that the stabilizer formalism is used to express the QECC. With the stabilizer formalism, QECCs are viewed as a group of quantum stabilizer operators, and we consider quantum operators rather than quantum states. A quantum stabilizer group S is an abelian subgroup of P_n, where the stabilizer group S is closed under multiplication. A set of stabilizer states from group S can be regarded as code C_S, such that $C_S = \{|\psi\rangle \in {H_2}^{\otimes n} : \mathbf{s}|\psi\rangle = |\psi\rangle, \forall \mathbf{s} \in S\}$. A QECC with parameter $[[n, k, d_{\min}]]$ corresponds to group S, which has a minimal representation in terms of $n - k$ independent generators $\{\mathbf{g}_1, \mathbf{g}_2, \ldots, \mathbf{g}_{n-k}\}$. The constraints for these generators are that any two elements in $\{\mathbf{g}_1, \mathbf{g}_2, \ldots, \mathbf{g}_{n-k}\}$ must be commutative to each other. It is known that for a QECC $[[n, k, d_{\min}]]$, we have the quantum singleton bound or Knill–Laflamme bound as $n - k \geq 2(d_{\min} - 1)$.[26] Then, the QECCs whose parameters satisfy can be the maximum distance separable codes or optimal QECCs.[27]

2.2. *From self-orthogonal trace-inner product codes to quantum stabilizer codes*

The Galois field with 2 elements (GF(2)) is defined over the set of 0 and 1, i.e., $\mathrm{GF}(2) = \{0, 1\}$, 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, 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 $x \in \mathrm{GF}(4) : \ \overline{x} = x^2$.
(2) Trace function: For any $x \in \mathrm{GF}(4), \mathrm{Tr} : \ \mathrm{GF}(4) \to \mathrm{GF}(2), \mathrm{Tr}(x) = x + \overline{x} = x + x^2$.
(3) Trace-inner product: For two vectors over $\mathrm{GF}(4) : \ \mathbf{u} = (u_1, u_2, \ldots, u_n)$ and $\mathbf{v} = (v_1, v_2, \ldots, v_n)$, trace-inner product: "$\bullet$": $\mathrm{GF}(4)^n \to \mathrm{GF}(2)$, we have $\mathbf{u} \bullet \mathbf{v} = \mathrm{Tr}(\mathbf{u} \times \overline{\mathbf{v}}) = \sum_{i=1}^{n} \mathrm{Tr}(u_i \times {v_i}^2)$.

Table 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

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).[22] The mapping between the Pauli operators and elements of GF(4) is shown in Table 1.

3. Code Construction Method

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

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

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

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

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

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

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

Then, the matrix $\mathbf{F}$ 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) as follows:

(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) $\mathrm{Tr}(a_k \times b_k) = 0$,
(5) $\mathrm{Tr}(\omega \times a_k + \omega^2 \times b_k) = \omega \times a_k + \omega^2 \times b_k + (\omega \times a_k + \omega^2 \times b_k)^2 = \omega \times a_k + \omega^2 \times b_k + \omega^2 \times a_k + \omega \times b_k = (\omega + \omega^2) \times (a_k + b_k) = a_k + b_k$.

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

$$\begin{aligned}
\mathbf{f}_0 \bullet \mathbf{f}_1 &= \mathrm{Tr}[(a_0 + \omega b_0) \times (a_{n-1} + \omega b_1)^2] + \mathrm{Tr}[(a_1 + \omega b_1) \times (a_0 + \omega b_2)^2] \\
&\quad + \cdots + \mathrm{Tr}[(a_{n-1} + \omega b_{n-1}) \times (a_{n-2} + \omega b_0)^2] \\
&= \mathrm{Tr}[(a_0 + \omega b_0) \times (a_{n-1} + \omega^2 b_1)] + \mathrm{Tr}[(a_1 + \omega b_1) \times (a_0 + \omega^2 b_2)] \\
&\quad + \cdots + \mathrm{Tr}[(a_{n-1} + \omega b_{n-1}) \times (a_{n-2} + \omega^2 b_0)] \\
&= \mathrm{Tr}[a_0 a_{n-1} + b_0 b_1 + \omega b_0 a_{n-1} + \omega^2 a_0 b_1] + \mathrm{Tr}[a_1 a_0 + b_1 b_2 + \omega b_1 a_0 + \omega^2 a_1 b_2] \\
&\quad + \cdots + \mathrm{Tr}[a_{n-1} a_{n-2} + b_{n-1} b_0 + \omega b_{n-1} a_{n-2} + \omega^2 a_{n-1} b_0] \\
&= \mathrm{Tr}[\omega b_0 a_{n-1} + \omega^2 a_0 b_1] + \mathrm{Tr}[\omega b_1 a_0 + \omega^2 a_1 b_2] \\
&\quad + \cdots + \mathrm{Tr}[\omega b_{n-1} a_{n-2} + \omega^2 a_{n-1} b_0] \\
&= (b_0 a_{n-1} + a_0 b_1) + (b_1 a_0 + a_1 b_2) + (b_2 a_1 + a_2 b_3) \\
&\quad + \cdots + (b_{n-1} a_{n-2} + a_{n-1} b_0) \\
&= b_0 a_{n-1} + (a_0 b_1 + b_1 a_0) + (a_1 b_2 + b_2 a_1) \\
&\quad + (a_2 b_3 + \cdots + b_{n-1} a_{n-2}) + a_{n-1} b_0 \\
&= b_0 a_{n-1} + a_{n-1} b_0 = 0.
\end{aligned} \tag{2}$$

Generally, to prove the self-orthogonal nature of matrix $\mathbf{F}$, we consider the trace-inner product of any two rows, such as the lth 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 lth row. Hence, without the 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 follows:

$$\begin{aligned}
\mathbf{f}_0 \bullet \mathbf{f}_k &= \mathrm{Tr}[(a_0 + \omega b_0) \times (a_{n-k} + \omega b_k)^2] + \mathrm{Tr}[(a_1 + \omega b_1) \times (a_{n-k+1} + \omega b_{k+1})^2] \\
&\quad + \cdots + \mathrm{Tr}[(a_{n-1} + \omega b_{n-1}) \times (a_{n-k-1} + \omega b_{k-1})^2]
\end{aligned}$$

$$
\begin{aligned}
&= \mathrm{Tr}[(a_0 + \omega b_0) \times (a_{n-k} + \omega^2 b_k)] + \mathrm{Tr}[(a_1 + \omega b_1) \times (a_{n-k+1} + \omega^2 b_{k+1})] \\
&\quad + \cdots + \mathrm{Tr}[(a_{n-1} + \omega b_{n-1}) \times (a_{n-k-1} + \omega^2 b_{k-1})] \\
&= \mathrm{Tr}[a_0 a_{n-k} + b_0 b_k + \omega b_0 a_{n-k} + \omega^2 a_0 b_k] \\
&\quad + \cdots + \mathrm{Tr}[a_{n-1} a_{n-k-1} + b_{n-1} b_{k-1} + \omega b_{n-1} a_{n-k-1} + \omega^2 a_{n-1} b_{k-1}] \\
&= \mathrm{Tr}[\omega b_0 a_{n-k} + \omega^2 a_0 b_k] + \cdots + \mathrm{Tr}[\omega b_{n-1} a_{n-k-1} + \omega^2 a_{n-1} b_{k-1}] \\
&= (b_0 a_{n-k} + a_0 b_k) + (b_1 a_{n-k+1} + a_1 b_{k+1}) \\
&\quad + (b_2 a_{n-k+2} + \cdots + (b_{n-1} a_{n-k-1} + a_{n-1} b_{k-1}) \\
&= \sum_{x=0}^{n-1} b_x a_{n-k+x} + \sum_{y=0}^{n-1} a_y b_{k+y} = \sum_{z+k-n=0}^{n-1} b_{z+k-n} a_z + \sum_{y=0}^{n-1} a_y b_{k+y} \\
&= \sum_{z=0}^{n-1} b_{z+k} a_z + \sum_{y=0}^{n-1} a_y b_{k+y} = 0. \qquad (3)
\end{aligned}
$$

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

3.2. *Generator matrix generation of the proposed quantum stabilizer codes with length from 7 to 12*

In this section, 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 1, we obtain the generator matrix of quantum stabilizer codes that corresponds 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.[28] 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.

3.2.1. *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 **u** and **v**, as shown in Construction 1, we have the following 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 **F** as the input, we get a quantum stabilizer code with parameter $[[7,1,3]]$, the standard form for its generators in as follows:

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

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.

3.2.2. *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:

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

3.2.3. *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 follows:

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 $[[11, 2, 3]]$

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

3.2.4. *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 in the following:

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 $[[12, 4, 3]]$

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

3.3. *Comparison between proposed codes with referenced codes*

In Table 2, we summarize some quantum stabilizer codes with lengths ranging from 7 to 12 over the proposed constructions. The optimal quantum stabilizer codes are defined as the codes where the parameters equalize the equation of the Knill–Laflamme bound; detailed discussion in Sec. 2. 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 construction and[29–32] are listed together in Fig. 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[29–32] respectively. The blanks in Fig. 1 mean that there are no existing quantum stabilizer codes with length n and dimension k.

As can be seen in Fig. 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.[29–32] The

Table 2. Quantum stabilizer codes from proposed construction.

Code length	Code parameters	Note
7	[[7,1,3]]	Optimal quantum stabilizer code
8	[[8,1,3]]	Optimal quantum stabilizer code
9	[[9,1,3]]	Optimal quantum stabilizer code
10	[[10,1,3]]	
11	[[11,1,3]]	
11	[[11,2,3]]	
12	[[12,1,4]]	
12	[[12,2,3]]	
12	[[12,3,3]]	
12	[[12,4,3]]	

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	-	-	-	-	-	-	-	-	-

Fig. 1. Minimum distance of proposed $[[n, k, d_{\min}]]$ in comparison with four referenced papers.

minimum distances of quantum stabilizer codes in[31] are larger than the ones of the proposed codes for a code length of 10; the code construction in[31] 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.[31] The constraint method, limitations of code length and distinct results of the proposed code constructions and the four referenced constructions are summarized in Table 3.

Table 3. Comparison between references and the proposed method.

Paper	Construction method	Limitation of code length	Main results
[29]	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.
[30]	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 $[[6, 1, 2]]$.
[31]	Based on nonresidue 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 $[[10, 4, 3]]$.
[32]	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.

4. Conclusion

In this paper, 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.

Acknowledgments

This work was supported by the Research Program through the National Research Foundation of Korea (NRF-2019R1A2C1005920).

References

1. C. H. Bennett and P. W. Shor, *IEEE Trans. Inf. Theory* **44**, 2724 (1998).
2. P. W. Shor, in Algorithms for quantum computation: Discrete logarithms and factoring, *Proc. 35th Annual Symp. Foundations of Computer Science* (IEEE Computer Society Press, 1994).
3. L. K. Grover, *Phys. Rev. Lett.* **79**, 325 (1997).
4. D. M. Nguyen and S. Kim, *Int. J. Theor. Phys.* **58**, 71 (2019).
5. P. W. Shor, *Phys. Rev. A* **52**, 2493 (1995).
6. A. R. Calderbank and P. W. Shor, *Phys. Rev. A* **54**, 1098 (1996).
7. A. M. Steane, *Phys. Rev. A* **54**, 4741 (1996).
8. D. Gottesman, Caltech Ph.D. thesis (1997), arXiv:quant-ph/9705052.
9. M. Zidan *et al.*, *Appl. Math. Inf. Sci.* **12**, 265 (2018).
10. M. Zidan *et al.*, *Results Phys.* **15**, 102549 (2019).
11. M. Zidan *et al.*, *Appl. Sci.* **9**, 1277 (2019).
12. D. M. Nguyen and S. Kim, *Mod. Phys. Lett. B* **33**, 1950270 (2019).
13. M. Steudtner and S. Wehner, *Phys. Rev. A* **99**, 022308 (2019).
14. D. M. Nguyen and S. Kim, *Int. J. Theor. Phys.* **58**, 2043 (2019).
15. D. M. Nguyen and S. Kim, *Symmetry* **10**, 655 (2018).
16. A. Naghipour *et al.*, *Int. J. Quantum Inform.* **13**, 1550021 (2015).
17. D. M. Nguyen and S. Kim, *Symmetry* **9**, 122 (2017).
18. D. M. Nguyen and S. Kim, Construction and complement circuit of a quantum stabilizer code with length 7, in *Proc. Eighth Int. Conf. Ubiquitous and Future Networks* (*ICUFN*) (Vienna, Austria, 2016).
19. G. Xu *et al.*, *Int. J. Mod. Phys. B* **31**(6), 1750034 (2017).
20. A. R. Calderbank *et al.*, *IEEE Trans. Inf. Theory* **44**, 1369 (1998).
21. J. Gao and Y. Wang, *IEEE Access* **7**, 26418 (2019).
22. J. L. Kim, *IEEE Trans. Inf. Theory* **47**, 1575 (2001).
23. D. M. Nguyen and S. Kim, *J. Comm. Net.* **20**, 309 (2018).
24. C. Y. Lai and C. C. Lu, *IEEE Trans. Inf. Theory* **57**, 7163 (2011).
25. R. Dastbasted, Quantum stabilizer codes, Master thesis, Sabanci University (2017).
26. I. Djordjevic, Quantum Information Processing and Quantum Error Correction: An Engineering Approach, *Academic Press*, 1st edn. (April 16, 2012).
27. M. Grassl, http://www.codetables.de accessed on12 July 2019.

28. MAGMA, http://magma.maths.usyd.edu.au/calc/.
29. Y. G. Zeng and M. H. Lee, *Adv. Math. Phys.* **2010**, 469124 (2010).
30. S. M. Zhao *et al.*, *Int. J. Quantum Inf.* **10**, 1250015 (2012).
31. Y. Xie, J. Yuan and Q. Sun, *IEEE Trans. Commun.* **66**, 9 (2018).
32. Y. Guo and M. H. Lee, *Quantum Inf. Process.* **8**, 361 (2009).