Text Encryption Algorithms based on Pseudo Random Number Generator

Mina Mishra
Ph. D. Scholar
Electronics & Telecommunication, Nagpur University, Nagpur, Maharashtra, India

V.H. Mankar
Lecturer
Department of Electronics Engineering, Government Polytechnic, Nagpur, Maharashtra, India

ABSTRACT
This paper presents algorithms for encryption and decryption developed using pseudo random number generator (PRNG) and non-Linear functions. PRNG used in the work are matlab random number generator (RNG) and Linear congruential generator (LCG). The developed algorithms are named according to PRNG used in it. State of PRNG is considered as secret key of the cipher. The encryption schemes have been cryptanalyzed for four different methods to test its strength like key space analysis, plaintext and key sensitive test. Known plaintext attack is also performed by taking into consideration a small string of plaintext and the complete cipher text for small text. The analysis is performed on different keys selected randomly from key space for various texts and files.

Key sensitivity up to 50 % and plaintext sensitivity ranging from 3% to 50 % have been obtained in the developed ciphers. It is concluded that proposed encryption algorithms have strength against linear, differential and statistical attacks.

Keywords
Cryptography, Pseudo random number generator (PRNG), Random number generator, Linear Congruential Generator.

1. INTRODUCTION
Cryptography has remained important over the centuries, used mainly for military and diplomatic communications. With the advent of the internet and electronic commerce, cryptography has become vital for the functioning of the global economy, and is something that is used by millions of people on a daily basis. Various schemes of encryption using different techniques have been proposed in recent years. The Pseudo random number generators have been used for design of ciphers and found to be fundamental tool in many cryptographic applications like key generation, encryption, masking protocols and for internet gambling [1]- [4].

A pseudorandom number generator (PRNG), also known as a deterministic random bit generator (DRBG) is an algorithm for generating a sequence of numbers that approximates the properties of random numbers. The sequence is not truly random in that it is completely determined by a relatively small set of initial values, called the PRNG's state, which includes a truly random seed. Good statistical properties are a central requirement for the output of a PRNG, and common classes of suitable algorithms include linear congruential generator, lagged fibonacci generator, and linear feedback shift register [5], [6].

This paper aims to develop a number of algorithms for encryption/decryption using PRNG's and non-linear functions. The encryption techniques are crypt analyzed for linear and differential attacks to test their validity. Various methods of cryptanalysis used in this work are key space, plaintext and key sensitive test. Known plaintext attack is also performed by taking into consideration a small string of plaintext and the complete cipher text for small text. Simple pseudorandom number generator like matlab random number generator (RNG) and linear congruential generator (LCG) have been used here. The developed algorithms are named according to PRNG used in it.

The analysis is performed on different keys selected randomly from key space for various texts and files. Key sensitivity up to 50 % and plaintext sensitivity ranging from 3% to 50 % have been obtained in the developed ciphers. It is concluded that proposed encryption algorithms have strength against linear, differential and statistical attacks.

The rest of the paper is organized as follow. Section II, presents the brief introduction of PRNG's used in the developed algorithm. Next section presents and discusses about the algorithm. In section IV, analysis and results have been discussed. Section V gives conclusion of the work.

2. PSEUDORANDOM NUMBER GENERATOR
PRBG plays an important role in cryptography. They have been frequently used in the designing of the ciphering methods. Two of the simple PRBG used in designing of encryption algorithm have been discussed in brief here:

A. Random Number Generator (RNG)
Matlab Rand produces uniformly distributed pseudorandom numbers, scalar value drawn from a uniform distribution on the unit interval.

Rand ('twister', k), r = rand, produces a pseudorandom number corresponding to state k, which is the seed (also known as initial condition) of generator, acting as key for cryptosystem.

B. Linear Congruential Generator (LCG)
A Linear Congruential Generator (LCG) represents one of the oldest and best-known pseudorandom number generator algorithms. The generator is defined by the recurrence relation:

$x(k+1) = (ax(k)+b) \bmod c$

Where, x(k) is the sequence of pseudorandom values, and c,0 < c - the "modulus", a, 0 < a < c - the "multiplier", b, 0 < = b < c - the "increment", x0, 0 < = x0 < c - the "seed" or" start value" are integer constants that specify the generator. The period of a general LCG is at most c, and for some choices of a much less than that. Provided that b is nonzero, the LCG will have a full period for all seed values if and only if:

- b and c are relatively prime,
- (a -1) is divisible by all prime factors of c,

- (a -1) is a multiple of 4 if c is a multiple of 4.

Parameter b is the best choice as encryption key for chaotic cryptosystem.

3. ENCRYPTION AND DECRYPTION ALGORITHM

This section discusses the algorithms of various encryption methods. Encryption process for each of the method has been explained using algorithm. Decryption process is just reverse of the encryption process. State of the PRNG's is used as a secret key in the method.

A. RNG Method
Encryption Algorithm:
Step-1. Read the plain text as p and key as k, which is the state of the random number generator.

Step-2. Convert each text into its ASCII values.

Step-3. Transform each character of text using the expressions given as:

$$y = p + 2 \sin (100)$$

$$c = y + 10\, r$$

$$k = k + 1.$$

Where, p is input text; c is output text; r = random number generated by the state, 'k' of Matlab random number generator;

Step-4. Plot output of the system.

Step-5. Convert integer values into its character values.

Step-6. Read c as output text as cipher text.

B. LCG Method
Encryption Algorithm:
Step-1. Read plaintext as p, key as b and length as n.

Step-2. Change the character values of text into its ASCII values.

Step-3. Each ASCII values are transformed into five corresponding values using the following transformations:

$$y = p + \sin (b);$$

$$c = y + r;$$

y is intermediate variable.

r is the random numbers generated corresponding to the key; b is state of LCG; c is any variable.

Step-4. Plot 'c' obtained from above step.

Step-5. The sequences of numbers in c are then converted into character values.

Step-6. Read the output text (cipher text).

Algorithm For Linear Congruential Generator:
Step-1. Read iteration as N, initial condition as x (1), values of parameters a, b and c where b is considered as key for the system.

Step-2. Calculate, for k=1: N

$$x(k+1) = \mathrm{mod}\ (a\, x(1) + b, c)$$

Step-3. Read output states as x.

A. Modified RNG Method:
Encryption Algorithm:
Step-1. Input plaintext as p, key as k.

Step-2. Convert p into ASCII form.

Step-3. Read length of plaintext as n.

Step-4. Read index of iterations as i and index of length of plaintext as m.

Step-5.

(a) For i = 1, do following steps:

i. Convert each number into its binary form, which forms matrix of dimension containing rows equal to the length of text and number of columns is eight.

ii. Above matrix dimension is changed into dimension where no. of rows is half the length of text and no. of columns is sixteen.

iii. For each column, elements of each row are circularly shifted by one in anticlockwise direction. Element of first row of each column is shifted to last row.

iv. From second row to last row, for each row elements of each column is shifted by one in right direction.

v. Again matrix dimension is changed into dimension of matrix as in (a) (i).

vi. Binary form of numbers is changed into decimal form.

vii. Read output sequence of numbers as p.

(b) For m = 1, do following steps:

i. Generate random number corresponding to key (state).

ii. Do Xor operation between random number and p and store the result in p.

iii. Increment key by one, go to (b)(i) and the loop continues till m = n is completed. The loop outputs new values of p having length n.

(c) i = i+1 and go to step (5) (a) (i). Repeat process till i = 16 is completed.

Step-6. For i = 1, do following steps.

i. Generate random number r corresponding to value of key (state).

ii. Calculate = p + mod (r, 128).

iii. Convert each number of p into its binary form, which forms matrix of dimension containing rows equal to the length of text and number of columns is eight.

iv. Above matrix dimension is changed into dimension where no. of rows is half the length of text and no. of columns is sixteen.

v. Matrix is partitioned into two equal halves, first and second.

vi. Mix both matrix and obtain new matrix of same dimension. Mixing is done in such a way that elements of second column of new matrix becomes elements of first column of second matrix, third columns elements becomes elements third column of first matrix, fourth columns elements becomes elements of second column of second matrix and so on.

vii. Dimension of new mixed matrix is changed into dimension i.e., equal to the dimension of matrix as instep (6) (iii).

viii. Binary forms of numbers in matrix are changed into decimal form.

ix. Read output as p.

x. Again key is incremented by one, k = k + 1.

xi. i = i + 1, go to step (6) (i). The process repeats until i = 16 is completed.

Step-5. Read output generated by completion of above steps as c and plot it.

Step-6. Convert integer values of c into character form.

Step-7. Read c as cipher text.

4. ANALYSIS AND RESULTS

The analysis part consists of testing the validity of methods against the most basic attacks like linear, statistical and differential attacks. Cryptanalysis is the necessary for testing the strength of the developed ciphers. The cryptanalytic procedures used in this work include:

A. Key Space Analysis: The size of the key space is the number of encryption/decryption key pairs that are available in the cipher system. In the proposed method, the key space (range of keys) is defined clearly. Once the key has been defined and key space has been properly characterized, the good key is chosen randomly from the large key ranges [7].

B. Plaintext sensitivity: This method corresponds to the percentage of change in bits of cipher text obtained after encryption of plaintext, which is derived by changing single bit from the original plaintext from the bits of cipher text obtained after encryption of original plaintext. With the change in single bit of plaintext, there, must be ideally 50% change in bits of cipher text to resist differential cryptanalysis (chosen-plaintext attack) and statistical analysis, corresponds to plaintext sensitivity test [8].

C. Key sensitivity: This method corresponds to the percentage of change in bits of cipher text obtained after encryption of plaintext using key, which is flipped by single bit from the original key, from bits of cipher text obtained after encryption of plaintext using original key, which requires ideally 50% change in cipher text bits to resist Linear and statistical attacks [9].

D. Known plaintext attack: It is assumed that the opponent knows everything about the algorithm; he/she has the corresponding cipher text of plaintext and some portion of plaintext. With this much information, the opponent tries to find out the secret key [10].

For each of the methods the analysis result is cited in tabular form as follows:

Table 1: Analysis Table for RNG method

Sl. No.	Plaintext	Key value	Cipher text	Plaintext sensitivity (in %).	Key sensitivity (in %).	Robustness against known plaintext attack for p = [p1 p2].
1.	What is your name?	0	%[lf}"qs(yvvs'siogE	1.5038	22.3684	R
2.	I am going to market.	19	N cr)ormuo&vw"meythy2	0.6494	25.3247	R
3.	My college name is s.s.c.e.t.	265	%Mz!eurneii)rbrf$lx(u/t3j0e0|4	0.9524	27.6190	R
4.	Hello!how are you?	4765	!Qgrmx#lry%ezl#ryF	1.3158	28.2895	R
5.	Sita is singing very well.	39846	%Uryg)mw&ukqkqri(zjr'jnu3	0.9259	28.7037	R
6.	Ram scored 98 marks in Maths.	874976	&Wdq"zlqvmd#?<"pfroy!or(Tfuo{3	0.4762	29.0476	R
7.	Jaycee publication.	1234567	"Kf{ffk!uydlmddvioq3	0.7143	27.8571	R
8.	Thank you,sir.	87649034	"Wjcwn%zxz5wjx7	0.9524	24.1667	R
9.	The match was very exciting.	945672345	!\nm$rgzhn)zfx!xmv|'jgl{lth7	0.8621	26.7241	R
10.	I will be leaving at 9p.m.	3760321854	)R'~msl!hi)mjbxjuj$e{)Bt6u1	1.0582	29.6296	R

[R- Robust; p [p1 p2…p n]- First 'n' characters of available plaintext string.]

Table 2: Analysis Table for LCG method

Sl. No.	Plaintext	Key value	Cipher text	Plaintext sensitivity (in %)	Key sensitivity (in %)	Robustness against known plaintext attack for p=[p1 p2].
1.	What is your name?	0	Xlgb]i}xsnbvqlgu‰„□z!50+&j~ytotˆƒ~ y!50+&zŽ‰„□p„□zuvŠ…€{s‡,}x!50+ &o*f*~ytbvqlgn,}xsfzupk@TOJE	1.25	24.8684	R
2.	I am going to market.	20	"1'6,KZP_U"1'6,crhwmo~ty"1'6,ixn}sqv {kzp□up□uzixn}s"1'6,v{qv{"1'6,o~tycr hwmty~m|r • wgvl{qv{0?5D:	1.0227	26.4773	R
3.	My college name is s.s.c.e.t.	301	!6-8%NcZeRz • ~!6-8%dyp{hp|tmyqmyqf{r}jh}t□lf{r}j!6-8%o{sbwnyfnzrf{r}j!6-8%j□v • ntx!6-8%tx/D;F3tx/D;F3dyp{h/D;F3f{r}j/D;F3u • y/D;F3	0.6667	27.7586	R
4.	Hello!how are you?	2465	JT^OYgq{lvnxs}nxs}q{v#-7(2jt~oyq{vy • ~",6'1cmwhrt~ygq{lv",6'1{ • q{vw • |AKUFP	1.3889	30.5556	R
5.	Sita is singing very well.	96785	TY^chjoty~uz□bglqv!&+05joty~ty~!&+05ty~joty~oty~hmrw|joty~oty~hmrw|!&+05w| • fkpuzsx}z□!&+05x}fkpuzmrw| • mrw| • /49>C	0.7692	25.9615	R
6.	Ram scored 98 marks in Maths.	123456	STi`Rbcxoano{m!"7. tu • sdezqcpq}ostrfg|seef{rd!"7. :;PG99:OF8!"7. no{mbcxoastrlmyktu • s!"7. jkwiop|n!"7. NOd[Mbcxoauvtij□vhtu • s/0E<.	0.5714	23.2381	R
7.	Jaycee publication.	2592398	K]`TRbtwkiz • • dvymkfx{omfx{om!36*(qzxv□}cuxljm□vtj|□sqdvymkbtwkiu~|j|□sqpywo • xv/AD86	1.0526	23.1579	R
8.	Thank you,sir.	40098645	UdZi_ixn}sbqgvlo~tyl{qv!0&5+z□p□uzv{-<2A7ty~jyo~tsx}/>4C9	2.1429	27.1429	R
9.	The match was very exciting.	120765956	UVkbTij□vhfg|se!"7. no{mbcxoauvtdezqcij□vh!"7. xywbcxoatu • s!"7. wx • vfg|sestrz{ • y!"7. fg|seyz • xdezqcjkwiuvtjkwiop|nhi~ug/0E<.	0.7143	38.0357	R
10.	I will be leaving at 9p.m.	4000000000000000	K_ZUP"61,'y • ~k□zupn}xsn}xs"61,'dxsnig{vql"61,'n}xsg{vqlcwrmhx}k□zupp□zui}xsn"61,'cwrmhv{"61,';OJE@r • |w0D?:5o~yt0D?:5>>	1.25	23.7500	R

Table 3: Analysis Table for Modified RNG method.

Sl. No.	Plaintext	Key value	Cipher text	Plaintext sensitivity (in %)	Key sensitivity (in %)	Robustness against known plaintext attack for p = [p1 p2 …p(length of p-1)]
1.	What is your name?	0	úY¸3+9ùìNÚâ9= Ó[	46.0526	49.3750	R
2.	I am going to market.	19	y&ý ½ÙV;¯U1ó°	48.2955	46.0227	R
3.	My college name is s.s.c.e.t.	265	"A°î$IMÀ`óö • *ZõäPy¤»õÏÀ¥	48.7500	50.4167	R
4.	Hello!how are you?	4765	Ã*[Í@w1ËÅVt • É=Ü!	43.4211	46.0526	R
5.	Sita is singing very well.	39846	*£¨gúµÇLÄH´C$×!¹-Õ¿LS	50	48.6111	R
6.	Ram scored 98 marks in Maths.	874976	ý¬u • U¹WG°Ú0ù2ÏmH*NÌ¢D®Ê^ÂÆô	34.1667	49.1667	R
7.	Jaycee publication.	1234567	îD=õñ*XáW¢¡÷□ª?	46.2500	52.5000	R
8.	Thank you,sir.	87649034	ä_¼5Qÿ(1ÖI2%E"	30.8333	52.5000	R
9.	The match was very exciting.	945672345	àW-7ÛË_lBM«¥wiVâßfëémâplT8	45.2586	53.4483	R
10.	I will be leaving at 9p.m.	3760321854	ÙðÖÄ*w¯UT4#÷rzðc ô©!}h	49.5370	51.3889	R

5. CONCLUSION

This paper proposes an encryption algorithm based on pseudo random number generator (PRNG). The performance of developed encryption scheme is evaluated by performing key space analysis, plaintext and key sensitive test and known plaintext attack on them. Known plaintext attack is performed by taking into consideration a small string of plaintext and the complete cipher text for small text.

The analysis is performed on different keys selected randomly from key space for various texts and files. Developed algorithm showed key sensitivity up to 50 %. Plaintext sensitivity in modified algorithm has been increased from 3% to 50 %.Thus, proposed encryption algorithm have shown strength against linear, differential and statistical attacks. A comparative result analysis is briefed in table 4. Modified RNG possess good plaintext and key sensitivity property.

Table 4: Comparison between the three methods

Name of cipher	Range of plaintext sensitivity	Range of key sensitivity	Robustness against known plaintext tattack
RNG	0.5 to 1.5 %	22 to 30 %	Yes
LCG	0.5 to 3 %	23 to 39 %	Yes
MODIFIED RNG	34 to 50 %	43 to 54%	Yes

6. ACKNOWLEDGMENTS
The authors would like to thank the anonymous reviewers for their valuable suggestions and the proposed references.

7. AUTHOR'S PROFILE
Mina Mishra, is pursuing Ph.D. (Engg) from Nagpur University, Maharashtra, India. She received M.E. degree specialization in communication in the year 2010. Her research area covers chaotic systems, chaotic cryptology, network security and secure communication.

Vijay H. Mankar received M. Tech. degree in Electronics Engineering from VNIT, Nagpur University, India in 1995 and Ph.D. (Engg) from Jadavpur University, Kolkata, India in 2009 respectively. He has more than 16 years of teaching experience and presently working as a Lecturer (Selection Grade) in Government Polytechnic, Nagpur (MS), India. He has published more than 45 research papers in international conference and journals. His field of interest includes digital image processing, data hiding and watermarking.

8. REFERENCES
[1] L. P. de Oliveira and M. Sobottka, 2008, "Cryptography with chaotic mixing" Chaos, Solitons and Fractals, vol. 35, pp. 466–471.

[2] Ruming Yin, Jian Yuan, Qiuhua Yang, Xiuming Shan, Xiqin Wang, 2009," Linear cryptanalysis for a chaos-based stream cipher," World Academy of Science, Engineering and Technology ,60, 799-804.

[3] Xianfeng Guo, Jiashu Zhang, Xianfeng Guo, 2010," An Efficient Cryptanalysis of a Chaotic Cryptosystem and Its Improvement", IEEE conference on Information Theory and Information Security, China, 578-581.

[4] Jiantao zho, au, o.c, 2010, "cryptanalysis of chaotic convolutional coder ", proceedings of ieee symposium circuits and systems (iscas) , 145-148.

[5] S. Li, X. Zheng, 2010, "Cryptanalysis of a chaotic image encryption method", Proceedings of the IEEE International. Symposium on circuits and systems, Scottsdale, AZ, USA.

[6] P. Xu; J. Zhao; D. Wang, 2011, "A selective image encryption algorithm based on hyper-chaos", IEEE 3rd International Conference on Communication Software and Networks (ICCSN), 2011, 376-379.

[7] Mishra Mina, Mankar V.H., 2012, "Design and Analysis of Cipher Based on Henon and Burger Maps" Proceedings of IEEE 4th International Conference on Electronics Computer Technology (ICECT- 2012), 978-1-4673-1850, 466-471.

[8] Mishra Mina, Mankar V.H., 2012, "A Chaotic encryption algorithm: Robustness against Brute-force attack" Advanced Intelligent and soft computing (AISC), Springer, 167, 169-179.

[9] Mishra Mina, Mankar V. H.,2012, "Hybrid Message-Embedded cipher using logistic map" published in International Journal of Security, Privacy and Trust Management (IJSPTM), 1(3/4), 81-91.

[10] Mishra Mina, Mankar V.H., 2012, "Message Embedded cipher using 2-D Chaotic map" published in International Journal of Chaos, Control, Modelling and Simulation (IJCCMS), 1(1), 13-23, July 2012.