CAPITAL UNIVERSITY OF SCIENCE AND TECHNOLOGY, ISLAMABAD

Modification and Improvement of Cryptosystem based on Non-Commutative Platform Groups

by

Sumaira Bibi

A thesis submitted in partial fulfillment for the degree of Master of Philosophy

in the

Faculty of Computing
Department of Mathematics

2018

To my parents, family and friends for their support and love.

CAPITAL UNIVERSITY OF SCIENCE & TECHNOLOGY
ISLAMABAD

CERTIFICATE OF APPROVAL

Modification and Improvement of Cryptosystem based on Non-Commutative Platform Groups

by

Sumaira Bibi

MMT163003

THESIS EXAMINING COMMITTEE

S. No.	Examiner	Name	Organization
(a)	External Examiner	Dr. Waqas Mahmood	QAU, Islamabad
(b)	Internal Examiner	Dr. Abdul Rehman Kashif	CUST, Islamabad
(c)	Supervisor	Dr. Rashid Ali	CUST, Islamabad

Dr. Rashid Ali
Thesis Supervisor
September, 2018

Dr. Muhammad Sagheer
Head
Dept. of Mathematics
September, 2018

Dr. Muhammad Abdul Qadir
Dean
Faculty of Computing
September, 2018

Author's Declaration

I, **Sumaira Bibi** hereby state that my MPhil thesis titled "**Modification and Improvement of Cryptosystem based on Non-Commutative Platform Groups**" is my own work and has not been submitted previously by me for taking any degree from Capital University of Science and Technology, Islamabad or anywhere else in the country/abroad.

At any time if my statement is found to be incorrect even after my graduation, the University has the right to withdraw my MPhil Degree.

(Sumaira Bibi)

Registration No: MMT-163003

Plagiarism Undertaking

I solemnly declare that research work presented in this thesis titled "*Modification and Improvement of Cryptosystem based on Non-Commutative Platform Groups*" is solely my research work with no significant contribution from any other person. Small contribution/help wherever taken has been dully acknowledged and that complete thesis has been written by me.

I understand the zero tolerance policy of the HEC and Capital University of Science and Technology towards plagiarism. Therefore, I as an author of the above titled thesis declare that no portion of my thesis has been plagiarized and any material used as reference is properly referred/cited.

I undertake that if I am found guilty of any formal plagiarism in the above titled thesis even after award of MPhil Degree, the University reserves the right to withdraw/revoke my MPhil degree and that HEC and the University have the right to publish my name on the HEC/University website on which names of students are placed who submitted plagiarized work.

(Sumaira Bibi)

Registration No: MMT-163003

Acknowledgements

First of all, I would like to thank **Almighty Allah** for His countless blessings in my life. He has gifted me a loving family and excellent teachers. He supports me in every path of life.

I would like to express my special thanks to my kind supervisor **Dr. Rashid Ali** for his motivation. His unfailing patience and encouragement kept me in good stead. I would never be able to forget his key contribution to one of the most fruitful endeavor of my life. I have appreciated the guidance for my supervisor and feeling proud to be a student of such great teacher.

Also, many thanks are due to all teachers of CUST Islamabad Dr. Muhammad Sagheer, Dr. Abdul Rehman Kashif, Dr. Shafqat Hussain, Dr. M. Afzal and Dr. Rashid Ali for their appreciation and support.

I am grateful to the management staff of Capital University of Science and Technology, Islamabad for providing a friendly environment for studies.

I am thankful to all of my family members for becoming a bulwark of patience and support during my research work. However, my deepest gratitude is reserved for my mother for her earnest prayers, unconditional love and unflinching support. I would also like to thank my brother M. Awais, and sisters for their support in completing my degree program. They supported and encouraged me throughout my life.

I would like to thank all of my friends Saba Majeed, Tehmas Zahra Kazmi, Farkhanda Shabbir, Fatima Ishfaq, and Sania Asim for supporting me during degree programs. Especially, I would like to thank Saba Majeed for motivating me during research work.

Finally, I am obliged to all people who have shared their knowledge and supported me all along.

"Mathematics reveals its secrets only to those who approach it with pure love, for its own beauty"

Archimedes

Abstract

In view of possible invent of quantum computers, there is a security threat to many modern cryptosystem that are based on hard problems in number theory like integer factorization problem. Therefore, the study of algebraic cryptosystem based on certain noncommutative structures is an active area of research in the present time. In this thesis, we review a noncommutative group based cryptosystem introduced recently by Kanwal and Ali [14]. The construction of cryptosystem is based on finite field $\mathbb{F}_p$. We mainly focused on the modification and improvement of the cryptosystem by suggesting a noncommutative structure of matrices over extended Galois field $GF(p^q)$. We implemented our modified encryption scheme using the computer algebra system ApCoCoA and formulated many examples to illustrate the scheme. It is observed that, with the modified structure we have not only increased the sizes of key space and message space but also the security against known attacks. In fact, the cryptanalysis of the modified scheme seem to be much harder than that of the actual structure.

Contents

List of Figures

List of Tables

Abbreviations

PKC	Public Key Cryptography
DLP	Discrete Log Problem
CSP	Conjugator Search Problem
PR	Private Key
PU	Public Key
IFP	Integer Factorization Problem

Symbols

M	Plaintext or Message
C	Ciphertext
E	Encryption Algorithm
D	Decryption Algorithm
PR	Private Key
$\mathbb{G}$	Group
$\mathbb{Z}$	Set of integers
$\mathbb{R}$	Set of real numbers
$\mathbb{Q}$	Rational numbers
$\mathbb{C}$	Complex numbers
$\mathbb{F}_p$,$\mathbb{Z}_p$	Finite field of order prime p
$\mathbb{F}_{p^q}$	Finite field extension
$\mathcal{K}$	Secret Key
$\mathcal{K}_{\mathcal{A}}$	Public Key of Allice

Chapter 1

Introduction

1.1 Cryptography

The safety and security of communication continued to be a major problem from very beginning. Roman generals understood some cryptographic methods and utilized a simple Shift Cipher or Caesar Cipher [6] while sending message to one another. As the time passes, new methods in cryptography were developed which provide authenticity and privacy of data. Cryptography is the practice of special tools of secret writing for its transmission over unprotected communication channels. For this purpose, we require a framework for changing original message into scrambled or ciphertext. Such frameworks are known as cryptosystems. A cryptosystem consists of five major components plaintext, ciphertext, encryption algorithm, decryption algorithm and key. Based on key design criteria of the crytosystem the subject of cryptography is divided into two main areas. One is called **symmetric key cryptography** and the other one is known as **asymmetric key cryptography**. In **Symmetric key cryptography** just a single key is used in both encryption and decryption algorithm that is just known to sender and receiver. Examples of such scheme includes DES [3], 2DES [42], Triple DES [42] and AES [24]. In this scheme the key distribution is a primary issue. When we have large number of users to communicate with each other at that point the

distribution of key among every one of the user turns into a significant issue. In 1970, researchers wants to overcome key distribution problem and their efforts become fruitful with the idea of **Public key cryptography or Asymmetric key cryptography** proposed by Diffie and Hellman [7] in 1976. It changes the way of cryptography and resolved the key distribution problem by a technique that uses two keys, such that knowledge of an encryption key is not exactly equal to the knowledge of the corresponding decryption key. The encryption key is known as public key and is freely distributed whereas the decryption key is kept secret and known as private key.

Examples of well known public key cryptosystem includes ElGamal cryptosystem [37], RSA [30] and Elliptic curve cryptosystems [11]. These modern encryption schemes are based on hard problems in "Number Theory" using the structure of some abelian groups for example see [9, 15, 40]. The hardness of these cryptosystems is based on difficulity of solving certain problems over finite abelian structure. Most common of these problems in public key cryptosystem are discrete logrithm problem (DLP) [22, 28] and Integer factorization problem (IFP) [19]. The security of well known cryptosystems RSA [30] and ElGamal [37] relies respectively on IFP and DLP over commutative group structure. Due to emerging technology, the computing power of computers increases day by day, so all of these cryptosystems are found to be suspectible to cryptographic attack known as quantum computers attack. Starting from the work of W. Shor [34, 35] these problems can be efficiently solved on quantum computers for abelian groups. Due to this all of these cryptosystems becomes less secure. As a consequence, there has been an active line of research to create and investigate new cryptosystems inview of noncommutative cryptographic platform. This line of research has been given the expansive title of noncommutative algebraic cryptography [5, 25].

1.2 Non-Commutative Algebraic Cryptography

As yet the main source for noncommutative algebraic cryptography has been noncommutative groups [10, 23]. Cryptosystems based on these objects utilize the

algebraic properties of the platform both in cryptanalysis and construction of cryptosystem. Wagner and Magyarik [41] were the first ones who used noncommutative groups in asymmetric key cryptography. Till today, research is in progress for using noncommutative groups in public key cryptography for examples see [2, 16, 26, 29, 33]. The security of cryptosystem based on noncommutative algebraic cryptography must rely on some hard problem. For instance, the conjugator search problem (CSP) [17]. Detailed survey of cryptography based on noncommutative group is given by Myannikov et al. [26].

1.3 Current research

In this research, we review a cryptosystem based on noncommutative platform groups introduced by Kanwal and Ali [14]. Their construction is based on finite field $\mathbb{F}_p$. We mainly focused on the modification and improvement of Kanwal and Ali [14] cryptosystem using a noncommutative structure of matrices over extended Galois field $GF(p^q)$. Using modified structure we have not only increased the size of key space and message space but also the security against known attacks. We constructed many examples for the illustration of our modified scheme. The algorithms for various computations of modified encryption scheme are implemented in computer algebra system ApCoCoA.

1.4 Thesis layout

The rest of the thesis is organised as follows:

1. In **Chapter 2**, we will introduce the fundamental ideas and definitions from cryptography that are required for the development of cryptosystem. Later on we will be discussing Stickel key exchange protocol.

2. In **Chapter 3**, a general framework and scheme of cryptosystem based on noncommutative platform groups [14] is described. We presented a toy example to illustrate the cryptosystem. The security claims of this cryptosystem are also highlighted.

3. In **Chapter 4**, we discussed the modified and improved form of the above cryptosystem by suggesting the use of noncommutative platform groups over extended Galois field $GF(p^q)$. The modified and improved cryptosystem is illustrated by examples.

Chapter 2

Preliminaries

In this chapter, we will present some basic definition from cryptography primitives and issues related to its key management. Furthermore, we will also highlight some basic definitions from algebra that are used in next chapters.

2.1 Cryptography

Cryptography is the science of protecting communication which transforms original message into unintelligble text or coded message for the transmission over the public networks in the presence of hackers. The original message is known as plaintext where as the coded message is called ciphertext. The plaintext is converted into ciphertext via the encryption algorithm. The ciphertext is then converted back to plaintext by the receiver or an authorized person via the decryption algorithm. Both sender and receiver use a secret information (known only to sender and receiver) for encryption and decryption algorithms, this secret informations known as a key. The entire process is called cryptosystem. The security of a cryptosystem relies only on the security of the key. On the basis of keys used, cryptographic techniques are divided into folowing two generic types.

- Private Key Cryptography

- Public Key Cryptography

2.1.1 Private Key Cryptography

Private key cryptography is also called the Symmetric key cryptography [12]. Its was the only cryptographic scheme that is used for message transmission over public networks before the development of public key encryption in 1976. In this method same key is used for both data encryption and decryption. A typical private key cryptography model is shown in Figure 2.1, in which both parties are using a common key K for data encryption and decryption which is not known to hackers.

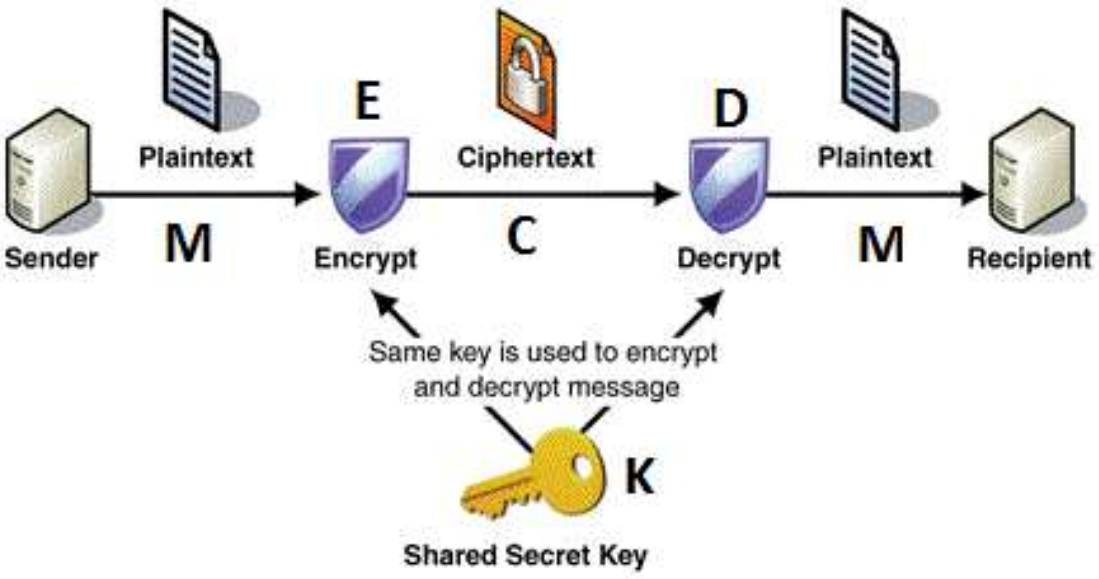

FIGURE 2.1: Private Key Cryptography.

Examples of private key cryptography includes (DES) Data Encryption Standard [42], (AES) Advanced Encryption standard [4] and Blowfish [27]. Demarits of private key cryptosystem are key management and security issues. Private encryption techniques may be categorized by either block cipher or stream cipher. Block cipher encrypt or decrypt a fixed length block of data into common block of ciphertext at a time while stream cipher encrypt or decrypt one byte of plaintext at a time.

2.1.2 Public Key Cryptography

To resolve the issues of symmetric key cryptography Diffie and Hellman [7] proposed the idea of Public key or Asymmetric key cryptography in 1976. Their notion is based on one-way trapdoor functions for transmitting the keys between two communicant. In the history of cryptography the evolution of public key cryptography is of great importance because the public key cryptosystem based on mathematical functions instead of substitution and permutation. In public key cryptography, encryption is performed by public key and decryption is performed by private key. The public key encryption scheme uses six main components as shown in Figure 2.2.

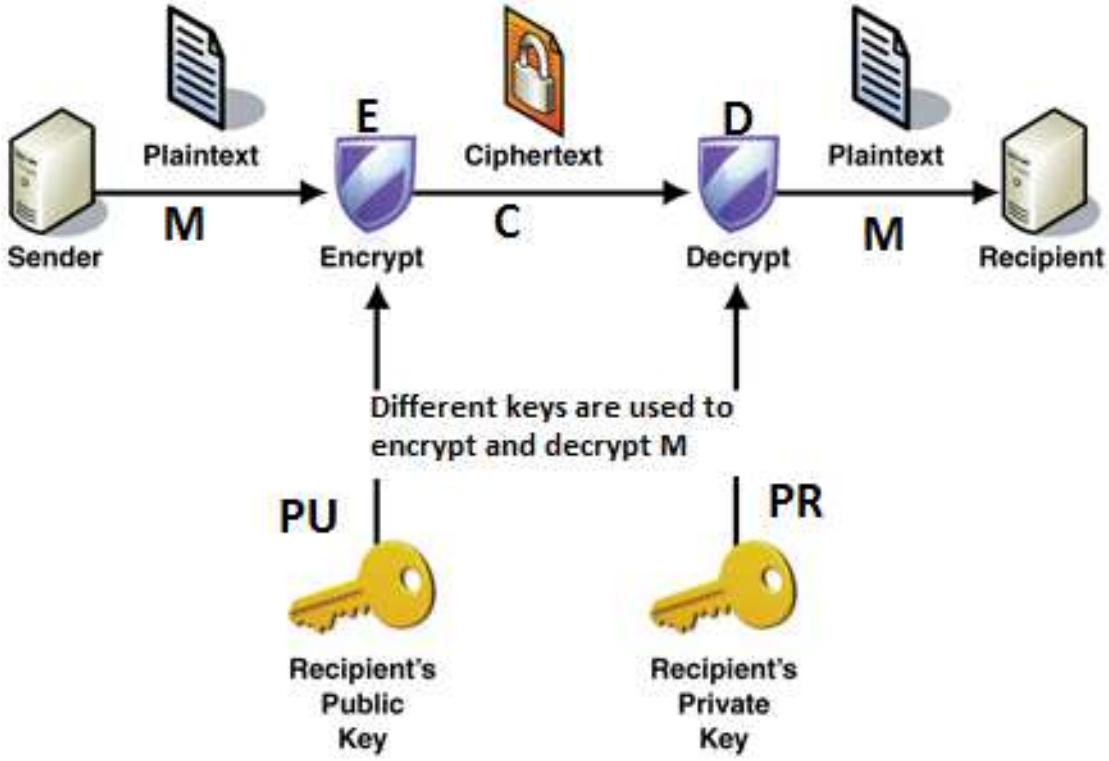

FIGURE 2.2: Public Key Cryptography.

The sender encrypts the plaintext M by using an encryption algorithm E and receiver's public key PU to get the ciphertext C. Then receiver uses his private key PR (that is only known to him) and decrypts ciphertext using decryption

algorithm D. Thus

$$C = E(PU, M) \tag{2.1}$$

$$M = D(PR, C) \tag{2.2}$$

Examples of public key cryptosystem includes RSA cryptosystem [30], ElGamal cryptosystem [37] and Elliptic curve cryptosystem [11].

2.2 Mathematical Background

In this section, we recall some basic definition from algebra that will be used in this thesis.

Definition 2.2.1 (Groups)
A **group** $\mathbb{G}$ [32] indicated by $(\mathbb{G}, *)$ is a nonempty set of elements under the binary operation $*$ that fulfills the following properties:

1. **Closure:** Binary operation $*$ is closed i.e $b * c \in \mathbb{G}$ for all $b, c \in \mathbb{G}$.

2. **Associativity:** $(b * c) * d = b * (c * d)$ for all $b, c, d \in \mathbb{G}$.

3. **Identity:** There exist an element $i \in \mathbb{G}$ that fulfills $b * i = i * b = b$ for all $b \in \mathbb{G}$. i is called the identity of $\mathbb{G}$.

4. **Inverse:** For each element $c \in \mathbb{G}$ $\exists\ c' \in \mathbb{G}$ that fulfills $c * c' = c' * c = i$. Where i is the identity element of $\mathbb{G}$.

Example 2.2.2 Following are the examples of groups.

i. Set of real numbers $\mathbb{R}$, rational number $\mathbb{Q}$, complex number $\mathbb{C}$ and integers $\mathbb{Z}$ all are group under usual binary operation addition " $+$ ".

ii. Set of real numbers $\mathbb{R} \setminus \{0\}$, rational number $\mathbb{Q} \setminus \{0\}$ and complex number $\mathbb{C} \setminus \{0\}$ are groups under usual binary operation multiplication " $\cdot$ ".

iii. Let $\mathbb{Z}_v = \{\overline{0}, \overline{1}, \overline{2}, ...\overline{v-1}\}$ be the set of residue classes under module v and $v > 0$ and $v \in \mathbb{Z}$ is group under addition $x * y = x + y$ where $x + y < v$. The binary operation " $+$ " is called addition modulo v.

iv. Set of integers $\mathbb{Z}$ does not form a group under usual multiplication due to lack of multiplicative inverses (Inverse of 3 is $\frac{1}{3}$ however $\frac{1}{3} \notin \mathbb{Z}$)

Definition 2.2.3 (Abelian Group)

A group $\mathbb{G}$ is called an abelian group [32], if it satisfies commutative law, that is for all $b, c \in \mathbb{G}$ we have $b * c = c * b$. Otherwise $\mathbb{G}$ is said to be a non-abelian group or noncommutative group.

Example 2.2.4 Following are the example of abelian groups.

i. Sets $\mathbb{Q}$, $\mathbb{R}$ and $\mathbb{Z}$ are abelian group under usual addition.

ii. Sets $\mathbb{Q} \setminus \{0\}$, $\mathbb{C} \setminus \{0\}$ and $\mathbb{R} \setminus \{0\}$ form abelian groups w.r.t usual multiplication.

iii. General Linear Group is characterized as $GL(v) = \{C \in T(v, v) | det(C) \neq 0\}$ where $T(v, v)$ is the set of all matrices of order $v \times v$. It is not an abelian group because matrix multiplication is not commutative.

Definition 2.2.5 (Order of an element of a group)

Order of an element $d \in \mathbb{G}$ is the smallest positive intger v such that $d^v = i$, where i is identity element of group $\mathbb{G}$.

Definition 2.2.6 (Ring)

A **Ring** [8] denoted by $(R, +, .)$ is a set of elements together with two binary operations addition " $+$ " and multiplication " $\cdot$ " that satisfies the following properties:

1. $(R, +)$ is an abelian group.

2. $(R, \cdot)$ is closed and associative.

3. Left and right distributive laws holds in R.

Example 2.2.7 Following are the examples of rings.

i. Set of integers $\mathbb{Z}$ under usual addition " $+$ " and multiplication " $\cdot$ " is a ring.

ii. Let $\mathbb{Z}_v = \{\overline{0}, \overline{1}, \overline{2}, ...\overline{v-1}\}$ and $v > 0$ and $v \in \mathbb{Z}$ is a ring under addition and multiplication modulo v.

Definition 2.2.8 (Field)

A triplet $\{\mathbb{F}, +, \cdot\}$ that is, a set $\mathbb{F}$ together with binary operations " $+$ " and " $\cdot$ " is called a field $\mathbb{F}$, if the following properties holds for all $a, b, c \in \mathbb{F}$:

1. $\mathbb{F}$ is an abelian group under addition having additive identity 0.

2. The nonzero elements of $\mathbb{F}$ forms an abelian group under multiplication.

3. In $\mathbb{F}$, multiplication is distributive over addition.

Example 2.2.9 Following are the examples of fields.

i. Set of real numbers $\mathbb{R}$ and set of rational numbers $\mathbb{Q}$ are field under usual addition and multiplication.

ii. For every prime p, set of residue classes $\mathbb{Z}_p$ under mod p is a field.

Definition 2.2.10 (Extension Field)

Let $\mathbb{F}$ and $\mathbb{T}$ be the two fields, then $\mathbb{F}$ is called the extension field of $\mathbb{T}$, if $\mathbb{T}$ is the subfield of $\mathbb{F}$. It is denoted by $\mathbb{F}/\mathbb{T}$.

Example 2.2.11 Followings are the examples of field extensions.

i. $\mathbb{C}$, the field of complex numbers is the extension field of $\mathbb{R}$, the field of real numbers, denoted by $\mathbb{C}/\mathbb{R}$.

ii. Let $p(r) = r^2 + r + 2 \in \mathbb{Z}_3(r)$ then there exist the extension field $\mathbb{T}$ of $\mathbb{Z}_3$ such that $\mathbb{T} = \mathbb{Z}_3(r)/\langle r^2 + r + 2\rangle$. The field $\mathbb{Z}_3(r)/\langle r^2 + r + 2\rangle$ is represented as $\{0, 1, 2, r, r+1, r+2, 2r, 2r+1, 2r+2\}$. Note that $(r^2+r+2)+(r^2+r+2) = 0$ this implies the fact $r^2 + r + 2 = 0$ so $r^2 = -r - 2 = 2r + 1$. Therefore, in $\mathbb{T}$ there exist the polynomials that are irreducible in mod $(r^2 + r + 2)$.

Definition 2.2.12 (Finite Field)

A field that contains only finitely many elements is known as a finite field.

2.3 Galois Field

Galois field is a finite field that was first introduced by the French mathematician Évariste Galois in 1830. In Galois field, the number of elements can be either prime or power of a prime. Let p be a prime number, a finite field with p elements denoted by $GF(p)$ or $\mathbb{Z}_p$ [23], consists of set of residue classes $\mathbb{Z}_p = \{\overline{0}, \overline{1}, \overline{2}, ...\overline{p-1}\}$ with arithmetic operations modulo p. As p is prime, so gcd(p,$\overline{u}$)=1 for each $\overline{u} \in \mathbb{Z}_p$.

Definition 2.3.1 (Polynomial over $GF(p)$)
A polynomial $f(r)$ with indeterminate 'r' over $GF(p)$ is an expression of the form

$$f(r) = a_i r^i + a_{i-1} r^{i-1} + \cdots + a_1 r + a_0 \qquad \text{for all} \quad i = 0, 1, \cdots, n,$$

where the coefficients a_i are taken from $GF(p)$.

Definition 2.3.2 (Irreducible polynomial)
A polynomial $m(r)$ of degree n is said to be irreducible, if it cannot be written as $m(r) = f(r)h(r)$ with non-constant polynomials $f(r)$ and $h(r)$ of degree less than n, otherwise it is known as reducible polynomial.
For example, $r^6 + r^4 + 2r^2 + 2$ is reducible polynomial over $GF(3)$, and $r^2 + 1$, $r^2 + r + 2$ are irreducible polynomials over $GF(3)$.

2.4 Extended Galois Field $GF(p^q)$

Extended Galois field $GF(p^q)$ [20, 21] is the finite field of order p^q, $\quad q \in \mathbb{Z}^+$. The elements of $GF(p^q)$ are represented by polynomials of degree less than q with coefficients from $GF(p)$, that is $GF(p^q)$ is the set of all polynomials over $GF(p)$ of degree less than q. Thus if $\mathbb{Z}_p[r]$ is the set of all polynomials over $\mathbb{Z}_p$ then $GF(p^q)$ can be represented by $\mathbb{Z}_p[r]/m(r)$ where $m(r)$ is an irreducible polynomial of degree q. In this section, we will represent elements of some Galois fields in tablular form. We have also develop a code to find the elements of $GF(p^q)$ given in Appendix A.

2.4.1 Elements of $GF(2^8)$

The elements of $GF(2^8)$ are polynomials of degree less than 8, with coefficients in $GF(2) = \{0, 1\}$ obtained by reducing the set of all polynomials modulo an irreducible polynomial of degree 8. It has 256 different elements. All the elements of $GF(2^8)$ are shown in the Table 2.1 given as follows :

Decimal	Polynomial	Binary
0	0	00000000
1	1	00000001
2	r	00000010
3	$r+1$	00000011
4	r^2	00000100
5	r^2+1	00000101
6	r^2+r	00000110
7	r^2+r+1	00000111
8	r^3	00001000
9	r^3+1	00001001
10	r^3+r	00001010
.	.	.
.	.	.
.	.	.
255	$r^7+r^6+r^5+r^4+r^3+r^2+r+1$	11111111

TABLE 2.1: Elements of Finite Field $GF(2^8)$

Note that all elements of $GF(2^8)$ have 8-bit binary representation as shown in the last column of Table 2.1.

2.4.2 Elements of $GF(2^{10})$

The elements of $GF(2^{10})$ are polynomials of degree less than 10, with coefficients in $GF(2) = \{0, 1\}$ obtained by reducing the set of all polynomials modulo an irreducible polynomial of degree 10. It has 1024 different elements with 10-bits binary representation. All the elements of $GF(2^{10})$ are shown in the Table 2.2 given as follows :

Decimal	Polynomial	Binary
0	0	0000000000
1	1	0000000001
2	r	0000000010
3	$r+1$	0000000011
4	r^2	0000000100
5	r^2+1	0000000101
6	r^2+r	0000000110
7	r^2+r+1	0000000111
8	r^3	0000001000
9	r^3+1	0000001001
10	r^3+r	0000001010
.	.	.
.	.	.
.	.	.
1024	$r^9+r^8+r^7+r^6+r^5+r^4+r^3+r^2+r+1$	1111111111

TABLE 2.2: Elements of Finite Field $GF(2^{10})$

2.5 Multiplicative Inverse in Galois field

As we have seen the elements of Galois fields are represented by polynomials. Extended Euclidean Algorithm is used to find inverse of any polynomial in Galois field $GF(p^q)$. By using this algorithm, we can find the multiplicative inverse of any polynomial $z(r) \in GF(p^q)$ modulo an irreducible polynomial $m(r)$, when $\gcd(z(r), m(r))= 1$. To find the inverse of $z(r) \mod m(r)$, the following steps are to be performed.

Algorithm 2.5.1 (Extended Euclidean Inverse algorithm)
Input: A polynomial $z(r)$ and an irreducible polynomial $m(r)$.
Output: $z^{-1}(r) \mod m(r)$.

1. Initialize six polynomials $A_i(r)$ and $B_i(r)$ for $i = 1, 2, 3$ as
   $(A_1(r), A_2(r), A_3(r)) = (1, 0, m(r))$
   $(B_1(r), B_2(r), B_3(r)) = (0, 1, z(r))$

2. If $B_3(r) = 0$, return $A_3(r)$=$\gcd(z(r), m(r))$; no inverse of $z(r)$ exist in mod $m(r)$

3. If $B_3(r) = 1$, so return $B_3(r)$=gcd$(z(r), m(r))$ and
$B_2(r) = z^{-1}(r) \mod m(r)$

4. Now divide $A_3(r)$ with $B_3(r)$ also find the quotient $Q(r)$ when $A_3(r)$ is divided by $B_3(r)$.

5. Set $(T_i(r) = (A_i(r) - Q(r).B_i(r))$; $i = 1, 2, 3$.

6. Set $(A_1(r), A_2(r), A_3(r)) = (B_1(r), B_2(r), B_3(r))$

7. Set $(B_1(r), B_2(r), B_3(r)) = (T_1(r), T_2(r), T_3(r))$

8. Goto step number 2.

For further details we refer to [42].

Definition 2.5.2 (Discrete logrithm problem DLP)
Let h be the generator of a finite cyclic group $\mathbb{G}$ and $b = h^m$ for some integer m the discrete log of b to the base h, is integer m. The problem of finding m, given $h, b \in \mathbb{G}$, is called discrete log problem [22, 28].

Definition 2.5.3 (Integer factorization problem IFP)
Let n be a given number, the problem of decomposition of n to the product of smaller primes p and q such that $n = pq$ is called integer factorization problem [19].

Definition 2.5.4 (Conjugator Search Problem)
Given elements x,y in a noncommutative group $\mathbb{G}$, the problem of determining the conjugator $z \in \mathbb{G}$ such that $x^z = z^{-1}xz = y$ is known as conjugator search problem (CSP) [17].

2.6 Stickel's key exchange protocol

A key exchange protocol based on noncommutative groups is proposed by Stickel [39] in 2005. The technique is easy to implement and would give more elevated amount of security. The method given in [39] is the generalization of Diffie and

Hellman technique to noncommutative groups. Main scheme of Stickel is explained in the following algorithm.

Algorithm 2.6.1 [39] [Stickel's Key Exchange]
Let $\mathbb{G}$ be a finite noncommutative group and $a, b \in \mathbb{G}$. let n_1 and n_2 be orders of a and b respectively. The underlying steps will explain the transmission of private key $\mathcal{K}$.

1. Bob chooses two random secret positive integers $s < n_1$ and $t < n_2$. He then calculates $d = a^s b^t$ and sends this outcome to Allice.

2. Allice chooses two random secret positive integers $u < n_1$ and $v < n_2$. She then calculates $e = a^u b^v$ and sends this outcome to Bob.

3. Bob calculate the secret key by using the formula $\mathcal{K} = a^s e b^t$.

4. Allice similarly calculate $\mathcal{K}$ as $\mathcal{K} = a^u d b^v$.

For details see [39]

2.6.1 A Variation of Stickel key exchange protocol

A variation of above public key exchange protocol as recommended by Stickel [39] can likewise be excuted in which the following steps are performed to transmit a secret key $\mathcal{K}$.

1. Bob chooses two secret positive integers s and t randomly such that $s < n_1$ and $t < n_2$. He then calculates $d = a^s b^t$ and sends this result to Allice.

2. In this step Allice randomly chooses two secret positive integers u and v with $u < n_1$ and $v < n_2$. Allice calculates $\mathcal{K} = a^u b^v$ and $m = a^u d b^v$
   m is transmitted to Bob and $\mathcal{K}$ is secret key.

3. Bob calculates the secret key $\mathcal{K}$ as $\mathcal{K} = a^{-s} m b^{-t}$.

In above key exchange method Stickel suggested to utilize group of invertible matrices over finite field. Several techniques based on matrix have an incredible capability to be utilized in cryptography. Different matrix based structures are

also used in cryptography for example [13, 18]. Shpilrain [36] and Sramka [38] examined the cryptanalysis of the Stickel scheme [39]. Sramka has presented an attack aimed to uncovers the secret exponents s, t, u and v. Shpilrain offered an efficient attack aimed to get the secret key $\mathcal{K}$. Due to these attacks Stickel key exchange protocol becomes less secure.

Chapter 3

Cryptosystem based on Non-Commutative Platform Groups

In this chapter, we review a new scheme which was proposed by Kanwal and Ali in [14] and is known as cryptosystem based on noncommutative platform groups.

3.1 The Proposed Cryptosystem

Several cryptosystems can be designed by utilizing the key establishment protocols. For example ElGamal [37] cryptosystem utilizes Diffie and Hellman key exchange scheme [7]. Here we describe the cryptosystem proposed by Kanwal and Ali in [14]. For designing the public key cryptosystem the authors utilized variation of Stickel key exchange protocol 2.6.1. The proposed cryptosystem can easily be implemented in any noncommutative structure which can be considered as its important feature.

The general scheme as proposed in [14] is explained and described as follow.

Setup: Suppose $\mathbb{G}$ is a noncommutative group that employs computational difficulty of DLP and CSP. Elements of group $\mathbb{G}$ has large order. Suppose a and b are elements of $\mathbb{G}$. Let n_1 be the order of a and n_2 be the order of b.

Algorithm 3.1.1 [14] (Proposed Cryptosystem)
Suppose that Bob and Allice wants secure communication through public channel. The communication between Bob and Allice goes as explained below.

Generation of key Allice chooses two random positive integers $s < n_1$ and $t < n_2$. The secret key PR_A of Allice is the pair (s, t). She computes $\mathcal{K}_\mathcal{A}$ as

$$\mathcal{K}_\mathcal{A} = a^s b^t.$$

Public key PU_A of Allice is given by

$$PU_A = (a, b, \mathcal{K}_\mathcal{A}).$$

Encryption process: Bob chooses two random positive integers $u < n_1$ and $v < n_2$ to send a plaintext $m \in \mathbb{G}$ to Allice. For encryption purpose Bob inputs plaintext m and public key $PU_A = (a, b, \mathcal{K}_\mathcal{A})$ of Allice. Then, he calculate a^u, b^v and performs the following sequence of steps to get the ciphertext.

1. Bob utilizes a^u, b^v and he computes y as

$$y = a^u b^v.$$

2. He finds inverse of y and then calculates C_1 and C_2 as

$$C_1 = {\mathcal{K}_\mathcal{A}}^y = y^{-1}\mathcal{K}_\mathcal{A} y.$$

$$C_2 = {\mathcal{K}_\mathcal{A}}^{y^{-1}} = y\mathcal{K}_\mathcal{A} y^{-1}.$$

3. After computing C_1 and C_2, he forms the part of the ciphertext C as

$$C = C_1 m C_2. \tag{3.1}$$

4. Then calculates C' as

$$C' = a^u \mathcal{K}_{\mathcal{A}} b^v. \tag{3.2}$$

The ciphertext transmitted to Allice is the pair (C', C).

Algorithm 3.1.2 Decryption: When Allice gets the ciphertext pair (C', C), she performs following steps for decryption to get original message. For this purpose she inputs ciphertext pair (C', C), her private key $PR_A = (s, t)$ and execute following steps.

1. In step 1, Allice form D as

$$D = a^{-s} C' b^{-t}. \tag{3.3}$$

2. She then computes inverse of D and calculates D_1 and D_2 as

$$D_1 = ({\mathcal{K}_{\mathcal{A}}}^{-1})^D = D^{-1} {\mathcal{K}_{\mathcal{A}}}^{-1} D. \tag{3.4}$$

$$D_2 = ({\mathcal{K}_{\mathcal{A}}}^{-1})^{D^{-1}} = D {\mathcal{K}_{\mathcal{A}}}^{-1} D^{-1}. \tag{3.5}$$

3. Allice can get the secret message m as

$$m = D_1 C D_2. \tag{3.6}$$

Correctness: The correctness of the above scheme is acknowledged as follows: From 3.3 we have,

$$D = (a^s)^{-1} C' (b^t)^{-1}$$

Using equations 3.2 and 3.1 in 3.3, we have

$$\begin{aligned} D &= (a^s)^{-1}(a^u \mathcal{K}_{\mathcal{A}} b^v)(b^t)^{-1}, \\ &= (a^s)^{-1}(a^u a^s b^t b^v)(b^t)^{-1}, \\ &= (a^s)^{-1}(a^{u+s} b^{v+t})(b^t)^{-1}, \\ &= a^u b^v, \\ &= y. \end{aligned}$$

Now the equations 3.4 and 3.5 can be written as,

$$D_1 = (\mathcal{K_A}^{-1})^D = y^{-1}\mathcal{K_A}^{-1}y$$
$$D_2 = (\mathcal{K_A}^{-1})^{D^{-1}} = yK_A^{-1}y^{-1}$$

In view of above we can write equation 3.6 as

$$\begin{aligned} D_1CD_2 &= [y^{-1}K_A^{-1}y]C[yK_A^{-1}y^{-1}], \\ &= [y^{-1}K_A^{-1}y]C_1mC_2[yK_A^{-1}y^{-1}], \\ &= [y^{-1}K_A^{-1}y]K_A^{y}mK_A^{y^{-1}}[yK_A^{-1}y^{-1}], \\ &= [y^{-1}K_A^{-1}y][y^{-1}K_Ay]m[yK_Ay^{-1}][yK_A^{-1}y^{-1}], \\ &= m. \end{aligned}$$

The authors [14] suggest the "Use of $GL(n, \mathbb{F}_p)$ for implementing the proposed scheme, where $GL(n, \mathbb{F}_p)$ is the general linear group of matrices of order n over the finite field $\mathbb{F}_p$ ". The complexity of the above cryptosystem depends both on matrix conjugator search problem 2.5.4 and discrete log problem 2.5.2.

3.2 A toy example

Choose $G = GL(3, \mathbb{F}_{103})$, that is the matrices of order 3 over $\mathbb{F}_{103}$ for using Algorithm 3.1.1.

Initial step: Let us consider two matrices

$$A = \begin{bmatrix} 31 & 57 & 47 \\ 95 & 63 & 23 \\ 21 & 19 & 13 \end{bmatrix} \in GL(3, \mathbb{F}_{103})$$

and

$$B = \begin{bmatrix} 21 & 46 & 17 \\ 69 & 24 & 27 \\ 33 & 18 & 51 \end{bmatrix} \in GL(3, \mathbb{F}_{103})$$

using computer algebra system ApCoCoA, we compute the orders n_1 and n_2 of the two matrices as

$$n_1 = |A| = 1326 \quad \text{and} \quad n_2 = |B| = 10608.$$

Key Generation: To obtain the public and secret keys, Allice chooses two random positive integers $s = 23 < 1326$ and $t = 31 < 10608$. She then calculates

$$\begin{aligned} \mathcal{K}_{\mathcal{A}} &= A^{23}B^{31} \mod 103, \\ &= \begin{bmatrix} 64 & 101 & 96 \\ 36 & 45 & 23 \\ 95 & 0 & 34 \end{bmatrix} \mod 103 \end{aligned}$$

The secret key is $PR_A = (23, 31)$ and public key is $PU_A = (A, B, \mathcal{K}_{\mathcal{A}})$.

Encryption:

To send a plaintext

$$M = \begin{bmatrix} 99 & 11 & 32 \\ 41 & 96 & 83 \\ 75 & 60 & 44 \end{bmatrix} \in GL(3, \mathbb{F}_{103})$$

to Allice, Bob chooses randomly $u = 45 < n_1$ and $v = 26 < n_2$. For encryption purpose Bob inputs plaintext M and public key $PU_A = (A, B, \mathcal{K}_{\mathcal{A}})$ of Allice. The following sequence of steps will be executed to get the ciphertext.

1. He calculates A^{45}, B^{26} to find Y as

$$\begin{aligned} Y &= A^{45}B^{26} \\ &= \begin{bmatrix} 1 & 35 & 65 \\ 44 & 36 & 32 \\ 28 & 58 & 45 \end{bmatrix} \mod 103 \end{aligned}$$

Now he calculates inverse of Y by using $Y^{-1} = Adj(Y)/det(Y)$. First he

finds $det(Y) = 75$, calculates its inverse by using Extended Euclidean Algorithm 2.5.1 as given in Table 3.1

Q	A_1	A_2	A_3	B_1	B_2	B_3
-	1	0	103	0	1	75
1	0	1	75	1	-1	28
2	1	-1	28	-2	3	19
1	-2	3	19	3	-4	9
2	3	-4	9	-8	11	1

TABLE 3.1: Extended Euclidean Algorithm $(75)^{-1} \mod 103$

So, $(75)^{-1} \mod 103 = 11$ and multiply inverse of $det(Y)$ with $Adj(Y)$ to get inverse of Y as

$$Y^{-1} = \begin{bmatrix} 82 & 43 & 73 \\ 24 & 45 & 2 \\ 92 & 48 & 39 \end{bmatrix} \mod 103$$

2. After finding Y^{-1} he forms C_1 by utilizing public key $\mathcal{K}_{\mathcal{A}}$ of Allice as

$$\begin{aligned} C_1 &= (Y^{-1})\mathcal{K}_{\mathcal{A}}Y \mod 103 \\ &= \begin{bmatrix} 40 & 19 & 9 \\ 21 & 50 & 27 \\ 56 & 30 & 53 \end{bmatrix} \mod 103 \end{aligned}$$

3. And also computes C_2 as,

$$\begin{aligned} C_2 &= Y\mathcal{K}_{\mathcal{A}}(Y^{-1}) \mod 103 \\ &= \begin{bmatrix} 37 & 69 & 33 \\ 7 & 48 & 51 \\ 52 & 70 & 58 \end{bmatrix} \mod 103 \end{aligned}$$

4. The ciphertext C is computed by multiplying C_1, plaintext M and C_2 respectively. The output is

$$C = C_1 M C_2 \mod 103$$
$$= \begin{bmatrix} 71 & 36 & 78 \\ 87 & 56 & 16 \\ 11 & 91 & 86 \end{bmatrix} \mod 103$$

5. He also calculates C' as,

$$C' = A^{45} \mathcal{K}_{\mathcal{A}} B^{26} \mod 103$$
$$= \begin{bmatrix} 18 & 45 & 60 \\ 84 & 10 & 75 \\ 56 & 16 & 94 \end{bmatrix} \mod 103$$

The ciphertext transmitted to Allice is the pair (C', C).

Decryption: When Allice gets the ciphertext pair (C', C), she computes following steps for decryption to get the original message. For this purpose, she inputs ciphertext pair (C', C), her private key $PR_A = (23, 31)$ and performs following calculations.

1. In step 1, Allice computes A^{-23} by taking inverse of A^{23}. For this she first computes the determinant $det(A^{23}) = 53$, calculate its inverse by using Extended Euclidean Algorithm 2.5.1 as:

Q	A_1	A_2	A_3	B_1	B_2	B_3
-	1	0	103	0	1	53
1	0	1	53	1	-1	50
1	1	-1	50	-1	2	3
16	-1	2	3	17	-33	2
1	17	-33	2	-18	35	1

TABLE 3.2: Extended Euclidean Algorithm $(53)^{-1} \mod 103$

So, $(53)^{-1} \mod 103 = 35$ and multiply inverse of $det(A^{23})$ with $Adj(A^{23})$ to get A^{-23} as

$$A^{-23} = \begin{bmatrix} 14 & 85 & 49 \\ 87 & 69 & 27 \\ 86 & 57 & 26 \end{bmatrix} \mod 103$$

she computes B^{-31} by taking inverse of B^{31}. For this she first computes the determinant $det(B^{31}) = 38$, calculate its inverse by using Extended Euclidean Algorithm 2.5.1 calculations of finding $38^{-1} \mod 103$ are given in Table 3.3

Q	A_1	A_2	A_3	B_1	B_2	B_3
-	1	0	103	0	1	38
2	0	1	38	1	-2	27
1	1	-2	27	-1	3	11
2	-1	3	11	3	-8	5
2	3	-8	5	-7	19	1

TABLE 3.3: Extended Euclidean Algorithm $(38)^{-1} \mod 103$

So, $(38)^{-1} \mod 103 = 19$ and multiply inverse of $det(B^{31})$ with $Adj(B^{31})$ to get B^{-31} as given below

$$B^{-31} = \begin{bmatrix} 22 & 96 & 48 \\ 34 & 40 & 66 \\ 100 & 13 & 20 \end{bmatrix} \mod 103$$

and then she calculates

$$\begin{aligned} D &= A^{-23} C' B^{-31} \\ &= \begin{bmatrix} 1 & 35 & 65 \\ 44 & 36 & 32 \\ 28 & 58 & 45 \end{bmatrix} \mod 103 \end{aligned}$$

2. Now she calculates inverse of D by using $D^{-1} = Adj(D)/det(D)$. First she computes $det(D) = 75$, calculates its inverse by using Extended Euclidean Algorithm 2.5.1 from Table 3.1 $(75)^{-1} \mod 103 = 11$ and multiply inverse of $det(D)$ with $Adj(D)$ to get inverse of D as

$$D^{-1} = \begin{bmatrix} 82 & 43 & 73 \\ 24 & 45 & 2 \\ 92 & 48 & 39 \end{bmatrix} \mod 103$$

Similarly she computes inverse of $\mathcal{K}_{\mathcal{A}}$ as given below

$$\mathcal{K}_{\mathcal{A}}{}^{-1} = \begin{bmatrix} 16 & 3 & 77 \\ 53 & 39 & 33 \\ 28 & 31 & 3 \end{bmatrix} \mod 103$$

And then she calculates D_1 and D_2 as

$$\begin{aligned} D_1 &= D^{-1}\mathcal{K}_{\mathcal{A}}{}^{-1}D \mod 103 \\ &= \begin{bmatrix} 63 & 72 & 77 \\ 7 & 41 & 44 \\ 83 & 97 & 57 \end{bmatrix} \mod 103 \end{aligned}$$

$$\begin{aligned} D_2 &= D\mathcal{K}_{\mathcal{A}}{}^{-1}D^{-1} \mod 103 \\ &= \begin{bmatrix} 35 & 95 & 99 \\ 90 & 22 & 36 \\ 66 & 41 & 1 \end{bmatrix} \mod 103 \end{aligned}$$

4. Allice can get the secret message M as

$$\begin{aligned} M &= D_1 C D_2 \mod 103 \\ &= \begin{bmatrix} 99 & 11 & 32 \\ 41 & 96 & 83 \\ 75 & 60 & 44 \end{bmatrix} \mod 103 \end{aligned}$$

3.3 Security claims

In, this section we will present the security claims of proposed cryptosystem. The scheme cover following features. To get adequate security and prevent the attacks on stickel main key exchange protocol [14] utilized variation of stickel key exchange protocol. The public matrices A and B has order n and can be choosen so that $2^n - 1$ is Mersenne prime. So, the product $A^s B^t$ has $(2^n - 1)^2$ different forms.

A hacker has to search this size of space for brute force attack. To extend space size the author suggest $n > 31$. The matrices A and B are selected so that these matrices have irreducible characteristic polynomials. By selecting such types of matrices the eigenvalue and eigenvector attacks becomes infeasible. Further, in encryption algorithm the matrices C_1 and C_2 involved are matrix conjugate of the matrix K_A with secret matrix Y. The problem of finding these matrices is conjugacy search problem. For directing conjugation attacks, a attacker first has to discover unknown matrices C_1 and C_2. That is the reason these kinds of attacks also remains infeasible.

3.3.1 Ciphertext only Attack

In ciphertext only Attack, the adversary is assumed to have access only to a set of ciphertext encrypted with secret key. His goal is to obtained the corresponding plaintext and even better the secret cipher key. Assume an adversary knows just the ciphertext pair (C', C). Equation (3.1) involves the product of three unknown matrices C_1, m and C_2. To get C_1 and C_2 an attacker has to solve the following system of equations.

$$\begin{cases} YC_1 = \mathcal{K}_{\mathcal{A}}Y \\ C_2Y = Y\mathcal{K}_{\mathcal{A}} \end{cases} \tag{3.7}$$

Where Y is unknown matrix. For proposed structure $GL(n, \mathbb{F}_p)$, system of equations 3.7 will translate into a large system of equations. The equation 3.2 involve the discrete log problem of the matrices A and B. The discrete log problem of propose cryptosystem relies upon the solutions of discrete log problem on $F_{p^{n^2}}$. When the field is large DLP becomes harder. Due to these facts, this kind of attacks ends up unworkable.

3.3.2 Known plaintext attack

In Known plainext attack the attacker is accepted to approach both plaintext and scrambled ciphertext. His goal is to reveal secret key. Assume the attacker knows the ciphertext (C'_j, C_j) corresponding to the plaintext M_j. From this data, he needs to discover next plaintext M_{j+1} from relating ciphertext (C'_{j+1}, C_{j+1}). This kind of attack can be made unworkable when we utilize different exponents u and v in each new message encryption process.

Chapter 4

A Modified Cryptosystem

In this chapter, we will present and discuss an improved form of the cryptosystem proposed in [14]. For this purpose we aim to use General linear group of matrices over Galois Field $GF(p^q)$. We explain this improvement with examples. The key generation algorithm, the encryption algorithm and the decryption algorithm for the new improved cryptosystem are implemented using the computer algebra system ApCoCoA [1]. Designed code for this cryptosystem is given in Appendix A.

4.1 Modified and Improved form of Cryptosystem

In this section, we will describe the modified and improved form of cryptosystem that was explained in Chapter 2. For this purpose first we will choose matrices over Galois Field $GF(p^q)$ as the platform of group that employs computational difficulty of the discrete log problem (DLP) and conjugator search problem (CSP). Sketch of the modified and improved scheme [14] is explained as follows.

Let $\mathbb{G}$ be a noncommutative group of matrices over Galois Field $GF(p^q)$ and elements in $\mathbb{G}$ has large order. Suppose A and B are elements of $GL(n, \mathbb{F}_{p^q})$ such that $AB \neq BA$ of very large order. Let n_1 be the order of matrix A and n_2

be the order of matrix B. Fix a polynomial $m(r)$ of degree q. All polynomial multiplications are reduced modulo an irreducible polynomial $m(r)$ of degree q.

Algorithm 4.1.1 (Modified and Improved Cryptosystem)
Suppose that Bob and Allice wants secret communication through public channel. The communication between them consists of some steps that are described below.

Generation of Key: Allice chooses two random positive integers $s < n_1$ and $t < n_2$. The ordered pair $PR_A = (s, t)$ is the secret key of Allice. She then computes $\mathcal{K}_{\mathcal{A}}$ as

$$\mathcal{K}_{\mathcal{A}} = A^s B^t \mod m(r). \tag{4.1}$$

and publish $PU_A = (A, B, \mathcal{K}_{\mathcal{A}})$ at public channels.

Encryption process: Bob chooses two random positive integers u and v such that $u < n_1$ and $v < n_2$ to send a plaintext $M \in GL(n, \mathbb{F}_{p^q})$ to Allice. For encryption purpose Bob inputs plaintext M and public key $PU_A = (A, B, \mathcal{K}_{\mathcal{A}})$ of Allice. He calculates A^u, B^v and performs the following sequence of steps to get the ciphertext.

1. Bob utilizes A^u, B^v and he then calculates Y as

$$Y = A^u B^v \mod m(r) \tag{4.2}$$

2. He finds inverse of Y using Extended Euclidean Algorithm and calculates C_1 by multiplying Y^{-1}, $\mathcal{K}_{\mathcal{A}}$ and Y. That is

$$C_1 = {\mathcal{K}_{\mathcal{A}}}^Y = Y^{-1}\mathcal{K}_{\mathcal{A}}Y \mod m(r) \tag{4.3}$$

3. And compute C_2 as

$$C_2 = {\mathcal{K}_{\mathcal{A}}}^{Y^{-1}} = Y\mathcal{K}_{\mathcal{A}}Y^{-1} \mod m(r) \tag{4.4}$$

4. After computing C_1 and C_2, he forms ciphertext C as,

$$C = C_1 M C_2 \mod m(r) \tag{4.5}$$

5. He also computes C'

$$C' = A^u \mathcal{K}_{\mathcal{A}} B^v \mod m(r) \tag{4.6}$$

The ciphertext transmitted to Allice is the pair (C', C).

Algorithm 4.1.2 Decryption: When Allice gets the ciphertext pair (C', C), she computes following steps for decryption to get original message. For this purpose, she inputs ciphertext pair (C', C), her private key $PR_A = (s, t)$ and performs following calculations.

1. In step 1, Allice calculates A^{-s}, B^{-t} and then computes

$$D = A^{-s} C' B^{-t} \mod m(r) \tag{4.7}$$

2. She then computes inverse of D, ${\mathcal{K}_{\mathcal{A}}}^{-1}$ and calculates D_1 by multiplying D^{-1}, ${\mathcal{K}_{\mathcal{A}}}^{-1}$ and D that is

$$D_1 = ({\mathcal{K}_{\mathcal{A}}}^{-1})^D = D^{-1} {\mathcal{K}_{\mathcal{A}}}^{-1} D \mod m(r) \tag{4.8}$$

3. And also calculate

$$D_2 = ({\mathcal{K}_{\mathcal{A}}}^{-1})^{D^{-1}} = D {\mathcal{K}_{\mathcal{A}}}^{-1} D^{-1} \mod m(r) \tag{4.9}$$

4. Allice can get the secret message M as

$$M = D_1 C D_2 \mod m(r). \tag{4.10}$$

4.2 Suggested parameters of the improved cryptosystem

In [14] authors suggested "The use of $GL(n, \mathbb{F}_p)$ for implementing their scheme, where $GL(n, \mathbb{F}_p)$ is general linear group of matrices of order n over the finite field $\mathbb{F}_p$". As we know the cryptosystem is more secure if key space is large enough. The most obvious reason for modification and improvement of that cryptosystem

is enlargement of key space and message space as $GF(p^q)$ is the extension of $GF(p)$. In [31], we have $L = |GL(n, \mathbb{F}_{p^q})| = \prod_{j=0}^{n-1}((p^q)^n - (p^q)^j)$ which is utilized to calculate size of space $GL(n, \mathbb{F}_{p^q})$ so we have size L space for selecting matrices A and B. Moreover, solution of DLP and CSP becomes harder as the field gets enormous.

4.3 Illustrative Examples

In this section we will explain the modification and improvement of cryptosystem based on noncommutative platform groups through examples.

Example 4.3.1 Let us take matrices of order 2 over Galois field $GF(2^8)$ for implementing modified and improved scheme. As we know in Galois field addition and multiplication is performed under certain irreducible polynomial. let us take $m(r) = (r^8 + r^4 + r^3 + r + 1)$ as irreducible polynomial and all the calculation given below are performed under mod $m(r)$.

Initial step: let us consider two matrices

$$A = \begin{bmatrix} r^7 + r^5 + 1 & r^2 + 1 \\ r^2 + r + 1 & r^7 + r^6 + r^5 \end{bmatrix} \in GL(2, GF(2^8))$$

and

$$B = \begin{bmatrix} r^3 + 1 & r^4 + r \\ r^7 + r^4 & r^6 + r^3 \end{bmatrix} \in GL(2, GF(2^8))$$

using computer algebra system ApCoCoA, we compute the orders n_1 and n_2 of the two matrices as

$$n_1 = |A| = 65535 \quad \text{and} \quad n_2 = |B| = 255.$$

Key Generation: To obtain the public and secret keys, Allice chooses two random positive integers $s = 59 < 65535$ and $t = 43 < 255$. She then calculates

$$\begin{aligned} \mathcal{K}_{\mathcal{A}} &= A^{59}B^{43} \mod (m(r)) \\ &= \begin{bmatrix} r^6 + r^5 + r^3 + r^2 & r^6 + r \\ r^6 + r^5 + r^3 + r + 1 & r^4 + r^3 + r^2 + r + 1 \end{bmatrix} \end{aligned}$$

The secret key is $PR_A = (59, 43)$, and public key is $PU_A = (A, B, \mathcal{K}_\mathcal{A})$.

Encryption:

To send a plaintext

$$M = \begin{bmatrix} r+1 & r^2+r \\ r^3 & 1 \end{bmatrix} \in GL(2, GF(2^8))$$

to Allice, Bob chooses randomly $u = 71 < n_1$ and $v = 39 < n_2$. For encryption purpose Bob inputs plaintext M and public key $PU_A = (A, B, \mathcal{K}_\mathcal{A})$ of Allice. The following sequence of steps will be executed to get the ciphertext.

1. He calculates A^{71}, B^{39} to find Y as

$$\begin{aligned} Y &= A^{71}B^{39} \mod (m(r)) \\ &= \begin{bmatrix} r^6+r^5+r^4+r & r^7+r^6+r^3+r^2+1 \\ r^6+r^4+r^2+1 & r^6+1 \end{bmatrix} \end{aligned}$$

Now he calculates inverse of Y by using $Y^{-1} = Adj(Y)/det(Y)$. First he find $det(Y) = r^7+r^6+r^4+r^2+r$, calculates its inverse by using Extended Euclidean Algorithm 2.5.1 as $(r^7+r^6+r^4+r^2+r)^{-1} \mod m(r) = r^7+r^6+r^5+r$ and multiply inverse of $det(Y)$ with $Adj(Y)$ to get inverse of Y as

$$Y^{-1} = \begin{bmatrix} r^6+r^4+r^3+r^2 & r^7+r^6+r^3+r^2+r+1 \\ r^6+r^4+r^3+r+1 & r^7+r^4+r^3+r^2 \end{bmatrix}$$

2. After finding Y^{-1} he forms C_1 by utilizing $\mathcal{K}_\mathcal{A}$ of Allice as

$$\begin{aligned} C_1 &= (Y^{-1})\mathcal{K}_\mathcal{A}Y \mod (m(r)) \\ &= \begin{bmatrix} r^7+r^3+1 & r^7+r^3 \\ r^6+r^4 & r^7+r^6+r^5+r^4+r^3+r \end{bmatrix} \end{aligned}$$

3. And also computes C_2 as,

$$\begin{aligned} C_2 &= Y\mathcal{K}_\mathcal{A}(Y^{-1}) \mod (m(r)) \\ &= \begin{bmatrix} r^7+r^3 & r^5+r^4+r+1 \\ r^7+r^5+r^3 & r^7+r^6+r^5+r^4+r^3+r+1 \end{bmatrix} \end{aligned}$$

4. The ciphertext C is computed by multiplying C_1, plaintext M and C_2 respectively. The output is

$$\begin{aligned} C &= C_1MC_2 \mod (m(r)) \\ &= \begin{bmatrix} r^7+r^5+r^3+r^2 & r^7+r^6+r^5+r^4+1 \\ r^7+r^6+r^5+r^3+r+1 & r^7+r^2+r \end{bmatrix} \end{aligned}$$

5. He also calculates $C^{'}$ as,

$$\begin{aligned} C^{'} &= A^{71}\mathcal{K}_{\mathcal{A}}B^{39} \mod (m(r)) \\ &= \begin{bmatrix} r^7+r^6+r^3+r^2 & r+1 \\ r^5+r^3+r+1 & r^6+r^4+r^3+r^2+r \end{bmatrix} \end{aligned}$$

The ciphertext transmitted to Allice is the pair $(C^{'}, C)$.

Decryption: When Allice gets the ciphertext pair $(C^{'}, C)$, she computes following steps for decryption to get the original message. For this purpose, she inputs ciphertext pair $(C^{'}, C)$, her private key $PR_A = (59, 43)$ to perform the following calculations.

1. In step 1, Allice computes A^{-59} by taking inverse of A^{59}. For this she first computes $det(A^{59}) = r^7 + r^6$, calculate its inverse by using Extended Euclidean Algorithm 2.5.1 as $(r^7+r^6)^{-1} \mod (m(r)) = r^3+r+1$ and multiply inverse of $det(A^{59})$ with $Adj(A^{59})$ to get A^{-59} as

$$A^{-59} = \begin{bmatrix} r^6+r^4+r^2+r+1 & r^7+r^5+r^2+r \\ r^4+r+1 & r^7+r^6+r^4+1 \end{bmatrix}$$

Similarly she computes inverse of B^{43} as given below

$$B^{-43} = \begin{bmatrix} r^6+r^5+r^4+r+1 & r^4+r \\ r^7+r^4 & r^5+r^4+r \end{bmatrix}$$

and then she calculates

$$\begin{aligned} D &= A^{-59}C'B^{-43} \mod (m(r)) \\ &= \begin{bmatrix} r^6+r^5+r^4+r & r^7+r^6+r^3+r^2+1 \\ r^6+r^4+r^2+1 & r^6+1 \end{bmatrix} \end{aligned}$$

2. Now she calculates inverse of D by using $D^{-1} = Adj(D)/det(D)$. First she finds $det(D) = r^7 + r^6 + r^4 + r^2 + r$, calculates its inverse by using Extended Euclidean Algorithm 2.5.1 as $(r^7+r^6+r^4+r^2+r)^{-1} \mod (m(r)) = r^7 + r^6 + r^5 + r$ and multiply inverse of $det(D)$ with $Adj(D)$ to get inverse of D as

$$D^{-1} = \begin{bmatrix} r^6+r^4+r^3+r^2 & r^7+r^6+r^3+r^2+r+1 \\ r^6+r^4+r^3+r+1 & r^7+r^4+r^3+r^2 \end{bmatrix}$$

Similarly she computes inverse of $\mathcal{K}_{\mathcal{A}}$ as given below

$$\mathcal{K}_{\mathcal{A}}{}^{-1} = \begin{bmatrix} r^6+r^4+r^3+r^2 & r^7+r^6+r^3+r^2+r+1 \\ r^6+r^4+r^3+r+1 & r^7+r^4+r^3+r^2 \end{bmatrix}$$

and then she calculates D_1 as

$$\begin{aligned} D_1 &= D^{-1}\mathcal{K}_{\mathcal{A}}{}^{-1}D \mod (m(r)) \\ &= \begin{bmatrix} r^6+r & r^7+r^6+r^4+r^3+r \\ r^5+r+1 & r^6+r^4+r \end{bmatrix} \end{aligned}$$

3. And also calculates

$$\begin{aligned} D_2 &= D\mathcal{K}_{\mathcal{A}}{}^{-1}D^{-1} \mod (m(r)) \\ &= \begin{bmatrix} r^7+r^6+r^3+r & r^6+r^5+r^3+r+1 \\ r^6+r^5+r^3+r & r^7+r^6+r^4+r^3+r \end{bmatrix} \end{aligned}$$

4. Allice can get the secret message M as

$$\begin{aligned} M &= D_1CD_2 \mod (m(r)) \\ &= \begin{bmatrix} r+1 & r^2+r \\ r^3 & 1 \end{bmatrix} \end{aligned}$$

Example 4.3.2 Let us take matrices of order 4 over extended Galois field $GF(2^{10})$ for implementing the improved scheme that is matrices are in $GL(4, GF(2^{10}))$ 2.2.4. As in Galois field addition and multiplication is performed under certain irreducible polynomial. let we take irreducible polynomial as $m(r) = (r^{10}+r^3+1)$. All the calculation are performed under mod $(m(r))$.

Initial step: let us consider two matrices

$$A = \begin{bmatrix} r^3+1 & r^2 & r^6+1 & r^7+r \\ r^2+r & r^3 & r^9+r & 1 \\ r^3+r^2 & r^6 & r^7+1 & 0 \\ r^5+r & r & r^6+r & r^7+r^3 \end{bmatrix} \in GL(4, GF(2^{10}))$$

and

$$B = \begin{bmatrix} r^2 & r^6+1 & r^7+r & 1 \\ r^3 & r^8+r & r^9+r & 1 \\ r^2+1 & r^4 & r^8+1 & 0 \\ r^5+r & r & r^6+r & r^7 \end{bmatrix} \in GL(4, GF(2^{10}))$$

using computer algebra system ApCoCoA, we compute the orders n_1 and n_2 of the two matrices as

$$n_1 = |A| = 1048575 \quad \text{and} \quad n_2 = |B| = 25575.$$

Key Generation: To obtain public and secret key, Allice chooses two positive integers $s = 34 < 1048575$ and $t = 21 < 25575$ randomly. She then calculates

$$\begin{aligned} \mathcal{K}_{\mathcal{A}} &= A^{34}B^{21} \mod (m(r)) \\ &= [J_i] \qquad \text{for } \ i = 1, 2, 3, 4. \end{aligned}$$

where J_i, represents columns of matrix $\mathcal{K}_{\mathcal{A}}$ and are given below[1].

$$J_1 = \begin{bmatrix} r^9 + r^7 + r^6 + r^3 + 1 \\ r^8 + r^6 + r^4 \\ r^9 + r^7 + r^5 + r^4 + r \\ r^9 + r^6 + r^4 + r^3 + r^2 + r + 1 \end{bmatrix}$$

$$J_2 = \begin{bmatrix} r^9 + r^6 + r^5 + r^3 + 1 \\ r^8 + r^7 + r^4 + r^3 + 1 \\ r^7 + r^6 + r^4 + r^3 + r^2 + 1 \\ r^6 + r^3 \end{bmatrix}$$

$$J_3 = \begin{bmatrix} r^9 + r^7 + r^5 + r^3 + r^2 + r \\ r^9 + r^8 + r^5 + r^3 + r^2 \\ r^9 + r^7 + r^6 + r^3 \\ r^9 + r^8 + r^6 + r^5 + r^2 + r + 1 \end{bmatrix}$$

$$J_4 = \begin{bmatrix} r^9 + r^6 + r^5 + r^4 + r^3 + r^2 + 1 \\ r^8 + r^7 + r^6 + r^4 + 1 \\ r^4 \\ r^9 + r^8 + r^6 + r^5 + r \end{bmatrix}$$

The secret key is $PR_A = (34, 21)$, and public key is $PU_A = (A, B, \mathcal{K}_{\mathcal{A}})$.

Encryption:

[1]We use this representation because matrix size is big.

To send a plaintext

$$M = \begin{bmatrix} r^4+1 & r^5 & r^8 & r+1 \\ r^5+1 & r^7 & r^2+r & r \\ r^4+r^2 & r^9 & r^7+r & 0 \\ r^6+r & r & r^9 & r^7+r^3 \end{bmatrix} \in GL(4, GF(2^{10}))$$

to Allice, Bob chooses randomly $u = 37 < n_1$ and $v = 51 < n_2$. For encryption purpose Bob inputs plaintext M and public key $PU_A = (A, B, \mathcal{K}_A)$ of Allice. The following sequence of steps will be executed to get the ciphertext.

1. He calculates A^{37}, B^{51} to find Y as

$$\begin{aligned} Y &= A^{37}B^{51} \mod (m(r)) \\ &= [J_i] \qquad \text{for } \ i = 1, 2, 3, 4. \end{aligned}$$

where J_i, represents columns of matrix Y and are given below.

$$J_1 = \begin{bmatrix} r^8+r^7+r^5+r^2+r \\ r^8+r \\ r^7+r^6+r^5+r^2+r \\ r^9+r^8+r^5+r^2+r+1 \end{bmatrix}$$

$$J_2 = \begin{bmatrix} r^7+r^6+r^4+r^2+r \\ r^9+r^6+r^4+r^2+r \\ r^8+r^7+r^6+r^3+r^2+1 \\ r^8+r^7+r^5+r^2+r \end{bmatrix}$$

$$J_3 = \begin{bmatrix} r^9+r^8+r^7+r^4+r^3+1 \\ r^9+r^8+r^7+r^6+r^4+r^3+1 \\ r^9+r^8+r^7+r^6+r^5+r+1 \\ r^8+r^6+r^5+r^3+r^2+r \end{bmatrix}$$

$$J_4 = \begin{bmatrix} r^8+r^6+r^5+r^4+r^3+r^2+r+1 \\ r^8+r^7+r^4+r^3+r^2+1 \\ r^9+r^8+r^3+r \\ r^8+r^7+r^5+r^3+r^2 \end{bmatrix}$$

Now he calculates inverse of Y by using $Y^{-1} = Adj(Y)/det(Y)$. First he find $det(Y) = r^9+r^8+r^7+r^6+r$, calculates its inverse by using Extended Euclidean

Algorithm 2.5.1 as $(r^9+r^8+r^7+r^6+r)^{-1} \mod (m(r)) = r^9+r^8+r^6+r^5+r^4+r^3$ and multiply inverse of $det(Y)$ with $Adj(Y)$ to get inverse of Y.

2. After finding Y^{-1} he forms C_1 by utilizing $\mathcal{K}_{\mathcal{A}}$ as follows

$$\begin{aligned} C_1 &= (Y^{-1})\mathcal{K}_{\mathcal{A}}Y \mod (m(r)) \\ &= [J_i] \qquad \text{for } i = 1,2,3,4. \end{aligned}$$

where J_i, represents columns of matrix C_1 and are given below

$$J_1 = \begin{bmatrix} r^9+r^7+r^6+r^3+r^2 \\ r^9+r^4+r^3+r+1 \\ r^2+1 \\ r^7+r^4+r^3 \end{bmatrix}$$

$$J_2 = \begin{bmatrix} r^8+r^7+r^5+r^4+r+1 \\ r^6+r^4+r^2+r \\ r^9+r^8+r^7+r^6+r^2+r+1 \\ r^5+r^2+r+1 \end{bmatrix}$$

$$J_3 = \begin{bmatrix} r^5+1 \\ r^5+r^4+r^3+r+1 \\ r^7+r^5+r^4+r^3+r^2+r+1 \\ r^6+r^2 \end{bmatrix}$$

$$J_4 = \begin{bmatrix} r^7+r^5+r^4+r^3 \\ r^8+r^6+r^5+r^3+r \\ r^7+r^4+r^2+r \\ r^7+r^6+r^3+r^2+r+1 \end{bmatrix}$$

3. He calculates C_2 as follows

$$\begin{aligned} C_2 &= Y\mathcal{K}_{\mathcal{A}}(Y^{-1}) \mod (m(r)) \\ &= [J_i] \qquad \text{for } i = 1,2,3,4. \end{aligned}$$

where J_i, represents columns of matrix C_2 and are given below.

$$J_1 = \begin{bmatrix} r^3 + r^2 + 1 \\ r^9 + r^5 + r^4 + r^3 + r^2 \\ r^9 + r^7 + r^4 + r^2 + 1 \\ r^9 + r^7 + r^6 + r^5 + r^2 \end{bmatrix}$$

$$J_2 = \begin{bmatrix} r^8 + r^5 + r^4 + r^3 + r^2 \\ r^9 + r^5 + r^2 + r + 1 \\ r^9 + r^6 + r^5 + r^2 \\ r^9 + r^6 + r^5 + r^4 + r^3 + r \end{bmatrix}$$

$$J_3 = \begin{bmatrix} r^9 + r^8 + r^7 + r^5 + r^4 + r + 1 \\ r^8 + r^7 + r^6 + r^5 + r^3 + r^2 + 1 \\ r^9 + r^8 + r^5 + r^3 + r^2 + r \\ r^9 + r^7 + r^6 \end{bmatrix}$$

$$J_4 = \begin{bmatrix} r^9 + r^3 + r^2 + 1 \\ r^9 + r^8 + r^6 + r^4 + r^3 + r^2 + r \\ r^7 + r^4 + r^3 + r^2 \\ r^9 + r^8 + r^7 + r^6 + r^5 + r^4 + r^3 + r^2 + r \end{bmatrix}$$

4. The ciphertext C is computed by multiplying C_1, plaintext M and C_2 respectively. The output is

$$\begin{aligned} C &= C_1 M C_2 \mod (m(r)) \\ &= [J_i] \qquad \text{for } \; i = 1, 2, 3, 4. \end{aligned}$$

where J_i, represents columns of matrix C and are given below.

$$J_1 = \begin{bmatrix} r^5 + r^4 + r^3 + r + 1 \\ r^8 + r^5 + r^2 + r + 1 \\ r^9 + r^7 + r^6 + r^4 + 1 \\ r^9 + r^7 + r^6 + r^4 + r^3 + r \end{bmatrix}$$

$$J_2 = \begin{bmatrix} r^9 + r^8 + r^5 + r^4 + r^3 + r^2 + 1 \\ r^7 + r^5 + r \\ r^9 + r^8 + r^7 + r^6 + r^4 + r^3 + r^2 \\ r^8 + r^7 + r^5 + r^4 + r^2 \end{bmatrix}$$

$$J_3 = \begin{bmatrix} r^9 + r^5 + r^4 + r^2 + r \\ r^9 + r^7 + r^5 + r^4 + r^3 + r^2 + r \\ r^8 + r^7 + r^5 + r^3 + r + 1 \\ r^9 + r^8 + r^7 + r^2 \end{bmatrix}$$

$$J_4 = \begin{bmatrix} r^7 + r^5 + r^2 + r \\ r^9 + r^8 + r^6 + r^4 + r + 1 \\ r^9 + r^6 + r^5 + r^4 + r^2 + r + 1 \\ r^9 + r^8 + r^5 + r^4 \end{bmatrix}$$

5. Then he calculates C' as

$$\begin{aligned} C' &= A^{37}\mathcal{K}_{\mathcal{A}}B^{51} \mod (m(r)) \\ &= [J_i] \qquad \text{for } i = 1, 2, 3, 4. \end{aligned}$$

where J_i, represents columns of matrix C' and are given below.

$$J_1 = \begin{bmatrix} r^9 + r^7 + r^4 + r + 1 \\ r^9 + r^6 + r^5 + r^4 + r^3 + r^2 \\ r^8 + r^7 + r^6 + r^2 + r \\ r^8 + r^7 + r^2 + r \end{bmatrix}$$

$$J_2 = \begin{bmatrix} r^9 + r^4 + r^2 + r \\ r^8 + r^4 + r^2 + r \\ r^7 + r^6 + r^4 + r^3 + r^2 \\ r^7 + r^6 + r^5 + r^4 + r^3 + r^2 + r \end{bmatrix}$$

$$J_3 = \begin{bmatrix} r^6 + r^4 + r^3 + r + 1 \\ r^8 + r^7 + r^6 + r^5 + r \\ r^8 + r^5 + r^3 + r^2 + r \\ r^8 + r \end{bmatrix}$$

$$J_4 = \begin{bmatrix} r^9 + r^8 + r^7 + r^6 + r^3 + 1 \\ r^8 + r^7 + r^6 + r^5 + r^2 + r \\ r^9 + r^7 + r^6 + r^4 + r^3 + r^2 \\ r^8 + r^6 + r^4 + r^2 + r + 1 \end{bmatrix}$$

The ciphertext transmitted to Allice is the pair (C', C).

Decryption: When Allice gets the ciphertext pair (C', C), she computes following

steps for decryption to get original message. For this purpose she inputs ciphertext pair $(C^{'}, C)$, her private key $PR_A = (34, 21)$ to perform the following calculations.

1. In step 1, Allice computes A^{-34}, B^{-21} and then, she forms D as

$$\begin{aligned} D &= A^{-34}C'B^{-21} \mod (m(r)) \\ &= [J_i] \qquad \text{for } i = 1, 2, 3, 4. \end{aligned}$$

where J_i, represents columns of matrix D and are given below.

$$J_1 = \begin{bmatrix} r^8 + r^7 + r^5 + r^2 + r \\ r^8 + r \\ r^7 + r^6 + r^5 + r^2 + r \\ r^9 + r^8 + r^5 + r^2 + r + 1 \end{bmatrix}$$

$$J_2 = \begin{bmatrix} r^7 + r^6 + r^4 + r^2 + r \\ r^9 + r^6 + r^4 + r^2 + r \\ r^8 + r^7 + r^6 + r^3 + r^2 + 1 \\ r^8 + r^7 + r^5 + r^2 + r \end{bmatrix}$$

$$J_3 = \begin{bmatrix} r^9 + r^8 + r^7 + r^4 + r^3 + 1 \\ r^9 + r^8 + r^7 + r^6 + r^4 + r^3 + 1 \\ r^9 + r^8 + r^7 + r^6 + r^5 + r + 1 \\ r^8 + r^6 + r^5 + r^3 + r^2 + r \end{bmatrix}$$

$$J_4 = \begin{bmatrix} r^8 + r^6 + r^5 + r^4 + r^3 + r^2 + r + 1 \\ r^8 + r^7 + r^4 + r^3 + r^2 + 1 \\ r^9 + r^8 + r^3 + r \\ r^8 + r^7 + r^5 + r^3 + r^2 \end{bmatrix}$$

2. Now she calculates inverse of D by using $D^{-1} = Adj(D)/det(D)$. First he find $det(D) = r^9 + r^8 + r^7 + r^6 + r$, calculates its inverse by using Extended Euclidean Algorithm 2.5.1 as $(r^9 + r^8 + r^7 + r^6 + r)^{-1} \mod (m(r)) = r^9 + r^8 + r^6 + r^5 + r^4 + r^3$ and multiply inverse of $det(D)$ with $Adj(D)$ to get inverse of D. Then she calculates

$$\begin{aligned} D_1 &= D^{-1}{\mathcal{K}_{\mathcal{A}}}^{-1}D \mod (m(r)) \\ &= [J_i] \qquad \text{for } i = 1, 2, 3, 4. \end{aligned}$$

where J_i, represents columns of matrix D_1 and are given below.

$$J_1 = \begin{bmatrix} r^8 + r^6 + 1 \\ r^9 + r^5 \\ r^8 + r^7 + r^6 + r \\ r^8 + r^6 + r^4 + r^2 \end{bmatrix}$$

$$J_2 = \begin{bmatrix} r^7 + r^5 + r^3 + r \\ r^9 + r^7 + r^3 + r \\ r^8 + r^7 + r^6 + r^5 + r^4 + r^3 + r^2 + r + 1 \\ r^9 + r^7 + r^3 + r^2 + r \end{bmatrix}$$

$$J_3 = \begin{bmatrix} r^9 + r^8 + r^7 \\ r^8 + r^7 + r^6 + r^5 + r^4 \\ r^9 + r^8 + r^7 + r^6 + r^3 + r + 1 \\ r^9 + r^8 + r^6 + r^5 + r^4 + r + 1 \end{bmatrix}$$

$$J_4 = \begin{bmatrix} r^9 + r^8 + r^5 + r^4 + r^3 + r + 1 \\ r^9 + r^8 + r^7 + r^6 + r^5 + r + 1 \\ r^9 + r^6 + r^4 + r^3 + 1 \\ r^9 + r^6 + r^4 + r^2 + 1 \end{bmatrix}$$

3. And calculates D_2 as

$$\begin{aligned} D_2 &= D\mathcal{K}_{\mathcal{A}}{}^{-1}D^{-1} \mod (m(r)) \\ &= [J_i] \qquad \text{for } \ i = 1, 2, 3, 4. \end{aligned}$$

where J_i, represents columns of matrix D_2 and are given below.

$$J_1 = \begin{bmatrix} r^8 + r^6 + r^5 + r^4 + r^2 + r \\ r^9 + r^7 + r^5 + r^4 + r^3 + r + 1 \\ r^6 + r^2 + r \\ r^3 + r^2 + r + 1 \end{bmatrix}$$

$$J_2 = \begin{bmatrix} r^9 + r^5 + r^4 + r + 1 \\ r^9 + r^7 + r^5 + r^3 + r \\ r^8 + r^7 + r^6 + r^3 + r + 1 \\ r^8 + r^7 + r^6 + r^5 + r^4 + r^3 + r^2 \end{bmatrix}$$

$$J_3 = \begin{bmatrix} r^9+r^8+r^5+r^2+1 \\ r^9+r^8+r^7+r^6+r^5+r^4+r^3 \\ r^9+r^6+r^5+r^4+r^3+1 \\ r^3+1 \end{bmatrix}$$

$$J_4 = \begin{bmatrix} r^9+r^6+r \\ r^7+r^5+r^4+r^3+r^2+1 \\ r^9+r^8+r^7+r^3+r^2+r \\ r^9+r^8+r^7+r^6+r^5+r^4 \end{bmatrix}$$

4. Allice can get the secret message M as

$$M = D_1CD_2 \mod (m(r))$$
$$= \begin{bmatrix} r^4+1 & r^5 & r^8 & r+1 \\ r^5+1 & r^7 & r^2+r & r \\ r^4+r^2 & r^9 & r^7+r & 0 \\ r^6+r & r & r^9 & r^7+r^3 \end{bmatrix}$$

Example 4.3.3 Let us take $GL(3, GF(2^{10}))$ for implementing the improved scheme. As in Galois field addition and multiplication is performed under certain irreducible polynomial. let we take irreducible polynomial as $m(r) = (r^{10}+r^3+1)$. All the calculation are performed under mod $(m(r))$.

Initial step: let us consider two matrices

$$A = \begin{bmatrix} 1 & r^2 & r \\ r & r^4 & 0 \\ 1 & r^5 & 0 \end{bmatrix} \in GL(3, GF(2^{10}))$$

and

$$B = \begin{bmatrix} 1 & r^3 & r^2 \\ r & r^7 & 0 \\ r^9 & r^5 & 0 \end{bmatrix} \in GL(3, GF(2^{10}))$$

using computer algebra system ApCoCoA, we compute the orders n_1 and n_2 of the two matrices as

$$n_1 = |A| = 1023 \quad \text{and} \quad n_2 = |B| = 209715.$$

Key Generation: To obtain public and secret key, Allice chooses two positive integers $s = 67 < 1023$ and $t = 89 < 209715$ randomly. She then calculates

$$\begin{aligned} \mathcal{K}_{\mathcal{A}} &= A^{67}B^{89} \mod (m(r)) \\ &= [J_i] \qquad \text{for } i = 1, 2, 3. \end{aligned}$$

where J_i, represents columns of matrix $\mathcal{K}_{\mathcal{A}}$ and are given below.

$$J_1 = \begin{bmatrix} r^7 + r^6 + r^5 + r^4 + r^3 + r^2 \\ r^8 + r^7 + r^5 \\ r^9 + r^6 + r^3 \end{bmatrix}$$

$$J_2 = \begin{bmatrix} r^8 + r^7 + r^4 + r + 1 \\ r^9 + r^6 + r^4 + r^2 + r + 1 \\ r^9 + 1 \end{bmatrix}$$

$$J_3 = \begin{bmatrix} r^9 + r^8 + r^3 + r^2 + r \\ r^9 + r^6 + r^4 + r^3 + r + 1 \\ r^8 + r^6 + r^5 + r \end{bmatrix}$$

The secret key is $PR_A = (67, 89)$, and public key is $PU_A = (A, B, \mathcal{K}_{\mathcal{A}})$.

Encryption:

To send a plaintext

$$M = \begin{bmatrix} r^8 + 1 & r^7 + r^2 & r^6 \\ r^3 + r & r^7 & r + 1 \\ r^5 + r^3 & r^9 + 1 & 1 \end{bmatrix} \in GL(3, GF(2^{10}))$$

to Allice, Bob chooses randomly $u = 76 < n_1$ and $v = 96 < n_2$. For encryption purpose Bob inputs plaintext M and public key $PU_A = (A, B, \mathcal{K}_{\mathcal{A}})$ of Allice. The following sequence of steps will be executed to get the ciphertext.

1. He calculates A^{76}, B^{96} to find Y as

$$\begin{aligned} Y &= A^{76}B^{96} \mod (m(r)) \\ &= [J_i] \qquad \text{for } i = 1, 2, 3. \end{aligned}$$

where J_i, represents columns of matrix Y and are given below.

$$J_1 = \begin{bmatrix} r^9+r^7+r^6+r^5+r^4 \\ r^8+r^4+r^2 \\ r^9+r^8+r^7+r^6+r^4+r+1 \end{bmatrix}$$

$$J_2 = \begin{bmatrix} r^9+r^4+r^2 \\ r^9+r^7+r^6+r^5+r^3+r^2 \\ r^6+r^5+r^4+r^3+r^2 \end{bmatrix}$$

$$J_3 = \begin{bmatrix} r^9+r^4+r^3+r^2 \\ r^7+r^5+r^3+r \\ r^9+r^8+r^5+r^4+r^2+1 \end{bmatrix}$$

Now he calculates inverse of Y by using $Y^{-1} = Adj(Y)/det(Y)$. First he find $det(Y) = r^9+r^7+r^6+r^5+r^4$, calculates its inverse by using Extended Euclidean Algorithm 2.5.1 as $(r^9+r^7+r^6+r^5+r^4)^{-1} \mod (m(r)) = r^7+r^5$ and multiply inverse of $det(Y)$ with $Adj(Y)$ to get inverse of Y.

2. After finding Y^{-1} he forms C_1 by utilizing $\mathcal{K}_{\mathcal{A}}$ as follows

$$\begin{aligned} C_1 &= (Y^{-1})\mathcal{K}_{\mathcal{A}}Y \mod (m(r)) \\ &= [J_i] \qquad \text{for } i = 1,2,3. \end{aligned}$$

where J_i, represents columns of matrix C_1 and are given below

$$J_1 = \begin{bmatrix} r^9+r^8+r^6+r^4+r^3 \\ r^7+r^4+r \\ r^9+r^7+r^6+r^4+r^2+1 \end{bmatrix}$$

$$J_2 = \begin{bmatrix} r^9+r^8+r^7+r^6+r^3+r^2+r \\ r^9+r^8+r^7+r^6+r^5+r^4+r^3+r \\ r^9+r^7+r^5+r^4 \end{bmatrix}$$

$$J_3 = \begin{bmatrix} r^8+r^3+r^2+1 \\ r^9+r^3+1 \\ r^9+r^8+r^6+r^5+r^3+r+1 \end{bmatrix}$$

3. He calculates C_2 as follows

$$C_2 = Y\mathcal{K}_{\mathcal{A}}(Y^{-1}) \mod (m(r))$$
$$= [J_i] \qquad \text{for } i = 1, 2, 3.$$

where J_i, represents columns of matrix C_2 and are given below.

$$J_1 = \begin{bmatrix} r^9 + r^8 + r^6 + r^5 + r^4 + r^3 + r^2 + 1 \\ r^8 + r^7 + r^3 + r^2 + r + 1 \\ r^8 + r^7 + r^6 + r^5 + r^3 + r^2 \end{bmatrix}$$

$$J_2 = \begin{bmatrix} r^9 + r^8 + r^7 + r^5 + r^3 + r^2 + r \\ r^9 + r^7 + r^2 + 1 \\ r^9 + r^7 + 1 \end{bmatrix}$$

$$J_3 = \begin{bmatrix} r^9 + r^8 + r^7 + r^3 + r^2 + r + 1 \\ r^9 + r^8 + r^7 + r^5 + r^4 + r^2 \\ r^9 + r^5 + r^4 + 1 \end{bmatrix}$$

4. The ciphertext C is computed by multiplying C_1, plaintext M and C_2 respectively. The output is

$$C = C_1 M C_2 \mod (m(r))$$
$$= [J_i] \qquad \text{for } i = 1, 2, 3.$$

where J_i, represents columns of matrix C and are given below.

$$J_1 = \begin{bmatrix} r^8 + r^5 + r \\ r^5 + r^4 + r^3 + r^2 + r + 1 \\ r^6 + r^4 + r + 1 \end{bmatrix}$$

$$J_2 = \begin{bmatrix} r^8 + r^7 + r^6 + r^5 + r \\ r^6 + r^5 + r^3 + 1 \\ r^8 + r^4 + r \end{bmatrix}$$

$$J_3 = \begin{bmatrix} r^5 + r^4 + r^2 \\ r^9 + r^7 + r^5 + r^4 + r^3 + r + 1 \\ r^7 + r^5 + r^2 + r \end{bmatrix}$$

5. Then he calculates C' as

$$C' = A^{76} K_A B^{96} \mod (m(r))$$
$$= [J_i] \qquad \text{for } i = 1, 2, 3.$$

where J_i, represents columns of matrix C' and are given below.

$$J_1 = \begin{bmatrix} r^9 + r^7 + r^5 + r^2 \\ r^3 + r^2 \\ r^9 + r^8 + r^7 + r^6 + r^5 + r \end{bmatrix}$$

$$J_2 = \begin{bmatrix} r^7 + r^6 + r^5 + r^3 + r \\ r^7 + r^5 + r^4 + r^3 \\ r^9 + r^8 + r^5 + r^4 + r^2 + r \end{bmatrix}$$

$$J_3 = \begin{bmatrix} r^9 + r^7 \\ r^9 + r^5 + r^4 \\ r^9 + r^8 + r^7 + r^3 + r + 1 \end{bmatrix}$$

The ciphertext transmitted to Allice is the pair (C', C).

Decryption: When Allice gets the ciphertext pair (C', C), she computes following steps for decryption to get original message. For this purpose she inputs ciphertext pair (C', C), her private key $PR_A = (67, 89)$ and performs following calculations.

1. In step 1, Allice computes A^{-67}, B^{-89} and then, she forms D as

$$D = A^{-67} C' B^{-89} \mod (m(r))$$
$$= [J_i] \qquad \text{for } i = 1, 2, 3.$$

where J_i, represents columns of matrix D and are given below.

$$J_1 = \begin{bmatrix} r^9 + r^7 + r^6 + r^5 + r^4 \\ r^8 + r^4 + r^2 \\ r^9 + r^8 + r^7 + r^6 + r^4 + r + 1 \end{bmatrix}$$

$$J_2 = \begin{bmatrix} r^9 + r^4 + r^2 \\ r^9 + r^7 + r^6 + r^5 + r^3 + r^2 \\ r^6 + r^5 + r^4 + r^3 + r^2 \end{bmatrix}$$

$$J_3 = \begin{bmatrix} r^9 + r^4 + r^3 + r^2 \\ r^7 + r^5 + r^3 + r \\ r^9 + r^8 + r^5 + r^4 + r^2 + 1 \end{bmatrix}$$

2. Now she calculates inverse of D by using $D^{-1} = Adj(D)/det(D)$. First he find $det(D) = r^9 + r^7 + r^6 + r^5 + r^4$, calculates its inverse by using Extended Euclidean Algorithm 2.5.1 as $(r^9 + r^7 + r^6 + r^5 + r^4)^{-1} \mod (m(r)) = r^7 + r^5$ and multiply inverse of $det(D)$ with $Adj(D)$ to get inverse of D. Similarly she calculates ${\mathcal{K}_{\mathcal{A}}}^{-1}$. Then she calculates

$$\begin{aligned} D_1 &= D^{-1}{\mathcal{K}_{\mathcal{A}}}^{-1}D \mod (m(r)) \\ &= [J_i] \qquad \text{for} \quad i = 1, 2, 3. \end{aligned}$$

where J_i, represents columns of matrix D_1 and are given below.

$$J_1 = \begin{bmatrix} r^9 + r^8 + r^6 + r^5 + r^4 + r^3 + r \\ r^8 + r^7 + r^6 + r^5 + r^2 + 1 \\ r^8 + r^6 + r^4 + r^3 + 1 \end{bmatrix}$$

$$J_2 = \begin{bmatrix} r^8 + r^7 + r^6 + r^3 + r + 1 \\ r^9 + r^7 + r^6 + r^5 + r^4 + r^3 + r^2 + r \\ r^9 \end{bmatrix}$$

$$J_3 = \begin{bmatrix} r^7 + r^6 + r^5 + r^4 + r^3 + r \\ r^9 + r^8 + r^6 + r^4 + r^3 + r + 1 \\ r^9 + r^6 + r^5 + r^2 + 1 \end{bmatrix}$$

3. And calculates D_2 as

$$\begin{aligned} D_2 &= D{\mathcal{K}_{\mathcal{A}}}^{-1}D^{-1} \mod (m(r)) \\ &= [J_i] \qquad \text{for} \quad i = 1, 2, 3. \end{aligned}$$

where J_i, represents columns of matrix D_2 and are given below.

$$J_1 = \begin{bmatrix} r^9 + r^8 + r^4 + r^3 + r \\ r^8 + r^6 + r^5 + r^4 + r^3 + r + 1 \\ r^9 + r^8 + r^5 + r^2 \end{bmatrix}$$

$$J_2 = \begin{bmatrix} r \\ r^8 + r^7 + r^6 + r^5 + 1 \\ r^9 + r^7 + r^3 + r^2 + r + 1 \end{bmatrix}$$

$$J_3 = \begin{bmatrix} r^8 + r^6 + r^5 + r^4 + r^2 \\ r^9 + r^8 + r^6 + r^5 + r^4 + r + 1 \\ r^8 + r^4 + r^3 + r \end{bmatrix}$$

4. Allice can get the secret message M as

$$\begin{aligned} M &= D_1 C D_2 \mod (m(r)) \\ &= \begin{bmatrix} r^8 + 1 & r^7 + r^2 & r^6 \\ r^3 + r & r^7 & r + 1 \\ r^5 + r^3 & r^9 + 1 & 1 \end{bmatrix} \end{aligned}$$

Example 4.3.4 Let us take matrices of order 3 over Galois field $GF(2^5)$ for implementing modified and improved scheme. As we know in Galois field addition and multiplication is performed under certain irreducible polynomial. let us take $m(r) = (r^5 + r^2 + 1)$ as irreducible polynomial and all the calculations given below are performed under mod $m(r)$.

Initial step: let us consider two matrices

$$A = \begin{bmatrix} r & r^2 + r & r^4 + r^3 + r^2 \\ r^3 + 1 & r^4 + r^2 & r^3 + r^2 + r + 1 \\ r^3 + r + 1 & r + 1 & r^4 + r^3 + r \end{bmatrix} \in GL(3, GF(2^5))$$

and

$$B = \begin{bmatrix} r^4 + r^2 + r & r^3 & r^3 + r \\ r^3 + r + 1 & r^4 + r^3 + r + 1 & r \\ r^3 + r & 1 & r^4 + r^3 + r^2 + r + 1 \end{bmatrix} \in GL(3, GF(2^5))$$

using computer algebra system ApCoCoA, we compute the orders n_1 and n_2 of the two matrices as

$$n_1 = |A| = 1023 \quad \text{and} \quad n_2 = |B| = 241.$$

Key Generation: To obtain the public and secret keys, Allice chooses two random positive integers $s = 23 < 1023$ and $t = 33 < 241$. She then calculates

$$\begin{aligned}\mathcal{K}_{\mathcal{A}} &= A^{23}B^{33} \mod (m(r)) \\ &= \begin{bmatrix} r^2+r+1 & 1 & 1 \\ r^3+r^2 & r^3+r^2+1 & r \\ r^4+r^3 & r^4+1 & r^3+r^2+r \end{bmatrix}\end{aligned}$$

The secret key is $PR_A = (23, 33)$, and public key is $PU_A = (A, B, \mathcal{K}_{\mathcal{A}})$.

Encryption:

To send a plaintext

$$M = \begin{bmatrix} r^2 & r^3+r^2 & r^4+r \\ r^4 & r+1 & r^3+r+1 \\ r^4+r^2+1 & r^3+r^2+r & 1 \end{bmatrix} \in GL(3, GF(2^5))$$

to Allice, Bob chooses randomly $u = 13 < n_1$ and $v = 20 < n_2$. For encryption purpose Bob inputs plaintext M and public key $PU_A = (A, B, \mathcal{K}_{\mathcal{A}})$ of Allice. The following sequence of steps will be executed to get the ciphertext.

1. He calculates A^{13}, B^{20} to find Y as

$$\begin{aligned}Y &= A^{13}B^{20} \mod (m(r)) \\ &= \begin{bmatrix} r^2+r & r^3+r^2+1 & r^4+r \\ r^4+r^2+r+1 & r^2+r & r^4+r^3+1 \\ r^3+1 & r^4+r^2+r+1 & r^3+r^2+r+1 \end{bmatrix}\end{aligned}$$

Now he calculates inverse of Y by using $Y^{-1} = Adj(Y)/det(Y)$. First he finds $det(Y) = r^4 + r^3 + r^2 + r$, calculates its inverse by using Extended Euclidean Algorithm 2.5.1 as $(r^4+r^3+r^2+r)^{-1} \mod (m(r)) = r^4+r^2$ and multiply inverse of $det(Y)$ with $Adj(Y)$ to get Y^{-1} as

$$Y^{-1} = \begin{bmatrix} r^3 & r^4+r^2+r & r+1 \\ r^3+r+1 & r^4+r+1 & r^3+r^2+r \\ r^3 & r^2 & r^3+r^2 \end{bmatrix}$$

2. After finding Y^{-1} he forms C_1 by utilizing $\mathcal{K}_{\mathcal{A}}$ as

$$C_1 = (Y^{-1})\mathcal{K}_{\mathcal{A}}Y \mod (m(r))$$
$$= \begin{bmatrix} r^4+r & r^3+1 & r^4+r^3 \\ r^3+r+1 & r^3+r^2+r & r^2 \\ r^4+r^3+r+1 & r^4 & r^4+r^3 \end{bmatrix}$$

3. And also computes C_2 as,

$$C_2 = Y\mathcal{K}_{\mathcal{A}}(Y^{-1}) \mod (m(r))$$
$$= \begin{bmatrix} r^4+r^3 & r^4+r^2 & r^4+r \\ r^4+r^3+1 & r^2+r+1 & r^4+r^3+r^2+1 \\ r^4 & r & r^4+r^3+r+1 \end{bmatrix}$$

4. The ciphertext C is computed by multiplying C_1, plaintext M and C_2 respectively. The output is

$$C = C_1MC_2 \mod (m(r))$$
$$= \begin{bmatrix} r^3+r^2 & r^2 & r^3+r^2+r+1 \\ r^2+1 & x^4+1 & r^2 \\ r^3+r^2+1 & r^3+r^2+r+1 & r^3 \end{bmatrix}$$

5. He also calculates C' as,

$$C' = A^{13}\mathcal{K}_{\mathcal{A}}B^{20} \mod (m(r))$$
$$= \begin{bmatrix} r^4+r+1 & r^4 & r^4+r^3+r^2+r \\ r^4+r^2+r+1 & r^4+r^3 & r^2+r+1 \\ r+1 & r^4+r^3 & r^4+r^3+r^2 \end{bmatrix}$$

The ciphertext transmitted to Allice is the pair (C', C).

Decryption: When Allice gets the ciphertext pair (C', C), she computes following steps for decryption to get the original message. For this purpose she inputs ciphertext pair (C', C), her private key $PR_A = (23, 33)$ and performs following calculations.

1. In step 1, Allice computes A^{-23} by taking inverse of A^{23}. For this she first compute $det(A^{23}) = r^4 + r^2 + 1$, calculate its inverse by using Extended Euclidean Algorithm 2.5.1 as $(r^4+r^2+1)^{-1} \mod (m(r)) = r^4+r^3+r$ and

multiply inverse of $det(A^{23})$ with $Adj(A^{23})$ to get A^{-23} as

$$A^{-23} = \begin{bmatrix} r^2+r+1 & r^4+r^3+r^2+r+1 & r^4+r^3+r \\ r^4+r^3+1 & r^4+r^3+1 & r^2+r \\ r^3+r^2+r+1 & r^4+r^3+r^2+1 & r^4+r^2 \end{bmatrix}$$

Similarly she computes inverse of B^{33} as given below

$$B^{-33} = \begin{bmatrix} r^4+r^3+r^2 & r^4+r^3 & r^3+r+1 \\ r^3+r^2+r+1 & r^4+r^3+r^2+r+1 & r^4+r^2 \\ 0 & r^3 & r^4+1 \end{bmatrix}$$

and then she calculates

$$\begin{aligned} D &= A^{-23}C'B^{-33} \mod (m(r)) \\ &= \begin{bmatrix} r^2+r & r^3+r^2+1 & r^4+r \\ r^4+r^2+r+1 & r^2+r & r^4+r^3+1 \\ r^3+1 & r^4+r^2+r+1 & r^3+r^2+r+1 \end{bmatrix} \end{aligned}$$

2. Now she calculates inverse of D by using $D^{-1} = Adj(D)/det(D)$. First she find $det(D) = r^4+r^3+r^2+r$, calculates its inverse by using Extended Euclidean Algorithm 2.5.1 as $(r^4+r^3+r^2+r)^{-1} \mod (m(r)) = r^4+r^2$ and multiply inverse of $det(D)$ with $Adj(D)$ to get inverse of D as

$$D^{-1} = \begin{bmatrix} r^3 & r^4+r^2+r & r+1 \\ r^3+r+1 & r^4+r+1 & r^3+r^2+r \\ r^3 & r^2 & r^3+r^2 \end{bmatrix}$$

Similarly she computes inverse of $\mathcal{K}_{\mathcal{A}}$ as given below

$$\mathcal{K}_{\mathcal{A}}{}^{-1} = \begin{bmatrix} r^3+r & r^4+r^3+1 & r^2 \\ r^2+1 & r^2+1 & r^2+r+1 \\ r^4+r^2+r+1 & 0 & r^4+r^3+r+1 \end{bmatrix}$$

and then she calculates D_1 as

$$\begin{aligned} D_1 &= D^{-1}\mathcal{K}_{\mathcal{A}}{}^{-1}D \mod (m(r)) \\ &= \begin{bmatrix} r^4+r^2+1 & r^4+r^3+1 & r^4+r^3+r^2+r+1 \\ r^4+r^3+r^2+1 & r^3+1 & r^3+r^2 \\ r^2+1 & r & r^3 \end{bmatrix} \end{aligned}$$

3. And also calculates

$$D_2 = D\mathcal{K}_{\mathcal{A}}{}^{-1}D^{-1} \mod (m(r)$$
$$= \begin{bmatrix} r^4+r+1 & r^4+r^2+1 & r^3+r^2+1 \\ r^4+r^3+r & r^2+r+1 & r^4+r^2+1 \\ r^4+r^3+r^2 & r^4+r^3+r & 0 \end{bmatrix}$$

4. Allice can get the secret message M as

$$M = D_1CD_2 \mod (m(r))$$
$$= \begin{bmatrix} r^2 & r^3+r^2 & r^4+r \\ r^4 & r+1 & r^3+r+1 \\ r^4+r^2+1 & r^3+r^2+r & 1 \end{bmatrix}$$

Example 4.3.5 Let us take matrices of order 3 over Galois field $GF(3^5)$ for implementing modified and improved scheme. As we know in Galois field addition and multiplication is performed under certain irreducible polynomial. let us take $m(r) = (r^5 + r + 1)$ as irreducible polynomial and all the calculation given below are performed under mod$m(r)$.

Initial step: let us consider two matrices

$$A = \begin{bmatrix} r^3 & r^3+r & 2 \\ 2r^4 & 0 & 2r^3 \\ 1 & r^4+1 & r \end{bmatrix} \in GL(3, GF(3^5))$$

and

$$B = \begin{bmatrix} r^2 & r & 1 \\ 2r & r+1 & r^3 \\ 0 & r^3+r & 2 \end{bmatrix} \in GL(3, GF(3^5))$$

using computer algebra system ApCoCoA, we compute the orders n_1 and n_2 of the two matrices as

$$n_1 = |A| = 29523 \quad \text{and} \quad n_2 = |B| = 39364.$$

Key Generation: To obtain the public and secret keys, Allice chooses two random positive integers $s = 21 < 29523$ and $t = 25 < 39364$. She then calculates

$$\mathcal{K}_{\mathcal{A}} = A^{21}B^{25} \mod (m(r))$$
$$= \begin{bmatrix} r^4+r^3+r+1 & 2r^2+2r & r^2+r+1 \\ r^3+r^2+2 & 2r^4+2r^2+2r & r^4+2r^3+r^2+2r+2 \\ r^4+2r^3 & r^4+2r^3+2r^2+r+1 & r^4+2r^3+2r^2 \end{bmatrix}$$

The secret key is $PR_A = (21, 25)$, and public key is $PU_A = (A, B, \mathcal{K}_{\mathcal{A}})$.

Encryption:

To send a plaintext

$$M = \begin{bmatrix} 2r^4+2r^2+2 & r^4+r^3 & r^4+2r \\ 2r+2 & 2r^3+r & r^4+r^3+2r \\ 2r^3+2r & r^3+r & 2r^4+r^2 \end{bmatrix} \in GL(3, GF(3^5))$$

to Allice, Bob chooses randomly $u = 31 < n_1$ and $v = 26 < n_2$. For encryption purpose Bob inputs plaintext M and public key $PU_A = (A, B, K_A)$ of Allice. The following sequence of steps will be executed to get the ciphertext.

1. He calculates A^{31}, B^{26} to find Y as

$$Y = A^{31}B^{26} \mod (m(r))$$
$$= \begin{bmatrix} 2r^3+r+1 & 2r^4+r^2+1 & 2r^4+2r^2+2 \\ 2r^3+2r^2+r+2 & 2r^4+r & 2r^4+2r^3+2r^2+r \\ r^4+r^2+2r+2 & r^2+r+1 & 2r^4+2r^2+1 \end{bmatrix}$$

Now he calculates inverse of Y by using $Y^{-1} = Adj(Y)/det(Y)$. First he finds $det(Y) = r^4 + 2r^3 + r^2 + 2r + 1$, calculates its inverse by using Extended Euclidean Algorithm 2.5.1 as $(r^4 + 2r^3 + r^2 + 2r + 1)^{-1} \mod (m(r)) = r^4$ and multiply inverse of $det(Y)$ with $Adj(Y)$ to get inverse of Y as

$$Y^{-1} = \begin{bmatrix} r^4+2r & r^3+r^2+2 & r^4+r^2+2 \\ r^3+2r^2+r & 2r^4+2r+1 & 2r^4+2r^3+2r^2+2 \\ r^3+r^2+r & r^3+2r^2+r+2 & 2r^4+2r^3+2r^2+2r \end{bmatrix}$$

2. After finding Y^{-1} he forms C_1 by utilizing $\mathcal{K}_{\mathcal{A}}$ as

$$C_1 = (Y^{-1})\mathcal{K}_{\mathcal{A}}Y \mod (m(r))$$
$$= \begin{bmatrix} 2r^4+2r^3+r & 2r^3+r^2+r & 2r^3+r \\ r^4+2r^2+r+2 & r^4+2r^2 & 2r^2+r+1 \\ r^4+r^3+r^2+2 & 2r^4+2r^2+r+1 & r^4+r^3+2r^2+2r+1 \end{bmatrix}$$

3. And also computes C_2 as,

$$C_2 = Y\mathcal{K}_{\mathcal{A}}(Y^{-1}) \mod (m(r))$$
$$= \begin{bmatrix} 2r^4+r^3+2r^2+2r & r^4+r^2+2 & 2r^4+r+2 \\ r^4+r^3 & 2r^3+r+2 & r^4+r^3+2r^2+2 \\ r^4+2r^3+2r^2+r+2 & r^3+2r^2+2r+2 & 2r^4+2r^2+2 \end{bmatrix}$$

4. The ciphertext C is computed by multiplying C_1, plaintext M and C_2 respectively. The output is

$$C = C_1MC_2 \mod (m(r))$$
$$= \begin{bmatrix} r^4+r^2+r+2 & r^3+r^2+2 & r^3+2r^2+r+1 \\ 2r^4+2r^3+1 & r^4+2r+2 & 2r^3+r^2+2r \\ 2r^4+2r^3 & r^2+2r+1 & r^4+r^2+2r+1 \end{bmatrix}$$

5. He also calculates $C^{'}$ as,

$$C^{'} = A^{31}\mathcal{K}_{\mathcal{A}}B^{26} \mod (m(r))$$
$$= \begin{bmatrix} r^2+r+2 & 2r^4+2r^2+2 & 2x^4+2 \\ r^4+2r^3+2r^2+2r & 2r^4+r^2+2r+2 & 2r^4+r^3+2r^2+r+2 \\ 2r^2+2r+1 & 2r^4+2r^2+2r+2 & 2r^4+r \end{bmatrix}$$

The ciphertext transmitted to Allice is the pair $(C^{'}, C)$.

Decryption: When Allice gets the ciphertext pair $(C^{'}, C)$, she computes following steps for decryption to get the original message. For this purpose, she inputs ciphertext pair $(C^{'}, C)$, her private key $PR_A = (21, 25)$ and performs following calculations.

1. In step 1, Allice computes A^{-21} by taking inverse of A^{21}. For this she first compute $det(A^{21}) = 2r^3+2r^2+2r+1$, calculate its inverse by using Extended Euclidean Algorithm 2.5.1 as $(2r^3+2r^2+2r+1)^{-1} \mod (m(r)) = 2r^4+2r^2$

and multiply inverse of $det(A^{23})$ with $Adj(A^{23})$ to get A^{-23} as

$$A^{-23} = \begin{bmatrix} 2r^4+2r^3+2r^2 & r^3+r^2+2 & 2r^4+r^3+1 \\ 2r^4+r^3+1 & r^3+2r^2+1 & 2r^3+2r^2+2r+1 \\ r^4+r^2 & r^4+r^3+2r & 2r^4+r^2+r+2 \end{bmatrix}$$

Similarly she computes inverse of B^{25} as given below

$$B^{-25} = \begin{bmatrix} 2r+2 & 2r^3+2r^2+2 & r^4+r^3+2r \\ 2r^2+2 & 2r^4+2r^2+2r+1 & 2r^4+r^3+r^2+2r+2 \\ r^4+2r^3+2r & r^3+2r+2 & r^4+r^3+r \end{bmatrix}$$

and then she calculates

$$\begin{aligned} D &= A^{-21}C'B^{-25} \mod (m(r)) \\ &= \begin{bmatrix} 2r^3+r+1 & 2r^4+r^2+1 & 2r^4+2r^2+2 \\ 2r^3+2r^2+r+2 & 2r^4+r & 2r^4+2r^3+2r^2+r \\ r^4+r^2+2r+2 & r^2+r+1 & 2r^4+2r^2+1 \end{bmatrix} \end{aligned}$$

2. Now she calculates inverse of D by using $D^{-1} = Adj(D)/det(D)$. First he find $det(D) = r^4+2r^3+r^2+2r+1$, calculates its inverse by using Extended Euclidean Algorithm 2.5.1 as $(r^4+2r^3+r^2+2r+1)^{-1} \mod (m(r)) = r^4$ and multiply inverse of $det(D)$ with $Adj(D)$ to get inverse of D as

$$D^{-1} = \begin{bmatrix} r^4+2r & r^3+r^2+2 & r^4+r^2+2 \\ r^3+2r^2+r & 2r^4+2r+1 & 2r^4+2r^3+2r^2+2 \\ r^3+r^2+r & r^3+2r^2+r+2 & 2r^4+2r^3+2r^2+2r \end{bmatrix}$$

Similarly she computes inverse of $\mathcal{K}_{\mathcal{A}}$ as given below

$${\mathcal{K}_{\mathcal{A}}}^{-1} = \begin{bmatrix} 2r^4+2r^3+2r+2 & 2r^3+r^2+2 & r^2+2r+1 \\ 2r^4+r^3+2r & 2r^4+r^3+r & 2r^4+r^3+r^2+2r+2 \\ r^4+2r^3+r^2+r & 2r^4+r^2+1 & 2r^4+r+1 \end{bmatrix}$$

and then she calculates D_1 as

$$\begin{aligned} D_1 &= D^{-1}{\mathcal{K}_{\mathcal{A}}}^{-1}D \mod (m(r)) \\ &= \begin{bmatrix} r^4+r^3+r^2+2r+1 & 2r^4+r^3+2r^2+r+1 & r^4+2r^2+2r \\ r^4+2r^2+1 & 2r^4+2r^2 & 2r^3+2r^2+1 \\ 2r^4+r^3+2r+1 & r^3+2r^2+r & 2r^3+2+2 \end{bmatrix} \end{aligned}$$

3. And also calculates

$$D_2 = D\mathcal{K}_{\mathcal{A}}{}^{-1}D^{-1} \mod (m(r))$$
$$= \begin{bmatrix} r^4+2r^3+2r^2+1 & 2r^3+r^2+r & 2r^4+r^3+2r^2+2 \\ r^3+2r & r^4+2r^3+2r+2 & r^4+2r^2+r+1 \\ 2r^4+r^3+2r & 2r^4+r^2+r+1 & r^4+2r^3+r^2+2r \end{bmatrix}$$

4. Allice can get the secret message M as

$$M = D_1CD_2 \mod (m(r))$$
$$= \begin{bmatrix} 2r^4+2r^2+2 & r^4+r^3 & r^4+2r \\ 2r+2 & 2r^3+r & r^4+r^3+2r \\ 2r^3+2r & r^3+r & 2r^4+r^2 \end{bmatrix}$$

4.4 Security Analysis

In platform group $GL(n, \mathbb{F}_p)$, n should be very large to provide a sufficiently large key space. The authors proposed to take $n > 31$. So for practical instances of this scheme, one has to perform computation with matrices of large orders. The modified cryptosystem which is defined on matrices over extented Galois field $GF(p^q)$ provides large key space and message space when all basic parameters are small. The discrete log problem DLP and CSP involved in modified cryptosystem is harder to solve due to the use of polynomials instead of integers. Hence the overall security of the cryptosystem is increased.

4.4.1 Algebraic key recovery attack:

In algebraic key recovery attack, the attacker aim is to obtain secret cipher key. Equation (4.1) involves the product of A^s and B^t. For algebraic key recovery attack, an attacker has to solve the following system of equations.

$$\begin{cases} \mathcal{K}_{\mathcal{A}} = HI \\ AH^{-1} = H^{-1}A \\ BI^{-1} = I^{-1}B \end{cases} \tag{4.11}$$

where H and I are unknown matrices. For modified cryptosystem Equations (4.11) will translate into systems of large equations. Due to this algebraic key recovery attack becomes infeasible.

4.4.2 Ciphertext only Attack

As we have explained earlier about ciphertext only attack. Consider the adversary has only ciphertext pair (C', C). For modified structure $GL(n, \mathbb{F}_{p^q})$, system of Equations (3.7) will translate into a large and more complex system of equations. As the structure $GL(n, \mathbb{F}_{p^q})$ involves polynomials so this type of attack becomes infeasible.

4.5 Conclusion

In this thesis, we review a research paper "Cryptosystem based on noncommutative platform groups" [14] that utilize matrices over $\mathbb{F}_p$. We modified and improved this cryptosystem by replacing group of matrices over $\mathbb{F}_p$ with group of matrices over Galois field $GF(p^q)$ that provides higher security than the actual cryptosystem. For execution, we create computer programs by utilizing platform of computer algebra system ApCoCoA [1]. We give examples of modified and improved scheme by utilizing matrices over extented Galois field. One can broaden our work by using several noncommutaive algebraic structures to build an asymmetric key cryptosystem having potential against known quantum attacks.

Appendix A

A Modified and Improved Cryptosystem

A.1 ApCoCoA Code for Modified Cryptosystem

This section contain the ApCoCoA code for calculation of modified and improved cryptosystem in finite field extension $GF(p^q)$. It consists of **ModInv**, **PolyMod**, **PolyInvM**, **MatGF**, **MatInv**, **OrderMat**, **KeyMatGF**, **Encrypt**,**Decrypt**.

A.1.1 GFP(P,Q)

This function calculates the elements of Galois field. It requires the input P, Q where P is the prime number and Q is positive integer.

```
Define GFP(P,Q);
  Re:=[];
  For A:= 0 To P-1 Do
    GF:=Poly(A);
    Append(Re,GF);
  EndFor;
  GF1:=Re;
```

```
  GF2:=Re;
  For A:= 1 To Q-1 Do
    For J:=1 To P-1 Do
      Foreach P In GF1 Do
        F:=J*r^A+P;
        Append(GF2,F);
      EndForeach;
    EndFor;
    GF1:=GF2;
  EndFor;
Return GF2;
EndDefine;
```

A.1.2 ModInv(N,P)

This function calculates the inverse of a number $N \mod P$ by implementing Extended Euclidean Algorithm.

```
Define ModInv(N,P)
  A1:=1;A2:=0;A3:=P;
  B1:=0;B2:=1;B3:=N;
  While B3<0 Do
    B3:=B3+P;
  EndWhile;
  While B3 <>1 Do
    Q:=Div(A3,B3);
    If Q=0 Then
    Error(" Q is 0");
    EndIf;
    T1:=A1-Q*B1;T2:=A2-Q*B2;T3:=A3-Q*B3;
    A1:=B1;A2:=B2;A3:=B3;
    B1:=T1;B2:=T2;B3:=T3;
```

```
    If B2<0 Then
    B2:=B2+P;
    EndIf;
    If B3=1 Then
    Return B2;
    EndIf;
    If B3=0 Then
    Return("Not Invertible!");
    EndIf;
  EndWhile;
  If B2<0 Then
  B2:=B2+P;
  EndIf;
Return B2;
EndDefine;
```

A.1.3 PolyMod(F,M)

This function gives the polynomial F that is reduced by some polynomial mod M.

```
Define PolyMod(F,M)
  If Type(F)=RATFUN Then
    If Mod(Den(LC(F.Num)),M)=0 Then
      D:=Den(LC(F.Num));
      D2:=D*F.Den-D*LPP(F.Den);
        If D2= 0 Then
          Error("Zero Denominator .  .  .");
        EndIf;
      F:=D*F.Num/(D2);
      Return PolyMod(F,M);
    EndIf;
    CoefNum:=Coefficients(F.Num);CoefDen:=Coefficients(F.Den);
```

```
    For I:= 1 To Len(CoefNum) Do
      If Type(CoefNum[I])=RAT Then
        CoefNum[I]:=Mod(CoefNum[I].Num*ModInv(CoefNum[I].Den,M),M);
      Else
        CoefNum[I]:=Mod(CoefNum[I],M);
      EndIf;
    EndFor;
    For I:= 1 To Len(CoefDen) Do
      If Type(CoefDen[I])=RAT Then
        CoefDen[I]:=Mod(CoefDen[I].Num*ModInv(CoefDen[I].Den,M),M);
      Else
        CoefDen[I]:=Mod(CoefDen[I],M);
      EndIf;
    EndFor;
    NewNum:=ScalarProduct(CoefNum,Support(F.Num));
    NewDen:=ScalarProduct(CoefDen,Support(F.Den));
      If NewDen= 0 Then
        Error("Zero Denominator .  .  .");
      EndIf;
    Return NewNum/NewDen;
  EndIf;
  Coef:=Coefficients(F);
  For I:= 1 To Len(Coef) Do
    If Type(Coef[I])=RAT Then
      Coef[I]:=Mod(Coef[I].Num*ModInv(Coef[I].Den,M),M);
    Else
      Coef[I]:=Mod(Coef[I],M);
    EndIf;
  EndFor;
Return ScalarProduct(Coef,Support(F));
EndDefine;
```

A.1.4 PolyInvM(F,M,Md)

This function calculates inverse of polynomial F using Extended Euclidean Inverse Algorithm.

```
Define PolyInvM(F,M,Md)
  F:=NR(F,[M]);
  If MakeSet(Log(F))=[0] Then
    Return ModInv(LC(F),Md);
  EndIf;
  A1:=1;A2:=0;A3:=PolyMod(M,Md);
  B1:=0;B2:=1;B3:=PolyMod(F,Md);
  While MakeSet(Log(B3))<>[0] Do
    D:=DivAlg(A3,[B3]);
    Q:=D.Quotients[1];
    Coef:=Coefficients(Q);
      For I:= 1 To Len(Coef) Do
        C:=Coef[I];
        Coef[I]:=Mod(C.Num*ModInv(C.Den,Md),Md);
      EndFor;
    Q:= ScalarProduct(Coef,Support(Q));
      If Q=0 Then Error(" Q is 0");
      EndIf;
    T1:=PolyMod(A1-Q*B1,Md);
    T2:=PolyMod(A2-Q*B2,Md);
    T3:=PolyMod(A3-Q*B3,Md);
    A1:=B1;A2:=B2;A3:=B3;
    B1:=T1;B2:=T2;B3:=T3;
      If B3=1 Then
        Return PolyMod(B2,Md);
      EndIf;
  EndWhile;
  If B3<>1 Then
```

```
    Return PolyMod(NR(ModInv(LC(B3),Md)*B2,[M]),Md);
  Else
    Return PolyMod(B2,Md);
  EndIf;
EndDefine;
```

A.1.5 MatGF(A,M)

This function is use to reduce elements of matrix A in extented Galois field under certain irreducible polynomial mod M.

```
Define MatGF(A,M)
  Rows:=NumRows(A);Cols:=NumCols(A);
  For I:=1 To Rows Do
    For J:=1 To Cols Do
      A[I][J]:=PolyMod(NR(A[I][J],[M]),Md);
    EndFor;
  EndFor;
Return A;
EndDefine;
```

A.1.6 MatInv(J,M)

This function calculates inverse of matrix J under some mod M in extended Galois field.

```
Define MatInv(J,M);
  Adj:=MatGF(Adjoint(J),M);
  Dt:=Det(J);
  Inv:=NR(PolyInvM(Dt,M,2),[M]);
  If Inv=0 Then
    Return "Matrix is not invertible"
  EndIf;
```

```
  MulInv:=MatGF(Inv*Adj,M);
Return MulInv;
EndDefine;
```

A.1.7 OrderMat(A,M)

This function find the order of matrix in extended Galois field.

```
Define OrderMat(A,M)
  N:=1;
  If NumRows(A)=NumCols(A) Then
    O:=NumRows(A)
  Else
    Return "Matrix is not square"
  EndIf;
  A1:=A;
    While A1<> Identity(O) Do
      A1:=MatGF(A1*A,M);
      N:=N+1;
    EndWhile;
Return N;
EndDefine;
```

A.1.8 KeyMatGF(A,B,T,ORD1,ORD2,S,M)

This function calculates the public key of Allice.

```
Define KeyMatGF(A,B,T,ORD1,ORD2,S,M)
  If 0<T AND T<ORD1 Then
  If 0<S AND S<ORD2 Then
    K:=(A^T)*(B^S);
    Key:=MatGF(K,M);
  EndIf;
```

```
  EndIf;
Return Key;
EndDefine;
```

A.1.9 Encrypt(MSG,V,W,A,B,ORD1,ORD2,K,M)

This function is used for encryption of plaintext (MSG). It requires inputs MSG, V, W, A, B, ORD1, ORD2, K and M. Where (V,W) is secret key of sender Bob.

```
Define Encrypt(MSG,V,W,A,B,ORD1,ORD2,K,M)
  If 0<V AND V<ORD1 Then
  If 0<W AND W<ORD2 Then
    X:=MatGF((A^V)*(B^W),M);
    C1:=MatGF(MatInv(X,M)*K*X,M);
    C2:=MatGF(X*K*MatInv(X,M),M);
    C:=MatGF(C1*MSG*C2,M);
    CD:=MatGF((A^V)*K*(B^W),M);
  EndIf;
  EndIf;
Return [C,CD];
EndDefine;
```

A.1.10 Decrypt(C,CD,A,B,T,S,K,M)

Finally, this function is used for decryption of message encrypted by Bob through public key K of Allice.

```
Define Decrypt(C,CD,A,B,T,S,K,M)
  D:=MatGF(MatInv((A^T),M)*CD*MatInv((B^S),M),M);
  D1:=MatGF(MatInv(D,M)*MatInv(K,M)*D,M);
  D2:=MatGF(D*MatInv(K,M)*MatInv(D,M),M);
  MSG:=MatGF(D1*C*D2 ,M);
```

```
Return MSG;

EndDefine;
```

Bibliography

[1] "Apcoca team,ApCoCoA". ApCoCoA :Applied computation in commutative algebra. Available at http://www.apcocoa.org.

[2] I. Anshel, M. Anshel, & D. Goldfeld, "An algebraic method for public-key cryptography". *Mathematical Research Letters*, 6:287–292, 1999.

[3] W. G. Barker, "Introduction to the analysis of the Data Encryption Standard (DES)". Aegean Park Press, 1991.

[4] L. M. Batten, "Public key cryptography: applications and attacks", volume 16. John Wiley & Sons, 2013.

[5] D. J. Bernstein, "Post-quantum cryptography". In *Encyclopedia of Cryptography and Security*, pages 949–950. Springer, 2011.

[6] C. Cipher & M. Cipher, "Introduction to cryptography". *EEC*, 484:584, 2004.

[7] W. Diffie & M. Hellman, "New directions in cryptography". *IEEE transactions on Information Theory*, 22(6):644–654, 1976.

[8] J. B. Fraleigh, "A first course in abstract algebra". Pearson Education India, 2003.

[9] P. Garrett, "Making, breaking, codes". *Prentice Hall*, 117:118, 2001.

[10] L. Gu, L. Wang, K. Ota, M. Dong, Z. Cao, & Y. Yang, "New public key cryptosystems based on non-abelian factorization problems". *Security and Communication Networks*, 6(7):912–922, 2013.

[11] D. Hankerson, A. J. Menezes, & S. Vanstone, "Guide to elliptic curve cryptography". Springer Science & Business Media, 2006.

[12] M. S. Iqbal, S. Singh, & A. Jaiswal, "Symmetric key cryptography: Technological developments in the field". *International Journal of Computer Applications*, 117(15), 2015.

[13] D. Kahrobaei, C. Koupparis, & V. Shpilrain, "Public key exchange using matrices over group rings". *Groups-Complexity-Cryptology*, 5(1):97–115, 2013.

[14] S. Kanwal & R. Ali, "A cryptosystem with noncommutative platform groups". *Neural Computing and Applications*, 29(11):1273–1278, 2016.

[15] J. Katz, A. J. Menezes, P. C. Van Oorschot, & S. A. Vanstone, "Handbook of applied cryptography". CRC press, 1996.

[16] K. H. Ko, S. J. Lee, J. H. Cheon, J. W. Han, J.-s. Kang, & C. Park, "New public-key cryptosystem using braid groups". In *Annual International Cryptology Conference*, pages 166–183. Springer, 2000.

[17] K. H. Ko, D.-H. Choi, M. S. Cho, & J.-W. Lee, "New signature scheme using conjugacy problem.". *IACR Cryptology ePrint Archive*, 2002:168, 2002.

[18] C. M. Koupparis, "Non-commutative cryptography: Diffie-Hellman and CCA secure cryptosystems using matrices over group rings and digital signatures". City University of New York, 2012.

[19] A. K. Lenstra, "Integer factoring". In *Towards a quarter-century of public key cryptography*, pages 31–58. Springer, 2000.

[20] R. Lidl & H. Niederreiter, "Introduction to finite fields and their applications". Cambridge university press, 1994.

[21] R. Lidl & H. Niederreiter, "Finite fields", volume 20. Cambridge university press, 1997.

[22] K. S. McCurley, "The discrete logarithm problem, volume 42 of". In *Proceedings of Symposia in Applied Mathematics*, pages 49–74.

[23] C. Mullan, "Some Results in Group-basedcrytography". PhD thesis, Citeseer, 2011.

[24] M. A. Musa, E. F. Schaefer, & S. Wedig, "A simplified aes algorithm and its linear and differential cryptanalyses". *Cryptologia*, 27(2):148–177, 2003.

[25] A. Myasnikov, V. Shpilrain, & A. Ushakov, "Group-based cryptography". *Advanced Courses in Mathematics-CRM Barcelona [Centre de Recerca Matematica]. Bâle*, 2007.

[26] A. Myasnikov, V. Shpilrain, & A. Ushakov. "Group-based cryptography. advanced courses in mathematics. crm barcelona", 2008.

[27] T. Nie & T. Zhang, "A study of des and blowfish encryption algorithm". In *Tencon 2009-2009 IEEE Region 10 Conference*, pages 1–4. IEEE, 2009.

[28] A. Odlyzko, "Discrete logarithms: The past and the future". In *Towards a Quarter-Century of Public Key Cryptography*, pages 59–75. Springer, 2000.

[29] S.-H. Paeng, K.-C. Ha, J. H. Kim, S. Chee, & C. Park, "New public key cryptosystem using finite non abelian groups". In *Annual International Cryptology Conference*, pages 470–485. Springer, 2001.

[30] R. L. Rivest, A. Shamir, & L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems". *Communications of the ACM*, 21 (2):120–126, 1978.

[31] J. J. Rotman, "The theory of groups: an introduction". 1973.

[32] J. J. Rotman, "A first course in abstract algebra". Pearson College Division, 2000.

[33] T. Satoh & K. Araki, "On construction of signature scheme over a certain non-commutative ring". *IEICE transactions on fundamentals of electronics, communications and computer sciences*, 80(1):40–45, 1997.

[34] P. W. Shor, "Algorithms for quantum computation: Discrete logarithms and factoring". In *Foundations of Computer Science, 1994 Proceedings., 35th Annual Symposium on*, pages 124–134. Ieee, 1994.

[35] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer". *SIAM review*, 41(2):303–332, 1999.

[36] V. Shpilrain, "Cryptanalysis of stickel's key exchange scheme". In *International Computer Science Symposium in Russia*, pages 283–288. Springer, 2008.

[37] R. Singh & S. Kumar, "Elgamal's algorithm in cryptography". *International Journal of Scientific & Engineering Research*, 3(12):1–4, 2012.

[38] M. Sramka, "On the security of stickel's key exchange scheme". *preprint*, 178, 2008.

[39] E. Stickel, "A new method for exchanging secret keys". In *Information Technology and Applications, 2005. ICITA 2005. Third International Conference on*, volume 2, pages 426–430. IEEE, 2005.

[40] D. R. Stinson, "Cryptography: theory and practice". CRC press, 2005.

[41] N. R. Wagner & M. R. Magyarik, "A public-key cryptosystem based on the word problem". In *Workshop on the Theory and Application of Cryptographic Techniques*, pages 19–36. Springer, 1984.

[42] S. William, "Cryptography and network security: principles and practices". Pearson Education India, 2006.