ORIGINAL PAPER

SYMMETRIC ENCRYPTION FROM CYCLIC CODES OVER NON-CHAIN RINGS

RABIA DERTLI[1], ŞENOL EREN[1]

Manuscript received: 14.04.2025; Accepted paper: 23.07.2025;
Published online: 30.09.2025.

Abstract. *This paper introduces a novel encryption scheme constructed over the finite commutative ring* $S = \mathbb{F}_2 + u\mathbb{F}_2 + v\mathbb{F}_2$, *where the indeterminates satisfy* $u^2 = u, v^2 = v, uv = vu = 0$. *Inspired by the principles of the one-time pad cryptosystem, the proposed method employs a unique, random key of equal length to the plaintext, ensuring theoretical perfect secrecy. A key feature of this design is its resistance to ciphertext-only attacks, as multiple plaintexts may map to the same ciphertext under different key instances. To assess the scheme's security against quantum adversaries, we model the key-recovery process and analyze its complexity under Grover's quantum search algorithm, which offers a quadratic speed-up compared to classical brute-force techniques. Experimental simulations in Python, using NumPy and Matplotlib, illustrate the performance gap between classical and quantum search across rings with varying cardinalities. The results reveal that while brute-force attacks quickly become infeasible as the key space grows, Grover's algorithm maintains computational viability. Overall, this study highlights the potential of integrating algebraic ring structures with post-quantum cryptographic analysis, offering a promising avenue for secure communication in quantum-aware security environments.*

Keywords: *Cyclic codes; One-time pad; Grover's algorithm; Encryption scheme.*

1. INTRODUCTION

Cryptology, the science of securing communication, has been a field of interest for researchers for centuries. Its significance is particularly evident in the military domain, where information security and confidentiality are of paramount importance. The foundations of modern cryptography were laid by Gilbert Vernam, who introduced the Vernam cipher, also known as the One-Time Pad (OTP) method [1]. In 1949, Claude Shannon rigorously proved that the OTP method provides perfect secrecy, provided that the key is randomly generated, used only once, and kept completely secret [2]. The One-Time Pad (OTP) encryption scheme operates by combining binary plaintext with a binary key of equal length. Let w represent the binary plaintext, k the binary key, and c the resulting ciphertext. The encryption and decryption processes are formally defined as:

$$c = w + k \text{ and } w = c + k, [3].$$

Due to the inherent randomness and uniqueness of the key, the system becomes computationally unbreakable. Even if an adversary intercepts the ciphertext and attempts to decode it by trying all possible keys, multiple meaningful plaintexts may emerge, making it

[1] Ondokuz Mayıs University, Faculty of Science, Department of Mathematics, 55139 Samsun, Turkey.
E-mail: rabia.alim06@gmail.com; seren@omu.edu.tr.

https://doi.org/10.46939/J.Sci.Arts-25.3-a08

practically impossible to determine the correct message without knowing the exact key. As a result, many researchers have utilized the OTP method to enhance the security of encryption systems. Çalkavur and Güzeltepe [4] applied this encryption scheme based on cyclic codes over the ring $\mathbb{F}_2 + v\mathbb{F}_2$ and developed a secure encryption method.

1.1. CYCLIC CODES AND THEIR APPLICATIONS IN CRYPTOGRAPHY

Cyclic codes, a prominent class of linear codes, play a significant role in error correction and cryptographic applications due to their inherent algebraic structure. These codes possess a cyclic property that simplifies their encoding and decoding processes, making them suitable for high-performance communication systems. Binary cyclic codes were first introduced by Prange in 1962 [5], and since then, they have been extensively studied and applied in various cryptographic frameworks. Among the many variations, negacyclic, constacyclic, quasi-cyclic, and skew-cyclic codes have been investigated to improve minimum distance and error correction capacity.

Recent studies have explored the use of cyclic codes over different algebraic structures, including finite commutative rings. For instance, Abualrub et al. studied skew cyclic codes over the ring [6]. Yildiz and Karadeniz [7] studied the codes over the ring $\mathbb{F}_2 + u\mathbb{F}_2 + v\mathbb{F}_2 + uv\mathbb{F}_2,\ u^2 = 0, v^2 = 0, uv = vu$. Dertli [8] studied codes over the ring $\mathbb{F}_2 + u\mathbb{F}_2 + v\mathbb{F}_2, u^2 = u, v^2 = v, uv = vu = 0$.

1.2. MOTIVATION AND CONTRIBUTION OF THIS STUDY

Inspired by these developments, this study proposes a new secure encryption scheme based on cyclic codes defined over the ring $S = \mathbb{F}_2 + u\mathbb{F}_2 + v\mathbb{F}_2$ where the relations $u^2 = u, v^2 = v, uv = vu = 0$ hold. The proposed scheme leverages the principles of the One-Time Pad method, ensuring theoretically perfect secrecy by using a randomly generated key of the same length as the plaintext that is employed only once. Additionally, the cyclic structure of the underlying ring enhances the security and efficiency of the encryption scheme by offering an increased minimum distance, thereby improving error correction and robustness against attacks.

To assess the cryptanalytic resilience of the proposed system, we apply Grover's quantum search algorithm, which provides a quadratic speed-up over classical brute-force methods. Grover's algorithm, introduced in 1996, efficiently reduces the search space size from $O(N)$ in classical brute-force to $O\sqrt{N})$ iterations. This algorithm has been shown to significantly reduce the time required to perform key recovery, making it a crucial tool in the post-quantum cryptanalysis of symmetric encryption schemes. Our analysis involves modeling the key recovery problem within the defined ring structure and conducting Python-based simulations to compare the performance of classical and quantum approaches. The results demonstrate that while classical brute-force becomes computationally impractical for large key spaces, Grover's algorithm maintains computational efficiency, particularly for rings with larger cardinality. The rest of the paper is structured as follows:

Section 2 provides the fundamental definitions and theoretical preliminaries. Section 3 introduces the proposed encryption scheme based on cyclic codes defined over the specified ring. Section 4 analyzes the security of the system and investigates potential attacks. Finally, Section 5 concludes the paper by presenting the results of simulations and comparisons between classical and quantum approaches and suggests possible directions for future research.

2. PRELIMINARIES

The definitions and theorem in this section are preliminary for a better understanding of the subject. From now on, S is defined as the ring $_2 + u\mathbb{F}_2 + v\mathbb{F}_2$, where $u^2 = u, v^2 = v, uv = vu = 0$.

2.1. CYCLIC CODES

Definition 2.1.1. [9] Let $\mathbb{F}_q$ is a finite field of order q. A linear code C of length n over $\mathbb{F}_q$ is a subspace of ${\mathbb{F}_q}^n$.

Definition 2.1.2. [10] A code C is cyclic if C is a linear code and any cyclic shift of a codeword is also a codeword, when ever $(a_0, a_1, \ldots, a_{n-1})$ is in C, then so is $(a_{n-1}, a_1, \ldots, a_{n-2})$. Let $\mathbb{F}_q$ be the set of polynomials in x whose coefficients are from the field $\mathbb{F}_q$. It is convenient to think of cyclic codes as consisting of polynomials as well as codewords. With every word $(a_0, a_1, \ldots, a_{n-1}) \in {\mathbb{F}_q}^n$ we can write the polynomial of degree less than n, $a(x) = a_0 + a_1 x + \cdots + a_i x^i + \cdots + a_{n-1}x^{n-1} \in \mathbb{F}_q[x]$. We know that every codeword can be written as a polynomial. Thus, each cyclic shift of a codeword is also expressed as a polynomial. Let $c(x)$ is a code polynomial and c' is the shifted codeword $c'(x) = c_{n-1} + c_0 x + c_1 x^2 + \cdots + c_i x^{i+1} + \cdots + c_{n-2}x^{n-1}$. Thus $c'(x)$ is equal to the product polynomial . More precisely, $c'(x) = xc(x) - c_{n-1}(x^n - 1)$. This means $c'(x)$ and $xc(x)$ are equal to polynomials in the ring $\mathbb{F}_q[x](mod\, x^n - 1)$. If $f(x)$ is any polynomial of $\mathbb{F}_q[x]$ whose remainder upon division by $x^n - 1$, belongs to C, then we may write $f(x) \in C(mod\, x^n - 1)$. Since each cyclic shift belongs to the cyclic code C, we can write $x^i c(x) \in C(mod\, x^n - 1)$ and indeed. The below statement is used to convert the structure of cyclic code into an algebraic one.

$$\theta : {\mathbb{F}_q}^n \to \mathbb{F}_q[x]/(x^n - 1)$$
$$(a_0, a_1, \ldots, a_{n-1}) \to a_0 + a_1 x + \ldots + a_{n-1}x^{n-1}$$

where the set of polynomials in x with coefficient in $\mathbb{F}_q$ is denoted by $\mathbb{F}_q[x]$.

Theorem 2.1.3. [9] Let θ be the linear map defined above. Then, any nonempty subset C of ${\mathbb{F}_q}^n$ is a cyclic code if and only if $\theta(C)$ is an ideal of $\mathbb{F}_q[x]/(x^n - 1)$. There is a relationship between the cyclic codes in ${\mathbb{F}_q}^{\mathrm{n}}$ and ideals of the ring $\mathbb{F}_q[x]/(x^n - 1)$. Let be a cyclic code of length n, where $g(x) = g_0 + g_1 x + \ldots + g_r x^r$ and $x^n - 1$ is divisible by $g(x)$. The code C can be expressed as follows: $C = \{a_i x g(x) : a_i \in \mathbb{F}_q[x]/(x^n - 1), deg(a_i(x)) < n - r\}$, where $i = p^{n-r}$.

Definition 2.1.4. [9] Let u be a word in ${\mathbb{F}_q}^n$ as $u = (u_1, u_2, \ldots, u_n)$. The number of nonzero coordinates of u is called the Hamming weight of u and is defined as follows:

$$w_H(u) = \begin{cases} 1 \; If \; u \neq 0 \\ 0 \; If \; u = 0 \end{cases}.$$

2.2. LINEAR CODES OVER S

The ring S was also defined by Dertli and Çengellenmiş [8]. $S = \mathbb{F}_2 + u\mathbb{F}_2 + v\mathbb{F}_2$ is a finite commutative ring with 8 elements, where $u^2 = u, v^2 = v, uv = vu = 0$ and $\mathbb{F}_2 = \{0,1\}$.

$$\begin{aligned} S \;&\cong \frac{\mathbb{F}_2[u,v]}{\langle u^2 - u, v^2 - v, uv = vu \rangle} \\ S =\; &\{a + ub + vc | a,b,c,d \in \mathbb{F}_2\} \\ =\; &\{0,1,u,v,1+u,1+v,u+v,1+u+v\}\,. \end{aligned}$$

S has four ideals such that:

$$\begin{aligned} I_0 \;&=\; \{0\}, I_1 = S \\ I_u \;&=\; \{0,u\}, I_v = \{0,v\}, I_{1+u+v} = \{0,1+u+v\} \\ I_{u+v} \;&=\; \{0,u,v,u+v\}, I_{1+u} = \{0,v,1+u,1+u+v\} \\ I_{1+v} \;&=\; \{0,u,1+v,1+u+v\}. \end{aligned}$$

Definition 2.2.1. [8] A linear code C over S length n is a S-submodule of S_n.

Definition 2.2.2. [8] For $a + ub + uc \in S$

$$\begin{aligned} \varphi :\; &S \to {\mathbb{F}_2}^3 \\ \varphi(a + ub + vc) \;=\; &(a, a+b, a+c) \end{aligned}$$

is defined as a Gray map. The Gray map of the elements is defined as,

$$\begin{aligned} \varphi(0) \;&=\; (000), \varphi(1+u) = (101), \\ \varphi(1) \;&=\; (111), \varphi(1+v) = (110), \\ \varphi(u) \;&=\; (010), \varphi(u+v) = (011), \\ \varphi(v) \;&=\; (001), \varphi(1+u+v) = (100). \end{aligned}$$

The projection map ψ is defined as follows

$$\begin{aligned} \psi :\; &S \to \mathbb{F}_2 \\ \psi(a + ub + vc) \;=\; &a\,. \end{aligned}$$

In Definition 2.1.4, the Hamming weight is defined. In the following definition, Lee weights will be constructed using the Gray map.

Definition 2.2.3. [11] Let

$$\begin{aligned} \varphi :\; &S \to \mathbb{F}_2^3 \\ \varphi(a + ub + vc) \;=\; &(a, a+b, a+c). \end{aligned}$$

By using this map, we can define the Lee weight. For any element $a + ub + vc \in S$, we define $w_L(a + ub + vc) = w_H(a, a + b, a + c)$, where w_H denotes the ordinary Hamming weight for binary codes. Lee weights are as follows.

$$w_L(0) = 0, w_L(1) = 3,$$
$$w_L(u) = w_L(v) = w_L(1 + u + v) = 1,$$
$$w_L(1 + u) = w_L(1 + v) = w_L(u + v) = 2.$$

Definition 2.2.4. [8] The cartesian product of vectors $s = (s_1, s_2, \dots, s_n) \in \mathbb{F}_2^n, r = (r_1, r_2, \dots, r_n) \in \mathbb{F}_2^n$ and $w = (w_1, w_2, \dots, w_n) \in \mathbb{F}_2^n$ is

$$(s \otimes r \otimes w) = (s_1, s_2, \dots, s_n) \otimes (r_1, r_2, \dots, r_n) \otimes (w_1, w_2, \dots, w_n)$$
$$= (s_1, s_2, \dots, s_n, r_1, r_2, \dots, r_n, w_1, w_2, \dots, w_n) \in \mathbb{F}_2^{3n}.$$

Definition 2.2.5. [8] Let A_1, A_2 and A_3 be any four codes. Then,

$$A^1 \otimes A^2 \otimes A^3 = \{(a^1, a^2, a^3): a^1 \in A^1, a^2 \in A^2, a^3 \in A^3\},$$
$$A^1 \oplus A^2 \oplus A^3 = \{a^1 + a^2 + a^3: a^1 \in A^1, a^2 \in A^2, a^3 \in A^3\}.$$

Let C be a linear code of length n over S. Define

$$C_1 = \{a \in \mathbb{F}_2^n : \exists\, b, c \in \mathbb{F}_2^n, a + ub + vc \in C\},$$
$$C_2 = \{a + b \in \mathbb{F}_2^n : \exists\, c \in \mathbb{F}_2^n, a + ub + vc \in C\},$$
$$C^3 = \{a + c \in \mathbb{F}_2^n : \exists\, b \in \mathbb{F}_2^n, a + ub + vc \in C\}.$$

Then $\varphi(C) = C_1 \otimes C_2 \otimes C_3 and\ |C| = |C_1|\,|C_2|\,|C_3|$.

3. SYMMETRIC ENCRYPTION SCHEME

A symmetric encryption scheme, also known as a secret key cryptosystem, requires both communicating entities to establish a mutual confidential key prior to initiating secure data exchange. This shared key serves as the cornerstone of encryption and decryption, ensuring that only authorized parties can interpret the transmitted information.

Secret key cryptosystems are predominantly classified into two structural categories: substitution-oriented and transposition-oriented mechanisms. In substitution-based methods, encryption is achieved by systematically replacing elements of the plaintext with alternative symbols based on a predetermined algorithm. This category encompasses two significant types:

Monoalphabetic substitution systems, in which each character from the plaintext consistently maps to a fixed symbol or sequence throughout the entire message. The substitution pattern remains unaltered during encryption.

Polyalphabetic substitution systems, on the other hand, involve dynamic character mappings that evolve during the encryption process. Consequently, identical plaintext characters may be encoded using different ciphertext symbols depending on their position in the message, enhancing resistance to cryptanalysis.

Conversely, transposition-based cryptosystems function by reordering the original characters of the plaintext without modifying their actual form. The encryption relies on a systematic permutation of character positions. A practical illustration of this technique is the transformation of the word permission into an impression, where the characters remain the same but their order is altered. Additionally, cryptographic systems may also be distinguished by the manner in which they handle input data during encryption. This gives rise to two principal paradigms:

Block ciphers, which partition the plaintext into fixed-length segments and apply encryption operations uniformly across each block using a single key. This method is well-suited for encrypting large volumes of data in a consistent and organized fashion.

Stream ciphers, by contrast, process data as a continuous flow, encrypting one character (or bit) at a time. Each unit of plaintext may be encoded using a unique keystream value. Due to their lightweight structure, stream ciphers are especially effective in latency-sensitive environments like digital communications, where speed and minimal error propagation are essential.

One Time Pad Cryptosystem: The One-Time Pad is an encryption scheme that encodes information using a key that is equal in length to the message. If m is the plaintext, s is key and c is cryptotext, then the encryption algorithm e_s is $c = c_s(m) = m + s$ and the decryption algorithm d_s is $m = d_s(c) = c + s$, [3].

3.1. ENCRYPTION SCHEMES

This section introduces two encryption schemes based on the One-Time Pad cryptosystem.

3.1.1. First encryption scheme

Our new encryption scheme can be explained as follows.

Key Generation Procedure:

i. Let m be a codeword selected from a cyclic code of length n, generated by a polynomial $g(x)$ of degree r.
ii. Generate s, a cyclic shift of the codeword m.
iii. Compute the sum $c = m + s$.
iv. Here, m serves as the plaintext, while s is considered the private key.

Encryption:

Plaintext: $m_i = a_i(x)g(x)$, where $0 \leq i \leq p^{n-r}$.
Key: $s_i = x^t a_i(x)g(x)$ where t is the number of shift and $s = s_1 s_2 \ldots s_n$.
Ciphertext: $c_i = m_i + s_i$.
We assume that $a_i(x)g(x) \neq a_j(x)g(x)\ for\ i \neq j, 0 \leq i, j \leq p^{n-r}$.

Decryption:

Ciphertext: c_i.
Plaintext: $m_i = c_i + s_i$.

Correctness of the Encryption Scheme

The correctness of the proposed encryption scheme fundamentally relies on the algebraic structure of cyclic codes, which are a subclass of linear error-correcting codes characterized by their invariance under cyclic shifts. A well-established property of cyclic codes is that any cyclic shift of a valid codeword results in another valid codeword belonging to the same code. This inherent property is leveraged in the construction of the encryption mechanism to ensure both integrity and consistency during the encryption and decryption processes. In the context of our scheme, each encryption key is derived from a cyclic shift of a codeword within the chosen cyclic code. Importantly, the length of each key is identical to that of the corresponding plaintext message, allowing a one-to-one mapping between the key elements and the plaintext symbols. This design guarantees that every symbol in the plaintext is encrypted using a unique symbol from the key, maintaining alignment across the entire message. Moreover, the scheme adopts a one-time usage policy for each key sequence, meaning that a specific key is employed only once per encryption session. This approach aligns with the principles of the one-time pad a theoretically unbreakable encryption method thereby enhancing the security posture of the system and preventing key reuse vulnerabilities. By capitalizing on the cyclic nature of the underlying code and enforcing non-reusability of the keys, the scheme ensures that any ciphertext generated can be accurately and deterministically decrypted using the corresponding key. This guarantees semantic correctness, i.e., that the decryption of a ciphertext under the appropriate key will always reproduce the original plaintext without error.

Example 3.1.1.1. Let us consider binary cyclic codes of length 9. The polynomial $x^9 - 1$ can be factorized over $\mathbb{F}_2$ as follows $x^9 - 1 = (x+1)(x^2+x+1)(x^6+x^3+1)$. There are 3 irreducible factors. The 8 generator polynomial are

$$\begin{gathered} 1, \\ x+1, \\ x^2+x+1, \\ x^6+x^3+1, \\ (x+1)(x^2+x+1) = x^3+1, \\ (x+1)(x^6+x^3+1) = x^7+x^6+x^4+x^3+x+1, \\ (x^2+x+1)(x^6+x^3+1) = x^8+x^7+x^6+x^5+x^4+x^3+x^2+x+1, \\ x^9-1. \end{gathered}$$

We try to construct an encryption scheme by using the generator polynomial $g(x) = x^7+x^6+x^4+x^3+x+1$. So, the generator matrix is

$$G = \begin{bmatrix} 110110110 \\ 011011011 \end{bmatrix}.$$

The codewords corresponding to the generator matrix are listed below:

$$C = \{000000000, 110110110, 011011011, 101101101\}.$$

If $g(x) = x^7+x^6+x^4+x^3+x+1$, then $a_i(x)g(x) \in \mathbb{F}_2[x]/(x^9-1)$. $deg(a_i(x) < 2$. So, $a_i(x) = \{0,1,x,x+1\}$.

Encryption: The encryption scheme derived from these codewords is presented below.
$m_i = a_i(x)g(x), s_i = x^t\, a_i(x)g(x)(let\ t = 1), c_i = m_i + s_i\ 1 \le i \le 4,$

$m_1 = a_1(x)g(x) = 1(x^7 + x^6 + x^4 + x^3 + x + 1) = x^7 + x^6 + x^4 + x^3 + x + 1$
$= 110110110,$
$s_1 = xa_1(x)g(x) = x(x^7 + x^6 + x^4 + x^3 + x + 1) = x^8 + x^7 + x^5 + x^4 + x^2 + x$
$= 011011011,$
$c_1 = m_1 + s_1 = x^8 + x^6 + x^5 + x^3 + x^2 + 1 = 101101101,$
$m_2 = a_2(x)g(x) = x(x^7 + x^6 + x^4 + x^3 + x + 1) = x^8 + x^7 + x^5 + x^4 + x^2 + x$
$= 011011011,$
$s_2 = xa_2(x)g(x) = x(x^8 + x^7 + x^5 + x^4 + x^2 + x) = x^8 + x^6 + x^5 + x^3 + x^2 + 1$
$= 101101101,$
$c_2 = m_2 + s_2 = x^7 + x^6 + x^4 + x^3 + x + 1 = 110110110,$
$m_3 = a_3(x)g(x) = (1 + x)(x^7 + x^6 + x^4 + x^3 + x + 1) = x^8 + x^6 + x^5 + x^3 + x^2 + 1$
$= 101101101,$
$s_3 = xa_3(x)g(x) = x(x^8 + x^6 + x^5 + x^3 + x^2 + 1) = x^7 + x^6 + x^4 + x^3 + x + 1$
$= 110110110,$
$c_3 = m_3 + s_3 = x^8 + x^7 + x^5 + x^4 + x^2 + x = 011011011,$
$m_4 = a_4(x)g(x) = 0(x^7 + x^6 + x^4 + x^3 + x + 1) = 000000000,$
$s_4 = 000000000,$
$c_4 = 000000000.$

Decryption:
$m_i = c_i + (p - 1)s_i \Rightarrow m_i = c_i + s_i,$
$m_1 = c_1 + s_1 = (x^8 + x^6 + x^5 + x^3 + x^2 + 1) + (x^8 + x^7 + x^5 + x^4 + x^2 + x)$
$= x^7 + x^6 + x^4 + x^3 + x + 1 = 110110110,$
$m_2 = c_2 + s_2 = (x^7 + x^6 + x^4 + x^3 + x + 1) + (x^8 + x^6 + x^5 + x^3 + x^2 + 1)$
$= x^8 + x^7 + x^5 + x^4 + x^2 + x = 011011011,$
$m_3 = c_3 + s_3 = (x^8 + x^7 + x^5 + x^4 + x^2 + x) + (x^7 + x^6 + x^4 + x^3 + x + 1)$
$= x^8 + x^6 + x^5 + x^3 + x^2 + 1 = 101101101,$
$m_4 = c_4 + s_4 = 0 + 0 = 000000000\,.$

Using the generator polynomial $g(x) = x^6 + x^3 + 1$, we now construct an alternative encryption scheme for the same cyclic code. The generator matrix is

$$G = \begin{bmatrix} 100100100 \\ 010010010 \\ 001001001 \end{bmatrix}.$$

The codewords corresponding to the generator matrix are listed below.

$$C = \begin{Bmatrix} 000000000,100100100,010010010,001001001, \\ 110110110,011011011,101101101,111111111 \end{Bmatrix}.$$

If $g(x) = x^6 + x^3 + 1$, then $a_i(x) \in \mathbb{F}_2[x]/(x^9 - 1)$. $deg(a_i(x) < 3$. So, $a_i(x) = \{0,1, x, x^2, 1 + x, x + x^2, 1 + x^2, 1 + x + x^2\}$.

Encryption:
$m_i = a_i(x)g(x), s_i = x^t\, a_i(x)g(x)(lett = 1), c_i = m_i + s_i\ 1 \leq i \leq 8,$
$m_1 = a_1(x)g(x) = 1(x^6 + x^3 + 1) = x^6 + x^3 + 1 = 100100100,$
$s_1 = xa_1(x)g(x) = x(x^6 + x^3 + 1) = x^7 + x^4 + x = 010010010,$
$c_1 = m_1 + s_1 = (x^6 + x^3 + 1) + (x^7 + x^4 + x) = x^7 + x^6 + x^4 + x^3 + x + 1$
$= 110110110,$

$m_2 = a_2(x)g(x) = x(x^6 + x^3 + 1) = x^7 + x^4 + x = 010010010,$
$s_2 = xa_2(x)g(x) = x(x^7 + x^4 + x) = x^8 + x^5 + x^2 = 001001001,$
$c_2 = m_2 + s_2 = (x^7 + x^4 + x) + (x^8 + x^5 + x^2) = x^8 + x^7 + x^5 + x^4 + x^2 + x$
$= 011011011,$
$m_3 = a_3(x)g(x) = x^2(x^6 + x^3 + 1) = x^8 + x^5 + x^2 = 001001001,$
$s_3 = xa_3(x)g(x) = x(x^8 + x^5 + x^2) = x^6 + x^3 + 1 = 100100100,$
$c_3 = m_3 + s_3 = (x^8 + x^5 + x^2) + (x^6 + x^3 + 1) = x^8 + x^6 + x^5 + x^3 + x^2 + 1$
$= 101101101,$
$m_4 = a_4(x)g(x) = (1 + x)(x^6 + x^3 + 1) = x^7 + x^6 + x^4 + x^3 + x + 1 = 110110110,$
$s_4 = xa_4(x)g(x) = x(x^7 + x^6 + x^4 + x^3 + x + 1) = x^8 + x^7 + x^5 + x^4 + x^2 + x$
$= 011011011,$
$c_4 = m_4 + s_4 = (x^7 + x^6 + x^4 + x^3 + x + 1) + (x^8 + x^7 + x^5 + x^4 + x^2 + x)$
$= x^8 + x^6 + x^5 + x^3 + x^2 + 1 = 101101101,$
$m_5 = a_5(x)g(x) = (x + x^2)(x^6 + x^3 + 1) = x^8 + x^7 + x^5 + x^4 + x^2 + x = 011011011,$
$s_5 = xa_5(x)g(x) = x(x^8 + x^7 + x^5 + x^4 + x^2 + x) = x^8 + x^6 + x^5 + x^3 + x^2 + 1$
$= 101101101,$
$c_5 = m_5 + s_5 = (x^8 + x^7 + x^5 + x^4 + x^2 + x) + (x^8 + x^6 + x^5 + x^3 + x^2 + 1)$
$= x^7 + x^6 + x^4 + x^3 + x + 1 = 110110110,$
$m_6 = a_6(x)g(x) = (1 + x^2)(x^6 + x^3 + 1) = x^8 + x^6 + x^5 + x^3 + x^2 + 1 = 101101101,$
$s_6 = xa_6(x)g(x) = x(x^8 + x^6 + x^5 + x^3 + x^2 + 1) = x^7 + x^6 + x^4 + x^3 + x + 1$
$= 110110110,$
$c_6 = m_6 + s_6 = (x^8 + x^6 + x^5 + x^3 + x^2 + 1) + (x^7 + x^6 + x^4 + x^3 + x + 1)$
$= x^8 + x^7 + x^5 + x^4 + x^2 + x = 011011011,$
$m_7 = a_7(x)g(x) = (1 + x + x^2)(x^6 + x^3 + 1) = x^8 + x^7 + x^6 + x^5 + x^4 + x^3 + x^2 + x + 1$
$= 111111111,$
$s_7 = xa_7(x)g(x) = x(x^8 + x^7 + x^6 + x^5 + x^4 + x^3 + x^2 + x + 1)$
$= x^8 + x^7 + x^6 + x^5 + x^4 + x^3 + x^2 + x + 1 = 111111111,$
$c_7 = m_7 + s_7 = (x^8 + x^7 + x^6 + x^5 + x^4 + x^3 + x^2 + x + 1) + (x^8 + x^7 + x^6 + x^5 + x^4$
$+ x^3 + x^2 + x + 1) = 000000000,$
$m_8 = a_8(x)g(x) = 0(x^6 + x^3 + 1) = 000000000,$
$s_8 = xa_8(x)g(x) = 000000000,$
$c_8 = m_8 + s_8 = 000000000\,.$

Decryption:
$m_i = c_i + (p - 1)s_i \Rightarrow m_i = c_i + s_i,$
$m_1 = c_1 + s_1 = (x^7 + x^6 + x^4 + x^3 + x + 1) + (x^7 + x^4 + x) = x^6 + x^3 + 1$
$= 100100100,$
$m_2 = c_2 + s_2 = (x^8 + x^7 + x^5 + x^4 + x^2 + x) + (x^8 + x^5 + x^2) = x^7 + x^4 + x$
$= 010010010,$
$m_3 = c_3 + s_3 = (x^8 + x^6 + x^5 + x^3 + x^2 + 1) + (x^6 + x^3 + 1) = x^8 + x^5 + x^2$
$= 001001001,$
$m_4 = c_4 + s_4 = (x^8 + x^6 + x^5 + x^3 + x^2 + 1) + (x^8 + x^7 + x^5 + x^4 + x^2 + x)$
$= 110110110,$
$m_5 = c_5 + s_5 = (x^7 + x^6 + x^4 + x^3 + x + 1) + (x^8 + x^6 + x^5 + x^3 + x^2 + 1)$
$= x^8 + x^7 + x^5 + x^4 + x^2 + x = 011011011,$
$m_6 = c_6 + s_6 = (x^8 + x^7 + x^5 + x^4 + x^2 + x) + (x^7 + x^6 + x^4 + x^3 + x + 1)$
$= x^8 + x^6 + x^5 + x^3 + x^2 + 1 = 101101101,$
$m_7 = c_7 + s_7 = 0 + (x^8 + x^7 + x^6 + x^5 + x^4 + x^3 + x^2 + x + 1)$
$= x^8 + x^7 + x^6 + x^5 + x^4 + x^3 + x^2 + x + 1 = 111111111,$

$m_8 = c_8 + s_8 = 0 + 0 = 000000000.$

It is evident that each encryption operation employs a unique, non-reusable key. Moreover, the encryption process possesses the capability to generate multiple valid ciphertexts from the same plaintext input. This feature is particularly advantageous, as it leads to the production of several meaningful ciphertext outputs, each corresponding to a distinct encryption key.

Such variability significantly increases the complexity of key inference for a potential adversary. Since different keys can yield different ciphertexts from identical plaintexts, the system inherently exhibits a form of probabilistic encryption. This property enhances security by introducing ambiguity, thereby reducing the likelihood of reverse-engineering the original key through cryptanalytic techniques.

Proposition 3.1.1.2. Let C be a cyclic code of length n with generator polynomial $g(x)$. If n is large enough, then the encryption scheme constructed based on C will be more reliable.

Proof: The security level of an encryption system is closely linked to the length of the cryptographic key. In the One-Time Pad cryptosystem, the key must be as long as the message to ensure perfect secrecy. In our proposed encryption scheme, the plaintext has length n, and consequently, the key is also of length n. Therefore, encrypting long messages necessitates the generation of equally long keys, which poses challenges in terms of both storage and secure transmission. However, if n is sufficiently large, recovering the key becomes computationally infeasible, thereby enhancing the overall reliability and robustness of the system.

3.1.2. Second encryption scheme

In this section, we implement the previously defined encryption scheme over the ring S, which is characterized by the relations $u^2 = u, v^2 = v, uv = vu = 0$. The objective is to demonstrate that the one-time pad encryption method remains perfectly secure when applied within the algebraic structure of the ring S.

Key Generation Procedure*:*

Let C be a cyclic code over S of length n. $C = (1 + u + v)C_1 \oplus uC_2 \oplus vC_3$, where $C_1 = \langle g_1(x)\rangle, C_2 = \langle g_2(x)\rangle, C_3 = \langle g_3(x)\rangle$ and $g_1(x)|x^n - 1, g_2(x)|x^n - 1, g_3(x)|x^n - 1$, [8].

We choose the codewords such that $u_i \in C_1, s_j \in C_2, m_k \in C_3$, while $0 \leq i < |C_1|, 0 \leq j < |C_2|, 0 \leq k < |C_3|$.

Encryption:

Plaintext: $p_{i+|C_1|j+|C_1||C_2|k} = u_i \times s_j \times m_k \in \varphi(C), 0 \leq i < |C_1|, 0 \leq j < |C_2|, 0 \leq k < |C_3|$.
Key: $m_k \in C_3, 0 \leq k < |C_3|$.
Ciphertext: $c_{i+|C_1|j+|C_1||C_2|k} = \varphi((1 + u + v)\, u_i + (u)\, s_j + (v)m_k)$.

Decryption:

Ciphertext: $c_{i+|C_1|j+|C_1||C_2|k}=\varphi((1 + u + v)\, u_i + (u)\, s_j + (v)m_k)$.
Plaintext: $p_{i+|C_1|j+|C_1||C_2|k} = \psi[\varphi^{-1}(c_{i+|C_1|j+|C_1||C_2|k}) + (v)m_k] \times s_j \times m_k$.

Example 3.1.2.1. Let us consider binary cyclic codes of length 3. The polynomial $x^3 - 1$ can be factorized over F_2 as follows $x^3 - 1 = (x+1)(x^2+x+1)$. Let us take as the generator polynomials $g_1(x) = x + 1, g_2(x)x^2 + x + 1$. These generator polynomials generate the binary cyclic codes are, respectively, $C_1 = C_2 = \{000,110,011,101\}, C_3 = \{000,111\}$. We choose $u^0 = s^0 = 000$, $u^1 = s^1 = 110, u^2 = s^2 = 011, u_3 = s_3 = 101$ and $m^0 = 000, m_1 = 111$ for $i = j = 0,1,2,3$ and $k = 0,1$.

Encryption:

Let $i = 0, j = 0, k = 0$. Then $u_0 = 000, s_0 = 000, m_0 = 000$. We get

$$p_0 = u_0 \times s_0 \times m_0 = 000 \times 000 \times 000 = 000000000,$$

$$c_0 = \varphi((1+u+v)\,u_0 + (u)\,s_0 + (v)m_0) = \varphi(000) = 000000000\,.$$

Let $i = 1, j = 0, k = 0$. Then $u_1 = 110, s_0 = 000, m_0 = 000$. We get

$$p_1 = u_1 \times s_0 \times m_0 = 110 \times 000 \times 000 = 110000000,$$

$$c_1 = \varphi((1+u+v)\,u_1 + (u)\,s_0 + (v)m_0) = \varphi((1+u+v)) = 100100100\,.$$

Let $i = 0, j = 1, k = 0$. Then $u_0 = 000, s_0 = 110, m_0 = 000$. We get

$$p_4 = u_0 \times s_1 \times m_0 = 000 \times 110 \times 000 = 000110000,$$

$$c_4 = \varphi((1+u+v)\,u_0 + (u)\,s_1 + (v)m_0) = \varphi(u) = 010010010\,.$$

Let $i = 0, j = 0, k = 1$. Then $u_0 = 000, s_0 = 000, m_1 = 111$. We get

$$p_{16} = u_0 \times s_0 \times m_1 = 000 \times 000 \times 111 = 000000111,$$

$$c_{16} = \varphi((1+u+v)\,u_0 + (u)\,s_0 + (v)m_1) = \varphi(v) = 001001001\,.$$

Decryption:

$c_0 = 000000000, m_0 = 000$. So $i = j = k = 0$ and
$p_0 = \psi[\varphi^{-1}(000000000) + (v)m_0] \times\ s_0 \times m_0$
$= \psi[(000) + (000)] \times 000 \times 000 = 000000000$
$c_1 = 100100100, m_0 = 000$. So $i = 1, j = k = 0$ and
$p_1 = \psi[\varphi^{-1}(100100100) + (v)m_0] \times\ s_0 \times m_0$
$= \psi[((1+u+v)) + (000)] \times 000 \times 000 = 110000000$
$c_4 = 010010010, m_0 = 000$. So $j = 1, i = k = 0$ and
$p_4 = \psi[\varphi^{-1}(010010010) + (v)m_0] \times\ s_1 \times m_0$
$= 000 \times 110 \times 000 = 000110000$
$c_{16} = 001001001, m_1 = 111$. So $k = 1, i = j = 0$ and
$p_{16} = \psi[\varphi^{-1}(001001001) + (v)m_1] \times\ s_0 \times m_1$
$= 000 \times 000 \times 111 = 000000111.$

In this scheme, a key with the same length as the data is utilized. As a result, even if the message is intercepted by an unauthorized party, its content remains unpredictable. An adversary attempting to decrypt the message would need to consider all possible n-bit combinations, rendering it infeasible to identify the correct plaintext. In this example, only a subset of the ciphertexts is presented; the remaining ciphertexts can be encrypted using the same method.

4. SECURITY ANALYSIS OF THE PROPOSED SCHEME

This work integrates cyclic codewords with the One-Time Pad (OTP) technique to establish a secure encryption framework. The sender encrypts the plaintext by XOR ing it bitwise with a randomly generated key of equal length. The resulting ciphertext is transmitted to the receiver, who retrieves the original message by applying the same XOR operation using an identical copy of the key. To ensure perfect secrecy, keys are used only once and are securely destroyed after transmission, eliminating any possibility of reuse. This approach guarantees information-theoretic security under the classical assumptions of OTP.

4.1. TYPES OF ATTACKS

In the proposed encryption schemes, the key utilized for message encryption is entirely random and matches the length of the plaintext. Due to this property, the most viable method of attack against the system is an exhaustive brute-force search.

The "Brute Force" Attack: A brute force attack operates by systematically attempting all possible key combinations to uncover the one used in the encryption process. This approach relies on an exhaustive search, testing each candidate key by decrypting the ciphertext until a plausible and intelligible plaintext is obtained. The valid key is identified as the one that yields a coherent and meaningful message upon decryption.

Grover's Algorithm: Grover's algorithm is a quantum algorithm developed within the framework of quantum information theory, designed to solve classical search problems that require linear-time scanning by achieving a quadratic speed-up through quantum computation. First proposed by Lov Grover in 1996, this algorithm is capable of solving problems that would require $O(N)$ steps classically in approximately $O(\sqrt{N})$ steps using quantum resources. Grover's algorithm is particularly effective in scenarios where a single unknown input must be located within a large database or solution space, and it provides a quantum advantage in applications such as cryptographic key recovery, inverse function computation, and unstructured data search. By employing a quantum oracle that inverts the phase of the target state and subsequently applying a reflection about the average (diffusion operator), the algorithm amplifies the probability amplitude of the correct solution. As a result, it enables significantly fewer steps than classical brute-force methods to reach the desired outcome with high probability.

Traditional brute-force search methods, which exhibit linear time complexity $O(N)$, can be sufficiently effective in small finite rings. However, their practical applicability significantly diminishes as the size of the search space increases. In particular, such methods become computationally infeasible in scenarios involving large key spaces, such as cryptographic key recovery. In contrast, Grover's algorithm leverages quantum parallelism to offer a significant improvement over classical approaches, reducing the search complexity to $O(\sqrt{N})$. For example, in a search space of size $N = 2^n$, a classical brute-force algorithm would require up to 2^n iterations to locate the correct solution, whereas Grover's algorithm can find the target with high probability in approximately $2^{n/2}$ steps. Thus, although brute-force techniques may appear more practical for small-scale problems, the quantum speed-up provided by Grover's algorithm becomes increasingly advantageous as the problem size grows, rendering it a powerful tool for cryptanalysis and quantum security assessments in modern cryptographic systems, [12].

This theoretical comparison is further illustrated in Fig. 1, which visualizes the expected number of iterations required by both methods over various search space sizes. The graph was produced using the Python programming language, specifically utilizing the matplotlib and numpy libraries for computational modeling and plotting. The average-case performance of the classical brute-force method was modeled using the expression $(N + 1)/2$, while Grover's algorithm was represented by the formula $(\pi/4)\sqrt{N}$.

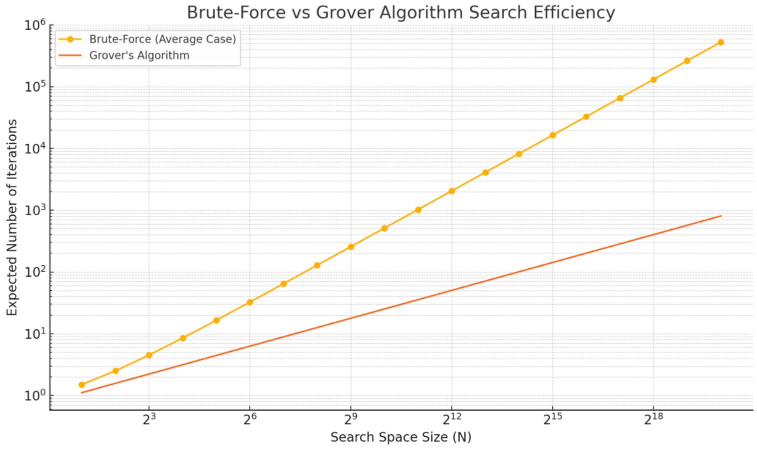

Figure 1. Expected iteration counts for classical brute-force and Grover's algorithm over different search space sizes N**.**

As depicted in the log-log scale of Fig. 1, the two methods perform similarly for small values of N; however, as the search space grows, Grover's algorithm clearly outperforms the classical method by requiring significantly fewer iterations. This visual evidence reinforces Grover's asymptotic superiority, particularly in cryptographic systems involving large key spaces or polynomial ring structures.

5. CONCLUSIONS

This study proposed and analyzed a novel encryption scheme defined over the finite commutative ring $S = \mathbb{F}_2 + u\mathbb{F}_2 + v\mathbb{F}_2$, characterized by the relations $u^2 = u, v^2 = v, uv = vu = 0$. The encryption model is based on the principles of the one-time pad cryptosystem, where a randomly generated key of the same length as the message is used exactly once for each encryption process. Due to this key randomness and uniqueness, the scheme provides theoretically perfect secrecy. After each encryption, multiple plausible plaintexts may correspond to a single ciphertext when different keys are applied, making it computationally infeasible to identify the original message without knowledge of the exact key.

To assess the cryptanalytic resilience of the proposed scheme, we examined the applicability of Grover's quantum search algorithm, which provides a quadratic speed-up over classical brute-force methods. Specifically, we modeled the key recovery problem within the defined ring structure and demonstrated that Grover's algorithm, when applied, significantly reduces the number of iterations required to discover a potential key compared to classical approaches. This comparison was conducted using the Python programming language, with the aid of libraries such as numpy and matplotlib, to visualize the expected number of iterations for both classical and quantum approaches over rings with varying numbers of elements. The findings clearly demonstrate that, for rings with large cardinality, the classical brute-force approach becomes impractical, whereas Grover's algorithm maintains its computational efficiency even as the scale of the problem increases.

The integration of algebraic ring theory with quantum-resistant cryptographic analysis underscores the robustness of our encryption framework. The results presented in this paper affirm that the proposed scheme is not only theoretically sound but also suitable for secure communication in environments where data confidentiality is of critical importance.

REFERENCES

[1] Vernam, G. S., *Journal of the AIEE*, **45**, 572, 1926.

[2] Shannon, C. E., *The Bell System Technical Journal*, **28**, 656, 1949.

[3] Kuklová, Z., *Coding theory, cryptography and cryptographic protocols-exercises with solutions*, Dissertation at Masarykova Univerzita, Fakulta İnformatiky, 2007.

[4] Çalkavur, S., Güzeltepe, M., *Sigma Journal of Engineering and Natural Sciences*, **40**, 380, 2022.

[5] Prange, E., *IRE Transactions on Information Theory*, **8**, 5, 1962.

[6] Abualrub, T., Seneviratne, P., *IMECS*, Hong Kong, 2010.

[7] Yildiz, B., Karadeniz, S., *Codes and Cryptography*, **58**, 221, 2011.

[8] Dertli, A., Cengellenmis, Y., Eren, S., *Palestine Journal of Mathematics*, **4**, 547, 2015.

[9] Ling, S., Xing, C., *Coding Theory: A First Course Overview and Key Concepts*, Cambridge University Press, Cambridge, England, 2004.

[10] Hill, R., *A First Course in Coding Theory*, Oxford University Press, Oxford, United Kingdom, 1986.

[11] Dertli, R., Eren, S., *Journal of Science and Arts*, **20**, 283, 2020.

[12] Grover, L.K., *Proceedings of 28th Annual ACM symposium on Theory of computing*, 212, 1996.