Malaysian Journal of Mathematical Sciences 20(1): 377–393 (2026)
https://doi.org/10.47836/mjms.20.1.18

Malaysian Journal of Mathematical Sciences

Journal homepage: https://mjms.upm.edu.my

A Modular Matrix Approach to Symmetric Encryption Inspired by NTRU Principles

Salem, M. H.[1] and Neamah, A. A.*[1]

[1]*Faculty of Computer Science and Mathematics, University of Kufa, Najaf, Iraq*

E-mail: ammara.meamah@uokufa.edu.iq
**Corresponding author*

Received: 24 May 2025
Accepted: 13 August 2025

Abstract

NTRU is an efficient quantum-resistant lattice-based public-key encryption scheme, but based on polynomial ring computations, which could introduce algebraic vulnerabilities and implementation cost to some uses. In this paper, we introduce a novel symmetric cryptosystem derived from the NTRU cryptography algorithm, using effective matrix computations over traditional polynomial algebra. The structure of the proposed system is based on the use of self-invertible matrices, which eliminate the need for explicit matrix inversion during decryption, ensuring more efficient calculation while retaining a strong mathematical security guarantee. Experimental results demonstrate that the system is lightweight and scalable and can be applied to data protection, image protection, and communication protection on constrained devices such as the Internet of Things (IoT) and embedded systems. In addition, the proposed construction offers an adaptive and secure cryptosystem, which is well-suited for the requirements of modern information security infrastructures that necessitate an efficient, secure, and lightweight encryption system.

Keywords: symmetric encryption; NTRU cryptosystem; self-invertible matrices; matrix-based cryptography.

1 Introduction

As the development of communication technologies is rapidly growing, and people and organizations are increasingly relying on common networks to exchange information, data security against cyberattacks and unauthorized access is highly important. Numerous research studies have revealed the acceleration of frequency and complexity, underscoring the need for more secure and effective encryption technologies [3]. Symmetric encryption systems serve as a fundamental component in data protection, as they utilize a single key for both encryption and decryption processes. This approach offers high performance and speed compared to asymmetric systems [19]. NTRU is a leading algorithm today because it is based on lattice-based cryptography over polynomial rings. In contrast to classical schemes such as ECC [16] and RSA [18], NTRU is resistant to quantum attacks and thus is a strong contender in the field of post-quantum cryptography.

Despite the performance advantage, the conventional NTRU that utilizes polynomial operations is computationally intensive, especially when it comes to the decryption step during which the inverse of a polynomial is computed [4]. This gap is dealt with in this paper by proposing a novel symmetric encryption scheme by storage of the matrix based structures in place of polynomials for enhanced throughput and computational complexity. The proposed system uses NTRU-based matrices instead of traditional polynomial and introduces the self-invertible matrices-which can not be used for encryption and decryption without inverting the matrix. This requires much less computational effort, and makes the entire decryption process more efficient [1]. This would represent a new era in the use of algebraic structures in cryptography and provide a universally flexible and secure mathematical foundation suitable for sensitive applications, such as IoT, smart systems, and military communications, which require both speed and security [12].

2 Related Works

Many derivatives and generalizations of the NTRU cryptosystem have been proposed over the years, with the aim of enhancing various aspects of its performance, security, or utility across a range of algebraic structures.

In 2002, the CTRU scheme was presented, which is also a modification of NTRU, working on binary finite fields [7]. Later, it was shown that CTRU is insecure and linear algebra vulnerable and is not of any performance benefit relative to the original NTRU. In 2005, the MaTRU cryptosystem extended NTRU to a $K \times K$ matrix ring structure, with a purported k-fold speedup over standard NTRU [2]. Two notable variants in 2008 were: Matrix NTRU, also employing matrix rings, which achieved faster encryption than standard NTRU [15]. GBNTRU, based on a ring of binary polynomials, unfortunately resulted in slower computations than the original scheme [5]. OTRU began using octonionic algebra in 2010 and provided the conceptually simpler version. However, its performance was not as excellent as the normal NTRU [14]. Lei and Liao [13] suggested a new NTRU-based key exchange scheme in 2013 by extending the NTRU lattice cryptosystem.

In 2015, the positive coefficient binary truncated polynomial ring known as DBTRU was used, which was much more efficient and had stronger security than the standard NTRU [20]. Gaithuru and Salleh [8] described the ITRU cryptosystem in 2017, which is based on the ring of integers rather than a truncated polynomial ring, and claimed that ITRU has some advantages over classical NTRU, including simpler parameter selection, guaranteed invertibility, and reliable message decryption. Later, it was shown in [10] that ITRU is vulnerable to basic frequency analysis attacks. The security of ITRU can be improved by using a different r for each ciphertext letter or by encoding the entire message as a single large integer instead of a sequence of integers.

D-NTRU was suggested as a security-provable alternative in the standard model in 2018. D-NTRU was better in the efficiency metric of ciphertext expansion, and its encryption/decryption was asymptotically faster than earlier provably secure variants of NTRU [21]. Yassin and Al-Saeedi [22] presented multidimensional systems called BITRU in 2019, that are bicartesian algebra based. In 2021, flattening-NTR prioritised binary polynomials over integer terms, enabling it to achieve higher efficiency and faster running time than that of the regular NTRU [6]. More recently, in 2024, the G-NTRU framework extended NTRU construction to various algebraic rings such as integers, complex numbers, and matrices [23]. This extension significantly facilitated NTRU's applications for post-quantum cryptography by offering tunable efficiency-security trade-offs. However, the framework also introduced issues: higher-dimension ring structures would reduce computational efficiency, while lower-dimension constructions are not necessarily guaranteed to possess secure security levels.

The structure of this article is as follows. Section 3 introduces the mathematical preliminaries on which the proposed work is based. Section 4 introduces the proposed encryption scheme from the perspective of integrating self-invertible matrices to enhance the conventional NTRU system. Section 5 demonstrates a comprehensive implementation example for better clarity of the application of the proposed scheme. In Section 6, a comparative study is conducted between the traditional NTRU scheme of encryption and the matrix-based scheme introduced, keeping in mind improved security and computational intensity. Finally, in Section 7, the paper concludes and outlines potential directions for future work.

3 Mathematical Preliminaries

This section introduces the foundational mathematical concepts and notations necessary to understand the proposed cryptographic scheme. We first define self-invertible matrices and discuss their algebraic properties. Then, we provide a brief overview of the classical NTRU cryptosystem, setting the stage for the modifications introduced later.

3.1 Self-invertible matrix construction

A fundamental requirement for the encryption scheme is a self-invertible matrix F such that,

$$F \cdot F \equiv I \pmod{p}.$$

The construction of this matrix follows a method closely derived from [1], detailed as follows:

- $A \in \mathbb{Z}_p^{n\times n}$ is a random square matrix over the ring $\mathbb{Z}_p$, where $p \in \mathbb{Z}^+$ is a positive integer modulus.
- k is a scalar constant such that $\gcd(k, p) = 1$, ensuring the existence of the modular inverse.
- z is the modular inverse of k modulo p, i.e., $z \equiv k^{-1} \pmod{p}$, with $z \in \{0, \dots, p-1\}$.

Let I be the $n \times n$ identity matrix. The self-invertible matrix $F \in \mathbb{Z}_p^{2n\times 2n}$ is composed of four

$n \times n$ blocks,

$$F \equiv \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \pmod{p},$$

where

$$\begin{cases} A_{11} \equiv A \pmod{p}, \\ A_{12} \equiv k(I - A) \pmod{p}, \\ A_{21} \equiv z(I + A) \pmod{p}, \\ A_{22} \equiv -A \pmod{p}. \end{cases}$$

This construction guarantees that $F^2 \equiv I \pmod{p}$, i.e., F is self-invertible modulo p, so the same matrix F can be used for both encryption and decryption. All matrix operations are performed modulo p, ensuring all elements lie in the range $\{0, \ldots, p-1\}$. The construction of these matrices is shown in Algorithm 1.

Properties:

- All eigenvalues of a self-invertible matrix lie in $\{1, -1\}$ (modulo p, when defined).
- The matrix must be non-singular in $\mathbb{Z}_p$.
- These matrices form a subgroup of the general linear group $GL_n(\mathbb{Z}_p)$.

Algorithm 1 Construct self-invertible matrix $F \in \mathbb{Z}_p^{2n \times 2n}$

```
Require: $A \in \mathbb{Z}_p^{n \times n}$ (square matrix), scalar $k \in \mathbb{Z}_p$, modulus $p \in \mathbb{Z}^+$.
Ensure: $F \in \mathbb{Z}_p^{2n \times 2n}$ such that $F^2 = I_{2n} \bmod p$,
$n \leftarrow$ size$(A, 1)$,
$m \leftarrow$ size$(A, 2)$,
if $n \neq m$ then
    error: "Matrix $A$ is not square."
    return
end if
Compute modular inverse: $z \leftarrow k^{-1} \bmod p$, where $z \in \{0, \ldots, p-1\}$,
$A_{11} \leftarrow A \bmod p$,
$A_{22} \leftarrow (-A_{11}) \bmod p$ ,
$A_{12} \leftarrow k \cdot (I_n - A_{11}) \bmod p$,
$A_{21} \leftarrow z \cdot (I_n + A_{11}) \bmod p$,
Form block matrix,
    $$F = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \bmod p.$$
return $F$
```

Relevance to cryptography: Self-invertible matrices are appealing in cryptographic design for the following reasons;

- They allow symmetric key operations where encryption and decryption use the same structure.
- They introduce additional algebraic complexity, strengthening resistance to structural and linear attacks. Specifically, the constraint $F^2 \equiv I \pmod{p}$ defines a non-linear matrix equation (quadratic), which cannot be accurately modeled or solved using linear methods without significant loss of structural information.
- They enable compact key representations and efficient matrix computations.

3.2 Overview of classical NTRU

The NTRU cryptosystem [11] is a lattice-based public key encryption scheme defined over polynomial rings of the form,

$$R = \mathbb{Z}[x]/(x^N - 1),$$

with parameters N, p and q, where $p \ll q$. NTRU relies on the hardness of the "short vector problem" in lattices derived from polynomial convolutions.

Key concepts:

- **Key generation:**
  Two polynomials f, g are chosen such that f is invertible modulo both p and q. This means that their inverses, $f_p \equiv f^{-1} \pmod{p}$ and $f_q \equiv f^{-1} \pmod{q}$, exist.
- **Encryption:**
  To encrypt a message polynomial m, the sender selects a small random polynomial r and utilizes the public key $h \equiv g \cdot f_q \pmod{q}$. The ciphertext is then computed as,
  $$e \equiv (r \cdot h + m) \pmod{q}.$$
- **Decryption:**
  The recipient uses the private key f to recover m by computing $a \equiv f \cdot e \pmod{q}$ and then $m \equiv f_p \cdot a \pmod{p}$.

Security basis: NTRU's security rests on the difficulty of finding short vectors in high-dimensional lattices constructed from the public key. Unlike RSA or ECC, it is believed to be resistant to quantum attacks.

3.3 Motivation for modification

While classical NTRU is efficient and quantum-resistant, its algebraic structure can sometimes be exploited in lattice reduction attacks. Integrating self-invertible matrices introduces additional algebraic obfuscation:

- It increases the hardness of deducing private keys from public keys.
- It allows a matrix-based transformation that further obscures the polynomial structure.
- It retains the speed and efficiency advantages of NTRU, while enhancing security.

4 Encryption Scheme

In this work, a modified version of the NTRU cryptosystem supported by self-invertible matrix is proposed for enhanced security and algebraic complexity, which is more secure against cryptanalytic attacks and its performance is studied.

4.1 Key generation using self-invertible matrices

Choice of parameters: Alice and Bob agree on public parameters shared between them, including a large prime q and a small positive integer p, and a matrix size n. These parameters together specify the ring $\mathbb{Z}_p$ and the security.

Generation of private key: Bob creates a secret matrix $F \in \mathbb{Z}_p^{n\times n}$ with the properties,

$$F^2 \equiv I \pmod{p}.$$

Such a matrix can be constructed using the method mentioned in Subsection 3.1.

Alice uses the matrix F and calculates its inverse modulo q, denoted by,

$$F_q \equiv F^{-1} \pmod{q}.$$

Bob keeps F, its self-inverse $F^{-1} \equiv F \pmod{p}$, and also

$$F_p \equiv F^{-1} \pmod{p}.$$

The encryption and decryption keys are F_q, F, and F_p.

The auxiliary matrix G: An important advantage of this approach is that G can be used as a part of the masking transformation during encryption.

Clarification on symmetric framework: Although this scheme draws inspiration from the public-key NTRU cryptosystem, it is implemented here as a fully symmetric system. The secret key consists mainly of the matrix F, from which the matrices $F_p \equiv F^{-1} \pmod{p}$ and $F_q \equiv F^{-1} \pmod{q}$ are derived using the publicly known parameters p and q. The matrix G, randomly generated for each encryption, acts as a masking element and is not required for decryption. Only the matrix F must be securely shared between the parties communicating. Consequently, the system does not employ any public keys.

4.2 Matrix-based encryption algorithm

Message preparation: Alice encodes her plaintext message as a matrix $M \in \mathbb{Z}_p^{n\times n}$.

Ciphertext computation: Using the inverse private matrix F_q and the matrix G, the ciphertext E is computed as,

$$E \equiv (p \cdot F_q \cdot G + M) \pmod{q}. \tag{1}$$

In (1), the term $p \cdot F_q \cdot G$ serves as a high-noise component that masks the message by dominating the ciphertext modulo q, while M is the low-noise component that carries the actual information and is recoverable after decryption by reduction modulo p.

Alice then transmits the ciphertext matrix E to Bob.

4.3 Decryption via dual-modular matrix inversion

Bob recovers the plaintext through a two-stage process:

Multiply with private key F: Bob computes,

$$A \equiv F \cdot E \pmod{q}. \tag{2}$$

This operation removes the F_q component due to the identity,

$$F \cdot F_q \equiv I \pmod{q}.$$

Multiply with inverse F_p: To retrieve the original message, Bob computes,

$$M' \equiv F_p \cdot A \pmod{p}, \tag{3}$$

where

$$F_p \equiv F^{-1} \pmod{p}.$$

This yields the original message,

$$M' = M,$$

ensuring successful decryption. Figure 1 depicts the workflow of the proposed matrix-based symmetric encryption system.

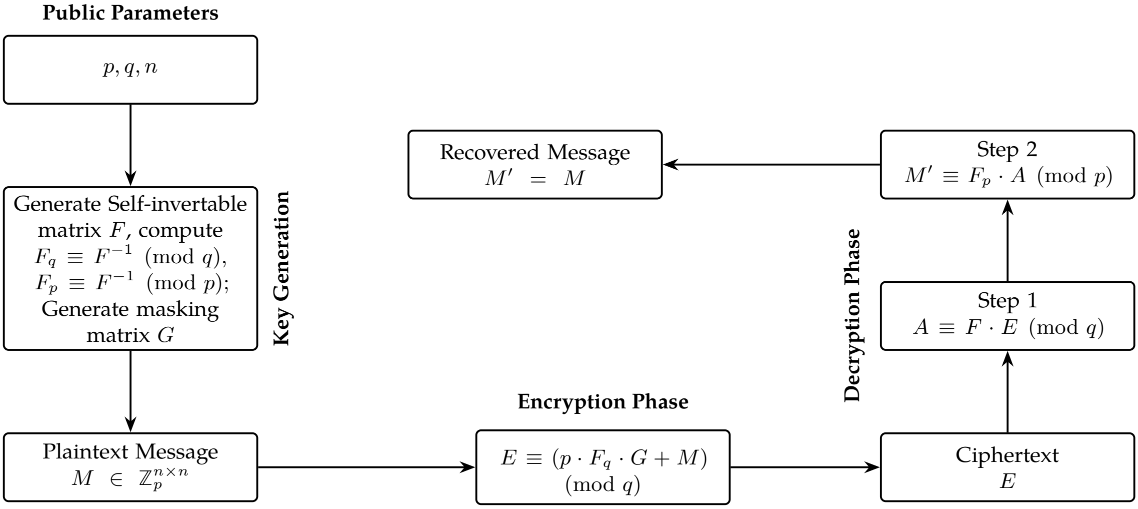
Public Parameters
p, q, n
Generate Self-invertable matrix F, compute $F_q \equiv F^{-1} \pmod q$, $F_p \equiv F^{-1} \pmod p$; Generate masking matrix G
Key Generation
Plaintext Message $M \in \mathbb{Z}_p^{n\times n}$
Encryption Phase
$E \equiv (p \cdot F_q \cdot G + M) \pmod q$
Ciphertext E
Step 1 $A \equiv F \cdot E \pmod q$
Step 2 $M' \equiv F_p \cdot A \pmod p$
Decryption Phase
Recovered Message $M' = M$

Figure 1: Workflow of the proposed matrix-based symmetric encryption system.

5 Implementations and Examples

The implementation details of the self-invertible matrix-based encryption scheme are described in this section with illustrative test examples for each phase (key generation, encryption, and decryption). We implemented the model in MATLAB(R2018b), and all the experiments were performed on a machine with an Intel Core i5-8265U CPU and 8GB of RAM.

5.1 Illustrative example

Let the public parameters be:

- $q = 1011107$ (it is a prime modulo),
- $p = 256$ (any arbitrary positive number such that $p \ll q$),
- $n = 4$ (matrix dimension).

5.1.1 Generation of private key

- F is a random 4×4 invertible matrix satisfying,

$$F^2 \equiv I \pmod p,$$

  constructed using the self-invertible matrix method described in Subsection 3.1.

  The matrix F is self-invertible modulo p, with inverse denoted by,

$$F^{-1} = F_p.$$

The matrix F and its inverse $F^{-1} = F_p$ are explicitly given by,

$$F = \begin{bmatrix} 206 & 8 & 51 & 248 \\ 252 & 137 & 4 & 120 \\ 207 & 8 & 50 & 248 \\ 252 & 138 & 4 & 119 \end{bmatrix}, \quad F_p = \begin{bmatrix} 206 & 8 & 51 & 248 \\ 252 & 137 & 4 & 120 \\ 207 & 8 & 50 & 248 \\ 252 & 138 & 4 & 119 \end{bmatrix}.$$

- F_q is computed as follows,

$$F_q \equiv F^{-1} \pmod{1011107} \equiv \begin{bmatrix} 691787 & 451287 & 287785 & 528284 \\ 691833 & 451416 & 287738 & 528156 \\ 691788 & 451287 & 287784 & 528284 \\ 691833 & 451417 & 287738 & 528155 \end{bmatrix}.$$

The private key F_q is used for encryption, while the private key pair $(F,\, F_p)$ is used for decryption.

- G is chosen at random as the following 4×4 matrix over $\mathbb{Z}_p$,

$$G = \begin{bmatrix} 22 & 240 & 136 & 35 \\ 205 & 4 & 226 & 55 \\ 253 & 175 & 230 & 46 \\ 17 & 200 & 160 & 10 \end{bmatrix}.$$

5.1.2 Encryption process

- Assume the plaintext message M is given as,

$$M = \begin{bmatrix} 27 & 105 & 253 & 62 \\ 157 & 251 & 196 & 75 \\ 240 & 242 & 86 & 174 \\ 90 & 173 & 169 & 135 \end{bmatrix}.$$

- The ciphertext E is computed as,

$$E \equiv (p \cdot F_q \cdot G + M) \pmod{q},$$

where explicitly,

$$\begin{aligned} E \equiv & \left(256 \cdot \begin{bmatrix} 691787 & 451287 & 287785 & 528284 \\ 691833 & 451416 & 287738 & 528156 \\ 691788 & 451287 & 287784 & 528284 \\ 691833 & 451417 & 287738 & 528155 \end{bmatrix} \cdot \begin{bmatrix} 22 & 240 & 136 & 35 \\ 205 & 4 & 226 & 55 \\ 253 & 175 & 230 & 46 \\ 17 & 200 & 160 & 10 \end{bmatrix} \right. \\ & \left. + \begin{bmatrix} 27 & 105 & 253 & 62 \\ 157 & 251 & 196 & 75 \\ 240 & 242 & 86 & 174 \\ 90 & 173 & 169 & 135 \end{bmatrix} \right) \pmod{1011107} \\ \equiv & \begin{bmatrix} 67624 & 701608 & 993743 & 932721 \\ 462273 & 56425 & 26192 & 257848 \\ 8701 & 718385 & 969512 & 930017 \\ 510334 & 6171 & 43061 & 269428 \end{bmatrix}. \end{aligned}$$

5.1.3 Decryption process

- Using private key matrix F to compute the matrix A as,

$$A \equiv F \cdot E \pmod q$$

$$A \equiv \left(\begin{bmatrix} 206 & 8 & 51 & 248 \\ 252 & 137 & 4 & 120 \\ 207 & 8 & 50 & 248 \\ 252 & 138 & 4 & 119 \end{bmatrix} \cdot \begin{bmatrix} 67624 & 701608 & 993743 & 932721 \\ 462273 & 56425 & 26192 & 257848 \\ 8701 & 718385 & 969512 & 930017 \\ 510334 & 6171 & 43061 & 269428 \end{bmatrix} \right) \pmod{1011107},$$

which yields,

$$A \equiv \begin{bmatrix} 47010 & 140324 & 134800 & 64686 \\ 92553 & 83599 & 169088 & 56875 \\ 105933 & 123547 & 159031 & 67390 \\ 44492 & 133853 & 152219 & 45295 \end{bmatrix}.$$

- Using F_p, the inverse of F modulo p, to obtain M',

$$M' \equiv F_p \cdot A \pmod p$$

$$M' \equiv \left(\begin{bmatrix} 206 & 8 & 51 & 248 \\ 252 & 137 & 4 & 120 \\ 207 & 8 & 50 & 248 \\ 252 & 138 & 4 & 119 \end{bmatrix} \cdot \begin{bmatrix} 47010 & 140324 & 134800 & 64686 \\ 92553 & 83599 & 169088 & 56875 \\ 105933 & 123547 & 159031 & 67390 \\ 44492 & 133853 & 152219 & 45295 \end{bmatrix} \right) \pmod{256},$$

which yields,

$$M' \equiv \begin{bmatrix} 27 & 105 & 253 & 62 \\ 157 & 251 & 196 & 75 \\ 240 & 242 & 86 & 174 \\ 90 & 173 & 169 & 135 \end{bmatrix} = M,$$

which confirms successful decryption.

5.2 Experimental results with fixed parameters

In this subsection, we demonstrate the performance of the encryption and decryption scheme using three different plaintext matrices, while keeping all cryptographic parameters fixed. The fixed parameters employed in the experimental results are;

- **Modulus parameters:** $p = 256$, $q = 1011107$.
- **Matrix dimension:** $n = 4$.
- **Self-invertible matrix (and its inverse modulo p):**

$$F = F_p = \begin{bmatrix} 206 & 8 & 51 & 248 \\ 252 & 137 & 4 & 120 \\ 207 & 8 & 50 & 248 \\ 252 & 138 & 4 & 119 \end{bmatrix}.$$

- **Matrix inverse modulo q:**

$$F_q = \begin{bmatrix} 691787 & 451287 & 287785 & 528284 \\ 691833 & 451416 & 287738 & 528156 \\ 691788 & 451287 & 287784 & 528284 \\ 691833 & 451417 & 287738 & 528155 \end{bmatrix}.$$

- **Auxiliary matrix:**

$$G = \begin{bmatrix} 22 & 240 & 136 & 35 \\ 205 & 4 & 226 & 55 \\ 253 & 175 & 230 & 46 \\ 17 & 200 & 160 & 10 \end{bmatrix}.$$

The results consistently confirm successful encryption and decryption under these identical key settings, as presented in Table 1.

Table 1: Test case results for encryption and decryption.

Plaintext M	**Ciphertext E**	**Recovered M'**	**Status**
$\begin{bmatrix} 105 & 141 & 184 & 88 \\ 154 & 149 & 255 & 226 \\ 192 & 131 & 90 & 116 \\ 149 & 21 & 248 & 105 \end{bmatrix}$	$\begin{bmatrix} 67702 & 701644 & 993674 & 932747 \\ 462270 & 56323 & 26251 & 257999 \\ 8653 & 718274 & 969516 & 929959 \\ 510393 & 6019 & 43140 & 269398 \end{bmatrix}$	$\begin{bmatrix} 105 & 141 & 184 & 88 \\ 154 & 149 & 255 & 226 \\ 192 & 131 & 90 & 116 \\ 149 & 21 & 248 & 105 \end{bmatrix}$	Success
$\begin{bmatrix} 77 & 184 & 236 & 252 \\ 11 & 224 & 204 & 183 \\ 50 & 149 & 73 & 214 \\ 184 & 18 & 139 & 110 \end{bmatrix}$	$\begin{bmatrix} 67674 & 701687 & 993726 & 932911 \\ 462127 & 56398 & 26200 & 257956 \\ 8511 & 718292 & 969499 & 930057 \\ 510428 & 6016 & 43031 & 269403 \end{bmatrix}$	$\begin{bmatrix} 77 & 184 & 236 & 252 \\ 11 & 224 & 204 & 183 \\ 50 & 149 & 73 & 214 \\ 184 & 18 & 139 & 110 \end{bmatrix}$	Success
$\begin{bmatrix} 95 & 95 & 171 & 104 \\ 239 & 151 & 52 & 170 \\ 212 & 223 & 167 & 239 \\ 217 & 238 & 18 & 207 \end{bmatrix}$	$\begin{bmatrix} 67692 & 701598 & 993661 & 932763 \\ 462355 & 56325 & 26048 & 257943 \\ 8673 & 718366 & 969593 & 930082 \\ 510461 & 6236 & 42910 & 269500 \end{bmatrix}$	$\begin{bmatrix} 95 & 95 & 171 & 104 \\ 239 & 151 & 52 & 170 \\ 212 & 223 & 167 & 239 \\ 217 & 238 & 18 & 207 \end{bmatrix}$	Success

5.3 Runtime comparison

To evaluate the efficiency of the matrix-based cryptosystem presented in this paper, we compared the running time of our scheme with the original NTRU encryption scheme. Key generation, encryption, and decryption were run 50 times for each minimal primitive operation, and the runtime was averaged in milliseconds. The conditions of comparison are as follows:

1. **Classical NTRU:**
   - Degree of polynomial ring: $N = 167$.
   - Small modulus: $p = 256$.
   - Large modulus: $q = 1011107$.

2. **Matrix-based scheme:**
    - Matrix core size: $n = 2$ (i.e., full matrix size $2n = 4$).
    - Small modulus: $p = 256$.
    - Large modulus: $q = 1011107$.

Table 2 shows that the proposed scheme outperforms the classical NTRU in all core operations. This improved efficiency comes from the avoidance of polynomial convolution and the use of optimized modular matrix multiplication, which significantly accelerates both the encryption and decryption phases.

Table 2: Average runtime comparison between classical NTRU and proposed scheme.

Method	**Key generation**	**Encryption**	**Decryption**
Classical NTRU	1.3346 ms	1.1551 ms	1.8803 ms
Proposed (matrix-based)	0.7737 ms	0.3030 ms	0.1527 ms

6 Comparison of Classical NTRU and Matrix-Based Scheme

In this section, we present a comparative analysis between the classical NTRU encryption scheme and the proposed matrix-based adaptation that incorporates a self-invertible matrix structure. The comparison is based on various cryptographic properties, algebraic structures, and operational differences.

6.1 Structural differences

This section presents a structural difference between the classical NTRU and the proposed scheme, as summarized in Table 3.

Table 3: Comparison of structural properties.

Aspect	**Classical NTRU**	**Proposed (Matrix-Based)**
Message Representation	Polynomial ring $\mathbb{Z}_q[x]/(x^N - 1)$.	Matrix ring $\mathbb{Z}_q^{n\times n}$.
Private Key	Two polynomials f, g	Self-invertible matrix. F and auxiliary matrix G.
Public Key	$h \equiv p \cdot g \cdot f^{-1} \pmod q$	–
Ciphertext Structure	Polynomial in $\mathbb{Z}_q[x]$.	Matrix in $\mathbb{Z}_q^{n\times n}$.
Decryption Core	Convolution and inversion in the ring.	Dual modular inverse matrix multiplications.

6.2 Security implications and resistance to lattice attacks

The proposed scheme improves security through a combination of algebraic design choices and cryptographic masking strategies, offering improved resistance to structural and lattice-based attacks.

- **Increased algebraic complexity:**
  The use of matrix rings over $\mathbb{Z}_p$ and $\mathbb{Z}_q$ introduces additional algebraic structure, with higher dimensionality and non-commutativity compared to polynomial rings. These properties increase resistance to algebraic and lattice-based attacks, which typically rely on the commutative and cyclic nature of convolution in classical NTRU.

- **Self-invertibility constraint:**
  The private matrix F is constructed to be self-invertible modulo p, i.e., $F^2 \equiv I \mod p$. This constraint complicates key recovery, as attackers would need to solve simultaneous congruences of modular matrix on distinct modulo p and q.

- **Obfuscation via multiplicative masking:**
  Each ciphertext includes a randomized multiplicative term $p \cdot F_q \cdot G$, where G is a fresh matrix per encryption. This acts as a structured but unpredictable noise component, obfuscating the message and strengthening resistance to chosen-ciphertext and plaintext-recovery attacks.

- **No public key exposure:**
  Unlike classical NTRU, the proposed scheme is fully symmetric and does not expose a public key. Consequently, attackers cannot construct a lattice based on a known transformation (e.g., $h \equiv g \cdot f^{-1} \pmod{q}$) a crucial step in most NTRU lattice attacks.

- **Resistance to ideal lattice embedding:**
  Classical NTRU relies on polynomial rings where ideal lattices can be formed due to ring homomorphism and the cyclic structure of convolution [11]. In contrast, the proposed scheme operates over non-commutative matrix rings, which lack the algebraic structure required to form ideal lattices, thereby disrupting standard lattice construction techniques and reducing the effectiveness of these lattice reduction attacks [17].

Although a complete reduction in complexity-theoretic security remains open, these structural and operational features contribute to improved resistance to known lattice-based cryptanalytic techniques. The use of randomized masking, self-invertible matrices, and modular arithmetic in a non-commutative setting increases the structural obfuscation and cryptographic strength scheme.

6.3 Performance comparison

This section provides a comparative analysis of the operational characteristics of the classical NTRU cryptosystem and the proposed matrix-based symmetric scheme. The comparison focuses on the arithmetic operations required for key generation, encryption, and decryption, highlighting the structural and computational distinctions between the two approaches. These details are summarized in Table 4.

Table 4: Comparison of cryptographic operations between classical NTRU and the proposed scheme.

Operation	Classical NTRU	Proposed (matrix-based)
Key generation	Polynomial inverses modulo p and q.	Matrix inverse modulo q, self-invertible construction modulo p.
Encryption	Polynomial multiplication.	Matrix multiplications and modular additions.
Decryption	Inverse polynomial multiplication.	Two matrix multiplications and reduction steps.
Computational characteristics	Fast due to convolution properties.	Faster than classical NTRU for small to moderate matrix sizes n, depending on the efficiency of underlying matrix operations such as multiplication and modular reduction.

6.4 Advantages of the proposed scheme

- **Enhanced structural obfuscation:**
  Matrix-based representations inherently resist attacks that exploit ring homomorphisms in polynomial-based schemes. The proposed matrix-based design inherently avoids the algebraic vulnerabilities present in polynomial ring-based schemes by operating over non-commutative modular matrix spaces. This prevents the application of ideal lattice constructions and ring homomorphisms typically used in classical NTRU attacks, enhancing structural obfuscation and resistance to lattice-based cryptanalysis.
- **Improved resistance to lattice attacks:**
  The matrix form enlarges the problem space and may make direct lattice recovery more computationally expensive.
- **Better key diversity:**
  With matrices, the number of potential key pairs increases factorially with dimension, offering a higher entropy for the same parameter sizes.

7 Conclusion, Discussion and Future Work

This study realizes a breakthrough in lattice-based cryptography by reformulating the traditional NTRU system into a more compact matrix-based cryptographic system. The proposed modification exploits the power of self-inverse matrices, a novel approach that simplifies key operations, particularly decryption, by replacing complex polynomial inversions with straightforward matrix multiplications. This modification enhances not only the computational efficiency in general but also the system stability, particularly in configurations that require high-speed processing and parallel computation. This performance gain is largely attributable to the substitution of intricate polynomial inversions with matrix multiplications that are highly suitable for parallel execution on contemporary architectures like multi-core processors. Hence, the scheme can utilize these architectures to facilitate efficient key operations and secure decryption.

The comparison with the classical and proposed NTRU schemes indicates that the matrix-based system is seen to preserve the robust security assurances of the original lattice-based scheme while attaining significant advances in practicality, particularly in the context of modern computing systems. The utilization of self-inverse matrices also introduces an additional aspect of flexibility in terms of scalability and parallelism, making the system increasingly suitable for practical implementation in post-quantum cryptographic systems.

Furthermore, the proposed scheme is more secure against classical and quantum attacks using hard lattice problems in the framework of matrices and by incorporating algebraic complexity using self-invertible vectors, which are resistant to both lattice reduction and quantum algebraic attacks. The fact that it maintains strong security while providing enhanced computational efficiency and scalability is a leap forward towards developing future-proof cryptographic protocols. In the future, we will study its side-channel security and add techniques for chosen-ciphertext attack (CCA) security, which is in favor of practical security and application of the scheme. We also intend to investigate the use of the new cryptosystem for image encryption by adopting a technique similar to that presented in [9].

Acknowledgement The authors thank the referees for their helpful comments and suggestions that contributed to improving the manuscript.

Conflicts of Interest The authors declare no conflict of interest.

References

[1] B. Acharya, G. S. Rath, S. K. Patra & S. K. Panigrahy (2007). Novel methods of generating self-invertible matrix for Hill Cipher algorithm. *International Journal of Security*, *1*(1), 14–21. http://dspace.nitrkl.ac.in/dspace/handle/2080/620.

[2] S. Akleylek & N. Kaya (2018). New quantum secure key exchange protocols based on MaTRU. In *2018 6th International Symposium on Digital Forensic and Security (ISDFS), 22–25 March 2018, Antalya, Turkey*, pp. 1–5. IEEE, Piscataway, New Jersey. https://doi.org/10.1109/ISDFS.2018.8355362.

[3] N. F. H. Al-Saffar, H. K. H. Alkhayyat & Z. K. Obaid (2024). A novel image encryption algorithm involving a logistic map and a self-invertible matrix. *Malaysian Journal of Mathematical Sciences*, *18*(1), 107–126. http:doi.org/10.47836/mjms.18.1.07.

[4] W. D. Banks & I. E. Shparlinski (2002). A variant of NTRU with non-invertible polynomials. In *Progress in Cryptology - INDOCRYPT 2002: Third International Conference on Cryptology in India Hyderabad, India, December 16–18, 2002*, volume 2551 of *Lecture Notes in Computer Science* pp. 62–70. Springer, Berlin, Heidelberg. http:doi.org/10.1007/3-540-36231-2_6.

[5] M. Caboara, F. Caruso & C. Traverso (2008). Gröbner bases for public key cryptography. In *Proceedings of the Twenty-First International Symposium on Symbolic and Algebraic Computation*, ISSAC '08 pp. 315–324. Association for Computing Machinery, New York, USA. https://doi.org/10.1145/1390768.1390811.

[6] Y. Doröz & B. Sunar (2020). Flattening NTRU for evaluation key free homomorphic encryption. *Journal of Mathematical Cryptology*, *14*(1), 66–83. https://doi.org/10.1515/jmc-2015-0052.

[7] P. Gaborit, J. Ohler & P. Solé (2002). *CTRU, a Polynomial Analogue of NTRU*. PhD thesis, Inria Center at Université Côte d'Azur, Valbonne, France. https://inria.hal.science/inria-00071964/.

[8] J. N. Gaithuru & M. Salleh (2017). ITRU: NTRU-based cryptosystem using ring of integers. *International Journal of Innovative Computing*, *7*(1), 33–38. https://doi.org/10.11113/ijic.v7n1.135.

[9] H. H. Hadi & A. A. Neamah (2024). An image encryption method based on modified elliptic curve Diffie-Hellman key exchange protocol and Hill Cipher. *Open Engineering*, *14*(1), Article ID: 20220552. https://doi.org/10.1515/eng-2022-0552.

[10] H. R. Hashim, A. Molnár & S. Tengely (2021). Cryptanalysis of ITRU. *Rad Hrvatske Akademije Znanosti i Umjetnosti. Matematičke Znanosti*, *25*(546), 181–193. https://doi.org/10.21857/yrvgqtexl9.

[11] J. Hoffstein, J. Pipher & J. H. Silverman (1998). NTRU: A ring-based public key cryptosystem. In *Algorithmic Number Theory: Third International Symposium, ANTS-III, Portland, Orgeon, USA, June 21-25, 1998, Proceedings*, volume 1423 of *Lecture Notes in Computer Science* pp. 267–288. Springer, Berlin, Heidelberg. https://link.springer.com/chapter/10.1007/BFb0054868.

[12] M. N. Khan, A. Rao & S. Camtepe (2020). Lightweight cryptographic protocols for IoT-constrained devices: A survey. *IEEE Internet of Things Journal*, *8*(6), 4132–4156. https://doi.org/10.1109/JIOT.2020.3026493.

[13] X. Lei & X. Liao. NTRU-KE: A lattice-based public key exchange protocol. Cryptology ePrint Archive, Paper 2013/718 2013. https://eprint.iacr.org/2013/718.

[14] E. Malekian & A. Zakerolhosseini (2010). OTRU: A non-associative and high-speed public key cryptosystem. In *2010 15th CSI International Symposium on Computer Architecture and Digital Systems*, pp. 83–90. IEEE, Piscataway, New Jersey. https://doi.org/10.1109/CADS.2010.5623536.

[15] R. Nayak, C. V. Sastry & J. Pradhan (2008). A matrix formulation for NTRU cryptosystem. In *Proceedings of the 2008 16th IEEE International Conference on Networks (ICON 2008), 12-14 December 2008, New Delhi, India*, pp. 1–5. IEEE, Piscataway, New Jersey. https://doi.org/10.1109/ICON.2008.4772602.

[16] A. A. Neamah (2015). New collisions to improve Pollard's Rho method of solving the discrete logarithm problem on elliptic curves. *Journal of Computer Science*, *11*(9), 971–975. https://doi.org/10.3844/jcssp.2015.971.975.

[17] C. Peikert (2016). A decade of lattice cryptography. *Foundations and Trends in Theoretical Computer Science*, *10*(4), 283–424. http://dx.doi.org/10.1561/0400000074.

[18] R. L. Rivest, A. Shamir & L. Adleman (1978). A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, *21*(2), 120–126. http:doi.org/10.1145/359340.359342.

[19] W. Stallings (2017). *Cryptography and Network Security: Principles and Practice*. Pearson, Upper Saddle River, New Jersey.

[20] C. M. Thang & N. Binh (2015). DBTRU, a new NTRU-like cryptosystem based on dual binary truncated polynomial rings. In *2015 2nd National Foundation for Science and Technology Development Conference on Information and Computer Science (NICS), 16-18 September 2015, Ho Chi Minh City, Vietnam*, pp. 11–16. IEEE, Piscataway, New Jersey. http://dx.doi.org/10.1109/NICS.2015.7302172.

[21] B. Wang, H. Lei & Y. Hu (2018). D-NTRU: More efficient and average-case IND-CPA secure NTRU variant. *Information Sciences*, *438*, 15–31. http:doi.org/10.1016/j.ins.2018.01.037.

[22] H. R. Yassein & N. M. G. Al-Saidi (2019). An innovative bicartesian algebra for designing of highly performed NTRU-like cryptosystem. *Malaysian Journal of Mathematical Sciences*, *13*(S), 29–43.

[23] X. Zhang, H. Wang & Y. Liu (2024). Generalized NTRU algorithms on algebraic rings. *Electronics*, *13*(21), Article ID: 4293. http:doi.org/10.3390/electronics13214293.