Efficient Code Based Cryptosystem with Dual Inverse Matrix

Farshid, Haidary Makoui[1], T. Aaron, Gulliver[1], and Mohammad Dakhilalian[2]

[1]*Department of Electrical and Computer Engineering, University of Victoria, Victoria, B.C., Canada. email: makoui@uvic.ca and agullive@ece.uvic.ca*

[2]*Department of Electrical and Computer Engineering, Isfahan University of Technology, Isfahan, Iran. email: mdalian@iut.ac.ir*

Abstract

The security of cryptographic primitives is an important issue. The Shor algorithm illustrates how quantum attacks threaten the security of these widely used primitives. Code-based cryptography is one of several approaches resistant to quantum attacks. To date, no attack has been able to break a code-based cryptosystem in polynomial time. Despite this level of security, these cryptosystems have not been considered for practical applications such as e-commerce, medical and industrial IoT, finance, blockchain, mobile services, and online banking. The main reason is the large public and private key sizes. This paper presents a new code-based cryptosystem based on inverse parity check matrices. The dual matrix provides both a parity check matrix transpose and a parity check matrix inverse. These are employed in the key generation, encryption, and decryption algorithms. The proposed scheme provides public and private key sizes smaller than the McEliece cryptosystem and has a higher level of security.

1 Introduction

Post-quantum cryptography [1] involves the development of cryptographic mechanisms [2–4] which are secure against quantum attacks. This is important because the Shor algorithm indicates that quantum attacks are a serious threat to cryptographic primitives [5]. Code-based cryptographic primitives [6] have been shown to

be resistant to quantum attacks. The first code-based cryptosystem was introduced by McEliece and is known as the McEliece cryptosystem [7]. The security of this cryptosystem is based on the hardness of the decoding and code distinguishability problems [8, 9]. The inability to distinguish between a scrambled parity check matrix and a random one is an NP-problem [9, 10], so decoding a linear code without knowledge of its algebraic structure is also an NP-problem [13].

This paper presents a code-based cryptosystem based on the McEliece cryptosystem. It employs a dual inverse matrix A in the key generation, encryption, and decryption algorithms. The key generation algorithm constructs public and private keys using A. The main advantage of the proposed approach is smaller public and private keys than the McEliece scheme. This addresses the main drawback of the McEliece cryptosystem and makes the proposed scheme suitable for applications in finance, medicine, and other areas.

1.1 Linear Block Codes

This section presents the required background on linear block codes. In communication systems, binary codes are commonly employed with redundant bits added to message bits to detect and correct errors. The encoder assigns a codeword $\boldsymbol{c} = (c_1, c_2, \ldots, c_n)$ to a message $\boldsymbol{m} = (m_1, m_2, \ldots, m_k)$. Thus, there are 2^k distinct messages and the corresponding 2^k codewords are referred to as a $C(n, k)$ block code. The length of this code is n and the dimension is k, $k \leq n$.

A block code is linear if its codewords form a k-dimensional vector subspace of the n-dimensional vector space. A set k linearly independent codewords $\boldsymbol{g}_1, \boldsymbol{g}_2, \ldots, \boldsymbol{g}_k$ defines a generator matrix G for the code. A systematic generator matrix has the form

$$G_{k\times n} = (I_k | P_{k\times(n-k)}), \tag{1}$$

where I_k is the $k \times k$ identity matrix. For every linear block code, there is a dual code denoted $C^{\perp}$ which is the $n-k$ dimensional dual space with generator matrix H. The matrix H is also called the parity check matrix of $C(n, k)$. It is an $(n-k) \times n$ matrix such that $GH^T = \mathbf{0}$ where ^{T} denotes transpose. A systematic parity check matrix has the form

$$H_{(n-k)\times n} = (P^T_{(n-k)\times k} | I_{n-k}). \tag{2}$$

1.2 The McEliece Cryptosystem

In 1978, McEliece introduced the first code-based cryptosystem and so it is called the McEliece cryptosystem [7]. In this cryptosystem, the plaintext bits are scrambled

and the corresponding codeword is permuted. Then up to t bits are flipped where t is the error correcting capability of the code. This is a public key cryptosystem where the public key is the product of a non-singular $k \times k$ scrambling matrix S, a $k \times n$ generator matrix of the code G, and an $n \times n$ permutation matrix P. The private key consists of these three matrices $pr_k = (S, G, P)$ and the public key is $pk = SGP$. The encryption and decryption algorithms are as follows.

Encryption Algorithm

1. For a plaintext $\boldsymbol{m}$ of length k, Alice uses Bob's public key to encode it as $\boldsymbol{c} = \boldsymbol{m}SGP$.

2. Next, she flips some of the bits of $\boldsymbol{c}$ by selecting a random vector $\boldsymbol{e}$ of length n such that $w(\boldsymbol{e}) \leq t$ where t is the error correcting capability of the code and $w(\cdot)$ denotes the Hamming weight. The ciphertext is

$$\boldsymbol{c'} = \boldsymbol{c} + \boldsymbol{e} = \boldsymbol{m}SGP + \boldsymbol{e}. \tag{3}$$

Decryption Algorithm

1. For a ciphertext $\boldsymbol{c'}$, find P^{-1} using the private key. Then multiply $\boldsymbol{c'}$ by P^{-1} to obtain

$$\boldsymbol{c'}P^{-1} = (\boldsymbol{m}SGP + \boldsymbol{e})P^{-1} = \boldsymbol{m}SG + \boldsymbol{e}P^{-1}. \tag{4}$$

2. As P is a permutation matrix, $P^{-1} = P^T$ is also a permutation matrix. Therefore, $\boldsymbol{e}P^{-1}$ is a vector with the same weight as $\boldsymbol{e}$. Thus $\boldsymbol{c'}P^{-1}$ can be decoded to obtain $\boldsymbol{m}S$.

3. Multiply $\boldsymbol{m}S$ by S^{-1} to obtain the plaintext $\boldsymbol{m}$.

In [12], it was shown that the probability of a successful ciphertext distinguishability attack against the McEliece cryptosystem is $\binom{n-t}{k}/\binom{n}{k}$. Therefore, the parameters $n = 1024$, $t = 50$, and $k \geq 524$ were recommended.

2 Dual Matrix A

Consider a matrix A such that $HA = I_{n-k}$ and $GA = \mathbf{0}$. Thus, A is an inverse parity check matrix and the transpose of a parity check matrix, so that $GH^T = GA$. Hence

A can be constructed using H^T and a non-singular matrix P' that satisfies $A = H^T P'$. Then

$$GA = \mathbf{0} \text{ and } GH^T = \mathbf{0},$$

so $A = H^T P'$ and

$$HA = H(H^T P') = (HH^T)P' = I_{n-k}.$$

Thus, $P' = (HH^T)^{-1}$ and A can be constructed only if the $(n-k) \times (n-k)$ matrix HH^T is non-singular. Let p_A denote the number of possible linear combinations of column vectors of A. Then

$$p_A = \prod_{i=0}^{n-k-1} (2^{n-k} - 2^i). \tag{5}$$

For example, the number of linear combinations of the column vectors of a dual inverse matrix A with $n - k = 3$ is

$$p_A = \prod_{i=0}^{3-1} (2^3 - 2^i) = (2^3 - 2^0) \times (2^3 - 2^1) \times (2^3 - 2^2) = 168.$$

3 Code-Based Cryptosystem Using A

In the proposed code-based cryptosystem, the dual matrix A is used in the key generation, encryption and decryption algorithms.

Proposed Code-based Cryptosystem Algorithms

---

1. Key Generation: $(pk, pr_k) \leftarrow Gen(\lambda)$, where λ denotes the key generation scheme.
2. Encryption: $\boldsymbol{c'} \leftarrow Enc(\boldsymbol{m}, pk)$, where $\boldsymbol{c'}$ and $\boldsymbol{m}$ denote the encrypted message (cipher) and message, respectively.
3. Decryption: $\boldsymbol{m} \leftarrow Dec(\boldsymbol{c'}, pr_k)$.

3.1 Key Generation

The key generation algorithm provides public and private keys using the generator matrix G of the code $C(n, k)$ and the dual matrix A. The generator matrix and parity check matrix are modified via scrambling. This provides resistance against structural attacks.

The following matrices are used by the key generation algorithm.

1. G, a generator matrix with dimensions $k \times n$.
2. H, a parity check matrix with dimensions $(n-k) \times n$.
3. A, a dual matrix with dimensions $n \times (n-k)$.
4. S, a non-singular scrambling matrix with dimensions $k \times k$.
5. P, a non-singular matrix with dimensions $n \times n$.
6. L, a non-singular matrix with dimensions $(n-k) \times (n-k)$.

Key Generation Algorithm $Gen(\lambda)$

---

1. Given the generator matrix G with non-singular HH^T.
2. Construct $P' = (HH^T)^{-1}$.
3. Public key: $pk \leftarrow (SGP, L^{-1}HP, P^{-1}AHP)$.
4. Private key: $pr_k \leftarrow (S^{-1}, P^{-1}, G, P^{-1}AL)$.

The dual matrix A is masked by a non-singular matrix L and a non-singular matrix P.

Theorem 1. *The public key $L^{-1}HP$ has many inverses, and the probability of constructing a particular inverse of $L^{-1}HP$ is trivial.*

Proof. The parity check matrix H is a full rank matrix and is not unique [11]. The inverse of H has $n-k$ columns, each of which can have 2^k different values, so the number of valid inverse matrices is $2^{k\times(n-k)}$ [11]. Therefore, the public key $L^{-1}HP$ is also a full rank matrix, hence the probability of constructing a particular inverse of the public key $L^{-1}HP$ is $\frac{1}{2^{k\times(n-k)}}$, which is negligible for an appropriate choice of parameters. □

3.2 Encryption Algorithm

The encryption algorithm transforms the message (plaintext) into ciphertext.

Encryption Algorithm $Enc(\boldsymbol{m}, pk)$

---

1. Encode a given plaintext $\boldsymbol{m}$ using the public key SGP

$$\boldsymbol{c} \leftarrow \boldsymbol{m}(SGP).$$

2. Let $\boldsymbol{s}$ denote a random $n-k$ bit vector

$$\boldsymbol{s} \leftarrow \text{ a random } n-k \text{ bit vector.}$$

3. Use the public key to construct $\boldsymbol{s}(L^{-1}HP)$

$$\boldsymbol{e} \leftarrow \boldsymbol{s}(L^{-1}HP).$$

4. Construct the ciphertext corresponding to $\boldsymbol{m}$

$$\boldsymbol{c'} \leftarrow \boldsymbol{c} + \boldsymbol{e} = \boldsymbol{m}(SGP) + \boldsymbol{s}(L^{-1}HP).$$

The error vector $\boldsymbol{e}$ can have weight 0 to n. Thus, the weight is not dependent on the error correction capability t of the code $C(n,k)$. Details are provided in Section 3.5.

3.3 Decryption Algorithm

The decryption algorithm decodes the ciphertext to obtain the plaintext.

Decryption Algorithm $Dec(\boldsymbol{c'}, pr_k)$

---

1. Find the vector $\boldsymbol{s}$ by multiplying the received ciphertext $\boldsymbol{c'}$ with the private key

$$\boldsymbol{s} \leftarrow \boldsymbol{c'}(P^{-1}AL).$$

$$\begin{aligned}
\boldsymbol{c'}(P^{-1}AL) &= [\boldsymbol{m}(SGP) + \boldsymbol{s}(L^{-1}HP)](P^{-1}AL) \\
\boldsymbol{c'}(P^{-1}AL) &= \boldsymbol{m}(SGP)(P^{-1}AL) + \boldsymbol{s}(L^{-1}HP)(P^{-1}AL) \\
\boldsymbol{c'}(P^{-1}AL) &= \boldsymbol{m}S(GA)L + \boldsymbol{s}L^{-1}(HA)L \\
\boldsymbol{c'}(P^{-1}AL) &= \boldsymbol{0} + \boldsymbol{s}(\boldsymbol{I}) \\
\boldsymbol{c'}(P^{-1}AL) &= \boldsymbol{s}
\end{aligned}$$

2. Construct $\boldsymbol{s}(L^{-1}HP)$ using the public key

$$\boldsymbol{s}(L^{-1}HP) \leftarrow \boldsymbol{s} \text{ and } (L^{-1}HP).$$

3. Find the codeword $\boldsymbol{c}$

$$\boldsymbol{c} \leftarrow \boldsymbol{c'} + \boldsymbol{s}(L^{-1}HP).$$

4. Decode the codeword $\boldsymbol{c}$ to obtain the plaintext $\boldsymbol{m}$

$$\begin{array}{rcl} \boldsymbol{m}SG & \leftarrow & (\boldsymbol{m}SGP)(P^{-1}) \\ \boldsymbol{m}S & \leftarrow & \text{decode } \boldsymbol{m}SG \\ \boldsymbol{m} & \leftarrow & (\boldsymbol{m}S)(S^{-1}). \end{array}$$

3.4 Example

Consider the generator matrix

$$G = (I_k | P_{k\times(n-k)}) = \left(\begin{array}{c|ccccccc} & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ I_k & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ & 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{array}\right),$$

with parity check matrix and dual matrix given by

$$H_{(n-k)\times n} = \left(\begin{array}{ccccc|c} 1 & 0 & 0 & 0 & 1 & \\ 0 & 1 & 0 & 1 & 1 & \\ 0 & 0 & 1 & 0 & 0 & \\ 1 & 0 & 1 & 1 & 1 & I_{n-k} \\ 0 & 1 & 1 & 0 & 0 & \\ 1 & 0 & 1 & 1 & 0 & \\ 1 & 0 & 1 & 0 & 1 & \end{array}\right) A_{n\times(n-k)} = \begin{pmatrix} 0 & 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 \end{pmatrix}.$$

The non-singular matrix L and matrix S are

$$L_{(n-k)\times(n-k)} = \begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix} S_{k\times k} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 \end{pmatrix}$$

and the non-singular matrix P is

$$P_{n\times n} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \end{pmatrix}$$

1. Bob encrypts the message $\boldsymbol{m} = 11010$ using Alice's public key SGP to obtain the ciphertext $\boldsymbol{c} = \boldsymbol{m}SGP = 110011101101$.

2. Bob randomly chooses an $n-k$ bit vector $\boldsymbol{s} = 1001011$.

3. He constructs an error pattern using $\boldsymbol{s}$ and Alice's public key $L^{-1}HP$

$$\boldsymbol{e} = \boldsymbol{s}(L^{-1}HP) = 1001011(L^{-1}HP) = 000000111011.$$

4. Bob constructs the codeword

$$\boldsymbol{c}^{'} = \boldsymbol{m}SGP + \boldsymbol{s}(L^{-1}HP) = 110011101101 + 000000111011 = 110011010110.$$

Bob transmits $\boldsymbol{c}^{'} = 110011010110$.

Alice receives the codeword and decrypts it as follows.

1. Alice uses her private key to find

$$\boldsymbol{s} = \boldsymbol{c'}(P^{-1}AL) = 110011010110(P^{-1}AL) = 1001011.$$

2. Using $\boldsymbol{s}$, Alice uses her public key $L^{-1}HP$ to construct

$$\boldsymbol{s}(L^{-1}HP) = 1001011(L^{-1}HP) = 000000111011.$$

3. Alice finds the codeword

$$\boldsymbol{c} = \boldsymbol{c}^{'} + \boldsymbol{s}(L^{-1}HP) = 110011010110 + 000000111011 = 110011101101.$$

4. Alice uses her private key P^{-1} to find

$$\boldsymbol{m}(SG) = \boldsymbol{m}(SGP)(P^{-1}) = 110011101101(P^{-1}) = 110110101100.$$

5. She decodes $\boldsymbol{m}(SG)$ to find

$$\boldsymbol{m}(S) = 11011.$$

6. She uses her private key S^{-1} to obtain

$$\boldsymbol{m} = \boldsymbol{m}(S)(S^{-1}) = 11011(S^{-1}) = 11010.$$

The following subsection examines the performance and security of the proposed scheme.

3.5 Performance and Security Analysis

There are two types of attacks on the McEliece cryptosystem, structural attacks and ciphertext distinguishability attacks. In a structural attack, an adversary tries to break the public key and find the generator matrix and private key. A distinguishability attack tries to recover the plaintext from a given ciphertext. The proposed algorithm masks the generator matrix using the scrambling matrix. In addition, from Theorem 1 the probability of breaking the public key and constructing the private key is negligible. Therefore, the proposed code-based cryptosystem is secure from structural attacks. As previously mentioned, the probability of a successful ciphertext distinguishability attack against the McEliece cryptosystem is $\binom{n-t}{k}/\binom{n}{k}$, which

is negligible. However, this depends on the error correction capability of the code t. Therefore, McEliece suggested using a Goppa code to make the probability of a ciphertext distinguishability attack negligible. The corresponding large key sizes affect the performance of the McEliece cryptosystem and limit the practical applications.

The error pattern $\boldsymbol{e} \in F_2^n$ whereas $\boldsymbol{s} \in F_2^{n-k}$. The proposed encryption algorithm selects an $n-k$ bit random vector $\boldsymbol{s}$ to obtain an n bit error pattern (steps 2 and 3 in the encryption algorithm $Enc(\boldsymbol{m}, pk)$). It also employs $L^{-1}HP$ with distinguishability probability $2^{-k(n-k)}$.

Consider an adversary that randomly selects an inverse of the public key as the private key, $(L^{-1}HP)^{-1} = (P^{-1}H^{-1}L)$. Then to decrypt the message, the ciphertext should be multiplied by the private key so that

$$\begin{aligned}
\boldsymbol{c'} &= \boldsymbol{m}(SGP) + \boldsymbol{s}(L^{-1}HP) \\
\boldsymbol{c'}(P^{-1}H^{-1}L) &= [\boldsymbol{m}(SGP) + \boldsymbol{s}(L^{-1}HP)](P^{-1}H^{-1}L) \\
\boldsymbol{c'}(P^{-1}H^{-1}L) &= \boldsymbol{m}(SGP)(P^{-1}H^{-1}L) + \boldsymbol{s}(L^{-1}HP)(P^{-1}H^{-1}L) \\
\boldsymbol{c'}(P^{-1}H^{-1}L) &= \boldsymbol{m}S(GH^{-1})L + \boldsymbol{s}L^{-1}(HH^{-1})L \\
\boldsymbol{c'}(P^{-1}H^{-1}L) &= \boldsymbol{m}(SGH^{-1}L) + \boldsymbol{s}(\boldsymbol{I}) \\
\boldsymbol{s} &= \boldsymbol{c'}(P^{-1}H^{-1}L) + \boldsymbol{m}(SGH^{-1}L).
\end{aligned}$$

Therefore, $\boldsymbol{m}(SGH^{-1}L) \neq \boldsymbol{0}$ and so $\boldsymbol{s}$ cannot be obtained. The decryption algorithm can construct $\boldsymbol{s}$ if and only if the randomly selected inverse of the parity check matrix is equal to A such that $GH^{-1} = GA = \boldsymbol{0}$. From Theorem 1, the probability of finding a specific inverse of the parity check matrix is negligible. Therefore, the probability of a successful ciphertext distinguishability attack against the proposed scheme is

$$Pr[(Adv, \gamma) = 1] = \frac{1}{2^{k(n-k)}} << \binom{n-t}{k} / \binom{n}{k} = \epsilon(\gamma),$$

so the security is not dependent on the error correction capability of the code. Therefore, the proposed schemes provide a higher level of security with no dependency on the error correction capability of the code.

It is suggested that $n^{'} = 256$ and $k^{'} = 128$ where $n - k \geq 128$. In this case, the public and private key sizes are much smaller than those of the McEliece cryptosystem. Conversely, a legitimate receiver can easily decode the given codeword, find the random value, and construct the plaintext.

It has been suggested that the McEliece cryptosystem employ a Goppa code with $n = 1024$ and $k = 524$. The public key SGP has dimensions $k \times n$ and the private key matrices S, G, and P have dimensions $k \times k$, $k \times n$, and $n \times n$, respectively. In total, this is $(n+k)^2 = 9 \times 2^{18}$ bits or 288 kB. The proposed scheme employs a code with

dimensions $n^{'} = 256$ and $k^{'} = 128$. The public key matrices SGP and $L^{-1}HP$ have dimensions $k^{'} \times n^{'}$ and $(n^{'} - k^{'}) \times n^{'}$, respectively. The private key matrices S, G, P, and $P^{-1}AL$ have dimensions $k \times k$, $k \times k$, $n \times n$, and $(n^{'} - k^{'}) \times n^{'}$, respectively. In total, this is $3n^{'2} + k^{'2} = 13 \times 2^{14}$ bits or 26 kB.

4 Conclusion

The McEliece cryptosystem has drawbacks such as dependency on the error correction capability of the code. In addition, the use of binary Goppa code results on a large key size which limits its applicability. Thus a new code-based cryptosystem algorithm was proposed with key generation, encryption, and decryption algorithms based on a dual matrix A. It was shown that this cryptosystem is secure against structural and ciphertext distinguishability attacks, and the security is better than that of the McEliece cryptosystem. Further, the security of the proposed cryptosystem is independent of the error correction capability of the code. It was demonstrated that this cryptosystem has 26 kB public and private key sizes compared to 288 kB with the McEliece cryptosystem.

References

[1] M. Baldi, "Post-quantum cryptographic schemes based on codes," in *Proc. Int. Conf. on High Perf. Computing and Simulation*, Genoa, Italy, pp. 908–910, 2017.

[2] T. N. R. Rao, K. H. Nam, "A private key algebraic coded cryptosystem," in *Advances in Cryptology, Proc. CRYPTO*, Lecture Notes in Computer Science, vol. 263, pp. 35–48, 1986.

[3] R. Hooshmand, M. R. Aref, "Efficient secure channel coding scheme based on low-density lattice codes," *IET Commun.*, vol. 10, no. 11, pp. 1365–1373, 2016.

[4] T. N. R. Rao, "Joint encryption and error correction schemes," in *Proc. Int. Symp. on Computer Architecture*, Ann Arbor, MI, USA, pp. 240–241, 1984.

[5] P. W. Shor, "Algorithms for quantum computation: Discrete logarithms and factoring," in *Proc. Annual Symp. on Foundations of Computer Science*, Santa Fe, NM, USA, pp. 124–134, 1994.

[6] N. Sendrier, "Code based cryptography: State of the art and perspectives," *IEEE Privacy Security*, vol. 15, no. 4, pp. 44–50, 2017.

[7] R. J. McEliece, "A public-key cryptosystem based on algebraic coding theory," *Jet Propulsion Lab.*, DSN Tech. Rep. 42-44, pp. 114–116, 1978.

[8] P. L. Cayrel, M. Meziani, "Post-quantum cryptography: Code-based signatures," in *Proc. Advances in Computer Science and Inform. Technology*, pp. 82–99, 2013.

[9] R. Nojima, H. Imai, K. Kobara, K. Morozov, "Semantic security for the McEliece cryptosystem without random oracles," *Design, Codes, Cryptogr.*, vol. 49, no. 1-3, pp. 289–305, 2008.

[10] P. L. Cayrel, P. Gaborit, M. Girault, "Identity based identification and signature schemes using correcting codes," in *Proc. Int. Workshop on Coding and Cryptogr.*, pp. 69–78, 2007.

[11] M. Esmaeili, *Application of Linear Block Codes in Cryptography*, Ph.D. Dissertation, Department of Electrical and Computer Engineering, University of Victoria, Victoria, BC, Canada, 2019.

[12] A. J. Menezes, P. C. van Oorschot, S. A. Vanstone, "Public-key encryption," Ch. 8 in *Handbook of Applied Cryptography*, CRC Press, Boca Raton, FL, USA, 1997.

[13] E. R. Berlekamp, R. J. McEliece, H. C. A. van Tilborg, "On the inherent intractability of certain coding problems," *IEEE Trans. Inform. Theory*, vol. 24, no. 3, pp. 384–386, 1978.