CRYPTOGRAPHIC SECURE PSEUDO-RANDOM GENERATION: THE CHAOTIC LINEAR CONGRUENTIAL GENERATOR (CLCG)

Behrouz Fathi Vajargah[1], Rahim Asghari[2]
[1]Department of statistic, Faculty of Mathematic science, university of Guilan, Rasht, Iran.
Email: fathi@guilan.ac.ir
[2]Department of Applied Mathematics, Faculty of Mathematic science, university of Guilan, Rasht, Iran.
Email: : meuisam.mathhome@gmail.com

ABSTRACT: *In the present paper, the properties of making a deterministic algorithm for generating a pseudo random number sequence for application in cryptography, is discussed. The Chaotic Linear Congruential Generator is proposed as a pseudo random number generator. It is shown that what chaotic features of the Discrete Logistic Map are useful for generating pseudo random numbers in cryptographic point of view. To evaluate the randomness as well as independence of the bit sequences, generated by the PRNGs, two well-known statistic tests were performed and the results approved that the generated numbers are statistically proper in cryptographic applications. A comparison is performed to illustrate the efficiency of the presented generator.*

Keywords: Chaotic function, Pseudorandom number sequence, Discrete Logestic Map, Linear Congruential Generator, Correlation Test, Chi-square Goodness-of-fit Test.

1. INTRODUCTION

Complexity of communications in human society is increased by developments in electronic communication technologies. Such complexity needs to provide authentication in prescribed communications. Cryptography is an attempt for this vital requirement. Creating efficient crypto algorithms are recently becoming the subject of researches in academic societies, which depicts human society worries about authentication. The security in most crypto systems is highly depends on generating unpredictable numbers, e.g. hidden key in DES, prime numbers p,q in RSA, and key stream in stream ciphers to guaranty that generated numbers are not guessable. As generating true random numbers (e.g., Johnson Noise) are very difficult in practice and regeneration of them is not possible, debugging and testing of programs become difficult, therefore pseudo random numbers (generated by mathematical algorithms) are applied in daily applications. Stream cipher
technique has special importance among other crypto algorithms and has maximum dependency to pseudo random number generators.

In designing stream cipher, a single pseudo random bit generator, plays the role of key stream generator for the stream cipher system which is indeed generator of key stream. From the cryptographically point of view, a key stream generator should have the following important parameters:

- The period of generating key should be sufficiently large to be consistent with the size of the sent message.
- Generating bit sequence should be practical and easy.
- Generated bits should be unpredictable.

Also, it should be noted that in order to guaranty unpredictability, the key stream should have two important properties: independence of generating numbers and having large period. These properties can be tested by statistical tests.

Todays most of practical stream ciphers are based on (LFSR) which makes stream ciphers practical and efficient. But LFSR and therefore stream ciphers are inefficient in implementation [1].

In 1984 Blum and Micali described how to generate a PRBG [2]. In 1999 Pascal Junod discussed and proved the security of The Blum-Blum-Shub Generator from cryptographical point of view [3], and it was implemented in 2014 with Aïssa et. al. [4,5]. In 2003 Edkal in his Ph.D thesis dicussed design and analysis of stream cipher based on LFSR [1]. In 2006, Parschi studied and analysed Chaos-based random number generators in the university of Bolonia [6].

In 2009 Krhovj´ak studied the relation between cryptography and PRNG in his Ph.D thesis [7]. And in 2013 Babu and Kumar described the design of a new stream cipher based on PRNG [8].

In this work, first, we present the notion of cryptographically secure pseudo random bit enumerators (PRBG), the discrete logistic map and linear congruential generator.

In the second part, the Chaotic Linear Congruential Generator (CLCG), a very simple and provably secure PRBG, is presented, with all the mathematical background needed to understand it. In the third part, the proof of its security is treated in details.

The paper is organized as follows: In section 2, pseudo random numbers and Discrete Logistic Map and the Linear Congruential Generator are introduced, in section 3, we present Chaotic Linear Congruential Generator, in section 4, statistical tests are implemented and in section 5 a numerical example is presented including some comparisons. Finally in section 6 we drive the conclusion.

2 Preliminareise

2.1 The Discrete Chaotic Logistic Map

The logistic map is a very simple mathematical model often used to describe the growth of biological populations. In 1976 May [10] showed that this simple model shows bewildering complex behaviour. Later Fiegenbaum [11,12] reported some of the universal quantitative features, which became the hallmark of the contemporary study of chaos. Because of its mathematical simplicity, this model continues to be useful test bed for new ideas in chaos theory as well as application of chaos in cryptography. The simple modified mathematical form of the logistic map is given as:

$$X_{n+1} = rX_n(1 - X_n) \qquad x = \frac{x_i}{x_{\max}} \quad 0 \le r \le 4 \qquad (1)$$

For $3.5 \le r \le 4$, x_n behaves chaotically, while for $3 \le r \le 3.45$ gradually n approaches a periodic motion of period 2. Logistic Map is in general, a one-dimensional map that can produce chaotic behavior [6]. Thus, it is in principle, interesting to see if such a map can describe the flow of ions into and out of electroactive polymeric films under the regimes of diffusion and migration promoted by a potential sweep in the electroactive potential region. The differential form of the quadratic logistic equation is:

$$\frac{dx}{dt} = rx(1 - x) \qquad (2)$$

Where $x = \frac{x_i}{x_{\max}}$ and $x_{\max}$ are the maximum value of x_i .

The right hand side function leads to a point of maximum and a drop in the rate of change of x; thus, the term $1 - x$ indeed introduces the non-linearity. Setting $r = 4$ and normalizing the x to its maximum values, the plots, as presented in Figure. 2, are obtained.

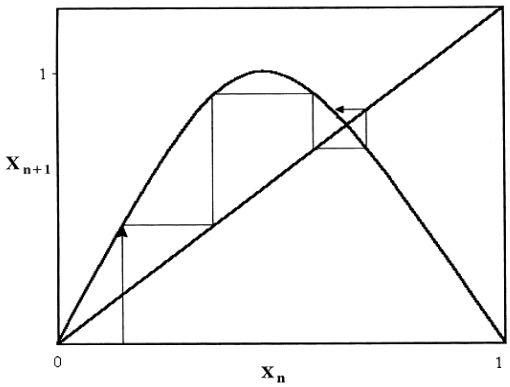

Figure1: Presentation of a Logistic Map. Arrows indicate the sequence of the iterations of the logistic equation for r = 4

2.2 The Linear Congruential Generator

Lehmer proposed a simple linear congruential generator as a source of random numbers [13]. Although these processes are completely deterministic, it can be shown that the numbers generated by the sequence appear to be uniformly distributed and statistically independent. Understanding its properties is necessary in order to use it to build better generators[14].

The form of the linear congruential generator is:

$x_i \equiv ax_{i-1} + c \pmod m \quad 0 \le x_i < m, i = 1, 2, \ldots \qquad (3)$

where the multiplier 'a', the increment 'c', and the modulus m are nonnegative integers.

Parameters:

a) *m*, the modulus, $m > 0$

b) a, the multiplier, $0 < a < m$

c) c, the increment, $0 < c < m$

d) x_0, the seed, $0 < x_0 < m$

Choice of a, c and m is important. m should be large, prime, e.g., $2^{31} - 1$. If c=0, few good values of a, e.g., $7^5 = 16807$.

Two LCGs can be combined to create a combined linear congruence generator, CLCG. With good constants in the underlying LCGs the generator has a period that is the product of the period of each LCG. The first CLCG was presented by the Wichmann and Hill generator [15]. The equivalence between this generator and an LCG was shown by Zeisel [16].

3. Chaotic Linear Congruential Generator (CLCG)

As mentioned before, a problem of Linear Congruential Generator (LCG) has considerably small period. That's why LCG is not sufficient for encryption. In this paper, we want to combine the LCG and discrete logestic map, until the generated number becomes suitable for application in encryption. Our purpose is to have keystream with high period.

As we know, the discrete logistic map is sensitive to initial value and has chaotic behavior. Because of this property we are going to use a discrete logistic map when the number, generated by LCG, is same. With this procedure, it is possible to generate suitable key stream.

Algorithm: CLCG

Select $x_0 \in (0,1)$
for i = 1 to n
$x_i = ax_{i-1} + b \pmod m$
for j = 1 to i-1
if $x_i == x_j$
$x_{i+1} = rx_i(1 - x_i)$
end if
end for
$y_i = x_i \pmod 2$
end for
The output sequence is $y_1, y_2, \ldots$

4. STATISTICAL TESTS RESULTS

Various statistical tests can be applied to a sequence to attempt to compare and evaluate the sequence to a truly random sequence. Randomness is a probabilistic property; that is, the properties of a random sequence. There are infinite number of possible statistical tests, each assessing the presence or absence of a "pattern" which, if detected, would indicate that the sequence is nonrandom. Because there are so many tests for judging whether a sequence is random or not, no specific finite set of tests is deemed "complete."

4.1 Correlation Test

This statistic test checks the "independence condition" of numbers generated by the pseudo random number generator. Correlation coefficient is a measure of association between two variables, and it ranges between -1 and 1. If the two variables are in perfect linear relationship, the correlation coefficient will be either 1 or -1.

We can set numbers in two subsets of x, y. Next, we calculate covariance of x, y and correlation coefficient by following:We can set numbers in two subset of x, y. Next, we calculate covariance of x, y and correlation coefficient by following:

$$\rho = \frac{cov(x,y)}{\text{var(x)var(y)}} \quad (4)$$

$$cov(x,y) = \frac{\sum_{i=1}^{n} x_i y_i}{\text{n}} - \bar{x}\bar{y} \quad (5)$$

$$var(x) = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n-1}} \quad (6)$$

As it is seen, If ρ be close to zero, then the generated numbers become independent, otherwise if ρ be away from zero, then the independence will fail [17].

4.1.1 Corrolation Test Result

We generate 500000 numbers by Chaotic Linear Congruential Bit Generator with r =3.999 , then Independency Test is applied for these numbers. We get $\rho = 0.0037$. This result shows that, generated numbers are approximately independent in the sense that to be applied in cryptography.

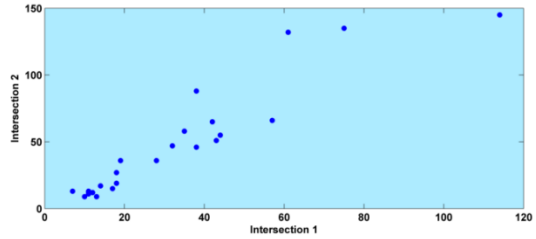

Figure2: Scatter plot of the Generated numbers by CLCG shows a positive correlation

4.2. Chi-square Gooduess-of-fit Test

The chi-square goodness-of-fit test, proposed by Pearson in 1900 is perhaps the best known among all other statistical tests. It is designed for testing discrete distributions and large samples. The test can be used for testing any distribution: uniform random number generators as well as random variate generators. The statistical test formula is of the form:

$$\sum_{i=1}^{k} \frac{(o_i - e_i)^2}{e_i} \pounds \chi^2_{[1-\alpha,k-1]} \quad (7)$$

Where;

a. k is the number of bins in the histogram.
b. o_i is the number of observed values in bin i in the histogram.
c. e_i is the number of expected values in bin i in the histogram.
d. The test results can be presented as follows: if the sum is less than $\chi^2_{[1-\alpha,i-1]}$, then the hypothesis that the observations come from the specified distribution cannot be rejected at a level of significance α [18].

4.2.1 Chi-square Gooduess-of-fit Test Result

100000 numbers are generated by Chaotic Linear Congruential Generator with r =3.999, then these numbers are tested via chi-square Goodness-of-Fit. The final result is obtained as follows:

$$\begin{cases} n = 100000 \\ k = 30 \\ e_i = 100000/30.0 \end{cases} \Rightarrow \sum_{i=1}^{30} \frac{(o_i - e_i)^2}{e_i} = 14.0781 \pounds \chi^2_{[0.1,29]} = 17.786$$

According to the result of the Chi-Square test, we cannot reject the null hypothesis that CLCG generates psedue random numbers with only 5% confidence.

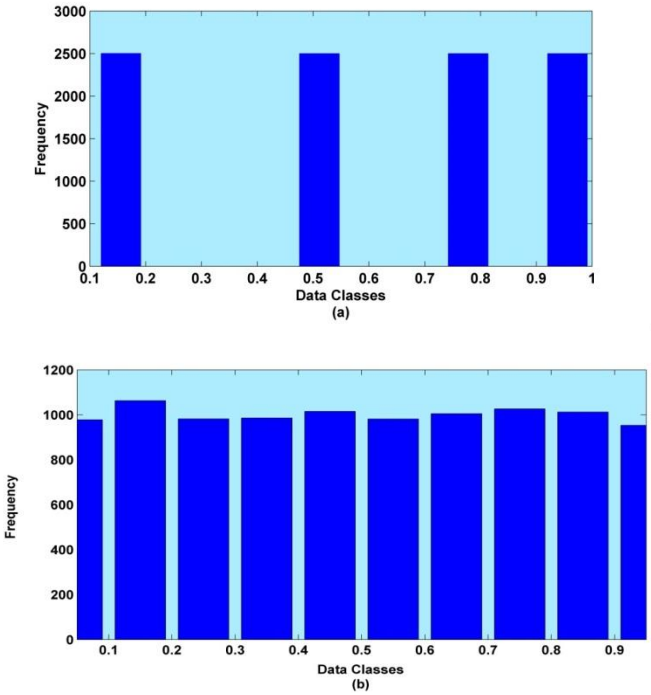

Figure 3: Histogram of frequency: frame (a) and frame(b) respectively, for 10000 generated numbers by LCG and CLCG

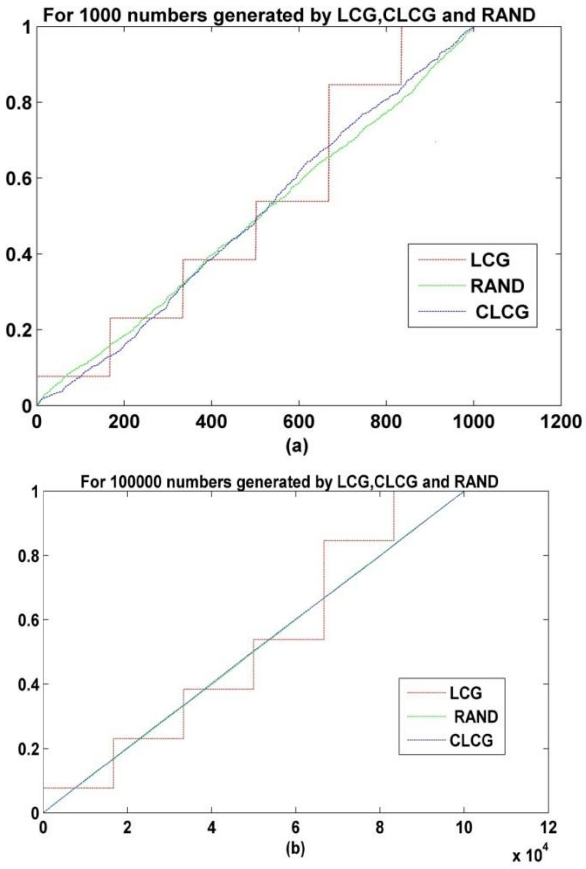

Figure4: frame (a), frame (b) and frame(c) respectively, 100000 generated number by LCG , CLCG and compire of LCG and CLCG.

Figures 3 and 4 are illustrating results of the statistical tests for uniformity of the generated numbers. Figure 3 compares uniformity of distribution of the generators, CLCG (b) w.r.t LCG (a) it can be seen that the presented method, i.e. CLCG is very close to uniform distribution, comparing with LCG method. In Figure 4 (a) and (b) we do the tests for 1000 and 100000 tests respectively. It can be seen that when we increase the number of samples, the CLCG method becomes very close to uniform distribution. In Figure 5 (b) we can see that numbers generated by CLCG are fall into those of RAND function.

5. Example

First we run Linear Congruential Generator (LCG) with $x_1 = 0.5, a = 3, b = 7, m = 5$

Generated numbers sequence is given in Table 1.

Table1: Generated numbers by CLCG

i	1	2	3	4	5
x(i)	0.1111	0.7778	0.5556	1.0000	0.1111

Next,we run new generator (CLCG) with

$x_1 = 0.5, a = 3, b = 7, m = 5$

Generated numbers sequence is given in Table 2.

Table2. Generated numbers by CLCG

i	1	2	2	4	5
x(i)	0.1111	0.7778	0.5556	1.0000	0.1111
i	6	7	8	9	10
x(i)	0.5401	0.6529	0.4263	0.5129	0.0061
i	11	12	13	14	15
x(i)	0.2976	0.3692	0.7463	0.1932	0.1056
i	16	17	18	19	20
x(i)	0.5074	0.4063	0.8931	0.2891	0.3567

RESULT:

The result is so surprised.At first Generated numbers sequence , period of numbers sequence is 4.but at new generator (CLCG), period of numbers sequence is very high, so that in 100000 Generated numbers, just one numbers was itratived as x(5).

6. CONCLUSION

he paper designs a new secure pseudo-random generation based on the chaotic linear congruential generator called CLCG. We believe that, the proposed generator is a secure PRNG from the cryptographic point of view. The quality of the new generator, are discussed with testing via the chi-square goodness-of-fit as well as correlation test. Based on these tests which are among the best and commonly applied statistical testers, we claim that CLCG provides high levels of security properties which are vital in cryptographic applications.

\noindent As mentioned in the context, weakness of regular LCGs is the shortness of the period. In addition to the proposed chaotic map, researchers can focus developing other forms of chaotic maps such as Lorenz attractor, Lozi maps ant etc.

7.REFERENCES

[1] P.Ekdahl, "On LFSR based Stream Ciphers*"*, *PhD Thesis*, Lund University, (2003).

[2] P.Junod, "Cryptographic Secure Pseudo-Random Bits Generation : The Blum-Blum-Shub Generator", *Note*, (1999).

[3] L.Blum, M.Blum, M.Shub, "Comparison of two pseudo-random number generators", *Proc. CRYPTO*, **82**,61-78,(1983).

[4] L. Blum, M. Blum, M. Shub, Mike, "*A Simple Unpredictable Pseudo Random Number Generator*", *SIAM Journal on Computing*, **15(2)**, ,364-83,(1986).

[5] B.Assa, M.Khaled, G.Lakhdar, *"*Implementation of Blum Blum Shub Generator for Message Encryption" , *International Conference on Control, Engineering and Information Technology* (**CEIT14**), (2014).

[6] F. Pareschi, "Chaos-Based Random Number Generator:Monotonic impelementation, Testing and Application*"*, *PhD Thesis*, Bologna University, (2009).

[7] J.Krhovjak, "Cryptographic random and pseudorandom data generators",*PhD Thesis*, Masaryk University,(2009).

[8] S. Dilli Babu, M. K.Patnala, "Design of a New Cryptography Algorithm using Reseeding-Mixing Pseudo Random Number Generator", *International Journal of Innovative Technology and Exploring Engineering*, **vol2,**284-286, (2013).

[9] M.Hoemmen, "Generating random numbers in parallel", *Note*, (2007).

[10] R.M. May, "Simple mathematical models with very complicated dynamics.Nature", vol **261** , pp. 459-467, (1976).

[11] M. J. Feigenbaum, "The universal metric properties of nonlinear transformations". *J. Stat. Phys*., **vol 21**,pp. 669-706,(1979) .

[12] M. J.Feigenbaum, Universal "behaviour in nonlinear systems". *Los Alamos Science*, **vol. 1**, pp. 4-27, (1980).

[13] D.H. Lehmer, "Mathematical methods in large-scale computing units", *In Proc. 2nd Sympos. on Large-Scale Digital Calculating Machinery*, Cambridge, MA,141-146, (1949),.

[14] B. Jansson, "Random number generators", *PhD thesis*, University of Stockholm, (1966).

[15] B.A.Wichmann, I.D. Hill, "An ecient and portable pseudo-random number generator, Applied Statistics" **31**, 188-190, (1984).

[16] H. Zeisel, "A remark on algorithm" *AS183, Applied Statistics*, **35**, (1986).

[17] R. A. Fisher, "Frequency distribution of the values of the correlation coecient in samples of an indenitely large population", *Biometrika*, **vol 10**, 507_521, (1915).

[18] S. Gorenstein, "Testiog a random number generator", *Comm. Assoc. Cow*, 111-118,(1976).