A Random Bit Generator Using Chaotic Maps

Narendra K Pareek[1], Vinod Patidar[2], and Krishan K Sud[3]
(Corresponding author: Krishan K Sud)

University Computer Centre, Vigyan Bhawan, New Campus[1]
M. L. Sukhadia University, Udaipur (Raj.), India
Department of Physics, Banasthali University, Banasthali (Raj.), India[2]
Sir Padampat Singhania University, Udaipur (Raj.), India[3]
(Email: {npareek, kksud}@yahoo.com, vinod_r_patidar@yahoo.co.in)
(Received Feb. 13, 2007; revised Aug. 2, 2007; and accepted Mar. 7, 2008)

Abstract

Chaotic systems have many interesting features such as sensitivity on initial condition and system parameter, ergodicity and mixing properties. In this paper, we exploit these interesting properties of chaotic systems to design a random bit generator, called CCCBG, in which two chaotic systems are cross-coupled with each other. To evaluate the randomness of the bit streams generated by the CCCBG, the four basic tests: monobit test, serial test, auto-correlation, Poker test and the most stringent tests of randomness: the NIST suite tests have been performed. As a result no patterns have been observed in the bit streams generated by the proposed CCCBG. The proposed CCCBG can be used in many applications requiring random binary sequences and also in the design of secure cryptosystems.

Keywords: Piecewise linear map, random bit generator, randomness

1 Introduction

Chaotic systems have a number of interesting properties such as sensitivity on initial condition and system parameter, ergodicity and mixing (stretching and folding) properties, etc. These properties make the chaotic systems a worthy choice for constructing the cryptosystems (block ciphers as well as stream ciphers) as sensitivity to the initial condition/system parameter and mixing properties respectively, are analogous to the confusion and diffusion properties of a good cryptosystem. A general way to design a chaotic stream cipher is to generate a random bit stream using chaotic system. In this paper, we propose a novel random bit generator through the cross-coupling of two chaotic systems which can be used in the design of a new chaotic stream cipher as well as in other engineering applications, where random bit sequences are required [15]. The first idea for designing pseudo-random number generator by making use of chaotic first order nonlinear difference equations was proposed by Oishi and Inoue [16] in 1982 and could construct a uniform random number generator with an arbitary Kolmogorov's entropy. After a long gap, in 1993 Lin and Chua [9] designed a pseudo random number generator by using a second-order digital filter and realised it on digital hardware. In 1996 Andrecut [1] suggested a method for obtaining a random number generator based on logistic map and also compared the congruential random generators, which are periodic, with the logistic random number generator, which is infinite and aperiodic. In 1999 Gonzalez and Pino [3] generalized the logistic map and designed a new function. The new chaotic function was truly unpredictable random function, which helped in the generation of truly random numbers. In 2001 Kolesov et al. [6] developed a digital random-number generator based on the discrete chaotic-signal algorithm. The suggested digital generator employed the matrix method of chaotic-signal synthesis. Further, Stojanovski and Kocarev [17, 18] analysed the application of a chaotic piecewise-linear one-dimensional map as random number generator. Li et al. [8] did a theoretical analysis, which suggests that piecewise linear chaotic maps have perfect cryptographic properties like: balance in the defined interval, long cycle length, high linear complexity, good correlation properties etc. They also pointed out that bit streams generated through a single chaotic system are potentially insecure as the output may leak some information about the chaotic system. To overcome this difficulty, they proposed a pseudo random bit generator based on a couple of chaotic systems, which are iterated independently and the bit streams are generated by comparing the outputs of these chaotic maps. They also justified their theoretical claims through a few numerical experimentations on the proposed pseudo random bit generator. In 2003 Kocarev and Jakimoski [5] discussed the different possibilities of using chaotic maps as pseudo-random number generators and also constructed a chaos-based pseudorandom bit generator. In 2004 Fu et al. [2] proposed a chaos-based random number generator using piecewise chaotic map. Further, a one-way coupled chaotic map lattice was used by Huaping et al. [4] for generating pseudo-random numbers. They showed that with suitable cooperative applications of both chaotic and conventional approaches, the output of the spatiotemporally chaotic system can meet the practical requirements of random numbers i.e. excellent random statistical properties, long periodicity of computer realizations and fast speed of random number generations. This pseudo-random number generator system can be used as ideal synchronous and self-synchronizing stream cipher systems for secure communications. In 2005 Li et al. [7] designed and analysed a random number generator based on a piecewise-linear map. Further, A new pseudo-random number generator (PRNG) based on a modified logistic map was proposed by Liu [10]. Based on this PRNG, a chaotic stream cipher was designed. Further, a chaotic random number generator was developed by Wang et al. [20] and realized it by an analog circuit. In 2006, Wang et al. [19] proposed a pseudo-random number generator based on z-logistic map, where the binary sequence through the chaotic orbit was realized under finite computing precision.

In the proposed random bit generator, two cross-coupled piecewise linear chaotic maps are employed (unlike to the pseudo random bit generator proposed in [8], where also two piecewise linear chaotic maps are employed but they are not coupled with each other) to generate random sequences and the set up is abbreviated as CCCBG (Cross-Coupled Chaotic random Bit Generator). In the CCCBG, random bit streams are generated by comparing the two orbits generated by cross coupled piecewise linear chaotic maps; therefore it is difficult for an eavesdropper to extract information about both chaotic systems. The rest of the paper is organised as follows: In the Section 2, we discuss the dynamics of the skew tent map in brief and the construction of the proposed CCCBG is presented in Section 3. In Section 4, we discuss the uniformity and randomness of the bit streams generated by CCCBG in detail and finally, in Section 5, we conclude the paper.

2 Dynamics of Skew Tent Map

The skew tent map is ergodic and has uniform invariant density function in its definition interval [18]. It is the simplest kind of one-dimensional chaotic map which is defined as:

$$x_{i+1} = F(\alpha\,,\, x_i) = \begin{cases} \frac{x_i}{\alpha} & x_i = [0, \alpha) \\ \frac{1-x_i}{1-\alpha} & x_i = (\alpha, 1] \end{cases} \qquad (1)$$

where α and x_i are system parameter and initial condition of the map respectively. It is a non-invertible transformation of unit interval onto itself and contains only one system parameter α, which determines position of the top of the tent in the interval [0,1]. A sequence computed by iterating $F(\alpha, x)$, is expansionary everywhere in the interval [0,1] and distributed uniformly in it. Orbits for system parameter values 0.4 and 0.8 are shown in Figure 1. In Figure 2, we have depicted the chaotic solutions of the Equation (1), which show sensitivity on initial condition as well as on system parameter.

3 Cross-coupled Chaotic Tent Map Based Bit Generator(CCCBG)

In this Section, we discuss the arrangement of chaotic systems in CCCBG. In the proposed CCCBG, we choose two skew tent maps which are piecewise linear chaotic maps and cross-coupled as shown in the Figure 3. The output generated by the first tent map is fed to the second tent map as the input (initial condition) and vice versa. The system parameter for the both chaotic maps is kept same and is in the chaotic regime. If $f_1(x_0, \alpha)$ and $f_2(y_0, \alpha)$ are two piecewise linear chaotic maps and are given as:

$$\begin{aligned} x_{i+1} &= f_1(\alpha\,,\, x_i)\ , \\ y_{i+1} &= f_2(\alpha\,,\, y_i)\ , \end{aligned}$$

where α is the system parameter and is same for both chaotic tent maps, x_i and y_i are the initial conditions and x_{i+1} and y_{i+1} are their new corresponding states. The CCCBG produces the binary sequences by comparing the outputs of the cross coupled piecewise linear chaotic maps (as shown in Figure 3) in the following way:

$$g(x_{i+1}, y_{i+1}) = \begin{cases} 0 & if\ x_{i+1} < y_{i+1}; \\ 1 & otherwise. \end{cases}$$

If the binary sequences generated by the CCCBG are random and have no pattern in them, we can use them for the development of new chaotic stream ciphers. In the next section, we discuss basic statistical tests as well as NIST suite tests for testing the randomness and uniformity of the binary sequences generated by CCCBG.

4 Analysis of Randomness of Bit Streams

We have studied the randomness and uniformity of the several binary sequences of large size, generated by the CCCBG for different sets of system parameter and initial conditions of cross-coupled tent maps. Here, we present the results for 10000 and 15000 sized binary sequences corresponding to the following parameter values of the five sets: (0.48999, 0.5006841, 0.538167586), (0.49045, 0.6410089, 0.505410089), (0.49493, 0.4417689, 0.754193089), (0.49951, 0.5166892, 0.273417389) and (0.49999, 0.1996892, 0.738567389), where the first parameter value represents the system parameter value which is same for both tent maps, and the second and third one as the initial condition for the two tent maps. For

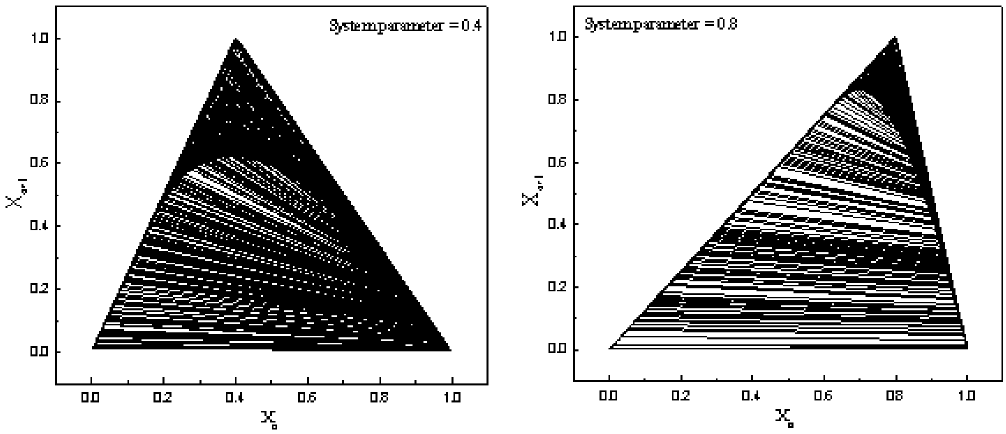

Figure 1: Shows the orbits of the skew tent map for system parameter values 0.4 and 0.8

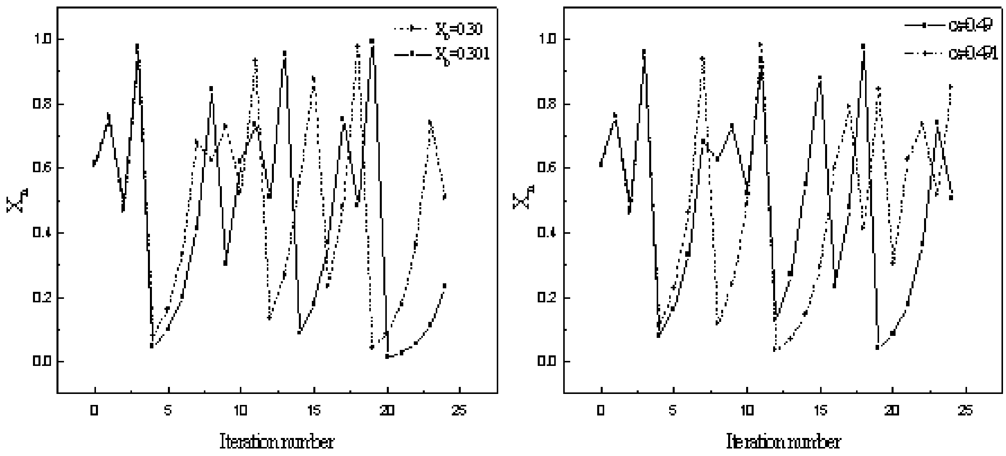

Figure 2: Shows the sensitivity of chaotic solution of skew tent map on initial condition and system parameter (α)

convenience, these five sets are designated as (α_1, x_1, y_1), (α_2, x_2, y_2), (α_3, x_3, y_3), (α_4, x_4, y_4) and (α_5, x_5, y_5). We discuss in the following paragraph of this Section the result and conclusions of our study of the different statistical tests to observe the randomness and uniformity of the binary sequences generated by the proposed CCCBG.

4.1 Monobit Test

The purpose of this test is to determine whether the frequency of 0's and 1's in binary sequences generated by the CCCBG are approximately same [11]. Let n_0, n_1 denote the number of 0's and 1's in binary sequences respectively. We calculate χ^2 by using the formula [11]:

$$\chi^2 = \frac{(n_0 - n_1)^2}{n},$$

which approximately follow a χ^2 distribution with one degree of freedom. The computed results are shown in Table 1. The calculated values of χ^2 are less as compared to the critical value of χ^2 at α=0.05 (5% level of significance) and $1df$ (one degree of freedom). It means that these binary sequences pass the monobit test and can be said to be satisfactorily random with respect to this test [11].

4.2 Serial Test

The purpose of this test is to determine whether the number of occurrence of pairs 00, 01, 10 and 11 in the bit streams generated by CCCBG is approximately same [11]. Let n_{00}, n_{01}, n_{10} and n_{11} denote the number of occurrence of pairs 00, 01, 10 and 11 respectively in the binary sequences. We calculate χ^2 by using the formula [11]:

$$\chi^2 = \frac{4}{n-1}(n_{00}^2 + n_{01}^2 + n_{10}^2 + n_{11}^2) - \frac{2}{n}(n_0^2 + n_1^2) + 1,$$

and the computed values are found to follow approximately the χ^2 distribution with 2 degrees of freedom. The

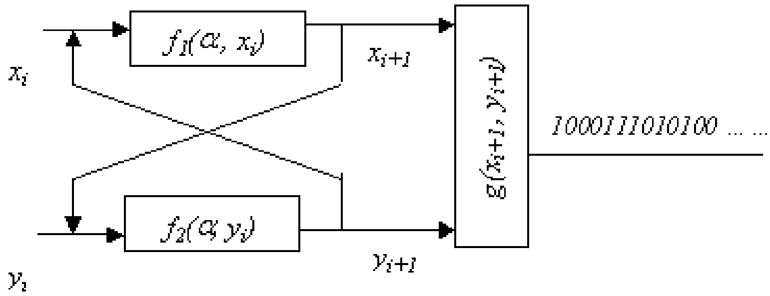

Figure 3: The block diagram of Cross-Coupled Chaotic random bit Generator (CCCBG)

Table 1: Shows the calculated χ^2 values for monobit test for two different large sized binary sequences having N=10000 and 15000 corresponding to five different sets of parameter values. The parameter values corresponding to five sets are (0.48999, 0.5006841, 0.538167586), (0.49045, 0.6410089, 0.505410089), (0.49493, 0.4417689, 0.754193089), (0.49951, 0.5166892, 0.273417389) and (0.49999,0.1996892, 0.738567389). The parameters α_i, x_i and y_i are respectively the system parameter (same for both maps), initial conditions for the first and second maps.

Size	Parameter	Numbers in binary sequences n_0	n_1	Calculated χ^2 value	Critical χ^2 value at α=0.005
N=10000	(α_1, x_1, y_1)	5030	4970	0.36	3.8415
	(α_2, x_2, y_2)	4987	5013	0.07	
	(α_3, x_3, y_3)	4995	5005	0.01	
	(α_4, x_4, y_4)	4950	5050	1.00	
	(α_5, x_5, y_5)	5029	4971	0.34	
N=15000	(α_1, x_1, y_1)	7536	7464	0.35	3.8415
	(α_2, x_2, y_2)	7504	7496	0.01	
	(α_3, x_3, y_3)	7496	7504	0.01	
	(α_4, x_4, y_4)	7419	7581	1.75	
	(α_5, x_5, y_5)	7567	7433	1.20	

results are shown in Table 2. The calculated values of χ^2 are less than critical value of χ^2 at α=0.05 (5% level of significance) and $2df$ (two degrees of freedom). It means that binary sequences pass the serial test and are satisfactorily random with respect to this test.

4.3 Auto Correlation

The purpose of this test is to check for correlations between the binary sequences generated by the proposed CCCBG. Let d be a fixed integer 1$\leq d \leq n/2$ where n is the size of binary sequence. The number of bits in binary sequences not equal to their d-shifts is

$$A(d) = \sum_{i=0}^{n-d-1} s_i \oplus s_{i+d} .$$

The statistical formula used is as follows [11]:

$$Z = 2(A(d) - \frac{n-d}{2})/\sqrt{n-d}.$$

The results are shown in Table 3 for $d = 25$. The calculated results fall within the accepted region $Z = \pm 1.96$ at α=0.05 (i.e. at 5% level of significance). Hence the binary sequences are random with respect to this test.

4.4 Poker Test

Let m be a positive integer such that $n/m \geq 5 \times (2^m)$ and let $k = n/m$ where n is the size of binary sequence. We divide the binary sequence into k non-overlapping parts each of length m and n_i is the number of occurrence of ith type of sequences of length m, where $1 \leq i \leq 2^m$. The Poker test determines whether m-bits long string appear approximately in same number of times a set of binary sequences [11]. We calculate χ^2 by using the formula [11]:

$$\chi^2 = \frac{2^m}{k}(\sum_{i=1}^{2^m} n_i^2) - k,$$

and computed values approximately follow the χ^2 distribution with (2m − 1) degree of freedom. The results are shown in Table 4. The calculated values of χ^2 are less than critical value of χ^2 at α=0.05 and $2^{m-1})df$ (degree of freedom). Hence the binary sequences also pass the Poker test and are satisfactorily random with respect to this test.

In the above analysis, we have examined the randomness of the binary sequences generated by the CCCBG for four basic statistical tests. It is observed that when the value of the system parameter (α) is between 0.49 and

Table 4: Shows the calculated χ^2 values for Poker test for two different large sized binary sequences having N=10000 and 15000 corresponding to five different sets of parameter values. The parameters α_i, x_i and y_i are respectively the system parameter (same for both maps), initial conditions for the first and second maps. The values of the parameters are same as given in the caption of Table 1.

Size	Block length (m) in bits	df (2^{m-1})	Calculated χ^2 value for					Critical χ^2 value at
			(α_1, x_1, y_1)	(α_2, x_2, y_2)	(α_3, x_3, y_3)	(α_4, x_4, y_4)	(α_5, x_5, y_5)	α=0.05
N=10000	2	3	07.0704	04.0752	00.2800	06.2368	04.1232	07.8147
	3	7	10.4647	05.8803	04.5506	12.6253	02.0015	14.0671
	4	15	19.3856	11.8080	09.7088	23.2256	12.9088	24.9958
N=15000	2	3	07.2981	05.5467	00.2976	02.0832	03.6299	07.8147
	3	7	04.7168	06.5920	07.9424	13.6160	03.9776	14.0671
	4	15	16.8011	14.4032	06.6293	07.2779	08.9419	24.9958

Table 2: Shows the calculated χ^2 values for serial test for two different large sized binary sequences having N=10000 and 15000 corresponding to five different sets of parameter values. The parameters αi, x_i and y_i are respectively the system parameter (same for both maps), initial conditions for the first and second maps. The values of the parameters are same as given in the caption of Table 1.

Size	Parameter	Calculated χ^2 value	Critical χ^2 value at α=0.005
N=10000	(α_1, x_1, y_1)	3.21234	5.9915
	(α_2, x_2, y_2)	2.10252	
	(α_3, x_3, y_3)	0.83838	
	(α_4, x_4, y_4)	3.28073	
	(α_5, x_5, y_5)	1.82412	
N=15000	(α_1, x_1, y_1)	2.09876	5.9915
	(α_2, x_2, y_2)	3.50070	
	(α_3, x_3, y_3)	0.44396	
	(α_4, x_4, y_4)	2.83491	
	(α_5, x_5, y_5)	1.55052	

Table 3: Shows the calculated Z-values for autocorrelation test for two different large sized binary sequences having N=10000 and 15000 corresponding to five different sets of parameter values. The parameters α_i, x_i and y_i are respectively the system parameter (same for both maps), initial conditions for the first and second maps. The values of the parameters are same as given in the caption of Table 1.

Size	Parameter	Calculated Z value
N=10000	(α_1, x_1, y_1)	-0.56034
	(α_2, x_2, y_2)	-0.49061
	(α_3, x_3, y_3)	1.49187
	(α_4, x_4, y_4)	0.71089
	(α_5, x_5, y_5)	-0.99124
N=15000	(α_1, x_1, y_1)	-0.09834
	(α_2, x_2, y_2)	-0.05720
	(α_3, x_3, y_3)	0.95610
	(α_4, x_4, y_4)	0.64557
	(α_5, x_5, y_5)	-0.22064

0.50, the distribution of the binary sequences, generated by the CCCBG, are uniform and random. Beyond this range of system parameter, the binary sequences may fail in one or more of the statistical tests described above. So for the value of α is between 0.49 and 0.50 and initial condition for both cross-coupled chaotic tent maps in the range [0,1], the CCCBG generates uniform and random binary sequences. We have done the calculation in double precision floating point numbers.

In addition to the statistical tests discussed above, the most stringent randomness tests, namely the NIST suite tests (issued by the National Institute of Standards and Technology, special publication 800-22) have also been performed to evaluate the randomness of arbitrarily long binary sequences produced by the proposed CCCBG. The NIST statistical tests suite (which can be freely downloaded from website http://csrc.nist.gov/rng/) for random sequences offers a battery of sixteen statistical tests. These tests assess the presence of a pattern which, if detected, would indicate that the sequence is non-random. The properties of a random sequence can be described in terms of probability. In each test a probability, called the P-value, is extracted. This value summarizes the strength of the evidence against the perfect randomness hypothesis. A P-value larger than 0.01, means that the sequence is considered to be random with a confidence of 99%. The NIST suite tests were performed on five binary sequences, each containing 15000 bits. The P-value as well as final results obtained from the NIST suite for five different sets are given in Table 5. The CCCBG successfully passes all randomness tests of NIST suite.

Table 5: Shows the P-values obtained from NIST suite for fourteen different tests. The P-values are obtained for five different sets of parameters for each test. The parameters α_i, x_i and y_i are respectively the system parameter (same for both maps), initial conditions for the first and second maps. The values of the parameters are same as given in the caption of Table 1.

Test Name	**P-values**					**Conclusion**
	(α_1, x_1, y_1)	(α_2, x_2, y_2)	(α_3, x_3, y_3)	(α_4, x_4, y_4)	$(\alpha_5, x5, y_5)$	
Approximate Entropy Test	0.113169	0.110449	0.032330	0.605333	0.080288	Success
Frequency Test within Block	0.571881	0.174253	0.736170	0.480171	0.734041	Success
Cumulative (forward) Sum Test	0.355713	0.360988	0.998247	0.261811	0.465759	Success
Cumulative (reverse) Sum Test	0.850139	0.405249	0.991189	0.112111	0.360988	Success
Discrete Fourier Transform Test	0.524923	0.915612	0.111961	0.791082	0.185326	Success
Frequency Test	0.556614	0.947919	0.947919	0.185927	0.273909	Success
Lempel-Ziv Compression Test	1.000000	1.000000	1.000000	1.000000	1.000000	Success
Linear Complexity Test	0.274193	0.485289	0.013392	0.438106	0.694499	Success
Longest Runs of ones in a Block Test	0.706404	0.706404	0.706404	0.031775	0.295889	Success
Non-overlapping Template Matching Test	Success	Success	Success	Success	Success	Success
Overlapping Template Matching Test	0.048349	0.932964	0.570019	0.622580	0.138687	Success
Rank Test	0.994872	0.473711	0.013928	0.187368	0.100749	Success
Run Test	0.008225f	0.626630	0.503137	0.296859	0.560992	Success
Serial Test P1 P2	0.532974 0.658087	0.191867 0.144224	0.236357 0.185718	0.291859 0.018966	0.847115 0.484094	Success

5 Conclusions

We have proposed a new binary sequence generator, called cross-coupled chaotic random bit generator (CCCBG), which exploits the interesting properties of a skew tent map. By using the cross coupling, we forcefully change the behavior of both the chaotic maps regularly. Hence by knowing the system parameter and initial condition of one of the chaotic map, one would not be able to identify the behavior of the CCCBG. The initial condition and system parameter for tent maps can also be generated by using the external secret key [12, 13, 14]. To evaluate the randomness and uniformity, we have employed four different statistical tests i.e. frequency test, Poker test, auto-correlation test and serial test on several large sized binary sequences, generated by the CCCBG. These binary sequences pass all four tests successfully. Further, the most stringent tests of randomness, the NIST suite tests have also been performed to evaluate the randomness of the bit streams generated by the CCCBG. The CCCBG successfully passes all the randomness tests of NIST suite. We suggest the use of the random binary sequences generated by the proposed CCCBG to design new secure cryptosystems.

References

[1] M. Andrecut, "Logistic map as a random number generator," *International Journal of Modern Physics B*, vol. 12, no. 9, pp. 921-930, 1998.

[2] S. M. Fu, Z. Y. Chen, and Y. A. Zho, "Chaos-based random number generators," *Computer Research and Development*, vol. 41, no. 4, pp. 749-754, 2004.

[3] J. A. Gonzailez, and R. Pino, "Random number generator based on unpredictable chaotic functions," *Computer Physics Communications*, vol. 120, no. 2-3, pp. , 109-114, 1999.

[4] L. Huaping, S. Wang, and H. Gang, "Pseudo-random number generator based on coupled map lattices," *International Journal of Modern Physics B*, 18(17-19), 2409-2414, 2004.

[5] L. Kocarev, and G. Jakimoski, "Pseudorandom bits generated by chaotic maps," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 50, no. 1, pp. 123-126, 2003.

[6] V. V. Kolesov, R. V. Belyaev, and G. M. Voronov, "A Digital random-number generator based on the chaotic signal algorithm," *Journal of Communications Technology and Electronics*, vol. 46, no. 11, pp. 1258-1263, 2001.

[7] X. M. Li, H. B. Shen, and X. L. Yan, "Characteristic analysis of a chaotic random number generator using piece-wise-linear map," *Journal of Electronics and Information Technology*, vol. 27, no. 6, pp. 874-878, 2005.

[8] S. Li, X. Mou and Y. Cai, "Pseudo random bit generator based on couple chaotic systems and its application in stream cipher cryptography," *Progress in cryptologyXIndocrypt '01*, LNCS 2247, pp. 316-329, Springer-Verlag, Berlin, 2001.

[9] T. Lin, and L. O. Chua, "New class of pseudo-random number generator based on chaos in digital filters," *International Journal of Circuit Theory and Applications*, vol. 21, no. 5, pp. 473-480, 1993.

[10] J. Liu, "Design of a chaotic random sequence and its application," *Computer Engineering*, vol. 31, no. 18, pp. 150-152, 2005.

[11] A. J. Menezes, P. C. V. Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, Boca Raton, Forida, USA, 1997.

[12] N. K. Pareek, V. Patidar, and K. K. Sud, "Discrete chaotic cryptography using external key," *Physics Letters A*, vol. 309, pp. 75-82, 2003.

[13] N. K. Pareek, V. Patidar, and K. K. Sud, "Cryptography using multiple one-dimensional chaotic maps," *Communication in Nonlinear Science and Numerical Simulation*, vol. 10, pp. 715-723, 2005.

[14] N .K. Pareek, V. Patidar, and K. K. Sud, "Image encryption using chaotic logistic map," *Image and Vision Computing*, vol. 24, pp. 926-934, 2006.

[15] G. S. Sandhu and S. Berber, "Theoretical model, simulation results and performances of a secure Chaos-based multi-user communication system," *International Journal of Network Security*, vol. 8, no. 1, pp. 25-30, 2009.

[16] O. Shin'ichi and I. Hajime, "Pseudo-random number generators and chaos," *Transactions of the Institute of Electronics and Communication Engineers of Japan*, vol. E-65, no. 9, pp. 534-541, 1982.

[17] T. Stojanovski, and L. Kocarev, "Chaos-based random number generators - Part I: Analysis," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 48, no. 3, pp. 281-288, 2001.

[18] T. Stojanovski, J. Pihl, and L. Kocarev, "Chaos-based random number generators - Part II: Practical realization," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 48, no. 3, pp. 382-385, 2001.

[19] L. Wang, F. P. Wang, and Z. J. Wang, "Novel chaos-based pseudo-random number generator," *Acta Physica Sinica*, vol. 55, no. 8, pp. 3964-3968, 2006.

[20] Y. Wang, H. Shen, and X. Yan, "Design of a chaotic random number generator," *Chinese Journal of Semiconductors*, vol. 26, no. 12, pp. 2433-2439, 2005.

N. K. Pareek received his M.Sc. from University of Rajasthan, Jaipur, India in 1986 and Ph.D. degree in Computer Science from M L Sukhadia University, Udaipur, India in 2005. Presently, he is working as a Programmer at the University Computer Centre of the M L Sukhadia University, Udaipur and has taught various courses of computer science to undergraduate and post graduate students. His research interest is in the field of information security, chaotic cryptology, data compression, and information retrieval systems. He has published one book and eight papers on chaotic cryptography and image encryption.

Vinod Patidar is currently an Assistant Professor of Physics at Banasthali University, Banasthali, India. He has received his M.Sc. (Physics) and Ph.D. (Nonlinear Dynamics) from M L Sukhaida University, Udaipur, India in 1999 and 2005 respectively. His research interest includes bifurcation & chaos in classical systems, control & synchronization of chaos, and application of chaotic dynamical systems in cryptography.

K. K. Sud (Ph.D., Ohio University) is a Senior Professor and Head, Basic Sciences at Sir Padampat Singhania University, Udaipur. He is a former professor and Director of the University Computer Centre and PG programme in computer application of the M L Sukhadia University, Udaipur. His research interest includes development and cryptanalysis of chaotic cryptosystems, image encryption and compression, radiation physics and electron-atom collision problems. He has published two books and over fifty research papers in the field of interest in international journals. He is also a senior member of the Computer Society of India.