Application of chaotic maps in designing cryptographic pseudo random number generators

B. FATHI VAJARGAH[a,*], R. ASGHARI[b]
[a,]*Department of Statistics, Faculty of Mathematical Sciences, University of Guilan, Rasht, Iran*
[b] *Department of Applied Mathematics, Faculty of Mathematical Sciences, University of Guilan, Rasht, Iran*

In this paper, the structure of chaotic maps and its applications in cryptography and designing cryptographic pseudo random number generators is studied. Three cryptographic pseudo random number generators based on chaotic functions are constructed. High sensitivity to initial conditions in chaotic maps has the main role in the design of proposed algorithms. Proposed algorithms are tested by correlation test and the NIST tests and compared by frequency histograms. The statistical analyses confirm the high quality of the algorithms such as uniformity, independency and randomness properties and the efficiency of proposed algorithms for cryptographic applications.

(Received November 30, 2015; accepted February 10, 2017)

Keywords: Chaotic maps; Pseudo random number generators; Cryptography; Correlation test; NIST tests

1. Introduction

All cryptography systems are depending on generating pseudo random numbers. These numbers have main role in cryptosystems under secret key. Generation of such numbers is an important subject for researchers in different scientific fields [1, 2]. Application of chaotic systems in cryptographic applications is a suitable tool we have at hand. As an example, we can mention to the following application; using new 5D hyper chaotic systems in secure communication [3], Sketch of a Chaotic based Cryptosystem by Carmen Pellicer-Lostao, Ricardo Lopez-Ruiz [4], design of Block Encryption Ciphers Based on Chaotic Maps [5] and In 2009, designed a fast encryption scheme based on chaotic map by Su Su Maung and Myint Sein [6].

Chaotic systems also have wide range of applications in digital image and video cryptography; a novel fast image Encryption scheme based on the 3D chaotic baker map has designed by Mao, Chen and Lian [7], and a new chaotic algorithm for image encryption has presented Gao, Zhang, Liang, and Li [8].

Generating pseudo random numbers applying chaotic maps is popular method. There are several pseudo random numbers generators (PRNG) have been developed based on chaotic properties as an example we can mention to the generating PRN by using cellular automata [1], chaotic Hénon map [9,10,11], a robust chaos-based [12], logistic map [13] and etc.

A good PRNG should work efficiently, which means it should be able to produce a large amount of random numbers in a short period of time. In addition to the conditions above RNGs for cryptographic applications must be resistant against attacks [14, 15]. In designing stream cipher, a single pseudo random bit generator, plays the role of the key stream generator for the stream cipher system which is indeed generator of key stream.

Today's most of practical stream ciphers are based on Linear Feedback Shift Register (LFSR) which makes stream ciphers practical and efficient. But LFSR and therefore stream ciphers are inefficient in implementation [14]. In 2006, Parschi studied and analyzed Chaos-based random number generators in the University of Bologna [16]. In 2009 Krhovj´ak studied the relation between cryptography and PRNG in his Ph.D. thesis [17]. And in 2013 Babu and Kumar described the design of a new stream cipher based on PRNG [18].

In this paper, three new PRNG's are designed based on the chaotic maps. First, the henon map is applied in the middle square generator for solving the middle square generator problem in a special state and named. Next, the discrete logistic map is used in the BBS generator to increase complication of the BBS generator and named the Chaotic Blum Blum Shub (CBBS). Also, the CBBS are modified by controlling of generated numbers by the CBBS and named Modified CBBS (MCBBS). In this paper, chaos behaviors of the henon map and the discrete logistic map has main role to design the proposed algorithms. The proposed algorithms are examined by the NIST tests and compared by frequency histograms.

The paper is organized as follows: In section 2, the Discrete Logistic map, the Chaotic Hénon map and the reciprocal cot squared map are introduced. In section 3, we present three new PRNG's algorithms based on chaotic systems. In section 4, statistical tests and some comparisons are implemented. Finally, in section 5 we drive the conclusion.

2. Preliminarily

In this section, first we introduce two chaotic maps. These chaotic maps are well known for high sensitivity to initial value and have been used in several applications.

The main goal of authors is to apply the chaotic maps for designing new PRNG's. In the following, Blum Blum Shub generator (BBS) is presented. The BBS is a well-known generator for cryptographic applications.

2.1. The discrete logistic map

Logistic map is a very simple mathematical model often used to describe the growth of biological populations. Historically in 1976 bewildering complex behavior of Logistic maps are shown by May [19]. Later Fiegenbaum [10, 20] reported some of the universal quantitative features, which then became the hallmark of the contemporary study of chaos. Because of the mathematical simplicity, this model continues to be useful test bed for new ideas in chaos theory and popular in cryptographic applications. A modified mathematical form of the logistic map is given as:

$$x_{n+1} = r x_n (1 - x_n), \quad x = \frac{x_i}{x_{\max}}, \quad 0 \le r \le 4 \tag{1}$$

For $3.5 \le r \le 4$, x_n behaves chaotically, while for $3 \le r \le 3.45$, x_n gradually approaches to a periodic motion of period 2. In general the logistic map is a one-dimensional map that can produce chaotic behavior [10, 12].

In order to summarize behaviors encountered when r increases, constructing a bifurcation diagram can be helpful in which local maximum of values of x_n for each value of r are reported. The transition from one regime to another is called a bifurcation. Such a diagram illustrates the value and stability of the steady state and periodic orbits. In Fig. 1, the diagram is obtained by computing for each value of r the steady state or the maximum and minimum of x_n after the transients, e.g. from x_{100} to x_{10000} [11].

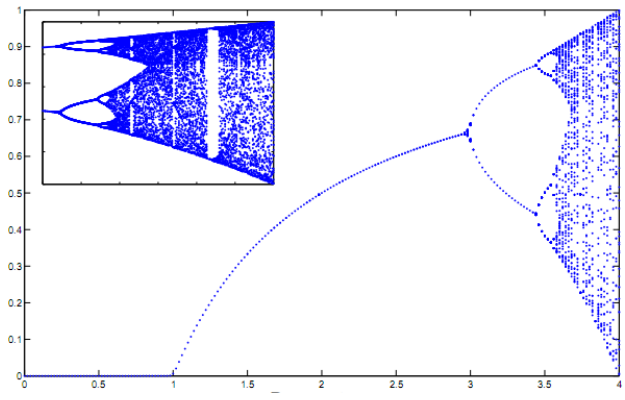

Fig.1. Bifurcation diagram. The inset is a zoom on the right part of the diagram [11]

For example graph of $f(x) = rx(x-1)$ is drawn (red curve in Fig. 2). Note that, $f(x)$ is a parabola passing through 0 at $x = 1$ and $x = 0$, and with a maximum at $x = \frac{1}{2}$. Choosing an initial value x_0, we can read $x_1 = f(x_0)$ directly from the parabolic curve, $x_2 = f(x_1)$, and so on. Analogously it is needed to evaluate $f(x)$ at each succeeding value of x_n. To achieve this goal the line $x_{n+1} = x_n$ is applied to reflect each value of x_{n+1} back to the x_n axis. This process, which is equivalent to bouncing between the curves $x_{n+1} = x_n$ (diagonal line) and $x_{n+1} = f(x)$ is called recursive graphical method (also called cobwebbing) [10, 11].

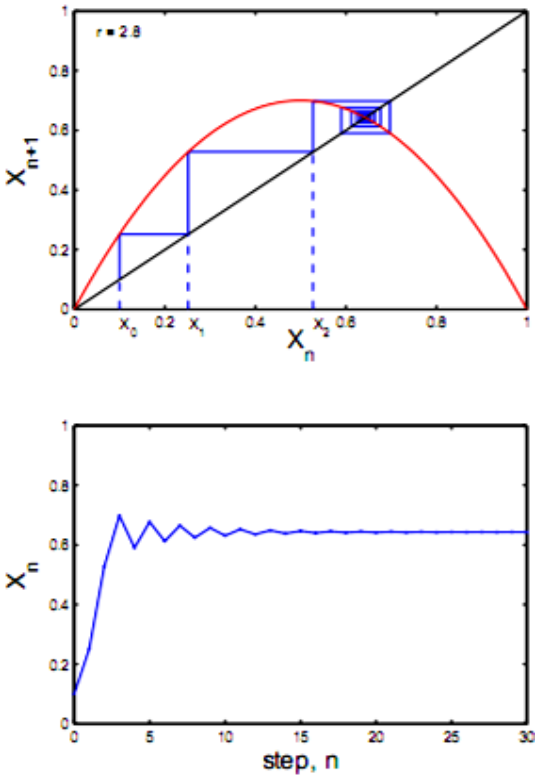

Fig. 2. cobweb diagram[11]

2.1.1. Sensitivity to initial conditions

Chaotic behaviors are characterized by a high sensitivity to initial conditions; starting from initial conditions arbitrarily close to each other, the trajectories will rapidly diverge as illustrate in Fig. 3. In other word, a small difference in the initial condition will produce large differences in the long-term behavior of the system. This property is sometimes called the "butterfly effect". In Fig. 3, both curves have been obtained for r = 3.8 but differ by their initial conditions: $x_0 = 0.4$ for the blue curve and $x_0 = 0.41$ for the red curve [11].

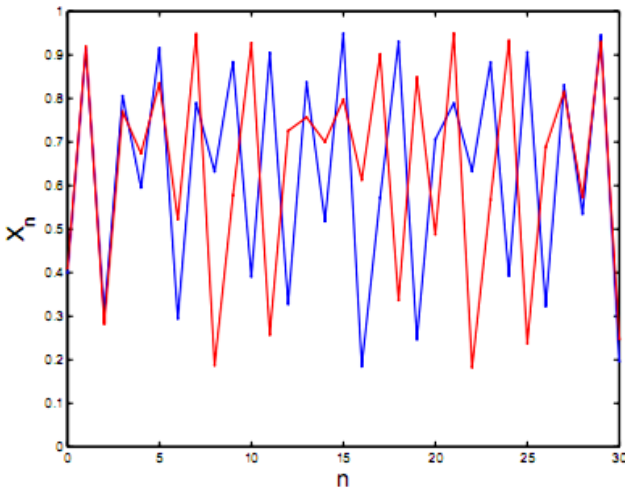

Fig. 3. Sensitivity to initial conditions [11]

2.2. The Hénon map

In 1989, Matthews Hénon suggested a new efficient way to deal with the intractable problem of fast and highly secure encryption by proposing the chaotic encryption algorithm. Chaotic system exhibits dynamics that are sensitive to initial conditions [21]. The basic property of a chaotic system, as mentioned earlier, is the sensitivity to initial conditions. Using chaotic cryptosystems in practical applications is successful to increase the security level for cryptographic systems. Hénon map is represented by the state equations with a chaotic attractor and is a simplified model of the Poincare map for the Lorenz equation proposed by Hénon in 1976 [21]. The two dimensional Hénon map is given in the following equation:

$$\begin{cases} x_{k+1} = -ax_k^2 + y_k + 1 \\ y_{k+1} = bx_k \end{cases} \quad (2)$$

The pair (x, y) is the two dimensional state of the system. The initial point is (x_0, y_0).

For $a = 1.4$ and $b = 0.3$ this map shows chaotic behavior. This chaotic behavior is known as Hénon attractor is the orbit of the iteration. The following pseudo-code can be used to explore the Hénon attractor on the computer. In Fig. 4, for $a = 1.4$, $b = 0.3$, $x_0 = 0.5$ and $y_0 = 0.5$ the Hénon attractor is shown.

If we zoom in on portions of this attractor, we can see a fractal structure.

The orbits are very sensitive to the initial conditions and the sign of chaos, but the attractor appears to be a stable geometrical object that is not sensitive to initial conditions.

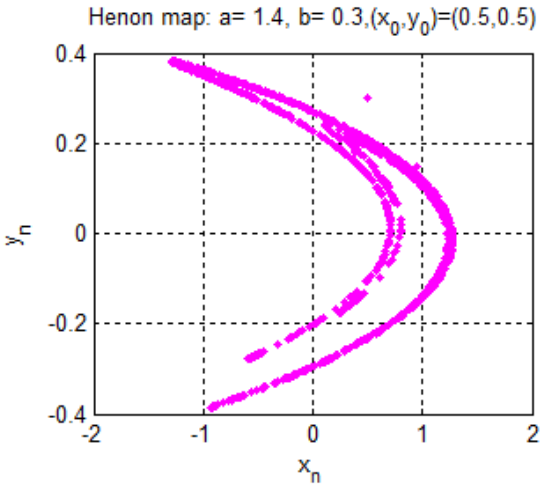

Fig. 4. The Hénon attractor

To study the evolution of that dynamic system, we plot the bifurcation diagram in the phase space in Fig. 4. That diagram allows visualizing the bifurcation phenomenon which is the transition of the orbit structure. Clearly the Hénon map is a bifurcation if $a = \frac{3}{4}(1-b)^2$. For different value of the parameter a, we plot a set of converged values of x, that means, we plot the Hénon map bifurcation diagram when $a = 1.4$, $b = 0.3$ and the initial conditions $x = y = 0$ are within the basin of attraction for this map. In Fig. 5, the bifurcation diagram of the Hénon map is shown.

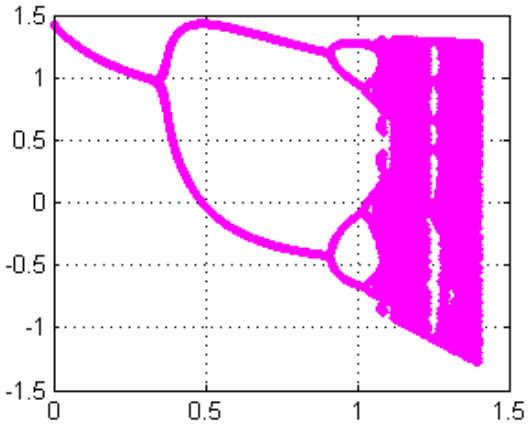

Fig. 5. The bifurcation diagram

This Hénon map receives a real number between 0 and1.4, then returns a real number in $[1.5, 1.5]$ again. The various sequences are yielded depending on the parameter a and the initial values x_0, y_0.

We can see that if the parameter a is taken between 0 and about 0.32, the sequence $\{x_n\}$ converges to a fixed point independent on the initial value x_0 , y_0. According the described properties, chaotic properties of the Hénon map and the discrete logistic map are useful to design an efficient cryptographic secure pseudo random number generator. The sensitive to initial condition property

playing important role in creating a suitable unpredictable random number sequence. In section 3, applications of the chaotic maps in the CSPRNG's are presented.

2.3. Blum Blum Shub generator

The Blum Blum Shub generator (BBS) is conceptually simple and provably secure PRBG [22]. In the BBS algorithm, we need an input S as seed for the PRN generator. The output sequence $\{y_i\}$ is provided which is called the pseudo random sequence. The BBS algorithm can summarize as follows [22,23]:

Select two k-bit random primes p and q that are 3 modulo 4, let $N = pq$, and let S be random integer that is relatively prime to N. The Algorithm is summarized in Algorithm 1.

Algorithm1: BBS generator
Select two k-bit random primes p and q that are 3 modulo 4, let $N = pq$, and let $s \in [1, N-1]$ be random integer that is relatively prime to N then: 1. $x_0 = s^2 \mod N$ 2. The sequence is defined as $x_i = x_{i-1}^2 \mod N$

It can be seen that this version of the algorithm slightly differs from the original algorithm as the outputs are not binary and we do not named it due to simplicity.

In order to generate a sequence of pseudo random number with length M, we simply can do M times iteration in step (2) of Algorithm 1. There are many considerations towards choosing proper M. Alexi et al. [23,24] proved that the BBS generator is secure if $M = O(loglogN)$. The security of a pseudo random generator is a characteristic that shows how hard it is to tell the difference between the pseudo random sequences and truly random sequences. For the BBS pseudo random generator distinguishing these two sequences is as hard as factoring a large composite integer. The BBS pseudo random generator is a CSPRBG under the assumption that integer factorization is intractable [25]. It forms the basis for the Blum-Goldwasser probabilistic public key encryption scheme [22, 26].

3. Proposed algorithms

In this section, three new PRNG's are proposed. The proposed algorithms are based on the Hénon map and the discrete logistic map. It is well known that the middle square generator and the linear congruential generator have a big problem and are not useful for cryptographic applications. On other hand, the chaotic maps have a suitable property. They are very sensitive to initial conditions. A small change in input value leads to create a very different output sequence. These notes are suitable reasons to design new PRNG's based on combination of the chaotic maps and other generators. Therefore, we propose three algorithms based on combinations a pseudo random number generator and a chaotic map. These proposed Algorithms are proper for cryptographic purposes, because we correct the deficiency of existing algorithms and improve them. These algorithms are tested by NIST test and comparison, in the next section.

3.1. First proposed chaotic PRNG

In 2011, Hamed Rahimov, Majid Babaie and Hassan Hassanabadi were proposing a new PRNG based on the discrete logistic map [27] and the middle square method. They tried to solve a big problem in the middle square method. The middle square method usually has a short period. This problem occurs when middle digits of generated number are all zero and the algorithm creates zero value forever. Because of this big problem, this generator is not efficient for applications such as cryptographic systems. That's why; the discrete logistic map was used in the middle square when an output number is zero [27]. In first proposed chaotic PRNG, the chaotic Hénon map has used instead of the discrete logistic map. This proposed algorithm named Chaotic Hénon Middle Square (CHMS). This algorithm is summarized in Algorithm 2:

Algorithm 2: The Chaotic Hénon Middle square
$C = A \times A \Rightarrow D = C\%10^{\frac{n}{2}}$ $E = (C-D) \Big/ 10^{\frac{n}{2}} \Rightarrow result = E\%10^n$ $Num = result \Big/ 10^n$ $if\ (Num == 0) \quad then \quad \{$ $\begin{cases} x_{n+1} = -ax_n^2 + y_n + 1 \\ y_{n+1} = bx_n \end{cases}$ $B = x_{n+1}$ $x_n = x_{n+1}$ $\}$

3.2. Second proposed chaotic PRNG

In this subsection, second proposed chaotic PRNG is presented. In this new PRNG, the discrete logistic map is used in the BBS generator and named Chaotic Blum Blum Shub (CBBS) and summarized in Algorithm 3. In this algorithm, first a number is generated by the BBS. Next, the generated numbers is used in the discrete logistic map as an input value. Finally, the output value is generated number by CBBS algorithm. The aim of this proposed algorithm is improvement of randomness properties in BBS generator. As we know, the logistic map is very

sensitive to initial value. Therefore, the logistic map helps to create an unpredicted random sequence.

Algorithm 3: CBBS algorithm
Select two k-bit random primes p and q that are 3 modulo 4, let $N = pq$, and let $s \in [1, N-1]$ be random integer that is relatively prime to N then: 1. $x_0 = s^2 \mod N$ 2. Compute $x_i = x_{i-1}^2 \mod N$ 3. The sequence is defined as $x_{i+1} = rx_i(1-x_i)$

3.3. Third proposed chaotic PRNG

In this subsection, third proposed chaotic PRNG is presented. By this Algorithm the goal is to goimprove the uniformity of the CBBS. To this aim proper controller is applied to regulate the distribution of PRNs generated by the CBBS.

The interval in which we are going to produce random numbers is divided into a number of subintervals, and at every step of the algorithm a subinterval is chosen randomly and then recall BBS to generate PRN at this interval. The subintervals are forced to control CBBS to generate PRN more uniform according to the statistical tests and histogram plots. Noting that subintervals should be chosen with same chance as much as possible to guarantee the uniformity. The algorithm is presented in Algorithm 4.

Algorithm4: MCBBS algorithm
Start: Choose Shub primes p; q, and seed S. Preparation: Divide the interval I into some subintervals I_j . Run: the CBBS generator for generating $y_1, y_2, \ldots$ $For\ j = 1, 2, \ldots$ 1. Choose a subinterval randomly and name itI^*. 2. Transform y_j into subinterval I^* put into xj . **Return**: $x_1, x_2, \ldots$ As generated random numbers by the MCBBS.

4. Statistical analyses

As we know, the cryptographic PRNG's should to have three important properties as follows:

1. Uniformity
2. Independency of sequence
3. Unpredicted

In this section, the correlation tests as well as the NIST tests are implemented on the proposed algorithms to investigate the strength of the algorithms. In addition to, frequency histograms in two and three dimensions are drawn for better presentation of the proposed algorithms.

All results are reported through tables and figures.

The results show that which algorithm is the best for cryptographic applications.

4.1. The correlation test results

In this subsection, the correlation test is used to investigate independency of generated number sequences. Numbers are more independent, the test results closer to zero. We generate 5000, 10000 and 20000 numbers by CHMS, CBBS and MCBBS generators, and then the correlation test is applied to these numbers. The results show that, the generated numbers by CBBS and CHMS have more independency comparing with MCBBS and are efficient in the sense that to be applied in cryptography from the correlation point of view. The final results are present in Table 1.

Table1. The correlation test results

Algorithm	**Results n=5000**	**Result n=10000**	**Result n=20000**
CHMS	-0.0028	0.0058	0.0055
CBBS	0.0065	0.0082	0.0053
MCBBS	0.0725	0.0640	0.0694

4.2. The frequency histograms

In this subsection, the histograms illustrate quality of the distribution of generated numbers by these algorithms. The uniform distribution of generated numbers is an ideal statistical property for cryptographic PRNG's. First, we generate 1000 and 10000 numbers by CHMS, CBBS and MCBBS and then, we draw the frequency histograms. In Figs. 6 and 7, histograms of MCBBS are completely uniform and histograms of CHMS are almost uniform. But, uniformity of CBBS isn't suitable.

As the uniformity of CBBS is not proper for cryptography applications, we modify CBBS such that modified version of CBBS, called MCBBS, passed the statistical tests which is required for cryptography proposed.

Also, in Fig. 8, the scatter plots of CHMS, CBBS and MCBBS are presented. The scatter plots show that scatter of CHMS is more than CBBS and MCBBS and also scatters of CBBS and MCBBS are similar. In Fig. 9, the comparison is clearer, In short the uniformity of CBBS is more than CHMS and also, uniformity of MCBBS is more than CBBS.

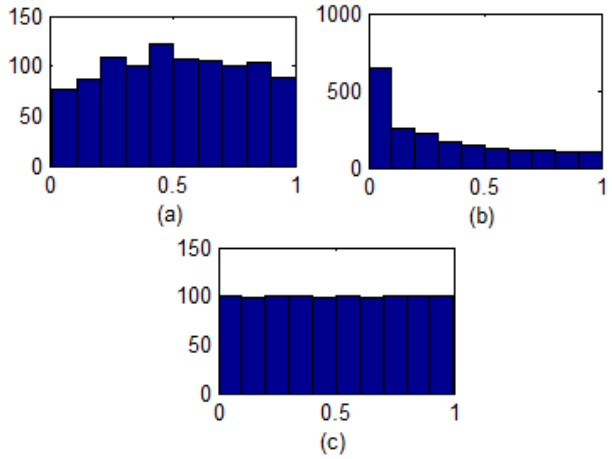

Fig. 6. Histogram of frequency: frame (a), frame (b) and frame(c) respectively, for 1000 generated numbers by CHMS, CBBS and MCBBS

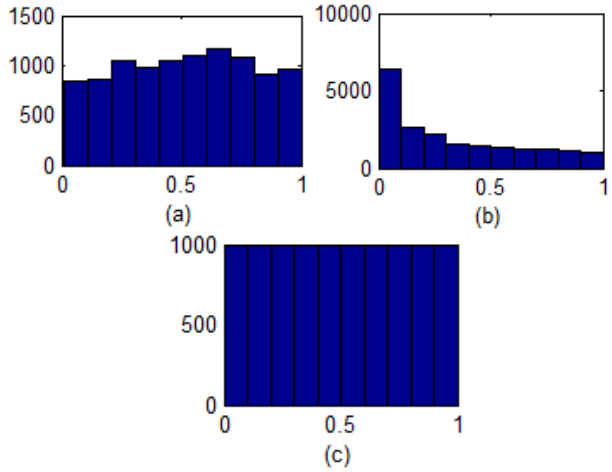

Fig. 7. Histogram of frequency: frame (a), frame (b) and frame(c) respectively, for 10000 generated numbers by CHMS, CBBS and MCBBS

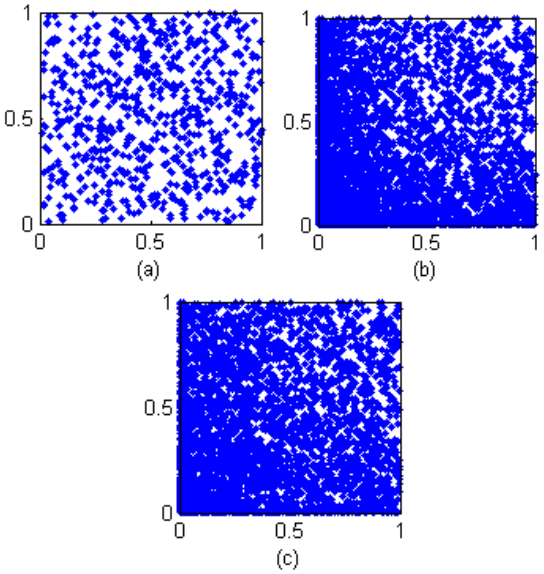

Fig. 8. Scatter plot: frame (a), frame (b) and (c) respectively, for CHMS, CBBS and MCBBS

4.3 The NIST tests results

For completeness and as a very basic test, frequency test is implemented to verify there were roughly the same number of 0s, 1s, 2s, 3s, etc. Then serial test which compares observed frequencies with their hypothetical predictions are also implemented. The poker test which designed based on hands in the game poker, tests certain sequences of five numbers of the results. In order to looking at the distances between zeroes we planned to apply Gap test. Final results including P-values and test result are summarized in Table 2. According to Table 2, we can infer that MCBBS performs almost better than CBBS and CBBS performs better than CHMS except in the some tests. The NIST tests are a statistical package. If the P-value for a test is determined to be equal zero, that means the sequence appears to be exactly non-random and a P-value to be 1 then the sequence is significantly random [27].

Table 2.The NIST test results

NIST test	CHMS (P-value)	CBBS (P-value)	MCBBS (P-value)
Frequency	0.45	0.69	0.76
Block-frequency	0.56	0.42	0.51
CuSums-forward	0.12	0.58	0.53
CuSums-backward	0.22	0.65	0.78
Rans	0.91	0.83	0.81
Long run	0.75	0.87	0.79
Rank	0.63	0.71	0.74
FFT	0.31	0.84	0.96
Approximate entropy	0.62	0.60	0.83
Universal	0.42	0.52	0.60
Serial	0.17	0.43	0.39
Poker	0.35	0.66	0.60
Gap	0.39	0.19	0.22

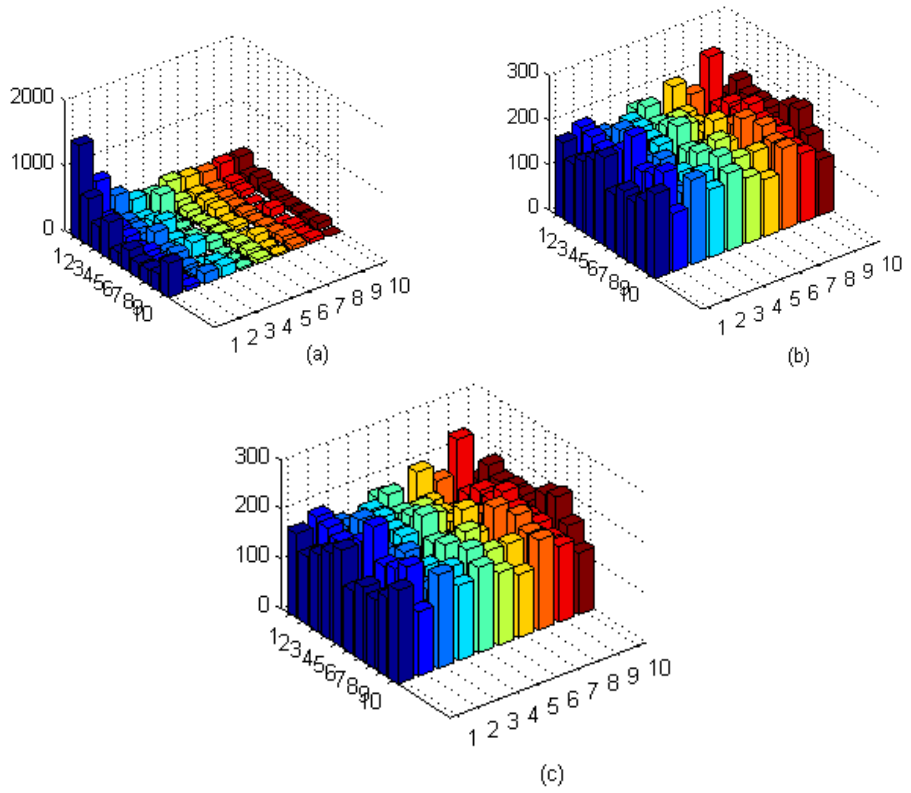

Fig. 9. Histogram of frequency in 2D: frame (a), frame (b) and frame (c) respectively, for generated numbers by CHMS, CBBS and MCBBS.

5. Conclusion

In this paper, the logistic map and the Hénon maps and their properties are introduced in detail. Next, BBS generator as a cryptographic PRNG is described. Then, three new cryptographic secure PRNG's are designed based on combination of the chaotic maps and a pseudo number generator. The high sensitive to initial conditions help to design a suitable CSPRNG with a long period, chaotic maps. For statistical analyses, the NIST tests and the correlation test are performed over CHMS, CBBS and MCBBS and the results are presented. Also, the frequency histograms are drawn in 2D and 3D. The results show that, the all proposed PRNG's are proper for all cryptographic applications.

We figured that MCBBS generator is performed better that CBBS and CHMS. For future researches we propose that these generators will use in cryptographic applications such as stream ciphers.

References

[1] F. Temiza, I. Siapa, H. Akinb, Acta Physica Polonica, **125**, 534 (2014).
[2] A. Beige, B. G. Engelert, Ch. Kurtsiefer, H. Weinfurter, Acta Physica Polonica **101**, 357 (2002).
[3] H. Bouraoui, K. Kemih, Acta Physica Polonica **123**, 259 (2013.
[4] C. Pellicer-Lostao, R. Lopez-Ruiz, Notions of Chaotic Cryptography: Sketch of a Chaos based Cryptosystem, Applied Cryptography and Network Seurity, Intec Books, 2012.
[5] G. Jakimoski, L. Kocarev, IEEE Transactions on circuits and systems—I: Fundamental Theory and Applications, **48**(2), 163 (2001).
[6] Su Su Maung, Myitnt Sein, GMSARN International Conference on Sustainable Development: Issues and Prospects for the GMS, Manchester, UK (2008).
[7] Y. B. Mao, G. Chen, S. G. Lian, Int. J. Bifurcate Chaos **14**, 3613 (2004).
[8] H. Gao, Y. Zhang, S. Liang, D. Li, Chaos, Solutions &Fractals **29**, 393, (2006).
[9] B. F. Vajargah, R. Asghari, International Journal of Physic Theory and Cryptographicy **9**, 19 (2015).
[10] B. F. Vajargah, R. Asghari, International Journal of Mechatronics, Electrical and Computer Technology (IJMEC) **5**(15), 2120 (2015).
[11] D. Gonze, The logistic equation, Oct. 4, (2013).
[12] M. Drutarovský, P. Galajda, Radio engineering **16**(3), 120 (2007).
[13] H. Rahimov, M. Babaie, H. Hassan Abadi, Applied Mathematics **2**, 412 (2011).
[14] C. Kenny, Trinity College Dublin, April 2005.
[15] M. S. Baptista, Physics Letters A **240**(1-2), 50 (1998).
[16] F. Pareschi, PhD Thesis, Bologna University, Dec. 2006-2009.
[17] J. Krhovjak, PhD Thesis, Masaryk University, Jan. 2009.
[18] S. Dilli Babu, M. K. Patnala, International Journal of Innovative Technology and Exploring Engineering **2**, 284, (2013).

[19] R. M. May, Nature **261**, 459 (1976).
[20] M. J. Feigenbaum, J. Stat. Phys. **21**, 669 (1979).
[21] S. Parker, O. Chua, Proceedings of the IEEE Transactions, **75**(8), 982 (1995).
[22] C. Ding, Electronics Letters **33**(8), 677 (1997); V. Umesh, V. Vijay V. Vazirani. Advances in Cryptology, Springer Berlin Heidelberg, 1985.
[23] B. Assa, M. Khaled, G. Lakhdar, International Conference on Control, Engineering and Information Technology (CEIT14), Berlin, 2014.
[24] M. Bellare, P. Rogaway, 2004, Introduction to modern cryptography. Notes.
[25] L. Blum, L. Blum, M. Shub, M. Mike, 1986, SIAM Journal on Computing **15**(2), 364 (1989).
[26] P. Junod, Note, 1999.
[27] N. Mondal, P. Ghoshit, arXiv preprint arXiv: 1203.5731, 1 (2012).
[28] K. H. Tsoi, K. H. Leung, P. H. W. Leong, IET Proceeding Computers & Digital Techniques **1**(4), 208 (2007).

*Corresponding author: fathi@guilan.ac.ir