ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 21, August 2017

# REVIEW ON APPLICATION OF MATHEMATICS IN CRYPTOGRAPHY

[1]M.Durga Ratnam,.[2]G. Srinivasa Rao

[1,2]Assistant Professors

Sir CRR College of Engineering, Eluru

**Abstract:**

Information processing by electronic devices leads to a multitude of security relevant challenges. With the help of cryptography, many of these challenges can be solved and new applications can be made possible in branches of Mathematics.Cryptography draws on many areas of mathematics, including number theory, abstract algebra, probability, and information theory. In this paper explained that cryptography has its foundations deeply embedded in Mathematics for encryption and decryption. The Mathematical background required for this purpose this paper consists of advanced topics such as Modular arithmetic, Finite fields and Prime numbers, Groups, Lattices reduction algorithm, Linear algebra and Geometry.

**Key words:** linear algerbrElliptic curve cryptography, Authenticated key exchange protocol,Function density problems, modulo's, Groups, Lattice

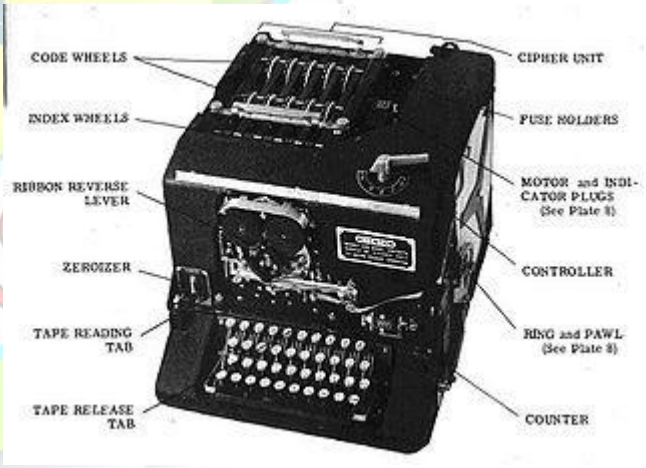
**Introduction:**

Cryptography: The word "cryptography" is derived from the Greek words "kruptoc", 'hidden', and "grafein", 'to write'; it is therefore about secret writing-"hidden secret".

Cryptography is a mechanism to encode and decode secret Messages for protecting message from unauthorized users to Access the messages. In a network environment cryptography Is playing a main role in data protection in applications Running.

In the past cryptography was used by the political Sectors of intelligence and military. Until the end of the First World War, cryptology developed slowly and the schemes used were, mathematically speaking, elementary from today's point of view.

But presently it is commonly utilized in the ATM cards, e-commerce, e-mail, Computer password, and other application over the years.




There are different algorithm have available to modify of a Message by an encrypting key which is known by sender and Receiver. The message could not be decrypted without using encrypting key. One of the issue is appeared with Cryptography is that the message always clear to intermediate Person that the message is encrypted form. This means that the sender of the message does not want it to be read by unauthorized person.

Today, there are many cryptography Techniques which are capable of encrypting data, one of the Most widely technique is affine algorithm. Affine has the

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 21, August 2017

ability to convert the information to a form not understandable by the intruder.

Secret agents, online stores and pupils exchanging "secret messages" consisting of nonsensical symbols all use it: cryptography. The sender encrypts the message and the receiver decrypts it with an agreed upon secret. The endeavor to read encrypted messages without knowledge of the secret is called cryptanalysis. It is common to summarize both aspects under cryptology.

Due to the technical development in the field of electronics, the notions of cryptography and cryptology are nowadays used more broadly; the goals of cryptography now cover all aspects of security in processing, transmission and use of information in the presence of an adversary.

In this way, cryptographic methods have entered many different areas. One can use them to ensure confidentiality in any kind of electronic communication. They are used for authentication when unlocking a car or releasing an immobilizer, withdrawing money with a bank card or identifying oneself at a border with a passport, for example. Documents are nowadays often signed digitally with cryptographic methods, for example by a notary; like this the non-repudiation of agreements can be guaranteed. With digital signatures one can also guarantee the integrity of electronic data, that is, that the data has not been tampered with; this is for example used in passports.

**New ideas**

According to the technical development the history of cryptology can be divided into
three periods:

a) The paper-and-pencil era until about the end of World War I.
b) The era of electric-mechanic cipher machines from about the end of World War I until about 1970.

c) The electronic era from about 1970.

As the name indicates, the first period was characterized by the fact that at most simple mechanical devices came to use for secret writing. For breaking
schemes, beside ad hoc approaches mainly statistical methods as described were applied.

In the second period, for encryption next to schemes for writing by hand, electric-mechanic machines like the German Enigma were used. The increase of sophistication in the electronic-mechanic encryption machines was countered by cryptanalytic methods which exceeded purely statistical techniques. Cryptanalysis was a driving force in the development and construction of the first electronic computing machines.

The electronic era started with the advent of data processing. As the methods developed in the beginning of the era are still being used or the current methods are direct successors of these methods, it can be seen as the present age of cryptology.

This period is characterized not only by the used technology but also by its strive
for scientific methods in cryptography, a strong connection to mathematics and a
high innovation speed.


**TYPES OF CRYPTOGRAPHIC ALGORITHMS**

There are several ways of classifying cryptographic algorithms. Here they will be categorized based on the number of keys that are employed for encryption and decryption

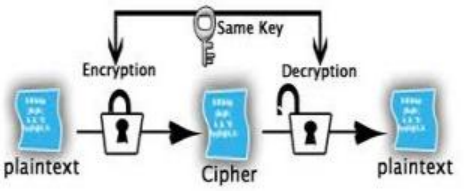
There three types of algorithms

**I. Secret key cryptography (SKC) :**

In this Cryptography, uses a single key for both encryption and decryption, also called symmetric encryption. Primarily used for privacy and confidentiality.

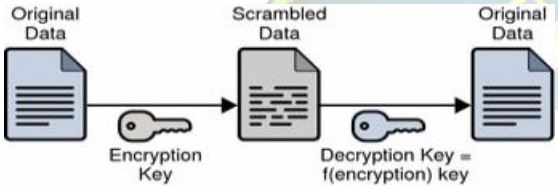
ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 21, August 2017




## II. Public-key cryptography (asymmetric-key cryptography)

Public-Key Encryption uses one key for encryption and another for decryption. Public-Key Encryption also called asymmetric encryption. Primarily used for authentication, non reputation and key exchange




Public key cryptography depends upon the existence, so called one-way function, or mathematical functions that are easy to compute where as their inverse function is relatively difficult to compute.
Let we give Two simple examples

(1). Multiplicative VS Factorization:

Suppose you have two prime numbers 3 and 7, and you need to calculate the product. It should take almost no time to calculate the value which is 21

Now suppose, instead of that you have a number i.e. product of two primes factors, then you will eventually come up with the solution but whereas calculating the product took milliseconds, factoring will take longer. The problem becomes harder if we start with primes that have, say 400 digits or so, because the product will have ~800 digits.

(2). Exponentiation VS Logarithms:

Suppose you take the number 3 to the power 6, again it takes relatively easy to calculate $3^6 = 729$.

But if you start with the number 729 and need to determine the two integers, x and y; so that $log_x 729 = y$. It will take larger to find the two values.

## III. Hash function

Hash functions can be used to verify digital signatures, so that when signing documents via the Internet, the signature is applied to one particular individual. Much like a hand-written signature, these signatures are verified by assigning their exact hash code to a person. Furthermore, hashing is applied to passwords for computer systems. Hashing for passwords began with the UNIX operating system. A user on the system would first create a password. That password would be hashed, using an algorithm or key, and then stored in a password file. This is still prominent today, as web applications that require passwords will often hash user's passwords and store them in a database (Ref-h)

Relations of Function Density Problems to some concrete topics in cryptography will be follows

Function Density Problems(FDP)**:** Let C be a set of some functions, and let C ′ be a subset of C. Let d(·, ·) be a distance function for the pairs of functions in C. In this setting, we define a Function Density Problem to be a problem of estimating the following quantity:

r(C, C ′ ) := sup{d(f, C ′ ) | f ∈ C} ------( 1 )

where, for each f ∈ C, d(f, C ′ ) := inf{d(f, g) | g ∈ C′} is the distance from f to C ′ . (The symbol 'r' stands for "radius", by an analogy as if C ′ is a single central point in the figure C, in which case the r is the radius of C in usual sense.)

Among very various situations covered by Definition 1 (where C in fact need not even to be a set of functions), in the applications of FDPs discussed in this paper we will focus on the following typical cases:

Function Density Problems – typical cases**:** Let C be the set of all functions f : X → Y from a given finite set X to a given finite set Y . Let C ′ ⊂ C. For any f, g

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 21, August 2017

∈ C, we define the distance between f and g by

$$dH(f, g) := |\{x \in X \mid f(x) \neq g(x)\}| . \quad (2)$$

In this setting, a Function Density Problem is a problem of estimating the quantity r(C, C ′ ) defined by (1) with d(·, ·) = dH(·, ·).

In the case of Definition (2), the "sup" and "inf" in Definition 1 can be simply replaced with "max" and "min", respectively. Moreover, the distance defined by (2) coincides with the (generalized) Hamming distance when members of C are identified with sequences of length |X| over the alphabet Y in a natural manner. Note that the quantity r(C, C ′ ) can be regarded as a special case of so-called Hausdorff distance for two subsets of a metric space, which would support that it is reasonable to consider r(C, C ′ ).

Here we propose a new framework for theoretical security evaluation of keyless hash functions based on FDPs. Although theoretical security evaluation of keyless hash functions is evidently an extremely difficult problem and our proposed framework is unfortunately not yet practical, we hope that our framework can be a clue to this problem (Ref i, j, k).

We consider a keyless hash function H : X → Y with possibly large but finite domain X and relatively small (finite) range Y . Among the major security requirements for hash functions, we focus on the collision resistance of H; we discuss how it is difficult to find a collision pair (x1, x2) for H (recall that (x1, x2) is called a collision pair for H if we have x1, x2 ∈ X, x1 ≠ x2 and H(x1) = H(x2)).

To show the relevance of FDPs to this problem, first we give a somewhat informal description of an abstract "typical" strategy for finding a collision pair:

i) Construct a close approximation H′ : X → Yof H in such a way that collision pairs for H′ can be found with reasonable computational time.

ii)Find randomly a collision pair (x ′1, x′2) for H′

iii) Construct from (x ′ 1 , x′ 2 ) a candidate (x1, x2) of a collision pair for H (in the simplest case, we just set (x1, x2) = (x′1,x′2 ).

iv) Check if (x1, x2) is a collision pair of H; if itis indeed a collision pair of H, then output (x1, x2) and stop the process.

v) If (x1, x2) is not a collision pair of H, go back to Step (2) and repeat the process

.

**Public-key Crypto vs Private-key Crypto**

| Private-Key Cryptography | Public-Key Cryptography |
|---|---|
| - Key distribution has to be done apriori. | + Key distribution can be done over public channel |
| - In multi-sender scenario, a receiver need to hold one secret key per sender | + One receiver can setup a single public-key/ secret key and all the senders can use the same public key |
| - Well-suited for closed organization (university, private company, military). Does not work for open environment (Internet Merchant) | + Very fast computation. Efficient Communication. Only way to do crypto in resource-constrained devices such as mobile, RFID, ATM cards etc |
| + Very fast computation. Efficient Communication. Only way to do crypto in resource-constrained devices such as mobile, RFID, ATM cards etc | + Very fast computation. Efficient Communication. Only way to do crypto in resource-constrained devices such as mobile, RFID, ATM cards etc |
| + Very fast computation. Efficient Communication. Only way to do crypto in resource-constrained devices such as mobile, RFID, ATM cards etc | - Relies on the fact that there is a way to correctly send the public key to the senders (can be ensured if the parties share some prior info or there is a trusted party) |

## MATHEMATICAL APPLICATIONS TO CRYPTOGRAPHY

Cryptography has its foundations deeply embedded in Mathematics. The Mathematical background required for this purpose this paper consists of Modular arithmetic, Finite fields and Prime numbers because cryptography is associated with additive and

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 21, August 2017

multiplicative binary operatoions and elliptic curve in cryptosystem

## 1. MODULAR ARITHMETIC

According to Modular arithmetic that deals with infinite set of integers denoted by the set Z = { ….,-2,-1,0,1,2,…..}, and the arithmetic operations addition, subtractions, multiplications and division.

Modulo-n based on

- Finite set of non negative integers from 0 to n-1 and
- Binary arithmetic operations - addition, subtractions, multiplications.

Note that we have not included division as a part of modulo arithmetic since it is not binary operation.

The Modular arithmetic operations are mapped to $Z_n$= { 0,1,2,…..n-1}.

Measured number-crunching is fundamentally doing expansion (and different operations) - the qualities "wrap around", continually remaining not exactly a settled number called the modulus.

To discover, for instance, 39 modulo 7, you basically compute 39/7 (= 5 4/7) and take the rest of. For this situation, 7 partitions into 39 with a rest of 4. Subsequently, 39 modulo 7 = 4. Take note of that the rest of (partitioning by 7) is constantly under 7. In this way, the qualities "wrap around," as should be obvious beneath:

| | |
|---|---|
| 0 mod 7=0 | 6 mod 7=6 |
| 1 mod 7=1 | 7 mod 7=0 |
| 2 mod 7=2 | 8 mod 7=1 |
| 3 mod 7=3 | 9 mod 7=2 |
| 4 mod 7=4 | 10 mod 7=3 |
| 5 mod 7=5 | etc. |

Examples:

(14+11) mod 17= 25mod 17= 8(mod17)

( 3-8 ) mod 17= 5 mod 17 = 5 (mod 17)

(3-7) mod 17 = (-4) mod 17 = 13 (mod 17)

( 14X2) mod 17 = 28 mod 17 = 11( mod 17)

**Problem (1):**

If $a = 58, b = 73, find (a \times b) mod\ 7$.

**Solution:** $(58 \times 73) mod\ 7 = \{(58\ mod\ 7) \times (73\ mod\ 7)\} mod\ 7$

$= (2 \times 3) mod 7 = 6.$

**Problem(2):** Compute $12^{10} mod\ 7$.

**Solution:** $12^{10} mod\ 7 = 5^{10} mod\ 7 = 25^{5} mod\ 7$

$= 4^{5} mod\ 7 = 16 \times 16 \times 4\ mod 7$

$= 2 \times 2 \times 4\ mod\ 7 = 2.$

## 2. Encryption and Decryption using Additive Inverse

To demonstrate a simple application of additive inverse for encryption and decryption, Let us represent each letter of English alphabet by an integer that corresponds to its position in the alphabet. Thus **7** corresponds to **' H'**if we starts with A=*0.* All the letters are represented by set $Z_{26.}$

Encryption is carried out by modulo 26 addition of numerical representation of letter and a fixed number called Key.

If the key is 20, **H** is encrypted as:( 7+20 ) mod 26 =1, which corresponds to letter **B**.

For Decryption, we use the encryption algorithm with the additive inverse of the key. The additive inverse of 20 is 6 in $Z_{26.}$ Therefore, letter **B** is Decrypted as

(1+6) mod 26= 7, which is letter **H.**

**Example:** Encrypt NET if the key is 11. And Decrypt YPE if the key is 11

**Solution:**

Encryption of Net:

$N = 13 \Rightarrow (13 + 11) mod\ 26 = 24 \Rightarrow Y$

$E = 4 \Rightarrow (4 + 11) mod\ 26 = 15 \Rightarrow P$

$T = 19\ (19 + 11) mod\ 26 = 4 \Rightarrow E$

Decryption of YPE : Additive inverse of 11 modulo 26 is 15

$Y = 24 \Rightarrow (24 + 15) mod\ 26 = 13 \Rightarrow N$

$P = 15 \Rightarrow (15 + 15) mod\ 26 = 4 \Rightarrow E$

$E = 4 \Rightarrow (4 + 15) mod\ 26 = 19 \Rightarrow T.$

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 21, August 2017

## 3. APPLICATIONS OF LINEAR ALGEBRA TO CRYPTOGRAPHY

Cryptography, to most people, is concerned with keeping communications private. Indeed, the protection of sensitive communications has been the emphasis of

cryptography throughout much of its history. Encryption is the transformation of data into some unreadable form. Its purpose is to ensure privacy by keeping the information hidden from anyone for whom it is not intended, even those who can see the encrypted data. Decryption is the reverse of encryption; it is the transformation of encrypted data back into some intelligible form.

Encryption and decryption require the use of some secret information, usually referred to as a key. Depending on the encryption mechanism used, the same key might be used for both encryption and decryption, while for other mechanisms, the keys used for encryption and decryption might be different.

Today governments use sophisticated methods of coding and decoding messages. One type of code, which is extremely difficult to break, makes use of a large matrix to encode a message. The receiver of the message decodes it using the inverse of the matrix. This first matrix is called the **encoding matrix** and its inverse is called the **decoding matrix.**

**Example** :Let the message be

**F R I E N D  R E Q U E S T**

and the encoding matrix be

$$\begin{bmatrix} -3 & -3 & -4 \\ 0 & 1 & 1 \\ 4 & 3 & 4 \end{bmatrix}$$

We assign a number for each letter of the alphabet. For simplicity, let us associate each letter with its position in the alphabet: A is 1, B is 2, and so on. Also, we assign the number 27 (remember we have only 26 letters in the alphabet) to a space between two words. Thus the message becomes:

**F R I E N D* R E Q U E S T**
**6 18 9 5 14 4 27 18 5 17 21 5 19 20**

Since we are using a 3 by 3 matrix, we break the enumerated message above into a sequence of 3 by 1 vectors:

$$\begin{bmatrix} 6 \\ 18 \\ 9 \end{bmatrix} \begin{bmatrix} 5 \\ 14 \\ 4 \end{bmatrix} \begin{bmatrix} 27 \\ 18 \\ 5 \end{bmatrix} \begin{bmatrix} 17 \\ 21 \\ 5 \end{bmatrix} \begin{bmatrix} 19 \\ 20 \\ 27 \end{bmatrix}$$

Note that it was necessary to add a space at the end of the message to complete the last vector. We now encode the message by multiplying each of the above vectors by the encoding matrix. This can be done by writing the above vectors as columns of a matrix and perform the matrix multiplication of that matrix with the encoding matrix as follows:

$$\begin{bmatrix} -3 & -3 & -4 \\ 0 & 1 & 1 \\ 4 & 3 & 4 \end{bmatrix} \begin{bmatrix} 6 & 5 & 2717 & 19 \\ 18 & 14 & 1821 & 20 \\ 9 & 4 & 5\ 5 & 27 \end{bmatrix}$$

which gives the matrix

$$\begin{bmatrix} -108 & -73 & -155 & -134 & -225 \\ 27 & 18 & 23 & 26 & 47 \\ 114 & 78 & 182 & 151 & 244 \end{bmatrix}$$

To decode the message, the receiver writes this string as a sequence of 3 by 1 column matrices and repeats the technique using the inverse of the encoding matrix. The inverse of this encoding matrix, the decoding matrix, is:

$$\begin{bmatrix} 1 & 0 & 1 \\ 4 & 4 & 3 \\ -4 & -3 & -3 \end{bmatrix}$$

Thus, to decode the message, perform the matrix multiplication

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 21, August 2017

$$\begin{bmatrix} 1 & 0 & 1 \\ 4 & 4 & 3 \\ -4 & -3 & -3 \end{bmatrix} \begin{bmatrix} -108 & -73 & -155 & -134 & -225 \\ 27 & 18 & 23 & 26 & 47 \\ 114 & 78 & 182 & 151 & 244 \end{bmatrix}$$

and get the matrix

$$\begin{bmatrix} 6 & 5 & 2717 & 19 \\ 18 & 14 & 1821 & 20 \\ 9 & 4 & 5 & 5 & 27 \end{bmatrix}$$

The columns of this matrix, written in linear form, give the original message:

| F | R | I | E | N | D* | R | E | Q | U | E | S | T |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 6 | 18 | 9 | 5 | 14 | 4 | 27 | 18 | 5 | 17 | 21 | 5 | 19 | 20 |

## 4. A COMMUNICATION GAME BY CRYPTOGRAPHY

The cryptographic application serves three purposes as following.,

- To provide an initial demonstration on the effectiveness and practically of using cryptography for solving subtle problems in applications
- To suggest an initial hint on the foundation of Cryptography
- To begin our process of establishing a required mindset for conducting the development of Cryptographic systems for information security.

According to a problem, it is trivially simple and solve it with an equal simple solution. The solution is a two-party game which is very familiar to all of us. However we will realize that our simple game soon becomes troublesome when our game playing parties are physically remote from each other. The physical separation of the Game playing parties eliminates the basis for the game to be played fairly. The trouble then is, the game playing parties cannot trust the other side to play the game fairly.

The need for a fair playing of the game for remote players will **inspire** us to strengthen our simple game by protecting it with a shield of armor.

Here a simple example, Imagine that the two friends Alice and Bob are trying to run this protocol over the telephone. Alice offers Bob, " You pick a side. Then I will toss the coin and tell you whether or not you have won."

Of course Bob will not agree, because he cannot verify the outcome of the coin toss.

However we can add a little bit of cryptography to this protocol and turn it into a version workable over the phone. The result will become a Cryptographic protocol, our first Cryptographic protocol, For the time being, let us just consider our Cryptography as mathematical function f(x) which maps over the integers and has the following properties

Properties : Magic function f

- For every integer x, it is easy to compute f(x) from x, while given any value f(x) it is impossible to find any information about a pre-image x, e.g., Whether x is an odd or even number.
- It impossible to find a pair of integers (x,y) satisfying $x \neq y and f(x) = f(y)$.

Protocol for the above, Coin flipping over the telephone,

Premise: Alice and Bob have agreed:

i. A magic function f with properties specified in above property
ii. An even number x in f(x) represents HEADS and the other case represents TAILS

Here, this protocol has a weakness

a. Alice picks a large random integer x and computers f(x); She reads f(x) to Bob over the phone;
b. Bob tells Alice his guess of x as even or odd;
c. Alice reads x to Bob;
d. Bob verifies f(x) and sees the correctness/incorrectness of his guess.

## 5. AN ALGEBRAIC METHOD FOR PUBLIC-KEY CRYPTOGRAPHY

A protocol is a multi–party algorithm, defined by a sequence of steps, specifying the actions required of two or more parties in order to achieve a specified objective. Furthermore, a key establishment protocol is a protocol whereby a shared secret becomes available to two or more parties, for subsequent cryptographic applications (see [c]).

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 21, August 2017

A compact algebraic key establishment protocol followed by a group–theoretic illustration, for secret key establishment between two individuals whose only means of communication is a public channel. The foundation of the method lies in the difficulty of solving equations over algebraic structures, in particular groups. The protocol requires each party to perform an algebraic computation (several multiplications followed by rewriting in a monoid or group).
The results of the computation are then exchanged between the parties over a public channel and a common shared secret key is then obtained by each party after a second computation is performed. The second computation involves an algorithm to solve the word problem in the monoid or group.
In the case that the protocol is group–based, we show that an adversary (observing all communication over the public channel) can break the scheme and determine the secret key provided a system of conjugacy equations over the associated group is feasibly solvable.
Further, there are many groups where the word problem is known to be solvable in polynomial time while there is no known polynomial time algorithm to solve the conjugacy problem. An example is the braid group on n strands where the word problem for a word w (of length |w|)
Recent developments in mathematical and computational cryptanalysis (see [d,e]) have renewed interest in developing new cryptographic methods. These methods include public–key cryptography based on hidden monomial systems, combinatorial–algebraic systems, and the theories of elliptic and hyperelliptic curves (see [f]).

## 6. GROUPS - CRYPTOGRAPHY

There are some widely used cryptographic algorithms which need a finite, cyclic group (a finite set of element with a composition law which fulfils a few characteristics), e.g. DSA or Diffie-Hellman.
The group must have the following characteristics:

- Group elements must be represent able with relatively little memory.
- The group size must be known and be a prime number (or a multiple of a known prime number) of appropriate size (at least 160 bits for the traditional security level of "80-bit security").
- The group law must be easy to compute.
- It shall be hard (i.e. computationally infeasible, up to at least the targeted security level) to solve discrete logarithm in the group.

DSA, DH, ElGamal... were primarily defined in the group of non-zero integers modulo a big prime *p*, with modular multiplication as group law. The characteristics we look for are reached as long as *p* is large enough, e.g. at least 1024 bits (that's the minimal size for discrete logarithm to be hard in such a group).

## 7. ELLIPTICCURVE CRYPTOGRAPHY (ECC)

Elliptic curve cryptography (ECC) is an approach to public-key Cryptography based on the algebraic structure of elliptic curves over finite fields. ECC requires smaller keys compared to non-ECC cryptography (based on plain Galois fields) to provide equivalent security.

Elliptic curve are another kind of group, appropriate for group-based cryptographic algorithm. An elliptic curve is defined with:

- A finite field, usually consisting in integers modulo some prime *p* (there are also other fields which can be used).
- A curve equation, usually $y^2 = x^3 + ax + b$, where a and b are constant values from the finite field.

The curve is the set of pairs of values (x,y) which match the equation, along with a conventional extra element called "the point at infinity".
Since elliptic curves initially come from a graphical representations (when the field consists in the real numbers RR), the curve elements are called "*points*" and the two values xx and y are their "*coordinates*".

Then we define a group law, called *point addition* and denoted with a "+" sign. The definition looks quite artificial, with all the business about tracing a line and computing the intersection of that line with the curve; but the bottom-line is that it has the characteristics required for a group law, and it is easily computable (there are several methods; as a

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 21, August 2017

rough approximation, it costs about 10 multiplications in the base field). The *curve order* (the number of points on the curve) is close to p (the size of the finite field): the curve order is equal to $p+1-t$ for some integer t such that $|t| \leq 2\sqrt{p}$. Compared to the traditional multiplicative group modulo a big prime, elliptic curve variants of cryptographic algorithms have the following practical features:

- **They are *small* and *fast***. There is no known efficient discrete-logarithm solving algorithm for elliptic curves, beyond the generic algorithms which work on every group. So we get appropriate security as soon as p is close to 160 bits. Computing the group law costs ten field operations, but on a field which is 6 times smaller; since multiplications in a finite field have quadratic cost, we end up with an appreciable speedup.
- **Creating a new curve is uneasy**. Generating a new big prime is a matter of a fraction of a second with a basic PC, but making a new curve is much more expensive (the hard part is figuring out the curve order). Since there is no security issue in using the same group for several distinct key pairs, it is customary, with elliptic curves, to rely on a handful of standard curves which have been created such that their order is appropriate (a big prime value or a multiple of a big enough prime value);. The implementations are thus specialized and optimized for these particular curves, which again considerably speeds things up.
- **Elliptic curves can be used to factor integers**. Lenstra's elliptic curve factorization method can find some factors in big integers with a devious use of elliptic curves. This is not the best known factorization algorithm, except when it comes to finding medium-sized factors in a big non-prime integer.
- **Some elliptic curves allow for pairings**. A pairing is a bilinear operation which can link elements from two groups into elements of a third group. A pairing for cryptography requires all three groups to be "appropriate" (in particular with a hard-to-solve discrete logarithm). Pairings are an active research subject because they can be used to implement protocols with three participants (e.g. in electronic cash systems, with the buyer, the vendor and the bank, all mathematically involved in the system). The only known practical pairings for cryptography use some special elliptic curves.

Elliptic curves are usually said to be the next generation of cryptographic algorithms, in order to replace RSA. Performance of EC computations is the main interest of these algorithms, especially on small embedded systems such as smartcards (in particular Koblitz curves over binary fields); the biggest remaining issue is that public-key operations with group-based algorithms are a bit slow (RSA signature *verification* or asymmetric *encryption*, as opposed to signature generation and asymmetric decryption, respectively, is extremely fast, whereas analogous operations in the group-based algorithms are just fast).

Also, involved mathematics are a bit harder than with RSA, and there have been patents, so implementers are a bit wary. Yet elliptic curves become more and more common.

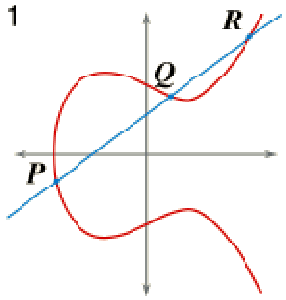
The equation of an elliptic curve is given as,

$y^2 = x^3 + ax + b$, Few terms that will be used,

**E -> Elliptic Curve**
**P -> Point on the curve**
**n -> Maximum limit ( This should be a prime number )**

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 21, August 2017




$$y^2 = x^3 + ax + b$$

**Key Generation**

Key generation is an important part where we have to generate both public key and private key. The sender will be encrypting the message with receiver's public key and the receiver will decrypt its private key.

Now, we have to select a number **'d'** within the range of **'n'**.
Using the following equation we can generate the public key

**Q = d * P**
**d** = The random number that we have selected within the range of ( **1 to n-1** ).
**P=** is the point on the curve.
**'Q' is the public key** and **'d' is the private key.**

**Encryption**

Let 'm' be the message that we are sending. We have to represent this message on the curve. This have in-depth implementation details. All the advance research on ECC is done by a company called certicom.

Conside *'m'* has the point *'M'* on the curve *'E'*. Randomly select 'k' from [1 – (n-1)].
Two cipher texts will be generated let it be **C1** and **C2**.

**C1 = k*P**
**C2 = M + k*Q**
C1 and C2 will be send.

**Decryption**
We have to get back the message 'm' that was send to us,

**M = C2 – d * C1**
M is the original message that we have send.

**Proof**
How does we get back the message,

M = C2 – d * C1

'M' can be represented as 'C2 – d * C1'

C2 – d * C1 = (M + k * Q) – d * ( k * P ) ( C2 = M + k * Q and C1 = k * P )

= M + k * d * P – d * k *P ( canceling out k * d * P )

= M ( Original Message ).

## 8. LATTICE-BASED CRYPTOGRAPHY

Lattice-based cryptographic constructions hold a great promise for post-quantum cryptography, as they enjoy very strong security proofs based on worst-case hardness, relatively efficient implementations, as well as great simplicity. In addition, lattice-based cryptography is believed to be secure against quantum computers. [8] discussed about a system,the effective incentive scheme is proposed to stimulate the forwarding cooperation of nodes in VANETs. In a coalitional game model, every relevant node cooperates in forwarding messages as required by the routing protocol. This scheme is extended with constrained storage space. A lightweight approach is also proposed to stimulate the cooperation.

Lattice based cryptography is starting to become quite popular in academia. The primary benefit of lattice based crypto is the resistance to quantum algorithms. (Ref –g)

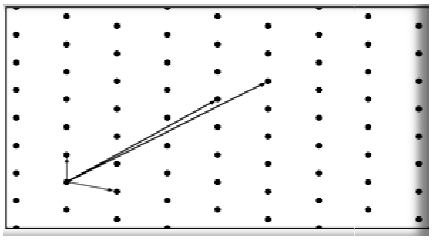
- **Post-quantum security:** As you note, quantum attacks are not known to break lattice-based cryptosystems. But some other proposals like McEliece, as well as most symmetric primitives

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 21, August 2017

are not known to be poly-time breakable on a quantum computer.

- **Security from *worst case* assumptions:** In security proofs for cryptosystems we typically assume that some problem is *hard on average* or more precisely *hard to solve for random instances drawn from some particular distribution*. For example, we may assume that factoring a product of two random primes cannot be done by a poly-time algorithm. While this is usually safe, it is in principle possible that someone will find an efficient factoring algorithm that works often on random instances but not on all instances. Lattice-based cryptography does not suffer from this drawback: Those schemes are proven secure assuming that lattice problems are hard in the *worst case*, meaning they are secure as long as no one can find, say, a poly-time algorithm for approximating shortest vectors in *every* lattice, not just random ones. This is a huge theoretical advance, but determining exactly what it will mean in practice is difficult for me to say.
- **Efficiency improvements:** I'll be a little sheepish on this point, but it's often noted that lattice-based schemes have a parallelizable structure that may make them faster in certain contexts. This is because the algorithms involved are usually simple matrix multiplication with relatively small modular arithmetic (i.e., not cryptographic-sized numbers). However, my understanding is that implementations of lattice-based schemes would have larger keys, and I'm also not aware of any studies that definitively compare lattice based schemes to traditional schemes.
- **New primitives:** We only know how to build fully homomorphic encryption from lattices or from very similar techniques. There is a ton of potential here, and we have no idea how to build such things from factoring or other traditional assumptions.



**CONCLUSION:**

In this paper, we discussed the mathematical models in several topics association with concepts of cryptography and its implementation in the computer algebra system. Among these methodologies we are invents and illustrates the number of areas in science and computations by the cryptography.

**References:**

[1] Introduction to Cryptography- Buchmann, Johannes. New York : Springer, 2001.
[2] Matrices Applications to Cryptography University of Ottawa http://aix1.uottawa.ca/~jkhoury/cryptography.htm
[3] A.J. Menezes, P.C. van Oorschot, and S.A. Vanstone, Handbook of applied cryptography,
[4] CRC Press Series on Discrete Mathematics and its Applications., CRC Press, Boca Raton, FL, 1997.
[5] D. Boneh, Twenty years of attacks on the RSA cryptosystem, Notices Amer. Math. Soc. 46 (1999), 203–213.
[6] P.C. van Oorschot and M.J. Wiener, Parallel collision search with cryptanalytic applications, J. Cryptology 12 (1999), 1–28.
[7] N. Koblitz, Algebraic aspects of cryptography, Algorithms and Computation in Mathematics, 3.,Springer-Verlag, Berlin, 1998.
[8] Christo Ananth, M.Muthamil Jothi, A.Nancy, V.Manjula, R.Muthu Veni, S.Kavya, “Efficient message forwarding in MANETs”, International Journal of Advanced Research in Management, Architecture, Technology and Engineering (IJARMATE), Volume 1,Issue 1, August 2015,pp:6-9

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 21, August 2017

[9] Grah, Joseph Sterling. "Hash Functions in Cryptography" (pdf). Retrieved 18 September 2013.
[10] J. L. Carter and M. N. Wegman, Universal classes of hash functions (extended abstract), Proc. STOC 1977 (1977), pp. 106–112.
[11] K. Nuida, T. Abe, S. Kaji, T. Maeno and Y. Numata, A mathematical problem for security analysis of hash
[12] functions and pseudorandom generators, Proc. IWSEC 2011 (2011), pp. 144–160.
[13] L. Blum, M. Blum and M. Shub, A simple unpredictable pseudo-random number generator,
[14] SIAM J. Comput. 15 (1986) 364–383.
[15] J. L. Carter and M. N. Wegman, Universal classes of hash functions (extended abstract), Proc.
[16] STOC 1977 (1977), pp. 106–112.
[17] K. Nuida, T. Abe, S. Kaji, T. Maeno and Y. Numata, A mathematical problem for security
[18] analysis of hash functions and pseudorandom generators, Proc. IWSEC 2011 (2011), pp.
[19] 144–160.
[20] E. Blais, J. H˚astad, R. A. Servedio and L.-Y. Tan, On DNF approximators for monotone
[21] boolean functions, Proc. ICALP 2014, Part I (2014), pp. 235–246.
[22] Cyclic Group Cryptography with Elliptic Curves, Brasov, May (2011).
[23] Hong Liu and Yanbing Liu, Cryptanalyzing an Image Encryption Scheme based on Hybrid
[24] Chaotic System and Cyclic Elliptic Curve, In *Optics and Laser Technology*, Elsevier, vol. 56, pp.
[25] 15-19, (2014).
[26] N.P. SmartAn identity based authenticated key agreement protocol based on the Weil
[27] PairingElectronics
Letters, 38 (2002), pp. 630-632
[28] N. Koblitz, Elliptic curve cryptosystem", *Journal of Mathematics of Computation* 1987;
[29] 48(177):203–209.
[30] S. Blake-Wilson, D. Johnson, A. Menezes., "Key agreement protocols and their security
[31] analysis", *Proc. of the 6th IMA International Conference on Cryptography and Coding*, LNCS,
[32] Springer-Verlag, 1997; 1335:30–45.
[33] C. Teoh, Two-dimensional barcodes for hardcopy document integrity verification
[34] (2008). URL http://eprints.utm.my/9467.
[35] S. S. Roy, et al., "Compact Ring-LWE Cryptoprocessor," Lecture Notes in Computer Science, Vol. 8731, pp. 371-391, 2014.