Quantum Information and Computation, Vol. 12, No. 3&4 (2012) 0181–0202
© Rinton Press

QUANTUM MCELIECE PUBLIC-KEY CRYPTOSYSTEM

HACHIRO FUJITA
Division of Information and Communications Systems, Tokyo Metropolitan University
6-6 Asahigaoka, Hino-city, Tokyo 191-0065, Japan

Received December 28, 2009
Revised November 26, 2011

The McEliece cryptosystem is one of the best-known (classical) public-key cryptosystems, which is based on algebraic coding theory. In this paper, we present a quantum analogue of the classical McEliece cryptosystem. Our quantum McEliece public-key cryptosystem is based on the theory of stabilizer codes and has the key generation, encryption and decryption algorithms similar to those in the classical McEliece cryptosystem. We present an explicit construction of the quantum McEliece public-key cryptosystem using Calderbank-Shor-Steane codes based on generalized Reed-Solomon codes. We examine the security of our quantum McEliece cryptosystem and compare it with alternative systems.

Keywords: Public-key cryptosystem, McEliece encryption, quantum information

Communicated by: R Cleve & M Mosca

1 Introduction

Multi-party quantum computation requires secure transmission of quantum information, i.e., quantum states. If the sender and receiver share Einstein-Podolsky-Rosen (EPR) pairs, they can use quantum teleportation [1] to privately transmit quantum states via a dual classical channel. Another possible solution to the problem is to use the *quantum one-time pad* (or *private quantum channel*) [2, 3]. The quantum one-time pad encrypts quantum states in an unconditionally secure way using shared classical keys. Encryption of a single qubit (two-level quantum system) requires two classical bits shared between the sender and receiver. To use the quantum one-time pad the sender and receiver have to share a secret key in advance. To establish a secret key between the sender and receiver they can use quantum key distribution such as the Bennett-Brassard 1984 (BB84) protocol [4]. Since BB84 has been proved to be unconditionally secure (see, e.g., [5]), the combination of the quantum one-time pad and BB84 achieves perfect secrecy. However, the BB84 protocol does not need entanglement but requires a preshared secret key for authentication.

Although the two solutions suggested above completely solve the problem of secure transmission of quantum states, we want a more efficient method for the problem which require neither shared entanglement nor common randomness. Recently, the concept and some realizations of a quantum public-key cryptosystem (QPKC) have been proposed [6, 7, 8, 9]. Most of the existing QPKCs encrypt classical information with a quantum public key. It is believed that they are quantum-computationally secure, i.e., they cannot be broken by a quantum computer. As opposed to symmetric cryptography, public-key cryptography does not need shared secret keys in advance but needs authentication of public keys. It seems that a quantum public key is more difficult to authenticate than a classical public key.

The McEliece public-key cryptosystem [10] is based on algebraic coding theory and its security relies on the difficulty of the problem of decoding general linear codes, an NP-hard problem in coding theory. It is believed that NP-hard problems cannot be solved in polynomial time on a quantum computer. Although Grover's search algorithm [11] can accelerate some classical attacks, the resulting computation cost is still exponential (see [12]). Hence the McEliece public-key cryptosystem seems to be immune to quantum computers.

In this paper we propose a quantum analogue of the McEliece public-key cryptosystem. Our quantum McEliece cryptosystem is based on quantum coding theory, more specifically, the theory of stabilizer codes (or additive quantum codes), and the security of the system relies on the hardness of the problem of decoding general stabilizer codes. The public and secret keys used in the system are classical and the message to be encrypted may be classical or quantum.

The structure of the paper is as follows. In Section 2 we review the classical McEliece and Niederreiter public-key cryptosystems, which is helpful in understanding our quantum McEliece public-key cryptosystem. In Section 3, for convenience of the reader we review the theory of stabilizer codes, which includes the definition, encoding and decoding of a stabilizer code. We show that the problem of decoding general stabilizer codes is intractable. We also review the Calderbank-Shor-Steane (CSS) codes and consider the decoding problem. In Section 4 we first present the quantum McEliece public-key cryptosystem in general and then give an explicit construction of the quantum McEliece cryptosystem using CSS codes derived from generalized Reed-Solomon (GRS) codes. In Section 5 we study the security of the quantum McEliece cryptosystem and show that the quantum McEliece cryptosystem is secure against conceivable classical and quantum attacks. In Section 6 we compare our quantum McEliece cryptosystem with an alternative system that uses the quantum one-time pad and the Niederreiter public-key cryptosystem. We also discuss Yang's quantum McEliece public-key cryptosystem. Section 7 concludes the paper. The appendix gives the detail of the construction of a family of CSS codes derived from GRS codes.

Notation: We denote by $\mathbb{F}_2$ the binary field $\{0,1\}$ and by $\mathbb{C}$ the field of complex numbers. A bold-faced italic symbol such as $\boldsymbol{a}$ denotes a binary vector. (In Appendix A we use the same notation for vectors with components from an extension field.) $\mathbf{0}$ denotes the zero vector of appropriate length or the zero matrix of appropriate size. For binary vectors $\boldsymbol{a}$ and $\boldsymbol{b}$, $\boldsymbol{a} \vee \boldsymbol{b}$ denotes the bit-wise OR of $\boldsymbol{a}$ and $\boldsymbol{b}$, $\boldsymbol{a} \cdot \boldsymbol{b}$ denotes the dot product of $\boldsymbol{a}$ and $\boldsymbol{b}$, and $\mathrm{wt}(\boldsymbol{a})$ denotes the Hamming weight of $\boldsymbol{a}$. For a matrix A, A^T denotes the transpose of A.

2 The McEliece/Niederreiter Public-Key Cryptosystem

Our quantum McEliece public-key cryptosystem (PKC) is a quantum analogue of the classical McEliece PKC [10]. For convenience of the reader we review the classical McEliece PKC and its dual version, the Niederreiter PKC [13]. For an extensive survey of code-based cryptography see [14, 15].

2.1 McEliece PKC

Let C be a binary linear code of length n, dimension k, and minimum distance $d \geq 2t+1$. We assume that C has an efficient decoder that can correct up to t errors. The McEliece cryptosystem is given as follows.

- *Setup:* Bob chooses three binary matrices, G, S, and P, where G is a $k \times n$ generator matrix of C, S is a random $k \times k$ nonsingular matrix, and P is a random $n \times n$ permutation matrix. Bob computes $\hat{G} = SGP$, publishes his public key $(\hat{G}, t)$, and keeps secret his private key (G, S, P).

- *Encryption:* Alice obtains Bob's public key $(\hat{G}, t)$ and encodes her message $\boldsymbol{m} \in \mathbb{F}_2^k$ into a ciphertext $\boldsymbol{c} = \boldsymbol{m}\hat{G} + \boldsymbol{e} \in \mathbb{F}_2^n$, where $\boldsymbol{e} \in \mathbb{F}_2^n$ is a randomly chosen binary vector of Hamming weight t. Alice sends the ciphertext $\boldsymbol{c}$ to Bob.

- *Decryption:* Bob multiplies the received ciphertext $\boldsymbol{c}$ on the right by the permutation P^{-1} to obtain $\boldsymbol{c}P^{-1} = \boldsymbol{m}SG + \boldsymbol{e}P^{-1}$. Note that $\boldsymbol{e}P^{-1}$ has Hamming weight t. Using the decoder for C Bob corrects the error $\boldsymbol{e}P^{-1}$ to obtain $\boldsymbol{m}S$. Multiplying $\boldsymbol{m}S$ on the right by S^{-1}, Bob obtains Alice's message $\boldsymbol{m}$.

If the generator matrix $\hat{G}$ is of systematic form then most of the bits of the message will be revealed. So $\hat{G}$ must be of non-systematic form and the public key size is kn.

The security of the McEliece PKC relies on the difficulty of the problem of decoding a general linear code. Consider the following decision problem:

COSET WEIGHTS
Instance: an $m \times n$ binary matrix A, a binary vector $\boldsymbol{y}$ of length m, and a positive integer w.
Question: does there exist a binary vector $\boldsymbol{x}$ of length n and Hamming weight up to w such that $\boldsymbol{x}A^T = \boldsymbol{y}$?

Berlekamp *et al.* [16] showed that the above problem is intractable.

Lemma 1 *COSET WEIGHTS is NP-complete.*

2.2 Niederreiter PKC

The Niederreiter public-key cryptosystem is a dual version of the McEliece cryptosystem. Let C be the binary linear code given in the previous subsection and let H be an $(n-k) \times n$ parity check matrix of C. Let P be as in the McEliece PKC and M be a random $(n-k) \times (n-k)$ nonsingular matrix. The message space of the Niederreiter PKC is identified with the set of all binary vectors of length n and weight t. The Niederreiter PKC is given as follows.

- *Setup:* Bob computes $\hat{H} = MHP$, publishes his public key $(\hat{H}, t)$, and keeps secret his private key (H, M, P).

- *Encryption:* Alice obtains Bob's public key $(\hat{H}, t)$ and encodes her message $\boldsymbol{e}$, which is a binary vector of Hamming weight t, into a ciphertext $\boldsymbol{s} = \boldsymbol{e}\hat{H}^T$. Alice sends the ciphertext $\boldsymbol{s}$ to Bob.

- *Decryption:* Bob multiplies the received ciphertext $\boldsymbol{s}$ on the right by the inverse of M^T to obtain $\boldsymbol{s}(M^T)^{-1} = \boldsymbol{e}P^T H^T$. Using the decoder for C Bob obtains the binary vector $\boldsymbol{e}P^T$ and multiplying $\boldsymbol{e}P^T$ on the right by P Bob obtains Alice's message $\boldsymbol{e}$.

Note that the Niederreiter PKC is deterministic, while the McEliece PKC is probabilistic.

Since the message space of the Niederreiter cryptosystem is the set of all binary vectors of length n and weight t, the number of messages is given by $\binom{n}{t}$, i.e., the message length is $\log_2\binom{n}{t}$ bits. Note that the ciphertext $\boldsymbol{c} = \boldsymbol{e}\hat{H}^T$ is no more than a syndrome of $\boldsymbol{e}$ and has length $n-k$. Hence the ratio of the message length to the ciphertext length is given by $\frac{\log_2\binom{n}{t}}{n-k}$. Since the public key $\hat{H}$ is a binary matrix of size $(n-k)\times n$, the public key size is $n(n-k)$. However, as opposed to the McEliece PKC, we may assume that $\hat{H}$ is of systematic form (see, e.g., [15, p. 132]). If we use a systematic form of $\hat{H}$, then the public key size reduces to $k(n-k)$.

Remark 1 *It has been shown [17] that the McEliece PKC is equivalent to the Niederreiter PKC with comparable parameters.*

3 Elements of the Theory of Stabilizer Codes

In this section we review the definition and the encoding and decoding of a stabilizer code. We then consider the decoding complexity of general stabilizer codes. We also review Calderbank-Shor-Steane (CSS) codes and consider the complexity of the problem of decoding CSS codes. For more information about stabilizer codes, see [18, 19, 20].

3.1 Definition

Quantum systems are described by *Hilbert spaces*, complex vector spaces with an inner product. Let $\mathcal{H} = \mathbb{C}^2$ be the two dimensional Hilbert space, which describes a two level quantum system which is called a *quantum bit* (*qubit* for short). Let $\mathcal{H}_n = \mathcal{H}^{\otimes n}$ be the n-fold tensor product of n copies of $\mathcal{H}$. Then $\mathcal{H}_n$ describes an n-qubit system. Let X and Z be the two of the Pauli matrices defined by

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \tag{1}$$

and let $Y = XZ$. X and Z correspond to a bit-flip error and a phase error, respectively, and Y to both errors. Note that $XZ = -ZX$. We define the *error group* G_n on n qubits to be the set of all n-fold tensor products of Pauli matrices X, Y, Z and the 2×2 identity matrix I_2 with multiplicative factors ± 1. For $\boldsymbol{a} = (a_1,\ldots,a_n), \boldsymbol{b} = (b_1,\ldots,b_n) \in \mathbb{F}_2^n$ we define

$$X(\boldsymbol{a}) = X^{a_1} \otimes \cdots \otimes X^{a_n} \tag{2}$$

$$Z(\boldsymbol{b}) = Z^{b_1} \otimes \cdots \otimes Z^{b_n}. \tag{3}$$

Then every element E of G_n can be written uniquely as $E = (-1)^c X(\boldsymbol{a})Z(\boldsymbol{b})$ for some $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{F}_2^n$ and $c \in \mathbb{F}_2$. Let $E = X(\boldsymbol{a})Z(\boldsymbol{b})$ and $E' = X(\boldsymbol{a}')Z(\boldsymbol{b}')$, where $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{a}', \boldsymbol{b}' \in \mathbb{F}_2^n$. Then we have

$$EE' = (-1)^{\boldsymbol{a}\cdot\boldsymbol{b}' + \boldsymbol{a}'\cdot\boldsymbol{b}} E'E. \tag{4}$$

The identity element of the error group G_n is given by the n-fold tensor product of I_2's, which is denoted by I_{2^n}. Let S be a commutative subgroup of the error group G_n not containing $-I_{2^n}$. The *stabilizer code* Q with stabilizer S is the simultaneous eigenspace of every operator in S with eigenvalue $+1$:

$$Q = \{|\psi\rangle \in \mathcal{H}_n \colon\ E|\psi\rangle = |\psi\rangle \text{ for all } E \in S\}. \tag{5}$$

We assume from now on that S is generated by $n-k$ independent generators, $E_1, \ldots, E_{n-k}$. Then Q has dimension 2^k. (Q encodes k qubits to n qubits.) Each generator E_i can be written uniquely as $E_i = (-1)^{c_i} X(\boldsymbol{a}_i) Z(\boldsymbol{b}_i)$ for some $\boldsymbol{a}_i, \boldsymbol{b}_i \in \mathbb{F}_2^n$ and $c_i \in \mathbb{F}_2$. The *generator matrix* H of the stabilizer S is defined to be the $(n-k) \times 2n$ matrix over $\mathbb{F}_2$ whose rows are given by $(\boldsymbol{a}_i | \boldsymbol{b}_i)$, $i = 1, \ldots, n-k$:

$$H = [H_X | H_Z] = \left[\begin{array}{c|c} \boldsymbol{a}_1 & \boldsymbol{b}_1 \\ \ldots & \ldots \\ \boldsymbol{a}_{n-k} & \boldsymbol{b}_{n-k} \end{array} \right]. \tag{6}$$

We define the alternate bilinear form on $\mathbb{F}_2^{2n}$ by

$$\langle (\boldsymbol{a}|\boldsymbol{b}), (\boldsymbol{a}'|\boldsymbol{b}') \rangle = \boldsymbol{a} \cdot \boldsymbol{b}' + \boldsymbol{a}' \cdot \boldsymbol{b}, \quad (\boldsymbol{a}|\boldsymbol{b}), (\boldsymbol{a}'|\boldsymbol{b}') \in \mathbb{F}_2^{2n}, \tag{7}$$

where the $\cdot$ operation is the usual dot product of two vectors. Then, from Eq. (4) the commutativity of the stabilizer S becomes the orthogonal relation between the vectors $(\boldsymbol{a}_j | \boldsymbol{b}_j)$, i.e., $H_X H_Z^T + H_Z H_X^T = \boldsymbol{0}$. The generator matrix H for the stabilizer code Q is very useful in encoding and decoding of Q, which we will review briefly in the next two subsections.

3.2 Encoding

Encoding can be performed on a quantum circuit of size $O(n^2)$ constructed from the *standard form* of the generator matrix for a stabilizer code [21]. The standard form is not unique, but if a standard form is given, the encoding circuit is completely specified by the standard form.

Remark 2 *It should be noticed that using Gaussian elimination one can compute a standard form of the generator matrix for a stabilizer code in $O(n^3)$ time on a classical computer. A Gaussian elimination procedure transforms $H = [H_X | H_Z]$ into $H' = [MH_XP | MH_ZP]$, where M is some $(n-k) \times (n-k)$ nonsingular binary matrix and P is some $n \times n$ permutation matrix.*

3.3 Decoding

Let Q be the stabilizer code with stabilizer S defined in Section 3.1 The *minimum distance* of Q is defined to be the minimum number of Pauli matrices (not I_2 factors) in an element of $C_{G_n}(S) \backslash S$, where $C_{G_n}(S)$ is the centralizer of S in G_n. We denote by $[[n, k, d]]$ the parameters of a stabilizer code of length n, dimension 2^k and minimum distance d. If the stabilizer code Q has minimum distance d, then Q can detect up to $d-1$ errors and correct up to $\lfloor (d-1)/2 \rfloor$ errors. To detect and correct errors we perform the measurement of the generators of the stabilizer S, which is called the *syndrome measurement*. If the errors that have occurred on the qubits are written as $E = X(\boldsymbol{e}_X) Z(\boldsymbol{e}_Z)$ for some $\boldsymbol{e}_X, \boldsymbol{e}_Z \in \mathbb{F}_2^n$, the syndrome $\boldsymbol{s}$ is given by

$$\boldsymbol{s} = (\boldsymbol{e}_Z | \boldsymbol{e}_X) H^T = \boldsymbol{e}_Z H_X^T + \boldsymbol{e}_X H_Z^T. \tag{8}$$

The syndrome gives the information about the errors that have occurred on qubits and using this information error detection and correction can be done, although it may take much time to compute the location and pattern (bit-flip or phase-flip or both) of an error. We consider the following decision problem:

DECODING STABILIZER CODES

Instance: two $m \times n$ binary matrices A and A' such that $AA'^T + A'A^T = \boldsymbol{0}$, a binary vector

$\boldsymbol{y}$ of length m, and a positive integer w.
Question: do there exist two binary vectors $\boldsymbol{x}, \boldsymbol{x}'$ of length n such that $\boldsymbol{x}A^T + \boldsymbol{x}'A'^T = \boldsymbol{y}$ and $\mathrm{wt}(\boldsymbol{x} \vee \boldsymbol{x}') \leq w$?

Lemma 2 *DECODING STABILIZER CODES is NP-complete.*

Proof. It is obvious that DECODING STABILIZER CODES is in NP. To prove its NP-completeness we will show that COSET WEIGHTS can be reduced to DECODING STABILIZER CODES. In fact, for a given instance $(A, \boldsymbol{y}, w)$ of COSET WEIGHTS we can set the instance $(A, A' = \mathbf{0}, \boldsymbol{y}, w)$ of DECODING STABILIZER CODES. □.

From the above theorem, for a general stabilizer code minimum distance decoding, i.e., the problem of finding a solution $(\boldsymbol{e}_X|\boldsymbol{e}_Z)$ of Eq. (8) given a syndrome $\boldsymbol{s}$ such that $\mathrm{wt}(\boldsymbol{e}_X \vee \boldsymbol{e}_Z)$ is minimum is NP-hard. In our applications we need bounded distance decoding, i.e., decoding up to half the minimum distance. We conjecture that bounded distance decoding of general stabilizer codes is NP-hard. A more general result on the complexity of decoding quantum error correcting codes can be found in [22].

3.4 Calderbank-Shor-Steane codes

Calderbank-Shor-Steane (CSS) codes discovered independently by Calderbank and Shor [23], and by Steane [24] are an important class of quantum error-correcting codes, which are constructed from classical binary linear codes. In fact, CSS codes are a special class of stabilizer codes. Let C_1 and C_2 be two binary linear codes of length n and dimensions k_1 and k_2, respectively, such that $C_2 \subset C_1$. The CSS code based on C_1 and C_2 is a stabilizer code with the generator matrix of the stabilizer defined by

$$H = \left[\begin{array}{c|c} H(C_2^\perp) & \mathbf{0} \\ \mathbf{0} & H(C_1) \end{array}\right], \tag{9}$$

where $H(C_1)$ and $H(C_2^\perp)$ are the parity check matrices for C_1 and $C_2^\perp$, respectively. Note that $H(C_2^\perp)$ is a generator matrix for C_2. Since $C_2 \subset C_1$, we have that $H(C_1)H(C_2^\perp)^T = \mathbf{0}$. The CSS code defined above has parameters $[[n, k = k_1 - k_2, d]]$, where $d = \min\{\mathrm{wt}(C_1 \setminus C_2), \mathrm{wt}(C_2^\perp \setminus C_1^\perp)\}$.

Encoding. Since a CSS code is a stabilizer code, it is encoded in the same way as presented in Section 3.2. Since the encoding method of Cleve and Gottesman is generic, it may not be optimum for CSS codes. It may be possible to use the CSS code structure in encoding. In fact, Grassl *et al.* [25] show an encoding circuit for a CSS code, which is a little simpler than the Cleve–Gottesman construction. Furthermore, it can be specified by a more compact data, which leads to a smaller public key of the quantum McEliece cryptosystem to be presented in the next section. Let $G(C_2)$ be a $k_2 \times n$ generator matrix for C_2 (we can take, e.g., $G(C_2) = H(C_2^\perp)$). Since $C_2 \subset C_1$, we can extend $G(C_2)$ to a generator matrix for C_1 by adding $k_1 - k_2$ row vectors in C_1. We denote by $G'(C_1)$ the matrix consisting of these rows. So $G(C_2)$ and $G'(C_1)$ form a $k_1 \times n$ generator matrix $G(C_1)$ for C_1:

$$G(C_1) = \left[\begin{array}{c} G(C_2) \\ G'(C_1) \end{array}\right]. \tag{10}$$

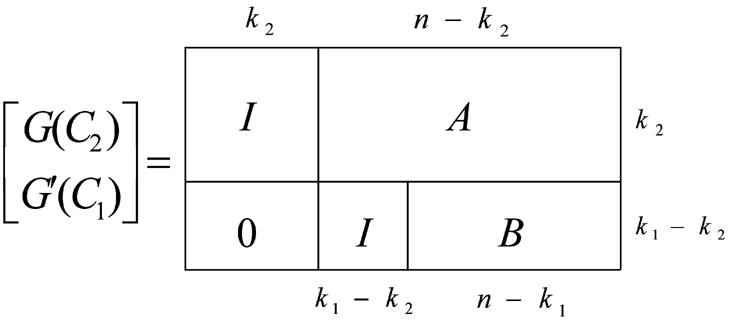

Fig. 1. Generator matrix for C_1.

By performing elementary row operations on $G(C_1)$ and exchanging the columns of $G(C_1)$, more explicitly, by multiplying on the left by an appropriately chosen $k_1 \times k_1$ nonsingular binary matrix S of the form

$$S = \left[\begin{array}{c|c} S_1 & \mathbf{0} \\ S_2 & S_3 \end{array}\right], \tag{11}$$

and on the right by an $n \times n$ permutation matrix P, we may assume that $k_1 \times n$ matrix $SG(C_1)P$ has the same form as shown in Fig. 1. This is a generator matrix for a permutation of C_1 and completely specifies an encoding circuit for the CSS code. For more details see [25].

Decoding. CSS codes can be decoded in the same way as stabilizer codes. One may think that the special structure on CSS codes may make the decoding of the codes easier than that of general stabilizer codes. To address this issue we consider the following decision problem:

DECODING CSS CODES
Instance: two binary matrices A and A' of sizes $m \times n$ and $m' \times n$, respectively, such that $AA'^T = \mathbf{0}$, two binary vectors $\boldsymbol{y}$ and $\boldsymbol{y}'$ of length m and m', respectively, and a positive integer w.
Question: do there exist two binary vectors $\boldsymbol{x}$ and $\boldsymbol{x}'$ of length n and Hamming weight at most w such that $\boldsymbol{x}A^T = \boldsymbol{y}$ and $\boldsymbol{x}'A'^T = \boldsymbol{y}'$?

Lemma 3 *DECODING CSS CODES is NP-complete.*

Proof. It is obvious that DECODING CSS CODES is in NP. To prove its NP-completeness we will show that COSET WEIGHTS can be reduced to DECODING CSS CODES, as in the proof of Lemma 2. For a given instance $(A, \boldsymbol{y}, w)$ of COSET WEIGHTS set the instance $(A, A' = \mathbf{0}, \boldsymbol{y}, \boldsymbol{y}' = \mathbf{0}, w)$ for the DECODING CSS CODES problem. □.

Remark 3 *We can construct a CSS code from a binary linear code C which contains its dual $C^\perp$, i.e., $C^\perp \subset C$. Taking $C_1 = C$ and $C_2 = C^\perp$ in Eq. (9) the generator matrix of the stabilizer becomes*

$$H = \left[\begin{array}{c|c} H(C) & \mathbf{0} \\ \mathbf{0} & H(C) \end{array}\right]. \tag{12}$$

Note that $H(C)$ has the property that $H(C)H(C)^T = \mathbf{0}$. The CSS code defined by H is compactly specified by $H(C)$ alone, but we cannot prove or disprove that the corresponding decision problem is NP-complete. We conjecture that the problem of decoding a general dual-containing code is NP-hard.

4 The Quantum McEliece Cryptosystem

In this section, we first present the general structure of the quantum McEliece cryptosystem (QMC) and then give an explicit construction of a QMC using the family of CSS codes derived from generalized Reed-Solomon (GRS) codes. Our QMC is a generalization of the classical McEliece cryptosystem.

4.1 Description of the QMC: The general construction

Suppose that Alice wants to send Bob a k-qubit message in (pure or mixed) state ρ. Let Q be an $[[n, k, d \geq 2t+1]]$ stabilizer code, and let $H = [H_X|H_Z]$ be the generator matrix of the stabilizer of Q, where H_X (resp. H_Z) denotes the $(n-k) \times n$ binary matrix corresponding to the X (resp. Z) part of the stabilizer generators. We assume that there exists a fast decoding algorithm for Q that solves Eq. (8). Our QMC is given as follows.

- *Setup:* Bob chooses a random $(n-k) \times (n-k)$ nonsingular binary matrix M' and a random $n \times n$ permutation matrix P', computes $[M'H_XP'|M'H_ZP']$, and transforms it into a standard form $[M''M'H_XP'P''|M''M'H_ZP'P'']$, where M'' is an appropriate $(n-k) \times (n-k)$ nonsingular binary matrix and P'' is an appropriate $n \times n$ permutation matrix. Let $M = M''M'$, $P = P'P''$ and $\hat{H} = [MH_XP|MH_ZP]$. Bob publishes his public key $(\hat{H}, t)$ and keeps secret his private key (H, M, P).

- *Encryption:* Alice obtains Bob's public key $(\hat{H}, t)$ and encodes her k-qubit message ρ to an n-qubit cipher state $\sigma = \mathcal{E}(\rho)$, where $\mathcal{E}$ is the encoder corresponding to $\hat{H}$. Alice randomly chooses t out of the n qubits and applies one of the Pauli matrices (X, Y and Z) to each of the t qubits. Alice sends Bob the resulting state σ'.

- *Decryption:* Bob performs the syndrome measurement corresponding to $\hat{H}$ on the received qubits σ' to obtain the syndrome $\boldsymbol{s}$, and computes $\boldsymbol{s}' = \boldsymbol{s}(M^T)^{-1}$. From $\boldsymbol{s}'$ he finds the error locations and patterns $(\hat{\boldsymbol{e}}_Z|\hat{\boldsymbol{e}}_X)$ with the help of the decoder for Q. Bob applies $Z(\hat{\boldsymbol{e}}_ZP)X(\hat{\boldsymbol{e}}_XP)$ to the received state σ' to obtain σ. Finally, Bob runs the encoder $\mathcal{E}$ backward and obtains $\mathcal{E}^{-1}(\sigma) = \mathcal{E}^{-1} \circ \mathcal{E}(\rho) = \rho$, which is Alice's k-qubit message.

Remark 4 *The standard form of a public key not only specifies the encoder circuit for a stabilizer code, but also reduces the effective size of the public key.*

Lemma 4 *Decryption in the cryptosystem works.*

Proof. Suppose the errors introduced by Alice is written as $E = X(\boldsymbol{e}_X)Z(\boldsymbol{e}_Z)$ for some $\boldsymbol{e}_X, \boldsymbol{e}_Z \in \mathbb{F}_2^n$. Then the syndrome $\boldsymbol{s}$ is given by

$$\boldsymbol{s} = (\boldsymbol{e}_Z|\boldsymbol{e}_X)\hat{H}^T = (\boldsymbol{e}_ZP^T|\boldsymbol{e}_XP^T)H^TM^T. \tag{13}$$

Bob computes $\boldsymbol{s}' = \boldsymbol{s}(M^T)^{-1} = (\boldsymbol{e}_ZP^T|\boldsymbol{e}_XP^T)H^T$, and by using the decoder for Q he can find the phase-flip errors and bit-flip errors $(\hat{\boldsymbol{e}}_Z|\hat{\boldsymbol{e}}_X) = (\boldsymbol{e}_ZP^T|\boldsymbol{e}_XP^T)$. He applies the permutation P to both the errors and obtains $(\hat{\boldsymbol{e}}_ZP|\hat{\boldsymbol{e}}_XP) = (\boldsymbol{e}_ZP^TP|\boldsymbol{e}_XP^TP) = (\boldsymbol{e}_Z|\boldsymbol{e}_X)$, since the permutation P satisfies $P^T = P^{-1}$. Finally, Bob applies the inverse $E^{-1} = Z(\boldsymbol{e}_Z)X(\boldsymbol{e}_X)$ to the received state σ' to obtain σ. □.

Remark 5 *The matrix M plays an important role in the security of the QMC. It hides the structure of a stabilizer code and makes it impossible for Eve to recover the private key. However, for Bob who knows the stabilizer code the matrix M is not important, since the stabilizer (not of operator type) is a additive group. Therefore, in the decryption Bob may use another generator matrix for the stabilizer of a permutation of Q defined by $\tilde{H} = [H_X P | H_Z P]$, which extracts the same error patterns $(\boldsymbol{e}_Z | \boldsymbol{e}_X)$ as $\hat{H}$. In this case Bob does not need to compute the inverse of M^T.*

4.2 Encryption of classical messages

Alice's message may be classical or quantum. In the quantum case a message is a pure or mixed quantum state, which may be unknown to Alice. There are some ways of encrypting classical messages. A naive way is to identify k-bit messages $\boldsymbol{m} \in \mathbb{F}_2^k$ with k-qubit computational basis states $|\boldsymbol{m}\rangle \in \mathcal{H}_k$. If Alice wants to send a classical message $\boldsymbol{m}$, she encodes $\boldsymbol{m}$ to $|\boldsymbol{m}\rangle$ and then encrypts it using the QMC presented in the previous section. Alice may use the first qubit of her k-qubit message as a flag bit to tell Bob whether the message is classical or not.

A more secure but less efficient method for encryption of classical messages is to use random Hadamard transforms. To encrypt a $k/2$-bit message $\boldsymbol{m}$ Alice randomly chooses a $k/2$-bit string $\boldsymbol{r}$ and prepares the state $|\psi_{\boldsymbol{m},\boldsymbol{r}}\rangle = |\boldsymbol{r}\rangle \otimes H(\boldsymbol{r})|\boldsymbol{m}\rangle$, where H is the Hadamard transform

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{14}$$

and we have used the same notation as $X(\boldsymbol{a})$ defined in Section 3.1. Note that $|\psi_{\boldsymbol{m},\boldsymbol{r}}\rangle$ is a superposition of computational basis states unless $\boldsymbol{r} = \boldsymbol{0}$. Then Alice encodes the state $|\psi_{\boldsymbol{m},\boldsymbol{r}}\rangle$ into a cipher state using the QMC. If he receives the cipher state, Bob decodes it to $|\psi_{\boldsymbol{m},\boldsymbol{r}}\rangle$. He measures the first $k/2$ qubits to obtain $\boldsymbol{r}$ and then applies $H(\boldsymbol{r})$ to the last $k/2$ qubits to obtain $|\boldsymbol{m}\rangle$ (and hence the message $\boldsymbol{m}$).

Although we show the security of the first method against some attacks in Section 5.3, we recommend the second method for secure transmission of classical messages.

4.3 The QMC based on CSS codes

In the previous section we showed the QMC in general. Since CSS codes have special structure, if we use a CSS code as a building block of the QMC, we can simplify the encryption and decryption in the QMC and reduce the key size of the system.

Let H be a generator matrix of the stabilizer for a CSS code Q defined by Eq. (9) and $G = G(C_1)$ be a generator matrix of the form Eq. (10). We assume that C_1 and $C_2^\perp$ can correct up to t errors.

- *Setup:* Bob chooses a random $k_1 \times k_1$ nonsingular binary matrix S' of the form Eq. (11) and a random $n \times n$ permutation matrix P', computes $S'GP'$, and transforms it into a matrix of the form shown in Fig. 1, $S''S'GP'P''$, where S'' is an appropriate $k_1 \times k_1$ nonsingular binary matrix of the form Eq. (11) and P'' is an appropriate $n \times n$ permutation matrix. Let $S = S''S'$, $P = P'P''$ and $\hat{G} = SGP$. Bob publishes his public key $(\hat{G}, t)$ and keeps secret his private key (G, S, P).

- *Encryption:* Alice obtains Bob's public key $(\hat{G}, t)$ and encodes her k-qubit message ρ to an n-qubit cipher state $\sigma = \mathcal{E}(\rho)$, where $\mathcal{E}$ is the encoder corresponding to $\hat{G}$.

Alice randomly chooses t positions and applies the Pauli-X operator to each of the t positions. Alice also randomly and independently chooses t positions and applies the Pauli-Z operator to each of the t positions. Alice sends Bob the resulting state σ'.

- *Decryption:* Bob performs the syndrome measurement corresponding to $H(C_2^\perp)P$ and $H(C_1)P$ on the received qubits σ' to obtain the syndrome $\boldsymbol{s}$. From $\boldsymbol{s}$ Bob finds the error locations and patterns $(\hat{\boldsymbol{e}}_Z|\hat{\boldsymbol{e}}_X)$ with the help of the decoder for Q. Bob applies $Z(\hat{\boldsymbol{e}}_Z P)X(\hat{\boldsymbol{e}}_X P)$ to the received state σ' to obtain σ. Finally, Bob runs the encoder $\mathcal{E}$ backward and obtains $\mathcal{E}^{-1}(\sigma) = \mathcal{E}^{-1} \circ \mathcal{E}(\rho) = \rho$, which is Alice's k-qubit message.

The proof that the decryption works is the same as that of the general case. Note that Bob does not use S, part of the private key, in the decryption (see Remark 5).

Remark 6 *In the encryption Alice may choose dependent bit-flip (Pauli-X) and phase-flip (Pauli-Z) errors to improve the probability of detecting an active adversary who introduces some errors to the quantum state sent from Alice to Bob. An example is as follows: Let* s *be a positive integer smaller than* t*, which is now part of the public key. Alice randomly chooses* t *positions for* X *errors and then randomly chooses* s *out of the* t *positions and* $t-s$ *out of the remaining* $n-t$ *positions for* Z *errors. Then the numbers of* X*,* Y *and* Z *errors is* $t-s$*,* s*,* $t-s$*, respectively. This dependence of* X *and* Z *errors reduces the decoding complexity, but enhances the capability of detecting such an adversary, since introducing some errors to the cipher state changes the dependence of* X *and* Z *errors, which leads to the changes of the numbers of* X*,* Y *and* Z *errors.*

4.4 Explicit construction

Using generalized Reed-Solomon (GRS) codes we can construct a family of CSS codes with parameters $[[n = mN, k = m(N-2K), d \geq K+1]]$, where m, N and K are positive integers such that $1 \leq K \leq N \leq 2^m$. There are $(2^m-1)^{N-1}$ CSS codes in the family. The detail of the construction is given in Appendix A.

Using the encoding method of Grassl *et al.* given in Section 3.4 the information about the encoding circuit for a CSS code in the family is encoded in $m^2K(2N-3K)$ bits, which is the public key of the QMC using the CSS code. It should be noticed that CSS codes based on GRS codes can be decoded efficiently with the Berlekamp–Massey algorithm [26, 27].

Example 1 *Taking* $m = 8$*,* $N = 256$ *and* $K = 100$ *we obtain a family of* $[[2048, 448, d \geq 101]]$ *CSS codes. The number of CSS codes in the family is about* $10^{613.7}$*. Another choice of the parameters is* $m = 9$*,* $N = 512$ *and* $K = 200$*, which define a family of* $10^{1384.0}$ $[[4608, 1008, d \geq 201]]$ *CSS codes.*

Remark 7 *The Niederreiter PKC based on GRS codes can be broken by using the Sidelnikov-Shestakov (SS) algorithm [28]. The SS algorithm computes another private key of the Niederreiter PKC in polynomial time from the public key which is obtained by symbol-wise scrambling and permutation (i.e.,* M *and* P*) of the parity check matrix for a GRS code (see Section 2.2). Since the public key of the QMC using CSS codes based on GRS codes is obtained by bit-wise scrambling and permutation (i.e.,* S *and* P*) of the generator matrix for the* binary expansion *of a GRS code (see Section 4.3 and Remark A.1 in Appendix A), the SS algorithm cannot apply directly to the QMC. This is the same as the reason why the SS algorithm does not* *apply to the (classical) McEliece PKC based on Goppa codes which are subfield subcodes of GRS codes (see Remark 3.1 of [14] and Remark 3 of [15]).*

5 Security of the QMC

In this section we examine the security of the QMC constructed in the previous section. To do this we first overview some attacks on the classical McEliece PKC. A good survey of this topic can be found in [14]. Many attacks on the McEliece PKC have been proposed so far. These attacks are classified into three categories: ciphertext-only attack, partially-known-plaintext attack, and chosen-ciphertext attack which includes the message-resend attack [29] and the reaction attack [30]. Note that since a public key is available, an eavesdropper can easily mount a chosen-plaintext attack. Since plaintexts and ciphertexts in the QMC are quantum states, we have to carefully consider whether quantum versions of these attacks exist or not. Note that even if a message state to be encrypted is classical i.e., a computational basis state, its corresponding ciphertext is a quantum state (a superposition of computational basis states). See Section 4.2 for encryption of classical messages. Since the QMC constructed in the previous section is based on a class of CSS codes, "ciphertext-only attack" reduces to a decoding problem for classical codes. We will show this type of attack on the QMC in the next section.

5.1 Stern's attack

Information set decoding (ISD) is a basic attack on the classical McEliece/Niederriter PKC. Some variants of ISD have been proposed so far: to name a few, the Adams-Meijer algorithm [31], the Lee-Brickell algorithm [32], Leon's algorithm [33] and Stern's algorithm [34] and its variants [35, 36]. Recently, Bernstein *et al.* [37] have succeeded in breaking the McEliece PKC with the original parameters suggested by McEliece. Their attack is an optimization of Stern's algorithm and the best lower bound on the work factor for ISD based attacks is derived from a generalization of Stern's algorithm [38]. So we use Stern's algorithm for cryptanalysis of our system. The original Stern algorithm takes as input an $(n-k) \times n$ binary matrix H and a positive integer w, and output a codeword of weight w, if any. Stern's algorithm is an iterative algorithm and each iteration takes T bit operations where

$$T = \frac{1}{2}(n-k)^2(n+k) + 2lp\binom{k/2}{p} + 2p(n-k)\binom{k/2}{p}^2/2^l, \tag{15}$$

where p and l are two parameters to be optimized. The probability P that each iteration succeeds in finding a codeword of weight w is given by

$$P = \frac{\binom{w}{2p}\binom{n-w}{k-2p}\binom{2p}{p}\binom{n-k-w+2p}{l}}{4^p\binom{n}{k}\binom{n-k}{l}}. \tag{16}$$

Hence the average number of iterations required is given by $1/P$ and the work factor (average-case complexity) of Stern's algorithm is given by T/P. For more details on Stern's algorithm see [34]. Note that Stern's algorithm can be applied to syndrome decoding, that is, the problem of finding, given an $(n-k)$-bit syndrome $\boldsymbol{s}$, an n-bit error vector $\boldsymbol{e}$ of weight t such that $\boldsymbol{e}H^T = \boldsymbol{s}$.

Table 1. Proposed parameters and security analysis of the quantum McEliece PKC

QMC with $[[n,k,d]]$ CSS	$[[2048, 448, \geq 101]]$	$[[4608, 1008, \geq 201]]$
# of qubits transmitted	2048	4608
# of information qubits	448	1008
Information rate	0.22	0.22
Public key size [bits]	1356800	6868800
Stern	$2 \cdot 2^{78.6}$ $(p = 3, l = 33)$	$2 \cdot 2^{145.1}$ $(p = 3, l = 37)$
Grover	$2 \cdot 2^{65.5}$	$2 \cdot 2^{103.4}$

Table 2. Some parameters of the Niederreiter PKC for key distribution

Niederreiter PKC with (n,t) Goppa	$(2600, 165)$	$(2700, 174)$
# of bits transmitted	1980	2088
# of information bits	881	926
Information rate	0.45	0.44
Public key size [bits]	1227600	1277856
# of qubits to be encrypted	440	463
Stern	$2^{79.7}$ $(p = 4, l = 37)$	$2^{79.5}$ $(p = 4, l = 37)$

Niederreiter PKC with (n,t) Goppa	$(6100, 365)$	$(6200, 373)$
# of bits transmitted	4745	4849
# of information bits	1987	2028
Information rate	0.42	0.42
Public key size [bits]	6429475	6550999
# of qubits to be encrypted	993	1014
Stern	$2^{146.2}$ $(p = 4, l = 42)$	$2^{146.3}$ $(p = 4, l = 42)$

In Table 1 we show the work factors of Stern's attack on the quantum McEliece PKCs using the CSS codes with the parameters given in Example 1. Note that in the decoding of a CSS code we can separately perform bit-flip error correction and phase error correction, which leads to the factor of 2 of the work factor in the table.

5.2 *Using Grover's algorithm to accelerate ISD*

The ISD based attack against the quantum McEliece PKC we have considered is basically a classical attack, although Eve have to perform the syndrome measurement on the transmitted qubits. We have to investigate more sophisticated quantum attacks on the quantum McEliece PKC. Although code based cryptography does not suffer from Shor's factoring algorithm, Grover's search algorithm may speed up some ISD based attacks on the McEliece PKC. In fact, Bernstein [12] shows that a quantum ISD algorithm is more efficient than a simple classical counterpart. Here we give the *basic quantum information set decoding* algorithm for a binary linear code C of length n, dimension k and minimum distance at least $2t+1$ using an $(n-k) \times n$ parity check matrix H for C instead of using a generator matrix as shown in [12]. Let $\boldsymbol{e}$ be an error vector of weight t and let $\boldsymbol{s} = \boldsymbol{e}H^T$. We first give a classical information set decoding algorithm and then speed up the algorithm via Grover's algorithm. Let S be an information set that is error-free (i.e., S contains no positions in error.) Then we can take an $(n-k) \times (n-k)$ nonsingular matrix U such that UH is a systematic parity check matrix with S being an information set (i.e., the submatrix corresponding to the complement of S is the $(n-k)\times(n-k)$ identity matrix except for the column ordering). From $\boldsymbol{s}=\boldsymbol{e}H^T$, $\boldsymbol{s}U^T$ has weight t and its nonzero components correspond to $\mathrm{supp}(\boldsymbol{e})$. The classical information set decoding algorithm is given as follows.

1. Randomly choose a k-set S from n positions and perform some elementary row operations on $[H|\boldsymbol{s}^T]$ to obtain a matrix such that the submatrix corresponding to the complement of S is the $(n-k)\times(n-k)$ identity matrix (this is just a Gaussian elimination procedure).
2. If the last column has weight t then the corresponding positions in the complement of S are in error.

There are $\binom{n}{k}$ ways of choosing k-sets from n positions and $\binom{n-t}{k}$ ways of choosing error-free k-sets. As shown in [12] almost 0.29 fraction of error-free k-sets are information sets. Hence the classical information set decoding algorithm needs $\binom{n}{k}/[0.29\binom{n-t}{k}]$ iterations on average and each iteration requires $T=(n-k)^2(n+k+1)/2+(n-k)$ bit operations. Hence the work factor of the algorithm is $T\binom{n}{k}/[0.29\binom{n-t}{k}]$. As shown in [12], by using Grover's algorithm one can reduce the average number of iterations into its square root and so the work factor of the quantum version of the algorithm is $T\sqrt{\binom{n}{k}/[0.29\binom{n-t}{k}]}$ qubit operations.

In Table 1 we show the work factor of Grover's attack on the quantum McEliece PKC with chosen parameters. The result shows that Grover's attack is more efficient than Stern's attack, although quantum computation differs from classical computation and they cannot be compared with each other in a naive way. Grover's attack on the quantum McEliece PKC based on the $[[2048,448,\geq 101]]$ CSS code requires $2^{66.5}$ qubit operations on average. This number is critically small with respect to nowadays (classical) computing resource, but it will be infeasible to perform such a number of qubit operations with an initial stage quantum computer. Note that both of the encoding and syndrome measurement of an $[[n,k]]$ quantum error-correcting code require at most $n(n-k)$ qubit operations. In our $[[2048,448,\geq 101]]$ CSS code case we need at most $2^{21.6}$ qubit operations, which is far below $2^{66.5}$ and will be achievable with a quantum computer that can perform up to three and a half million of qubit operations in real time.

5.3 On (in)security of classical messages against some attacks

In this section we will consider some attacks on the QMC based on GRS-CSS codes for classical messages using the encoding method of Grassl *et al.* presented in Section 3.4 and the first method for encryption of classical messages presented in Section 4.2. As explained there, Alice may use a flag bit to tell Bob whether the message is classical or not. The flag bit and the message are encoded together into a quantum state. Hence, even if Eve intercepts the quantum state, Eve cannot decide whether the message is classical or not until she decodes the quantum state and measures the flag bit. Below we assume that Eve somehow knows that Alice sends a classical message to Bob.

5.3.1 Ciphertext-only attack

When Eve decodes the quantum state into which Alice encoded a classical message, Eve does not need to correct phase errors. To obtain the message Eve first measures the quantum state in the computational basis to obtain $\boldsymbol{c}_1+\boldsymbol{c}_2+\boldsymbol{e}$, where $\boldsymbol{c}_1$ is the codeword of C_1 corresponding to the message, $\boldsymbol{c}_2$ is a random codeword of C_2 and $\boldsymbol{e}$ is a bit-flip error pattern that Alice has added. Eve has only to correct the bit-flip errors $\boldsymbol{e}$ introduced by Alice. So the complexity of information set decoding for the classical message case is half of the complexity for the quantum message case.

As presented in Section 3.4, by using the generator matrix $G(C_1)$ of the standard form in Fig. 1 $\boldsymbol{c}_1$ and $\boldsymbol{c}_2$ can be written as $\boldsymbol{c}_1 = \boldsymbol{m}G'(C_1)$ and $\boldsymbol{c}_2 = \boldsymbol{r}G(C_2)$, respectively, where $\boldsymbol{m}$ is a $(k_1 - k_2)$-bit message and $\boldsymbol{r}$ is a random k_2-bit vector. Since $G(C_2)$ is of systematic form, most of the bits of $\boldsymbol{r}$ are revealed, although Eve does not know which bits are correct and which ones are in error. On the other hand, since the block in $G(C_2)$ above the identity matrix in $G'(C_1)$ is nonzero, $\boldsymbol{c}_1 + \boldsymbol{c}_2 + \boldsymbol{e}$ does not reveal the massage $\boldsymbol{m}$ apparently. It is well-known that if the distribution on the message space is uniform then we may put the public key into the systematic form (see, e.g., [15, pp. 128–129]).

Remark 8 *Since $\boldsymbol{r}$ is not relevant to the message $\boldsymbol{m}$, Eve may not decode $\boldsymbol{r}$ and she has only to find a k_2-bit vector $\boldsymbol{r}'$ and an n-bit error vector $\boldsymbol{e}'$ such that $\boldsymbol{c}_1 + \boldsymbol{c}_2 + \boldsymbol{e} = \boldsymbol{c}_1 + \boldsymbol{r}'G(C_2) + \boldsymbol{e}'$. It is possible for Eve to find such a pair of $\boldsymbol{r}'$ and $\boldsymbol{e}'$ if she knows the syndrome of $\boldsymbol{e}$ with respect to C_2, but in fact she only knows the syndrome of $\boldsymbol{e}$ with respect to C_1 which is part of the syndrome of $\boldsymbol{e}$ with respect to C_2. So, what Eve can do is to find a C_1-codeword $\boldsymbol{c}_1 + \boldsymbol{c}_2$ (and hence $\boldsymbol{m}$ and $\boldsymbol{r}$) and the true error vector $\boldsymbol{e}$.*

Remark 9 *To find the bit-flip errors $\boldsymbol{e}$ Eve may use the generator matrix version of Stern's algorithm instead of the parity check matrix version as presented in Section 5.1. For more details on the generator matrix version of Stern's algorithm see [14]. The problem reduces to the one of finding a codeword of weight t in the $[n, k_1+1, t]$ binary code that is the binary code C_1 augmented by $\boldsymbol{e}$. For the CSS code of length 2048 in Example 1 the constituent binary codes C_1 and C_2 have parameters $[2048, 1248, \geq 101]$ and $[2048, 800, \geq 157]$, respectively. In this case the problem is to find a codeword of weight 50 in the $[2048, 1249, 50]$ code. The complexity of the generator matrix version of Stern's algorithm is $2^{78.8}$, which is almost the same as the parity check matrix version (see Table 1).*

Remark 10 *Since the first (naive) encryption method for classical messages using the CSS code based QMC encrypts a classical message into bit values of a quantum state, the reader may think that phase errors introduced by Alice do not play any role. However, introducing phase errors as well as bit errors increases the security of a message and it is important that Bob measures the syndrome for phase error correction and checks whether the phase errors are correctable or not, since Eve may introduce some errors beyond the error-correction capability of the CSS code. Only if both bit-flip and phase errors are corrected, the message is independent of Eve with high probability and hence is probably secure. The same comment, of course, applies to the security of a quantum message (cf. Example 2).*

5.3.2 Partially known plaintext attack

If Eve knows part of a classical message, then she can easily perform a partially known plaintext attack as in the classical case (see, e.g., [14]). Partial knowledge on the message reduces the complexity of information set decoding.

5.3.3 Message-resend attack

We briefly review the *message-resend attack* on the classical McEliece cryptosystem presented by Berson [29]. We use the notation introduced in Section 2. Alice sends a ciphertext $\boldsymbol{c} = \boldsymbol{m}\hat{G} + \boldsymbol{e}$ to Bob, where $\boldsymbol{m}$ is a message and $\boldsymbol{e}$ is an error vector of weight t. Eve who pretends to be Bob intercepts the ciphertext $\boldsymbol{c}$ and asks Alice to resend a ciphertext. Alice sends another ciphertext $\boldsymbol{c}' = \boldsymbol{m}\hat{G} + \boldsymbol{e}'$ to Bob, where $\boldsymbol{e}'$ is an error vector which differs from $\boldsymbol{e}$ with high probability. Eve again intercepts $\boldsymbol{c}'$ and compute the bit-wise XOR (modulo 2 sum) $\boldsymbol{c} + \boldsymbol{c}'$ to obtain $\boldsymbol{e} + \boldsymbol{e}'$. The nonzero components of $\boldsymbol{e} + \boldsymbol{e}'$ give (part of) the error locations of $\boldsymbol{e}$ or $\boldsymbol{e}'$. The key observation is that since $\boldsymbol{e}$ and $\boldsymbol{e}'$ are random error vectors (of weight t), the position where $\boldsymbol{e} + \boldsymbol{e}'$ is zero is error-free with high probability. This observation makes it possible for Eve to accelerate ISD. For more details on the message-resend attack see [29].

We now consider a quantum version of the message-resend attack. If Eve measures the quantum state corresponding to a classical message in the computational basis, then the measurement outcome takes the form $\boldsymbol{c}_1 + \boldsymbol{c}_2 + \boldsymbol{e}$, where $\boldsymbol{c}_1$ is the codeword of C_1 corresponding to the message, $\boldsymbol{c}_2$ is a random codeword of C_2 and $\boldsymbol{e}$ is an error vector that Alice has added. Eve who pretends to be Bob asks Alice to resend another quantum state that encrypts the same classical message as before. If Eve measures the quantum state in the computational basis, then the measurement outcome also has the same form $\boldsymbol{c}_1 + \boldsymbol{c}_2' + \boldsymbol{e}'$, where $\boldsymbol{c}_2'$ is another codeword of C_2 that is different from $\boldsymbol{c}_2$ with high probability and $\boldsymbol{e}'$ is an error vector added by Alice that is also different from $\boldsymbol{e}$ with high probability. Taking the bit-wise XOR of the two outcomes Eve obtains $\boldsymbol{c}_2 + \boldsymbol{c}_2' + \boldsymbol{e} + \boldsymbol{e}'$. Since $\boldsymbol{c}_2$ and $\boldsymbol{c}_2'$ are different from each other, the sum $\boldsymbol{c}_2 + \boldsymbol{c}_2'$ is a nonzero codeword of C_2. So Eve cannot obtain $\boldsymbol{e} + \boldsymbol{e}'$. Thus, the message-resend attack does not apply to this case.

Remark 11 *Eve may find the sum* $\mathbf{e} + \mathbf{e}'$ *with Stern's algorithm. Consider again the CSS code of length* 2048 *in Example 1. In this case we have* $t = 50$. *From the same analysis as done by Berson [29] the average weight of* $\mathbf{e} + \mathbf{e}'$ *is* 97.6. *Suppose that* $\mathbf{e} + \mathbf{e}'$ *has weight* 98 *(note that* $\mathbf{e} + \mathbf{e}'$ *has even weight) and consider the* $[2048, 801, 98]$ *binary code obtained from* C_2 *by adding* $\boldsymbol{c}_2 + \boldsymbol{c}_2' + \mathbf{e} + \mathbf{e}'$ *to it. Note that* C_2 *has parameters* $[2048, 800, \geq 157]$. *Then* $\mathbf{e} + \mathbf{e}'$ *is a minimum weight codeword of the augmented code and we can find such a codeword by using (the generator matrix version of) Stern's algorithm. The complexity of finding such a codeword is* $2^{82.4}$*, which is higher than the complexity of a ciphertext-only attack. Similarly, if* $\mathbf{e} + \mathbf{e}'$ *has weight* 96*, then the complexity is* $2^{81.0}$.

5.3.4 More on attacks on the QMC for classical messages

It may be interesting to further investigate quantum analogs of classical attacks on the classical McEliece cryptosystem (CMC). The *reaction attack* [30] is an example of a chosen ciphertext attack on the CMC. Suppose that Alice sends a ciphertext to Bob. Eve intercepts the ciphertext and flips a few bits of it to generate a new ciphertext which is sent to Bob. Bob receives the modified ciphertext and try to decode it. If he fails to decode the received ciphertext, he asks Eve who pretends to be Alice to resend the ciphertext again. This reaction of Bob gives Eve some information about the error vector which Alice generated in her encryption.

We have to carefully consider whether a quantum analog of the reaction attack exists or not. The problem depends on the definition of a quantum protocol which exchanges quantum information as well as classical information between two distant parties. A good definition is not trivial and we here do not give any definition.

Finally, we comment on the second method for encryption of classical messages presented in Section 4.2. The second encryption method resists the ciphertext-only attack, the partially known plaintext attack and the message-resend attack presented above. In fact, if Eve measures the cipher state in the computational basis then the original message cannot perfectly be reconstructed. So, the second method is more secure (though less efficient) than the first one.

5.4 Malleability

A stabilizer code with stabilizer S cannot detect an error in the centralizer $C_{G_n}(S)$ of its stabilizer S. Suppose that Alice sends a (classical or quantum) message to Bob using the QMC. Eve intercepts the quantum state sent by Alice, applies a unitary transformation in $C_{G_n}(S) \setminus S$ to the state, and finally resends the modified version of the quantum state to Bob. In this case Bob cannot detect Eve's cheating.

A simple way to protect classical messages against the malleability attack is to use a hash function to check the validity of a message, that is, Alice computes the hash value of a message, concatenates it with the message and then encrypts the modified message using the quantum McEliece cryptosystem. After decoding Bob computes the hash value of the message and checks whether or not the value is the same as the one added by Alice.

6 Comparison of the QMC and Alternative Systems

Let us recall the task that we want to perform: Alice wants to send a quantum message to Bob. They can use an insecure classical channel and an insecure quantum channel, but they have neither entanglement nor shared randomness. An adversary, Eve, wants to obtain the quantum message that Alice sends to Bob. Our QMC can accomplish this task. The task can also be performed by the quantum one-time pad (QOTP) with a classical PKC.

6.1 QOTP with Niederreiter PKC

Any public-key cryptosystem can be used to distribute a key for QOTP encryption. Although we have a large list of public key encryption schemes (see, e.g., [39]), most of the currently used PKCs such as RSA and elliptic curve cryptography cannot be used since these can be broken by Shor's factoring algorithm [40]. Some PKCs resistant to quantum computers exist: lattice-based cryptography, code-based cryptography and multivariate cryptography and so on (see, e.g., [41]). For fair comparison we restrict ourself to code-based cryptography, since our quantum McEliece PKC can be thought of as code-based cryptography.

The McEliece PKC is vulnerable to some advanced attacks (e.g., the message resend attack [29] and the reaction attack [30]) and some modifications have been proposed. The Kobara-Imai conversions [42] make the McEliece PKC secure against chosen ciphertext attack (CCA) under the random oracle model. Since the random oracle model assumes an ideal random function that does not exist in the real world (even in the future), we restrict ourself to public-key encryption schemes in the standard model for key distribution.

We compare our quantum McEliece PKC with the combination of the Niederreiter PKC and the quantum one-time pad. Niederreiter encryption has some advantages over the McEliece PKC (see, i.e., [36]). Note that although the Niederreiter PKC does not suffer from the message-resend attack, it is vulnerable to the reaction attack. For comparison, in Table 2 we show competitive Niederreiter PKCs with almost the same work factor of Stern's algorithm as the quantum McEliece PKC. We use the same system parameters used in the description of the McEliece/Niederreiter PKC based on Goppa codes in [14, 15]. A binary Goppa code is completely specified by three positive integers, m, n and t, where n is the length of the code which is smaller than or equal to 2^m. The binary Goppa code with parameters (m, n, t) has dimension $k \geq n - mt$ and minimum distance $d \geq 2t + 1$ (i.e., one can correct t errors with bounded distance decoding). Note that the binary Goppa code is a subfield subcode of a GRS code over the extension field $\mathbb{F}_{2^m}$ with 2^m elements. For more details on Goppa codes see [43]. As in [14, 15], we omit parameter m if it is clear from context.

We first compare the public-key sizes of the two systems. The proposed quantum McEliece PKCs each have a larger key size than the competing system. This is a drawback of our system. We next compare the communication costs required for the systems. The quantum McEliece PKCs with proposed parameters need a larger amount of quantum communication, but require no classical communication. On the other hand, the competing system only needs the same amount of quantum communication as the message quantum state, but needs a larger amount of classical communication. Although classical communication is cheaper and more reliable than quantum communication, if the cost of quantum and classical communication is the same (it may be possible in the future), the cost of (quantum) communication in the quantum McEliece PKC is smaller than the total cost of quantum and classical communication in the competing system.

Remark 12 *The quantum computation cost for the QMC is higher than that for the competing system, since the QMC requires the storage and processing of a large number of qubits. For classical computation the QMC has an advantage over the competing system, since the size of the finite field used in the QMC is generally smaller than that used in the competing system and so is the number of errors to be corrected (t in our notation). Note that the complexity of the Berlekamp–Massey algorithm for GRS and Goppa codes is* $O(t^2)$.

6.2 Yang's quantum McEliece PKC

Yang [44] proposes another quantum version of the classical McEliece PKC, which is an encryption method for quantum massages. The basic idea of Yang is to use the computational basis to perform McEliece encryption. Using the notation used in Section 2 we review Yang's quantum McEliece PKC. The private key and the public key of the system are exactly the same as those of the classical McEliece PKC: the private key is (G, S, P) and the public key is $(\hat{G}, t)$, where $\hat{G} = SGP$ (see Section 2 for the notation). If Alice sends a k-qubit message $\sum_{\boldsymbol{m}} \alpha_{\boldsymbol{m}} |\boldsymbol{m}\rangle$ to Bob, she transforms $\sum_{\boldsymbol{m}} \alpha_{\boldsymbol{m}} |\boldsymbol{m}\rangle$ into $\sum_{\boldsymbol{m}} \alpha_{\boldsymbol{m}} |\boldsymbol{m}\hat{G} + \boldsymbol{e}\rangle$ using the public key $(\hat{G}, t)$, where $\boldsymbol{e}$ is an error vector of weight t. Alice sends the resulting quantum state to Bob. Bob receives it and performs some unitary operations which are essentially the same as the decryption algorithm of the classical McEliece PKC. See [44] for more details. Yang's quantum McEliece PKC can be regarded as a special case of our QMC based on CSS codes. In fact, defining C_1 to be a code generated by the generator matrix G and taking $C_2 = \{\boldsymbol{0}\}$, the trivial code, we obtain an equivalent of Yang's system. Note that the encryption and decryption algorithms of Yang's system are different from ours. Although Yang's system is more efficient than a QMC with nontrivial C_2, it may leak some information about the message state as shown in the following example.

Example 2 *Suppose that Alice sends Bob the cipher state $\sum_{\boldsymbol{m}} \alpha_{\boldsymbol{m}} |\boldsymbol{m}\hat{G} + \boldsymbol{e}\rangle$ corresponding to the message state $\sum_{\boldsymbol{m}} \alpha_{\boldsymbol{m}} |\boldsymbol{m}\rangle$. Eve intercepts the cipher state and without performing error correction she applies to the cipher state a controlled gate based on a generalized inverse $\hat{G}^{-1}$ of $\hat{G}$ (see [44]). Then she obtains the state of the form $\sum_{\boldsymbol{m}} \alpha_{\boldsymbol{m}} |\boldsymbol{m} + \boldsymbol{e}\hat{G}^{-1}\rangle = X(\boldsymbol{e}\hat{G}^{-1}) \sum_{\boldsymbol{m}} \alpha_{\boldsymbol{m}} |\boldsymbol{m}\rangle$. Although the state obtained in this way contains some bit-flip errors, Eve can perform the X basis measurement on the state to obtain the same measurement statistics as that obtained from the original message state $\sum_{\boldsymbol{m}} \alpha_{\boldsymbol{m}} |\boldsymbol{m}\rangle$, since the X basis measurement and the bit-flip operator $X(\boldsymbol{e}\hat{G}^{-1})$ commute. This vulnerability is due to the fact that Alice introduces no phase errors. Note that Bob cannot detect and correct any phase error, since C_2 is trivial. A naive way to protect the system against the above attack is to apply the Hadamard transform to the message state before encryption on Alice's side and after decryption on Bob's side. In this case, however, Eve can perform the Z basis measurement freely. A more secure way of protection might be to use a random Hadamard transform as introduced in Section 4.2, though this reduces the efficiency of the system by half. The random Hadamard transform enables Yang's system to encrypt a classical message, although the original system assumes quantum messages. Note that when a classical message with uniform distribution is encrypted into a quantum cipher state using the combination of Yang's system and the randomization, Eve can still perform the above attack to obtain a bit string which is the same as the original message at a portion of about 3/4. This is substantially large compared to a random guess, i.e., Eve generates a random bit string where each bit equals the corresponding bit of the original message with probability 1/2.*

In [9] Yang *et al.* propose a quantum analog of the Niederreiter PKC. Using the notation used in Section 2 we review their quantum Niederreiter PKC. In this case the private key is (H, M, P) and the public key is $(\hat{H}, t)$, where $\hat{H} = MHP$ (see Section 2 for the notation). In their quantum Niederreiter PKC a quantum message $\sum_{\boldsymbol{m}} \alpha_{\boldsymbol{m}} |\boldsymbol{m}\rangle$ is encrypted into $\sum_{\boldsymbol{m}} \alpha_{\boldsymbol{m}} |\boldsymbol{m} + \boldsymbol{e}\rangle |\boldsymbol{m}\hat{H}^T\rangle$. Since their quantum Niederreiter PKC has the same weakness as shown in the above example, we do not compare it with our system.

7 Conclusion

In this paper we have presented the quantum analogue of the classical McEliece cryptosystem. We have considered the quantum McEliece cryptosystem for qubits, but the extension to arbitrary prime dimensions is straightforward. Although the QMC is not unconditionally secure, the system with properly chosen parameters is resistant to conceivable classical and quantum attacks and so it may provide a practical solution to secure transmission of quantum information. Quantum states are not easy to manipulate, but this property is preferable in cryptographic applications, since an adversary faces unknown quantum states which are difficult to deal with than bit strings of ciphertexts in conventional public-key encryption schemes. So our QMC may be a good candidate for a post-quantum public-key encryption scheme especially for classical information.

Since public-key cryptography for quantum information is still in its infancy, much effort should be made. In particular, more on quantum attacks (e.g., attacks using entanglement) should be examined, which is our future research problem.

Acknowledgments

The author would like to thank the referee for his/her useful comments that greatly improved the presentation and results of the paper. He is also grateful to Dr. Min-Hsiu Hsieh for his comments on the manuscript of the paper.

References

1. C. H. Bennett, G. Brassard, C. Crépeau, R. Jozsa, A. Peres, and W. K. Wootters (1993), *Teleporting an unknown quantum state via dual classical and Einstein-Podolsky-Rosen channels*, Phys. Rev. Lett., vol. 70, no. 13, pp. 1895–1899.
2. A. Ambainis, M. A. Mosca, A. Tapp, and R. de Wolf (2000), *Private quantum channels*, in Proc. IEEE Symposium on Foundations of Computer Science, pp. 547–553.
3. P. O. Boykin and V. Roychowdhury (2003), *Optimal encryption of quantum bits*, Phys. Rev. A, vol. 67, 042317.
4. C. H. Bennett and G. Brassard (1984), *Quantum cryptography: public key distribution and coin tossing*, in Proc. IEEE International Conference on Computers, Systems, and Signal Processing, pp. 175–179.
5. P. W. Shor and J. Preskill (2000), *Simple proof of security of the BB84 quantum key distribution protocol*, Phys. Rev. Lett., vol. 85, no. 2, pp. 441–444.
6. T. Okamoto, K. Tanaka, and S. Uchiyama (2000), *Quantum public-key cryptosystems*, in Advances in Cryptology–CRYPTO 2000 (LNCS 1880), pp. 147–165.
7. A. Kawachi, T. Koshiba, H. Nishimura, and T. Yamakami (2005), *Computational indistinguishability between quantum states and its cryptographic application*, in Advances in Cryptology–EUROCRYPT 2005 (LNCS 3494), pp. 268–284.
8. G. M. Nikolopoulos (2008), *Applications of single-qubit rotations in quantum public-key cryptography*, Phys. Rev. A, vol. 77, 032348.
9. Li Yang, Min Liang, Bao Li, Lei Hu, and Deng-Guo Feng (2010), *Quantum public-key cryptosystems based on induced trapdoor one-way transformations*, e-print arXiv:1012.5249v1.
10. R. J. McEliece (1978), *A public-key cryptosystem based on algebraic coding theory*, DSN Progress Report 42-44, pp. 114–116, JPL, Caltech.
11. L. K. Grover (1997), *Quantum mechanics helps in searching for a needle in a haystack*, Phys. Rev. Lett., vol. 79, no. 2, pp. 325–328.
12. Daniel J. Bernstein (2010), *Grover vs. McEliece*, in PQCrypto 2010 (LNCS 6061), pp.73-80.
13. H. Niederreiter (1986), *Knapsack-type cryptosystems and algebraic coding theory*, Problems of Control and Information Theory, vol. 15, no. 2, pp. 159–166.
14. D. Engelbert, R. Overbeck, and A. Schmidt (2007), *A summary of McEliece-type cryptosystems and their security*, J. Math. Crypt., vol. 1, pp. 151–199.
15. R. Overbeck and N. Sendrier (2009), *Code-based cryptography*, in [41].
16. E. R. Berlekamp, R. J. McEliece, and H. C. van Tilborg (1978), *On the inherent intractability of certain coding problems*, IEEE Transactions on Information Theory, vol. 24, no. 3, pp. 384–386.
17. Y. X. Li, R. H. Deng, and X. M. Wang (1994), *On the equivalence of McEliece's and Niederreiter's public-key cryptosytems*, IEEE Transactions on Information Theory, vol. 40, no. 1, pp. 271–273.
18. D. Gottesman (1997), *Stabilizer Codes and Quantum Error Correction*, Ph.D. thesis, Caltech. Available at http://arxiv.org/abs/quant-ph/9705052.
19. A. R. Calderbank, E. M. Rains, P. W. Shor, and N. J. A. Sloane (1998), *Quantum error correction via codes over GF(4)*, IEEE Transactions on Information Theory, vol. 44, no. 4, pp. 1369–1387.
20. M. A. Nielsen and I. L. Chuang (2000), *Quantum Computation and Quantum Information*, Cambridge University Press.
21. R. Cleve and D. Gottesman (1997), *Efficient computations of encodings for quantum error correction*, Phys. Rev. A, vol. 56, 76–82.
22. Min-Hsiu Hsieh and Francois Le Gall (2011), *NP-hardness of decoding quantum error correction*

codes, Phys. Rev. A, vol. 83, 052331.

23. A. R. Calderbank and P. W. Shor (1996), *Good quantum error-correcting codes exist*, Phys. Rev. A, vol. 54, no. 2, pp. 1098–1105.
24. A. M. Steane (1996), *Multiple particle interference and quantum error correction*, Proc. Roy. Soc. Lond. A, vol. 452, pp. 2551–2577.
25. M. Grassl, M. Rötteler, and T. Beth (2003), *Efficient quantum circuits for non-qubit quantum error-correcting codes*, International Journal of Fundations of Computer Science, vol. 14, no. 5, pp. 757–775.
26. E. R. Berlekamp (1968), *Algebraic Coding Theory*, McGraw-Hill (New York).
27. J. L. Massey (1969), *Shift-register synthesis and BCH decoding*, IEEE Transactions on Information Theory, vol. 15, no. 1, pp. 122–127.
28. V. M. Sidelnikov and S. O. Shestakov (1992), *On insecurity of cryptosystems based on generalized Reed-Solomon codes*, Discrete Mathematics and Applications, vol. 2, no. 4, pp. 439–444.
29. T. Berson (1997), *Failure of the McEliece public-key cryptosystem under message-resend and related-message attack*, in Advances in Cryptology–CRYPTO '97 (LNCS 1294), pp. 213–220.
30. C. Hall, I. Goldberg, and B. Schneier (1999), *Reaction attacks against several public-key cryptosystems*, in Proc. of the 2nd International Conference on Information and Communications Security (LNCS 1726), pp. 2–12.
31. C. M. Adams and H. Meijer (1988), *Security-related comments regarding McEliece's public-key cryptosystem*, in Advances in Cryptology–CRYPTO '87 (LNCS 293), pp. 224–228.
32. P. J. Lee and E. F. Brickell (1989), *An observation on the security of McEliece's public key cryptosystem*, in Advances in Cryptology–EUROCRYPT '88 (LNCS 330), pp. 275–280.
33. J. S. Leon (1988), *A probabilistic algorithm for computing minimum weights of large error-correcting codes*, IEEE Transactions on Information Theory, vol. 34, pp. 1354–1359.
34. J. Stern (1989), *A method for finding codewords of small weight*, in Coding Theory and Applications (LNCS 388), pp. 106–113.
35. F. Chabaud (1995), *On the security of some cryptosystems based on error-correcting codes*, in Advances in Cryptology–EUROCRYPT '94 (LNCS 950), pp. 131–139.
36. A. Canteaut and F. Chabaut (1998), *A new algorithm for finding minimum-weight words in a linear code: Application to primitive narrow-sense BCH-codes of length 511*, IEEE Transactions on Information Theory, vol. 44, no. 1, pp. 367–378.
37. Daniel J. Bernstein, Tanja Lange, Christiane Peters (2008), *Attacking and defending the McEliece cryptosystem*, in PQCrypto 2008 (LNCS 5299), pp. 31–46.
38. M. Finiasz and N. Sendrier (2009), *Security bounds for the design of code-based cryptosystems*, in Advances in Cryptology–ASIACRYPT 2009 (LNCS 5912), pp. 88–105.
39. A. Menezes, P. van Oorschot, and S. Vanstone (1996), *Handbook of Applied Cryptography*, CRC Press.
40. Peter W. Shor (1997), *Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer*, SIAM Journal on Computing, vol. 26, pp. 1484–1509.
41. Daniel J. Bernstein, Johannes Buchmann, Erik Dahmen (editors) (2009), *Post-Quantum Cryptography*, Springer.
42. K. Kobara and H. Imai (2001), *Semantically secure McEliece public-key cryptosystems–conversions for McEliece PKC*, in Practice and Theory in Public Key Cryptography–PKC '01 (LNCS 1992), pp. 19–35.
43. F. J. MacWilliams and N. J. A. Sloane (1977), *The Theory of Error-Correcting Codes*, North-Holland.
44. L. Yang (2005), *A public-key cryptosystem for quantum message transmission*, Proceedings of the SPIE 5631(1), pp. 233-236 and see also e-print arXiv:quant-ph/0310076.
45. M. Grassl, W. Geiselmann, and T. Beth (1999), *Quantum Reed-Solomon codes*, in Applied Algebra, Algebraic Algorithms and Error-Correcting Codes (LNCS 1719), pp. 231–244.

Appendix A. CSS Codes from GRS Codes

We will follow the notation and definition of a generalized Reed-Solomon (RS) code in [43]. Let $\mathbb{F}_{2^m}$ be the finite field with 2^m elements, where m is a positive integer. We first review the generalized RS codes over $\mathbb{F}_{2^m}$ and then present the construction of quantum RS codes using generalized RS codes. Let $1 \le K \le N \le 2^m$. Let $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_N)$, where the α_i are distinct elements of $\mathbb{F}_{2^m}$, and $\boldsymbol{v} = (v_1, v_2, \ldots, v_N)$, where the v_i are nonzero (but not necessarily distinct) elements of $\mathbb{F}_{2^m}$. Then the *generalized RS code*, denoted by $\mathrm{GRS}_K(\boldsymbol{\alpha}, \boldsymbol{v})$, consists of all vectors

$$(v_1 f(\alpha_1), v_2 f(\alpha_2), \ldots, v_N f(\alpha_N)), \tag{A.1}$$

where $f(x)$ ranges over all polynomials of degree $< K$ with coefficients from $\mathbb{F}_{2^m}$. $\mathrm{GRS}_K(\boldsymbol{\alpha}, \boldsymbol{v})$ is an $[N, K]$ linear code over $\mathbb{F}_{2^m}$. It can be shown that $\mathrm{GRS}_K(\boldsymbol{\alpha}, \boldsymbol{v})$ has minimum distance $D = N - K + 1$.

Lemma A.1 ([43]) *(a) The dual of* $\mathrm{GRS}_K(\boldsymbol{\alpha}, \boldsymbol{v})$ *is* $\mathrm{GRS}_{N-K}(\boldsymbol{\alpha}, \boldsymbol{v}')$ *for some* $\boldsymbol{v}'$.

(b) $\mathrm{GRS}_K(\boldsymbol{\alpha}, \boldsymbol{v}) = \mathrm{GRS}_K(\boldsymbol{\alpha}, \boldsymbol{w})$ *if and only if* $\boldsymbol{v} = \lambda \boldsymbol{w}$ *for some nonzero* $\lambda \in \mathbb{F}_{2^m}$.

Now we construct CSS codes from generalized RS codes. Our construction is a generalization of the Grassl-Geiselmann-Beth construction [45] of quantum RS codes using classical cyclic RS codes. Let $\mathcal{B} = \{\beta_1, \beta_2, \ldots, \beta_m\}$ be a basis of $\mathbb{F}_{2^m}$ over $\mathbb{F}_2$. Using the basis $\mathcal{B}$ each element α of $\mathbb{F}_{2^m}$ can be written uniquely as $\alpha = \sum_{j=1}^{m} a_j \beta_j$ for some $(a_1, a_2, \ldots, a_m) \in \mathbb{F}_2^m$. We define the map $\phi_{\mathcal{B}}$ from $\mathbb{F}_{2^m}$ to $\mathbb{F}_2^m$ by $\phi_{\mathcal{B}}(\alpha) = (a_1, a_2, \ldots, a_m)$. If no confusion arises, the subscript $\mathcal{B}$ will be omitted.

Definition A.1 ([45]) *The* binary expansion *of a code C of length N over $\mathbb{F}_{2^m}$ is a binary code obtained from C by replacing each component of each codeword $(c_1, c_2, \ldots, c_N)$ of C by its binary representation $(\phi(c_1), \phi(c_2), \ldots, \phi(c_N))$. The binary code obtained from C via ϕ is denoted by $\phi(C)$.*

The binary expansion $\phi(\mathrm{GRS}_K(\boldsymbol{\alpha}, \boldsymbol{v}))$ of $\mathrm{GRS}_K(\boldsymbol{\alpha}, \boldsymbol{v})$ with respect to a basis of $\mathbb{F}_{2^m}$ over $\mathbb{F}_2$ is a binary linear code of length mN, dimension mK and minimum distance $\ge D = N - K + 1$.

Remark A.1 *Let $\beta \in \mathbb{F}_{2^m}$. Multiplication by β in $\mathbb{F}_{2^m}$ induces a linear map on $\mathbb{F}_2^m$. Assuming that $\beta \cdot \beta_j = \sum_{i=1}^{m} b_{ij} \beta_i$ for some $b_{ij} \in \mathbb{F}_2$, we define $M(\beta) = [b_{ij}]$, an $m \times m$ matrix over $\mathbb{F}_2$. This is the so-called* matrix representation *of $\mathbb{F}_{2^m}$ over $\mathbb{F}_2$. Note that $M(\beta)$ depends on the basis of $\mathbb{F}_{2^m}$ over $\mathbb{F}_2$ used in the construction of ϕ. Then, for $\alpha \in \mathbb{F}_{2^m}$ we have $\phi(\alpha\beta) = \phi(\alpha) M(\beta)^T$. If* $\mathrm{GRS}_K(\boldsymbol{\alpha}, \boldsymbol{v})$ *has generator matrix $G = [g_{ij}]$ where $g_{ij} \in \mathbb{F}_{2^m}$, then the binary code* $\phi(\mathrm{GRS}_K(\boldsymbol{\alpha}, \boldsymbol{v}))$ *has generator matrix $\tilde{G} = [M(g_{ij})^T]$, which is obtained from G by replacing each entry g_{ij} by its corresponding $M(g_{ij})^T$. Similarly, if* $\mathrm{GRS}_K(\boldsymbol{\alpha}, \boldsymbol{v})$ *has parity check matrix $H = [h_{ij}]$ where $h_{ij} \in \mathbb{F}_{2^m}$, then the binary code* $\phi(\mathrm{GRS}_K(\boldsymbol{\alpha}, \boldsymbol{v}))$ *has parity check matrix $\tilde{H} = [M(h_{ij})]$.*

For any basis $\mathcal{B} = \{\beta_1, \beta_2, \ldots, \beta_m\}$ of $\mathbb{F}_{2^m}$ over $\mathbb{F}_2$, there exists a *dual basis* $\mathcal{B}^\perp = \{\beta_1', \beta_2', \ldots, \beta_m'\}$, i.e., $\mathrm{tr}(\beta_i \beta_j') = \delta_{ij}$, where $\mathrm{tr}(\cdot)$ is the trace from $\mathbb{F}_{2^m}$ to $\mathbb{F}_2$: $\mathrm{tr}(\alpha) = \sum_{i=0}^{m-1} \alpha^{2^i}$ for $\alpha \in \mathbb{F}_{2^m}$, and δ_{ij} is the Kronecker delta.

Lemma A.2 ([45]) *The dual of the binary expansion of a code C over $\mathbb{F}_{2^m}$ with respect to the basis $\mathcal{B}$ is equal to the binary expansion of the dual $C^\perp$ of C with respect to the dual basis $\mathcal{B}^\perp$, i.e.,*

$$\phi_{\mathcal{B}}(C)^\perp = \phi_{\mathcal{B}^\perp}(C^\perp). \tag{A.2}$$

Suppose now that $2K < N$. Consider two GRS codes, $\mathrm{GRS}_K(\boldsymbol{\alpha}, \boldsymbol{v})$ and $\mathrm{GRS}_{N-K}(\boldsymbol{\alpha}, \boldsymbol{v})$. We have the natural inclusion $\mathrm{GRS}_K(\boldsymbol{\alpha}, \boldsymbol{v}) \subset \mathrm{GRS}_{N-K}(\boldsymbol{\alpha}, \boldsymbol{v})$.

Lemma A.3 *If* $\mathrm{GRS}_K(\boldsymbol{\alpha}, \boldsymbol{v})^\perp = \mathrm{GRS}_{N-K}(\boldsymbol{\alpha}, \boldsymbol{v}')$ *for some* $\boldsymbol{v}'$*, then we have*

$$\mathrm{GRS}_{N-K}(\boldsymbol{\alpha}, \boldsymbol{v})^\perp = \mathrm{GRS}_K(\boldsymbol{\alpha}, \boldsymbol{v}'). \tag{A.3}$$

Proof. Consider

$$(v_1 f(\alpha_1), v_2 f(\alpha_2), \ldots, v_N f(\alpha_N)) \in \mathrm{GRS}_{N-K}(\boldsymbol{\alpha}, \boldsymbol{v}) \tag{A.4}$$

and

$$(v_1' g(\alpha_1), v_2' g(\alpha_2), \ldots, v_N' g(\alpha_N)) \in \mathrm{GRS}_K(\boldsymbol{\alpha}, \boldsymbol{v}'), \tag{A.5}$$

where $f(x)$ (resp. $g(x)$) is a polynomial of degree $< N - K$ (resp. $< K$) with coefficients from $\mathbb{F}_{2^m}$. The inner product of the two vectors is

$$\sum_{i=1}^{N} (v_i f(\alpha_i))(v_i' g(\alpha_i)) = \sum_{i=1}^{N} (v_i' f(\alpha_i))(v_i g(\alpha_i)) = 0, \tag{A.6}$$

where the last equality follows from the fact that

$$(v_1 g(\alpha_1), v_2 g(\alpha_2), \ldots, v_N g(\alpha_N)) \in \mathrm{GRS}_K(\boldsymbol{\alpha}, \boldsymbol{v}) \tag{A.7}$$

and

$$(v_1' f(\alpha_1), v_2' f(\alpha_2), \ldots, v_N' f(\alpha_N)) \in \mathrm{GRS}_{N-K}(\boldsymbol{\alpha}, \boldsymbol{v}') = \mathrm{GRS}_K(\boldsymbol{\alpha}, \boldsymbol{v})^\perp. \tag{A.8}$$

Hence we have $\mathrm{GRS}_K(\boldsymbol{\alpha}, \boldsymbol{v}') \subseteq \mathrm{GRS}_{N-K}(\boldsymbol{\alpha}, \boldsymbol{v})^\perp$. Since $\mathrm{GRS}_K(\boldsymbol{\alpha}, \boldsymbol{v}')$ and $\mathrm{GRS}_{N-K}(\boldsymbol{\alpha}, \boldsymbol{v})^\perp$ have the same dimension, we have $\mathrm{GRS}_K(\boldsymbol{\alpha}, \boldsymbol{v}') = \mathrm{GRS}_{N-K}(\boldsymbol{\alpha}, \boldsymbol{v})^\perp$. □.

To construct a CSS code we need a pair of binary linear codes. Let $C_1 = \phi_{\mathcal{B}}(\mathrm{GRS}_{N-K}(\boldsymbol{\alpha}, \boldsymbol{v}))$ and $C_2 = \phi_{\mathcal{B}}(\mathrm{GRS}_K(\boldsymbol{\alpha}, \boldsymbol{v}))$. Then we have $C_2 \subset C_1$. From Lemma A.1 (a) and Lemmas A.2 and A.3 we have that

$$C_1^\perp = \phi_{\mathcal{B}}(\mathrm{GRS}_{N-K}(\boldsymbol{\alpha}, \boldsymbol{v}))^\perp = \phi_{\mathcal{B}^\perp}(\mathrm{GRS}_{N-K}(\boldsymbol{\alpha}, \boldsymbol{v})^\perp) = \phi_{\mathcal{B}^\perp}(\mathrm{GRS}_K(\boldsymbol{\alpha}, \boldsymbol{v}')). \tag{A.9}$$

Similarly, we have $C_2^\perp = \phi_{\mathcal{B}^\perp}(\mathrm{GRS}_{N-K}(\boldsymbol{\alpha}, \boldsymbol{v}'))$. Note that C_1 and $C_2^\perp$ have minimum distance $\geq K + 1$. The pair of binary codes, C_1 and C_2, defines a CSS code with parameters $[[n = mN, k = m(N - 2K), d \geq K + 1]]$. From Lemma A.1 (b), for a fixed $\boldsymbol{\alpha}$ there are $(2^m - 1)^{N-1}$ ways of choosing $\boldsymbol{v}$. So we have a family of $(2^m - 1)^{N-1}$ CSS codes with parameters $[[n = mN, k = m(N - 2K), d \geq K + 1]]$.