Def Sci J, Vol 40, No 1, January 1990, pp 39-54.

Some New Mathematical Tools in Cryptology

I.J. Kumar and Meena Kumari

Scientific Analysis Group, Metcalfe House, Delhi-110 054

ABSTRACT

In this paper some new mathematical techniques used in the design and analysis of cipher systems have been reviewed. Firstly, some modern cryptosystems like stream ciphers, permutation-based systems and public key encryption systems are described and the mathematical tools used in their design have been outlined. Special emphasis has been laid on the problems related to application of computational complexity to cryptosystems. Recent work on the design of the systems based on a combined encryption and coding for error correction has also been reviewed.

Some recent system-oriented techniques of cryptanalysis have been discussed. It has been brought out that with the increase in the complexity of the cryptosystems it is necessary to apply some statistical and classification techniques for the purpose of identifying a cryptosystem as also for classification of the total key set into smaller classes. Finally, some very recent work on the application of artificial intelligence techniques in cryptography and cryptanalysis has been mentioned.

1. INTRODUCTION

The desire of man to keep his communications with some of his fellow beings secret from others is almost as old as the communication itself. From times immemorial, various methods including invisible writing have been practised for this purpose[1-3]. At the same time, it has also been a strong desire of man to read somebody elses' secret writing. While the former process of keeping the communications secure has been named as 'cryptography', the art or science of reading the secret writings without the total knowledge of the method of secret writing has been termed as 'cryptanalysis.' Put together, cryptography and cryptanalysis have been given the name 'cryptology'.

Received 29 December 1989

This discipline received great importance due to its military applications. It has been recognised by various authors that cryptography and cryptanalysis are essentially very highly mathematical disciplines. While in cryptography mathematics has been used to ensure that a certain minimum effort (beyond the power of the adversary) would have to be put in for reading the secret writing, from the angle of the cryptanalyst, mathematics has been used to develop general methods of reading the secret writings of a particular class (prepared by a given set of transformations). More and more sophisticated mathematical tools have been used in this continuous fight for supremacy between the cryptographer and the cryptanalyst. Starting from the simple substitution, mono- and poly-alphabetic substitutions and various kinds of transpositions, the cryptosystems based on composite and randomised permutations, block enciphering and bit by bit encryption by non-linear binary sequences came into being. Some of these methods of cryptography were based on ensuring mathematically that a certain minimum number of trials would have to be carried out by any adversary to get the clear message. On the other hand, with the development of fast computer systems, more and more sophisticated algorithmic and analytical approaches were developed to meet the challenge of cryptography. While the choice of means of securing one's messages depends upon the available technology, the ease of the management of key distribution and various other factors, and the methods used by the cryptanalyst have certain short cuts available based on some inherent mathematical or statistical structures in the crypts as also the information about the system of encryption. This information available to the analyst is mainly the (a) knowledge available about the system, (b) compromise of plain and cipher text, or (c) a large amount of cipher text only to provide information about the system.

In what follows, some of the very recent methods of encryption pointing out the requirements for these systems to be secure and the mathematical tools to ensure the security of these methods are discussed. On the cryptanalysis side, some methods applied to certain cryptosystems are discussed. Some general methods based on pattern recognition and Key classification have also been touched upon.

In section 2.1, stream ciphers based on linear feedback and non-linear feed-forward shift registers bringing out mathematically the requirements of the systems based on this philosophy to have good crack-resistance have been dealt. The generation of the binary sequences using feedback based on Boolean functions is discussed in section 2.2. Section 2.3 formulates some properties of permutations and how the cryptosystems based on this philosophy are designed. Some public key encryption systems based on factorisation of large composite numbers and the knapsack functions are discussed in section 2.4. Elementary complexity theory has been discussed with a view to point out its use in evaluation of cryptosystems in section 2.5. The recently formulated problem of combined encryption and encoding for error correction has been discussed in relation to public-key systems and the stream ciphers in section 2.6. Section 3.1 deals with some specific cryptanalytic methods of attack on some of the systems discussed above. Some applications of the methods based on pattern recognition and artificial intelligence (AI) in cryptanalysis are discussed in section 3.2.

2. CRYPTOGRAPHY

2.1 Stream Ciphers

In Modulo-2 Addition Systems for secure communication, the message sequence is added bit by bit modulo-2 to a randomly generated sequence. Thus for a message sequence $\{a(n)\}$ and the randomly generated sequence $\{b(n)\}$ we get the encrypted sequence as $\{a(n)\}+\{b(n)\}=\{c(n)\}$. This is the principle of stream ciphers. The design of such systems therefore, basically depends on ensuring certain characteristics in the generated sequence. These characteristics are listed below :

Large period : For every key, the sequence should have a very large period so that no part of enciphering sequence $\{b(n)\}$ is used repeatedly within a reasonable time.

Complexity : Given a segment of the sequence, it should not be possible to predict the following segment. The complexity of the sequence is measured in terms of length of the sequence required to predict correctly the rest of the sequence.

Good statistical properties : To ensure proper distribution of ones and zeros in the sequence and also good autocorrelation properties.

Variability : A large amount of variability to ensure that a brute force attack becomes infeasible.

The above properties can be ensured in a binary enciphering sequence generated by a well-designed shift register. The properties of the shift register sequences lend themselves to analysis through a number of mathematical methods like the linear algebra and the theory of difference equations. A binary sequence $\{a(n)\}$ generated by r-stage shift register satisfies the linear recurrence relation

$$a(n)=c_1a(n-1)+c_2a(n-2)+\ldots.+c_ra(n-r)$$

where c_i for $i=1,2,\ldots.,r$ is 1 or 0 according as the ith register is or is not involved in the feedback circuit. There are three methods of studying the linear recurring sequence generated by shift registers, viz., (a) algebraic method, (b) matrix method, and (c) classical method[4]. If $f(x)$ is the characteristic or the feedback, polynomial of the generated binary sequence $\{a(n)\}$, the maximal period of $\{a(n)\}$ depends on the irreducibility of $f(x)$ over the Galois field, GF(2). If $f(x)$ is a primitive polynomial over GF(2), then it generates the maximal period. Number of primitive polynomials of degree r is $\phi(2^r-1)/r$, where ϕ is the Euler ϕ function. This gives the variability of sequences generated through a linear r-stage feedback shift register. By applying m linear transformations successively to generate a sequence[5], we can increase the period of the sequence m times to $m(2^r-1)$ and also there is increase in the variability of the system to $\underline{2^{r-1}}/\underline{2^{r-1}-m+1}$.

The classical method is based on the result of the analytical theory of difference equations[6], according to which the general solution of linear homogenous difference equation with constant coefficients can be represented explicitly in terms of the roots of the characteristic polynomial. Selmer[4] represented the sequence generated by a feedback shift register as a linear recurrence in terms of the roots of the characteristic polynomial. Based on this representation, Key[7] analysed the increase in the complexity of non-linear feed-forward sequence in terms of the increase in number of roots of the generator polynomial. These roots of the generator polynomial are formed by taking the products of roots of the characteristic polynomial. This process of multiplication of the roots of the feedback polynomial, taken a fixed number at a time can easily be understood in the general setting of the theory of matrices[8]. The distinct products of the roots of the feedback polynomial are actually the roots of the minimum polynomial of the compound matrix formed from the companion matrix of the feedback polynomial. This property holds good for all the cases when the feedback polynomial is an irreducible polynomial or a power of an irreducible polynomial. Using this property Meena[9] has proposed a unified method of analysing the complexity of binary feed-forward sequence generated by any kind of generator and any level of feed-forward logic. The generators of the non-linear feed-forward sequences are the products of the various factors of the minimum polynomial of the compound matrix and the feedback polynomial. This approach enables us to predict the maximum as well as all other possible complexities of non-linear feed-forward sequences for any level. If the feedback polynomial is primitive, then one can attain the maximum complexity (equal to the period) of the sequence as well as sufficient variability by applying non-linear feed-forward logic in layers. One can also ensure good statistical properties in such sequences by using a Langford arrangements.

A Langford arrangement is an arrangement of numbers 112233....*gg* in a sequence (without gaps) in such a way that for $h=1,2,...,g$ the two *h*s are separated by exactly *h* places[10].

It may be seen from Table 1, how, by applying a number of logics successively and using non-linear feed-forward logic based on Langford arrangements, highly complex binary enciphering sequences with good statistical properties and large variability can be produced.

Table 1. Comparison of characteristics of sequences

	Maximum period	Complexity	Statistical properties	Number of sequences of maximum period (variability)
LFSR *r*-stage	2^r-1	r	Ideal	$(2^r-1)\,\phi\,(2^r-1)/r \approx 10^4$ for $r=10$
NLFFS's with Langford arrangement in *s* layers	2^r-1	2^r-1 Maximum complexity	Good	$(2^r-1)\;\phi\;\dfrac{(2^r-1)}{r}\left(\sum_{n=1}^{2} {}^rC_n\right)\dfrac{s(s+1)}{2}$ {sum of Langford arrangements for each layer $\approx 10^{10}$ for $s=9$
Multilogic generator (*m* logic on *r*-stage generator)	$m(2^r-1)$	mr	Good	$\dfrac{2^{r-1}}{2^{r-1}-m+1} \approx 10^3$ for $m=5$

2.2 De Bruijn Sequences

Removal of the restriction that the feedback logic be linear (involving only modulo-2 addition of certain bits) increases the number of maximal length shift register sequence of degree n from $\phi(2^n–1)/n$ to exactly $2^{2^{n-1}-n}$ (this formula was discovered by N.G. de Bruijn). This astronomical increase in the number of good sequences justifies, in itself, the quest for a non-linear shift register where apart from modulo-2 addition, multiplication and complementation of bits is also permitted.

The non-linear shift register sequence of maximal length 2^n is called de Bruijn sequence. In order to ensure that the sequence should achieve the length 2^n without cycles, its logic is to be of the form[11]

$$a_k = f(a_{k-n}, a_{k-n-1}, \ldots, a_{k-1}) + a_{k-n}.$$

De Bruijn sequences satisfy the first two randomness characteristics but its autocorrelation is a three-valued function.

$$C(0) = 2^n$$

$$C(k) = 0 \quad 1<k<n-1$$

$$C(n) \neq 0$$

The value of $C(n)$ is small in most cases. As against sequences generated by linear shift register, no analytical estimates are available for shorter cycle lengths and it is therefore important to stick to maximal length sequences. In fact, the distribution of cycle lengths for fixed initial vector and variable logic is a flat distribution for all lengths between 1 and 2^n. Golomb's suggestion about the generation of all de Bruijn sequences based on preference[9] function technique had the drawback that a large amount of storage was needed in any mechanization of generation of sequences from preference tables.

Games[12] constructed a class of de Bruijn sequences of degree $n+1$ from two (perhaps different) de Bruijn sequences of degree n, thus generalising the earlier work of Leach[13] and Lampel[14]. Etzion and Lampel[15] have given two algorithms for generating two classes of de Bruijn sequences. First algorithm generates $2^{k \cdot g(n,k)}$ sequences of period 2^n using $3n+k.g\ (n,k)$ bits of storage where k is a free parameter in the range $1 \leq k \leq 2^{n-4)/2}$ and $g\ (n,k)$ is of order $n-2 \log k$. The second algorithm generates about $2^{n^2/4}$ sequences of period 2^n using about $n^2/2$ bits of storage space. Games and Chan[16] have discussed the complexity of de Bruijn sequences. Etzion and Lampel[17,18] constructed de Bruijn sequences of given minimal complexity for its use in stream ciphers.

2.3 Permutation-Based Systems

Permutations of n objects number $\underline{|n}$ which is an exponential function of n. Such permutations offer therefore a very good basis for design of cryptosystems with large variability. The possibility of implementing randomised permutations through an electromechanical rotor system led to design of machines like Enigma and Typex during World War II. It is necessary to discuss some very basic results of theory of permutation group which have a direct bearing on the design and analysis of such systems.

The set of all permutations of n objects forms a group P_n of order $\underline{|n}$, in which the product of two permutations A and B is obtained by first carrying out A and then B. The group P_n is called[19] the symmetric group of degree n. A permutation C which shifts m objects cyclically ($m < n$) is called a circular permutation or a cycle of degree m. It can be shown that every permutation can be uniquely resolved into cycles which operate on mutually exclusive sets of objects. This resolution is unique save for the order in which the cycles occur and for the alternative ways in which each cycle may be expressed. If n is the degree of the permutation and $\mu_1, \mu_2, \ldots, \mu_r$ the degree of the cycles then,

$$\mu_1+\mu_2+ \quad +\mu_r=n$$

2.3.1 Class of Permutations

Thus every permutation of degree n is associated with a partition of n into positive integers namely the degrees of the cycles into which it is decomposed. Two permutations which correspond to the same partition are said to belong to the same class of P_n. Two permutations A and B of degree n are said to be similar (or conjugate) with respect to P_n if there exists a permutation S in P_n such that $B=S^{-1}AS$. It can be shown that two permutations are conjugate with respect to P_n if and only if they belong to the same class.

If G is a group with elements $g_0, g_1, \ldots$ and M a group of matrices $m_0, m_1, \ldots$ such that for each g_i there corresponds an m_j and also the product of two g_is corresponds to the product of the corresponding m_js then we say that M defines a representation of G. Matrices which are transforms of one another are called equivalent matrices. It is known that the equivalent matrices A and $B^{-1}AB$ have the same characteristic equation. Hence it follows that to a class of permutations, as defined above, there corresponds a class of matrices which have the same characteristic equation. The spur of a matrix which is the sum of the diagonal elements is called the character of the representation. All the members in the class have the same character and the group characters satisfy orthogonality relation[20]. The class structure of permutation groups can thus be studied more easily through representation theory using matrix algebra. The class structure of the permutations forming a permutation group are not only important for design and analysis of the crypto machines based on rotors but also basic to cryptography in general and many other physical applications. A masterly exposition of such structures has been given by Bhagavantham and Venkatarayudu[21].

A rotor (wired code wheel) is the basic building block of a permutation based system. It is an insulating disk on which electrical contacts, one for each letter of the alphabet, are placed uniformly around the periphery on each side of the disk. An internal conducting path through the insulating material connects contacts in pairs, one point on each side of the disk. An electric current enters on the left hand side of the rotor cross-section and emerges at one of the contacts on the right hand side of the rotcr. Thus the rotor implements a permutation electrically. This is the permutation of alphabet set from one side of the disk to the alphabet set on the other side. Rotating the rotor counter clockwise k places yields a second substitution defined by $\pi_k=C^{-k}\pi_0C^k$ where C^k and C^{-k} represent the shifts through k points in counter clockwise and clockwise directions and π_0 is the initial permutation.

Rotor systems are built using a number n of rotor permutations $\pi_1, \pi_2, \ldots, \pi_n$ and by rotating the individual rotors after encipherment of each plain-text letter, effectively changing the rotation displacements by k_1, k_2,...., k_n shifts. The cryptographic transformation is obtained by the composition of the n rotors

$$C^{-k_i}\pi_j C^{k_i} \quad \begin{array}{l} 1<j<n \\ 0<k_i<m \end{array}$$

where m is the cardinality of the alphabet set[22].

In order to achieve randomness in the shifts k_1, k_2,...., k_n a mechanism called a set of notch points is used. Notch points are identified with the letters against which they appear on the metallic frame containing the rotor. When the latter is in motion, a notch point induces a shift to the adjacent rotor also. At all other positions of the motion, the adjacent rotor remains stationary. A very complicated set of polyalphabetic substitution is thus generated through rotor-based systems. The famous Enigma and Typex machines belong to this category. These machines, although of World War II wintage, are still considered secure.

Permutations can also be implemented electronically. A 64-bit block cipher based on modulo-2 additions and permutation of 32-bit blocks called DES (data encryption standards) has been discussed in literature at length. The system was cleared by the National Bureau of Standards (NBS), USA for data security[23].

2.4 Public Key Encryption System

Recently a new class of cryptosystems have been discussed in literature. Such systems are fairly secure and it is not necessary to securely distribute the key used for encryption operation. Such systems have been named as public key cryptosystems. Mathematically speaking, such systems are based on one way functions. A function $F(X)$ is a one way function if (a) it is easy to compute $F(X)$ given X in the domain of F, and (b) it is hard to find X if any Y is given such that $Y=F(X)$. Two sub-classes of one way function, namely, the knapsack problem and the factorisation of a large number have specifically been used in the design of Merkle-Hellman Trapdoor Knapsack System and the RSA Public Key System respectively. The RSA systems makes use of the following result.

If $R=pq$, where p and q are distinct primes and $\phi(R)=(p-1)(q-1)$ then $x^{\phi(R)}=1$ (mod R) for x which is not divisible by either p or q. The RSA cryptosystem is based on the selection of two large (about 100 digits) prime numbers p and q and calculation of $R=pq$. We then select the random value e (less than R), such that the greatest common divisor of e and $\phi(R)$ is one and solve congruence $de=1\{\text{mod } \phi(R)\}$ such that $0 < d < R$. This is a simple procedure requiring 0(log R) operations. For this scheme the public encryption key is $k_1=(e,R)$ and the secret decryption key is $k_2=d$. In order to send a message to B on this system, the sender A sends on the channel $C=E_{k1}$ $(M)=M^e$ (mod R) where $0 < C < R$. B calculates $D(C)=C^d=M^{ed}=M$ (mod R) and receives M uniquely as $M<R$. There is a very simple and fast method of doing this which takes 0(log E) operations to complete. The security of this scheme depends on the difficulty of factoring R. The problem for the cryptanalyst is therefore, to factor R and to calculate d in order to obtain the original text. There are large number of factoring methods currently known[24-26] but the most powerful of these methods requires about $e^{\sqrt{\log N \log \log N}}$operations to factor N. A very fast computer (capable

of carrying out an operation in 10^{-9} seconds) might require about 4×10^6 years to factor 200 digits numbers.

The other scheme of public key encryption is the Merkle-Hellman Trapdoor Knapsack System[23]. The knapsack problem is: Given a set of k-rods of various length a_1, a_2,....., a_k find a sub-set of these rods that exactly fills the knapsack length $S \leq a_1+a_2+a_3+\ldots+a_k$. If S is the length of the knapsack, then $S=a_1x_1+a_2x_2+\ldots+a_kx_k=\vec{A}.\vec{X}$, where $x_i=0$ or 1. Merkle and Hellman have used the idea that it is easy to compute S from vectors $\vec{A}$ and $\vec{X}$ but it is computationally infeasible to find $\vec{X}$ given S and $\vec{A}$, to design a cryptosystem, which is a public key system described below.

Choose a vector $\vec{a}\,' = (a_1', a_2',\ldots.)$ such that each element is larger than the sum of the preceeding elements, for example, $\vec{a}\,' = (2, 6, 14, 31, 61)$. Choose any integer m such that $>\sum_{i=1}^{200} a_i'$. Choose w such that $gcd(w, m)=1$. Compute w^{-1} such that $ww^{-1} = 1$ (mod m). Compute $\vec{a} = \vec{a}\,'.w$ (mod m). Then each user makes vector $\vec{a}$ public and keeps $\vec{a}\,'$, w and m secret. In order to encrypt, we convert the message into binary form, say by using TP code or ASCII code and divide it into blocks of 200 bits. Let a block be denoted by $\vec{X}$. Compute $S=\vec{a}.\vec{x}$. Then S represent the crypt of $\vec{x}$. Each block can be enciphered in the same way. To decrypt, the designer knows, $\vec{a}\,'$, w,m,w^{-1}. From S he computes $S' = S.w^{-1}$ (mod m). Knowing S' and $\vec{a}\,'$, to compute vector $\vec{x}$ is an easy knapsack problem. For example, if $\vec{a} = (14\ 42\ 98\ 90\ 46)$ and $\vec{x} = (1\ 1\ 0\ 1\ 0)$, then encryption $S=14+42+90=146$. Decryption $S' = 146 \times 109$ (mod 127) = 39, here $w=7$, $w^{-1}=109$, $m=127$, $\vec{a}\,'=(2, 6, 14, 31, 61)$. Since $39<61 \Rightarrow x_5=0$, $39 > 31 \Rightarrow x_4=1, 39-31=8<14 \Rightarrow x_3=0$, $8>6 \Rightarrow x_2=1$, $8-6=2 \Rightarrow x_1=1$. Therefore decryption can be easily obtained. It requires 2^{200} trials to compute $\vec{x}$ if $\vec{a}\,'$, w and m are not known. The above scheme has not been used in any military system because of the following two reasons.

(i) In spite of the progress in VLSI technology, it is still difficult to implement these systems with desirable parameters so as to achieve appropriate security.

(ii) No definite proof of the computational infeasibility has been achieved.

May be in future someone might develop new methods that can be used to factor numbers of a certain type. It should not be forgotten that till today no such thing as proveably secure public key system exists. This aspect has been discussed in the next section.

2.5 Complexity of Cryptosystems

Looking back at various cryptosystems just discussed, namely the stream ciphers, permutation-based systems and public key systems, it comes out that the basic concern of the cryptographer is to ensure some minimum effort to be necessary to get the message out of the crypt. In order to achieve the above aim each system is designed to have two components, namely, an algorithm say E, and key k. The crypt is formed by applying E to a message M with key k. The crypt C is therefore, given $C=E_k(M)$. M can be achieved from C by authorised person knowing the key k by applying the inverse tranform and getting $M=E_k^{-1}(C)$. For the cryptanalyst the effort to get M consists of two parts, i.e., getting the inverse algorithm E^{-1} and the key k. The minimum effort to get the message M is ensured by making the inverse algorithm hard and making the set k of key very large. Even the knowledge of the algorithm makes it necessary for the cryptanalyst to try a large number of keys. While in symmetric systems the encryption and decryption keys are the same and are secured, in public key systems a part of the key is made public and other part is secured. In systems discussed above, one can achieve the desired minimum effort by adjusting various parameters. However, it is necessary to study the application of computational complexity theory to put the classification of cryptosystems on a firm basis. We say that a cryptosystem is *unconditionally secure*[27], if the cryptanalyst cannot determine how to get M regardless of how much cipher text and computer power is made available to him. The only system of this type is 'One Time Pad'. Unfortunately such a system requires a key length equal to that of the message and therefore cannot be brought in general use. One has therefore to content with systems which are *computationally secure*. In such systems, the cryptanalyst cannot solve a message in useful time even when provided with very large computational power. The above notions can be put on a firm basis of computational complexity theory.

The problem of determining the complexity of an algorithm is related to two important aspects, (a) the most efficient methods of obtaining the solution of a problem, and (b) the number of operations needed to perform this task.

It is necessary to define certain terms before going deeper into this question. We say that a function $f(n)$ is $0(g(n))$ if there exists a constant C such that $f(n) < C\,|g(n)|$ for $n \geq 0$. The polynomial time algorithm is defined to be an algorithm which solves any instance of a particular problem in time $0(p(n))$ for some polynomial function p of the input length n. Any algorithm, where time requirements are not so bounded, is called an exponential time algorithm. Denoting the class of all problems which can be solved by polynomial time algorithm by P, we find from Table 2 that for small values of n, a given polynomial function can accede a given exponential function. However, as n increases, the exponential function will greatly accede the polynomial function. The problems which are not in P are termed as hard or intractable[28,29]. Let NP (non-deterministic polynomial) denote the class which consists of all problems such that any guess solution of any instance of the problem can be checked for validity in a period of time which is $0(p(n))$. For example,the factoring problem is NP because any guess for any factor of n can be checked for trial division and division is a problem in P. This also is the case with the solution of a particular instance of the knapsack problem to be checked by addition, which is a problem in P. Thus the knapsack problem is also in NP which can be easily seen in P, as a sub-class of NP. An important unsolved problem in complexity theory is whether P=NP. If this is not the case, it proves existence of the problem for which really no efficient method of solution can ever be developed. A very remarkable result concerning this question has been the discovery of a special sub-class problem of NP called NP complete or NPC problems. If any problem on NPC can be shown to be in P also, then P=NP. The knapsack is one of the several problems which have shown to be in NPC. If any one could develop any polynomial algorithm for solving this problem then polynomial algorithm exists which will solve all problems in NP. Despite the fact that there are still a number of

Table 2. Polynomial and exponential functions for various n

$f(n)$	Type	n=1	n=2	n=5	n=10	n=20	n=50
n	Polynomial	1	2	5	10	20	50
n^2	Polynomial	1	4	25	100	4000	2500
n^3	Polynomial	1	8	125	1000	8000	125000
2^n	Exponential	2	4	32	1024	1048576	1.1258×10^{15}
$\underline{|n}$	Exponential	1	2	120	3628800	2.4329×10^{18}	3.0414×10^{64}

important unanswered problems in complexity theory, we have the means of identifying what are the hard problems. Naturally it would be desirable to base the encryption techniques on these problems.

This idea of using computationally intensive problems in the design of cryptosystems seems to be very attractive. However, Shamir[30] has shown that there are a number of difficulties associated with doing this.

(i) Complexity value deals with worst possible case of any problem which could be only one or few instances. A cryptosystem however should not be secure sometimes but always.

(ii) It is difficult to quantify the complexity of crypto problems because the exact amount of information available with the cryptanalyst varies from time to time.

(iii) It is not always possible to convert any particular difficult problem into a cryptosystem.

It has been shown by Even and Yacobi[31] that the problem of breaking a public key cryptosystem is not as hard NPC problem. Thus, at the moment, the complexity theory is inadequate to demonstrate the computational infeasibility of any cryptosystem. The only method currently available for the evaluation of a cryptosystem, even at design stage, is to ensure that even under the most favourable circumstances for the cryptanalyst, the problems of finding the message under the cryptogram is computationally very expensive.

2.6 Combined Encryption and Encoding

The transmission of encrypted blocks of data over noisy channels requires an additional step of error correction coding. This has led to the problem of joint encryption and coding for error correction. This problem can be formulated either as encoding problem followed by encryption or encryption followed by encoding.

Mc Eliece[32] used the first approach and designed a public key cryptosystem based on algebraic coding theory using t-error correcting Goppa codes. Recently Rao and Nam[33] introduced a new approach to the private key algebra coded cryptosystems using only small distance (d<6) codes. This scheme results in a very strong cryptosystems with high information rate and low overhead for encoding and decoding.

Considering the approach of encryption followed by encoding, Kak[34] described a method based on D-sequences. D-sequences are obtained in expansion of a fraction or a rational number and are decimal sequences to arbitrary basis. He used the scheme for secure and error-free transmission of the keys in Diffie-Hellman encryption system.

In the present context of broad band communication, it is essential to introduce a technique of error-free transmission for stream ciphers. Here the bit encryption is followed by blockwise encoding. After encryption, the message sequence is encoded for error correction using (m, k) block code by dividing the encrypted sequence into blocks of k bits and then adjoining to each k bits of this sequence $(m\text{-}k)$ parity bits, which are the linear combination of the k bits.

It is known that the capability of error correction depends upon introduction of redundancy into the sequence which goes against the property of unpredictability or complexity required for a good encrypting sequence. Thus the addition of redundant bits for error correction appear to affect the security adversely. However, a deeper analysis by Kumar and Meena[35] has shown that, breaking the encryption sequence (generated by a primitive polynomial of degree n) into blocks and then introducing the redundancy bits for error correction actually increases the complexity of the encryption sequence from $2n$ to $2mn$ provided some care is taken in selection of the block size. The above result is derived by making use of an earlier result of Berlekamp[36]. A number of problems in the field of combined encryption and coding have been discussed in a recent monograph[37].

3. CRYPTANALYSIS

3.1 Some Systems Dependent Approaches

In the context of availability of the plain and cipher text of sufficient length (this in stream ciphers means availability of the encryption sequence), an important method of attack is to evaluate a minimum Boolean function for which some of the bits are considered known and the rest are taken as no-care conditions. Let N bits be available from the systems. We may choose S such that $(N\text{-}S)$ bits are considered as known bits and S bits considered as no-care conditions. A Boolean function which satisfies the given N bits in this way may generate further bits leading to the continuation of the given meaningful message. Based on the algorithm of Quine[38] and Macluskey[39] presented in Hu[40], Bedi[41] has established a method of analysis for stream ciphers and has applied the same to a number of shift register based systems.

Although the cryptanalysis of the German Engima machine which is a World War II vintage rotor-based system, has been mentioned in a number of publications[42,43], the methods of attack used on such machines are still classified. Some general methods of attack on such systems deserve attention. Konheim[44], Andleman and Reeds[45], and De Laurentis[46] are important contributors in this area. The method of Andleman and Reeds is based on considering the cipher text C, as a sequence of random variables generated by a probability distribution $Pr(C, K)$ parameterised by the key k. The problem of cryptanalysis is thus reduced to a statistical point estimation problem, where the parameter is to be estimated in the key. Using an efficient iterative maximisation technique with convergence properties given earlier by Baum and

Eagon[47], the authors have illustrated the use of the technique to rotor-based systems as well as to substitution permutation networks.

The advent of public key encryption systems based on factorisation of composite numbers into primes and the knapsack problem have created a new interest in number theory. A number of attacks have been suggested on the RSA System[48,49] and Merkle-Hellman Systems[50,51].

3.2 Application of Pattern Recognition and AI Techniques

The present day electronic cipher systems are characterised by highly non-linear systems with a very large number of keys. In such systems even when the cryptanalyst has complete knowledge about the algorithm, a bruteforce method to try all keys is impossible even on the fastest computer. Any attack on such systems must therefore, aim classification of the keys in smaller sets and the capability to reach the correct sub-set for any given cryptogram. To achieve this mathematical and statistical techniques based on pattern recognition, cluster analysis and other classification techniques[52-57] can be applied. The problem of the cryptanalyst is further compounded if the intercepts come from a mixed source using more than one cryptosystem with no apparent clues to segregate them. Rao[58] and Khanna[59] have developed a number of techniques to segregate the traffic based on pattern learning.

To formulate the cryptosystem identification problem as a pattern recognition problem, a crypt is expressed as a set of d real numbers x_1, x_2, x_3,...., x_d. Such a set of measurements is called a pattern x and the individual components are features taken from the crypt such as single and digraph frequencies and jumps between two successive letters in the cryptogram. Any pattern can be represented as a point in a d-dimensional Ecludian space called the pattern space. A pattern classifier is a device which maps the points of the pattern into category numbers 1, 2, 3,...., R. The decision surfaces of any pattern classifier can be implicitly defined by scalar and single valued functions containing r members, $g_1(x)$, $g_2(x)$,...$g_R(x)$ called discriminant functions. These discriminant functions are chosen such that for all x in the ith category $g_i(x)>g_j(x)$, $i, j=1, 2,...R$ and $i\neq j$. The decision surface separating contigious regions i and j is given by $g_i(x)-g_j(x)=0$ when $R=2$, we obtain $g(x)=g_1(x)-g_2(x)$ and if $g(x)$ is positive, we place x in category 1 and if $g(x)$ is negative we place x in category 2. A large number of patterns (crypts) are chosen whose desired classification (system of encryption) is known. Discriminant functions are chosen which perform adequately on the training set and then these discriminant functions are used to predict the category of the unknown pattern.

Let M_i represent the prototype vector of the ith category, such as the mean feature vector obtained from the training set. A minimum distance classifier places a pattern x into that category which is associated with the nearest of the prototype points. The minimum distance classification can be effected by comparing the values of the expression for various x in the above expression $g_i(x)=M_i'x-1/2M_i'M_i$ for $i=1, 2,..., R$ and selecting the largest value. In this expression M_i' is the transpose of M_i.

Apart from the above, another important and useful method is the probabilistic method of discriminant functions. Let the parameters in each R categories be probability function $P(X/i)$, i=1, 2,..., R and $P(i)$ which denotes the prior probability of the ith category.

Let $\lambda(i,j)$ be the loss incurred when a machine places a pattern belonging to category i. For any specific X, we calculate the conditional average loss $L_i(X)$ and place X wherever the loss is minimum. Define the loss function $\lambda(i,j)=1-\delta_{ij}$ where $\delta_{ij}=1$ and $\delta_{ij}=0$ for $i \neq j$ then $L_i(X)=P(X)-P(X/i)$. $P(i)$ and minimising $L_i(x)$ means maximising $P(X/i)$ $P(i)$ and therefore $g_i(x)=P(X/i)$ $P(i)$.

In case X is normally distributed with mean vector M and covariance matrix Σ and taking $\Sigma=\Sigma_i=\Sigma_j$, i.e., assuming same covariance matrix for all groups.

$$P(X/i) = \frac{1}{2\pi^{d/2}|\Sigma|^{1/2}} \exp\left[\left(-\frac{1}{2}(X-M_i)'\,\Sigma\,(X-M_i)\right)\right]$$

After estimating the parameters M_i and $P(i)$ for each category, an unknown pattern is placed in that category, where the discriminant function yields the largest value.

As newer and more complex systems are given for analysis, more sophisticated methods of discriminant analysis had to be developed. One such approach is based on misclassified observations. In this approach, as a second stage, discriminant functions were developed based on misclassified patterns of one category and the original patterns of the other category and exact rules for classification were framed. This method is found to be very useful when there is a structure in the patterns and ordinary linear discriminant analysis has limitations in dealing with such data[59].

In another approach the data sets were partitioned into different sub-sets based on the norms of the vectors in all the categories. Discriminant functions were then developed for each range of the norm and an unknown pattern is classified by computing its norm to select the appropriate partition and then applying the associated discriminant function.

Very recently the techniques of artificial intelligence have been applied in cryptography and cryptanalysis. Carrol and Martin[60] have used expert systems to break simple substitution ciphers. These expert systems have used knowledge based on language characteristics. The authors have suggested the extension of such system for solution of polyalphabetic systems using regression analysis to separate out the segments of the text enciphered in each component alphabet. Kumar *et al.*[61] have suggested AI based systems for design of stream ciphers and for automatic location of the correct English text. The authors have also suggested Discovery Systems based on AI techniques to discover the statistical laws for classification of N keys into approximately $\sqrt{N}$ classes in any cryptosystem. The theoretical and practical difficulties connected with uncertainty in Induction Systems have been discussed by Kumar[62].

4. CONCLUSION

In this paper we have reviewed a number of techniques used recently for cryptography and cryptanalysis. While most of the techniques used in cryptography are based on discrete mathematical structures and number theoretic concepts, cryptanalysis of systems based on these techniques require additionally the application of statistical and classification techniques. The development of public key system has led to renewed interests in the classical problems of primality testing and factorisation. There are a number of open problems in the application of the results of complexity theory to the evaluation of cryptosystems. The design of systems using combined encryption and encoding is another promising area of research work. Application of AI techniques in cryptography is also bound to become a very important area.

ACKNOWLEDGEMENTS

The authors are grateful to their colleagues Sh. P.N. Sundaram, Dr. Laxmi Narain, Dr. S.S. Bedi, Sh. T.L. Rao and Dr. R.K. Khanna for many useful discussions and help in preparation of this script.

REFERENCES

1 Kahn, D., The Codebreakers : The Story of Secret Writing, (Macmillan, New York), 1967.

2. Smith, L.D., Cryptography, (Dover Publ. Inc.), 1943.

3. Denning, D.E.R., Cryptology and Data Security, (Addision-Wesley, London), 1982.

4. Selmer, E.S., Linear Recurrence Relations over Finite Fields, (Univ. of Bergen, Norway), 1966.

5. Reed, I.S. & Turn, R.J., *J. Assoc. Comp. Mach.*, **16** (1969), 461-473.

6. Milne-Thomson, The Calculus of Finite Differences, (Macmillan, London), 1960.

7. Key, E.L., *IEEE Trans. Inf. Theo.*, **22**(6), (1976), 732-736.

8. Wedderburn, J.H.M., Lectures on Matrices American Mathematical Society, (Colloquim Publications), Vol.XVIII, 1984.

9. Meena Kumari, *Discrete Math.*, **56**(382), (1985), 203-215.

10. Growth, E.J., *IEEE Trans. Inf. Theo.*, **17**(3), (1971), 288-296.

11. Golomb, S.W., Shift Register Sequences, (Aegean Park Press, California), 1982.

12. Games, R.A., *IEEE Trans. Inf. Theo.*, **29** (1983), 843-849.

13. Leach, E.B., *Proc. Amer. Math. Soc.*, **11** (1960), 566-574.

14. Lampel, A., *IEEE Trans. Comput.*, **19** (1970), 1024-1209.

15. Etazion, T. & Lampel, A., *IEEE Trans. Inf. Theo.*, **29** (1983), 480-484.

16. Games, R.A. & Chan, A.H., *IEEE Trans. Inf. Theo.*, **29** (1983), 144-146.

17. Etazion, T. & Lampel, A., *IEEE Trans. Inf. Theo.*, **30** (1984), 611.

18. Etazion, T. & Lampel, A., *IEEE Trans. Inf. Theo.*, **30** (1984), 705.

19. Herstein, I.N., Topics in Algebra, (Vikas Publishing House, New Delhi), 1975.

20. Aigner, M., Combinatorial Search, (John Wiley, New York), 1988.

21. Bhagavantham, S. & Venkatarayudu, T., Theory of Groups and its Application to Physical Problems, (Andhra Univ., Waltair), 1948.

22 Reed, J., *Cryptologia*, **1**(2), (1977), 186-194.

23 Baker, H. & Piper, F., Cipher Systems, (Northwood Book, London), 1982.

24 Dixon, J.D., *Mathematics of Computation*, **36** (1981), 255-260.

25 Lenstra, H.W. Jr., Report 86-89, Mathematics Institute Universitait Van Amsterdam, 1986.

26. Lehman, R.S., *Mathematics of Computation*, **28** (1974), 637-646.

27. Williams, H.C., *In* Secure Communication and Asymmetric Cryptosystems, G.J. Simmons, (Ed), (West View Press), 1982, pp. 11-39.

28 Garey, M.R. & Johnson, D.S., Computers and Interactability : A Guide to the Theory of NF Completness, (W.H. Freeman and Company, New York), 1979.

29. Galude, C., Theories of Computational Complexity, *Annal. of Discrete Math.*, **35** (North Holland, Amsterdam), 1988.

30. Shamir, A., On the Cryptocomplexity of Knapsack Systems, Tech. Rept. MIT/LCS/TM-129, (MIT Lab. Comput. Sci., Cambridge), 1979.

31 Even, S. & Yacobi, Y., On the Cryptocomplexity of a Public Key System, Preprint, 1979.

32. Mc. Eliece, R.J., A Public Key Cryptosystem Based on Algebraic Coding Theory, DSN Progress Report, (Jet Propulsion Laboratory, Pasadena), 1978, pp. 114-116.

33. Rao, T.R.N. & Nam, K.H., *IEEE Trans. Inf. Theo.*, **35**(4), (1989), 829-833.

34. Kak, S.C., *IEEE Trans. Comput.*, **C-34**(9), (1985).

35. Kumar, I.J. & Meena Kumari, Cryptographic Complexity of Encoded Shift Register Sequences, (to be published).

36. Berlekamp, E.R., Algebraic Coding Theory, (McGraw Hill, New York), 1968.

37. Beker, H.J. & Piper, F.C., (Eds), Cryptography and Coding, (Clarendon Press, Oxford), 1969.

38. Quine, W.A., *Amer. Math. Month.*, **62** (1955), 27-631.

39. Macluskey, E.J., Minimization theory, *In* A Survey of Switching Theory, E.J Maceluskey Jr. & T.C. Bartee, (Eds), (New York), 1962, pp. 81-83.

40. Hu, S.T., Mathematical Theory of Switching Circuit and Automata, (Univ. of California Press), 1968.

41 Bedi, S.S., Analysis of Stream Cipher Based on Non-linear Shift Register, Unpublished report, (SAG, DRDO, New Delhi), 1986.

42. Winterbotham, F.W., The Ultra Secret, (Wedenfelf and Nicholson), 1974.

43. Lewin, R., Ultra Goes to War : The Secret Story, (Hutchinson and Co.), 1978.

44. Konheim, A.G., Cryptography; Lecture Notes in Computer Science, 149, T. Beth, (Ed), (Springer-Verlag, Berlin), 1983, pp. 49-68.

45. Andelman, D. & Reeds, J., *IEEE Trans. on Inf. Theo.*, **28**(4), (1982), 578-584.

46. De Laurentis, J.M., Advances in Cryptology; Crypt 87, Lecture Notes in Computer Science; 293, C. Pomerance, (Ed), (Springer-Verlag, Berlin), 1987.

47. Baum, L.E. & Eagon, J.A., *Bull. Amer. Math. Soc.*, 73 (1967), 360-363.

48. Lidl, R. & Müller, W.B., Advances in Cryptology; Proceedings of Crypto 83, Chaum David, (Ed), (Plenum Press, New York), 1984, pp. 293-301.

49. Beth, T., (Ed), Cryptography; Lecture Notes in Computer Science 149, (Springer-Verlag, Berlin), 1982, pp. 325-375.

50. Beth, T., (Ed), Cryptography; Lecture Notes in Computer Science, 149, (Springer-Verlag, Berlin), 1982, pp. 307-322.

51. Chaum, D., Rivest, R.L. & Sherman, A., Advances in Cryptology; Proceedings of Crypto 82, (Plenum Press, New York), 1983, pp. 279-303.

52. Anderberg, M.R., Cluster Analysis for Applications, (Academic Press, New York), 1973.

53. Fu, K.S., Syntactic Pattern Recognition and Applications, (Prentice Hall, Englewood Cliffs), 1982.

54. Gorden, G.D., Classification, (Chapman and Hall, London), 1981.

55. Meisel, W.S., Computer Oriented Approaches to Pattern Recognition, (Academic Press, New York), 1972.

56. Oja, E., Subspace Methods of Pattern Recognition, (John Wiley, New York), 1983.

57. Rao, C.R., Linear Statistical Inference and its Applications, (Wiley-Eastern, New Delhi), 1973.

58. Rao, T.L., Identification of Cryptosystems from Cipher Text Only, Unpublished report, (SAG, DRDO, New Delhi), 1987.

59. Khanna, R.K., Two New Methods of Classification of Structured Patterns, (to be published).

60. Carrol, J.M. & Martin, S., *Cryptologia,* 10(4), (1986), 193-209.

61. Kumar, I.J., Meena Kumari & Saxena, P.K., Proceedings of Second National Seminar on Computer Applications in Defence, 8-10 December 1988, Bangalore, Restricted Section 3, (DRDO CC, Bangalore), 1988, pp. 10-25.

62. Kumar, I.J., *The Mathematics Student,* 57 (1989), 209-218.