MPRI-JSD | INTERNATIONAL JOURNAL OF AFRICAN SUSTAINABLE DEVELOPMENT
VOL. 10 NO.2 DEC-2019 ISSN: 2760-4106

CHAOS BASED PSEUDORANDOM SEQUENCES: CRYPTOSYSTEMS' SECURITY PERSPECTIVE.

[1]RAKIYA MK ADAMU, [2]ALIYU DANLADI HINA, [3]ISA YAHAYA
Department of Mathematics & Statistics, Federal Polytechnic, Bauchi.

Abstract

The generation of pseudo-random numbers (bits) plays a critical role in a large number of applications such as statistical mechanics, numerical simulations, gaming industry, communication or cryptography.The choice of secret keys for cryptographic primitives largely depends on the quality of random numbers used. These random numbers are fundamental tools in the generation of secret keys and initialization variables of encryption for cryptographic application, masking protocols, or for internet gambling. Chaotic Pseudorandom numbers were found to be very efficient in this aspect. The relevance of chaotic pseudorandom sequences in ensuring security in cryptosystems is considered, at the same time reviewing statistical tests required to make such sequences cryptographically secure. This paper intends to review the development of chaotic pseudorandom number generators through the years and the statistical tests they are required to pass as a measure of their randomness.

KEYWORDS:
Chaos, deterministic algorithm, Non-linear map, Pseudorandom, statistical test.

Introduction

Probably one of the earliest applications of chaos came from the observation of its natural pseudorandomness, either as a sampled form of continuous chaos, or straight from the appropriate nonlinear map.[1]

A pseudorandom sequence generator is a device or a deterministic algorithm that generates a long sequence of bits that are statistically independent and unbiased, upon the receipt of a fairly short sequence of input called the seed, Thesequence generated must be of sufficient size and be "random". The probability of any particular value being selected must be sufficiently small, this id to preclude an

adversary from gaining advantage through optimizing a search strategy based on such probability.

Deterministic in the sense that the sequence appears random even though a careful observation of a reasonable number of the outputs reveals its pattern, hence the name "pseudorandom". The output of such an algorithm is referred to as a pseudorandom sequence.The output of a pseudorandom numbers generator could be made to be numbers or digits, referred to as pseudorandom numbers or bits.

We take "randomness" in this sense to mean that a sequence of pseudorandom numbers should have the same probability of passing a "statistical test"as truly random numbers would have, (Not better!). A statistical test may be based on the value of any function of the sequence of pseudorandom numbers. It is sufficient that the expected distribution of that value be known (or calculable numerically) for a truly random distribution, then by considering the valueof the function for the given pseudorandom sequence, compared with the known expected distribution ofthat value for truly random numbers, one obtains a confidence level for the test. If many tests are applied and the confidence levels are calculated correctly, and if the tests are independent, the confidence levels should be uniformly distributed between zero and one if the pseudorandom generator is "good". The formal difficulty arises mostly from the fact that the number of possible tests is uncountably infinite, and in addition they are of course not all independent.The pseudorandom Number generator is cryptographically secure if, given the mapping that defines the generator and an arbitrary sequence of numbers generated by the generator, but not knowing the seed of the generator, it is hard to compute the next and the previous numbers in the sequence. [2].

Suitable metrics are needed to investigate the degree of randomness for number (binary) sequences produced by random number generators (RNGs) for cryptographic purposes. Today, researchers are developing new hardware and software based RNGs. However, few standards address statistical analysis techniques that should be employed in practice. [5] describes several empirical tests which include the: frequency, serial, gap, poker, coupon collector's, permutation, run, maximum-of-t, collision, birthday spacing, and serial correlation. Researchers at the Information Security Research Centre of Queensland University of Technology in Australia, developed a suit called the Crypt-XS suite of statistical tests. Crypt- XS tests include the frequency, binary derivative, change point, runs, sequence complexity and linear complexity tests.

INTERNATIONAL JOURNAL – ASD
VOL. 10 NO.2 DEC-2019 ISSN: 2760-4106

The National Institute of Standards and Technology (NIST) came up with a Statistical tests package which includes tests like the: frequency, block frequency, cumulative sums, runs, long runs, Marsaglia's rank, spectral (based on the Discrete Fourier Transform), nonoverlapping template matches, overlapping template matches, Maurer's universal statistical, approximate entropy (based on the work of Pincus, Singer and Kalman), random excursions (due to Baron and Rukhin), Lempel-Ziv complexity, linear complexity, and serial.

A more detailed description for those tests can be found in [3]. Where as a brief introduction of the most frequently used tests contained in the NIST suit is give viz:.

Frequency (Monobit) Test is to determine whether the number of ones and zeros in a sequence are approximately the same as would be expected for a truly random sequence. Frequency Test within a Block is to determine whether the frequency of ones in an M-bits block is approximately M/2, as would be expected under an assumption of randomness (M is the length of each block). **Runs Test** is to determine whether the number of runs of ones and zeros of various lengths is as expected for a random sequence. In particular, this test determines whether the oscillation between such zeros and ones is too fast or too slow. Test for the Longest Run of Ones in a Block is to determine whether the length of the longest run of ones within the tested sequence is consistent with the length of the longest run of ones that would be expected in a random sequence. **BinaryMatrix Rank Test** is to check for linear dependence among fixed length substrings of the original sequence.

Discrete Fourier Transform (Spectral) Test is to detect periodic features (i.e., repetitive patterns that are near each other) in the tested sequence that would indicate a deviation from the assumption of randomness.

Non-overlapping Template Matching Test is to detect generators that produce too many occurrences of a given non- periodic (aperiodic) pattern. Overlapping Template Matching Test is the number of occurrences of pre-specified target strings. **Maurer's "Universal Statistical" Test** is to detect whether or not the sequence can be significantly compressed without loss of information.

Linear Complexity Test is to determine whether or not the sequence is complex enough to be considered random. Serial Test is to determine whether the number of occurrences of the 2m m-bit (m is the length in bits of each block) overlapping patterns is approximately the same as would be expected for a random sequence.

INTERNATIONAL JOURNAL – ASD
VOL. 10 NO.2 DEC-2019 ISSN: 2760-4106

Approximate Entropy Test is to compare the frequency of overlapping blocks of two consecutive/adjacent lengths (m and m+1) against the expected result for a random sequence (m is the length of each block). **Cumulative Sums (Cusum) Test** is to determine whether the cumulative sum of the partial sequences occurring in the tested sequence is too large or too small relative to the expected behavior of that cumulative sum for random sequences. **Random Excursions Test** is to determine if the number of visits to a particular state within a cycle deviates from what one would expect for a random sequence. Random Excursions Variant Test is to detect deviations from the expected number of visits to various states in the random walk.

Over the years, [4] considerable experience has indicated what kinds of tests are likely to find the weaknesses of typical generators, and modern tests are much more stringent than most of the older ones. Modem generators are expected to pass all the old tests as well as those tests which traditional generators are known to fail. Probably the most extensive presentation of pseudorandom number testing is given by Knuth [5], but should be updated by the more severe tests, who suggests that any pseudorandom generator likely to have a "lattice structure" should be subjected to the "spectral test" among other tests.

Chaotic systems are widely reported in the literature for use as pseudorandom number generators have proposed a pseudorandom number generator based on the Chen chaotic system. The advantage of the proposed algorithm compared to others is that the generated pseudorandom sequence shows a uniform distribution. Security analysis of the proposed generator was carried out using a variety of statistical tests. [6]

For a cryptographic system, having satisfactory statistical properties is one of the necessary conditions in order to achieve security of the system, but it is not sufficient by itself. In this study, the security of the proposed generator is discussed from a different perspective.

A minimum security requirement for a pseudorandom bit generator is that the length k of the random seed should be sufficiently large so that a search over 2^k elements (the total number of possible seeds) is infeasible for the adversary.

A pseudorandom number generator is said to pass all polynomial-time statistical tests if no polynomial-time algorithm can correctly distinguish between an output sequence of the generator and a truly random sequence of the same length with probability significantly greater than the probability of flipping a coin [7].

INTERNATIONAL JOURNAL – ASD
VOL. 10 NO.2 DEC-2019 ISSN: 2760-4106

While it is impossible to give a mathematical proof that a generator is indeed a random bit generator, a test to detect certain kinds of weaknesses of the generator must be conducted. This is accomplished by taking a sample output sequence of the generator and subjecting it to various statistical tests. Each statistical test determines whether the sequence possesses a certain attribute that a truly random sequence would be likely to exhibit; the conclusion of each test is not definite, but rather probabilistic.[7]

Pseudorandom Number (Bits) Generators (PRNGs)

The need for random and pseudorandom numbers arises in many cryptographic applications. For example, common cryptosystems employ keys that must be generated in a random fashion.

The nature of randomness has attracted an increasing amount of interest in recent years. Many [8] applications require random input. Sources of random numbers can be broadly divided into two classes.

Pseudorandom number generators (PRNGs) and thetrue random number generators (TRNGs). The primary difference between random and pseudorandom numbers is that pseudorandom numbers are necessarily periodic derived from deterministic algorithms, whereas truly random numbers are not periodic and are derived from truly random sources.

A pseudo-random number generator is a deterministic method, usually described with a mapping, to produce from a small set of "random" number(s), called the seed, a larger set of random- looking numbers called pseudorandom numbers. Such mappings are preferred to be one-way. Several researches have been conducted using various kinds of chaotic mappings ranging from one dimensional two and three dimensional mappings. A number of authors considered combining to kinds of mappings so as to remedy the shortcoming of stability points of some mapings, such regions where the function is not chaotic.

In 1986, [9] two pseudorandom number generators, The 1/p generator and the x^2(mod N) generators were considered. The later was found to be unpredictable even though it was earlier thought to be weak and inefficient by researchers, a surprising development! the security of these generators were based on the assumed intractability of some number theoretic problems by probabilistic polynomial time procedures. However, the current standard in cryptographically secure random bits is the Blum BlumShub (BBS) algorithm [8]. The security of the BBS algorithm is based on the difficulty of factoring prime numbers. [10]proposed

in their paper an appropriate way to generate a cryptographically secured pseudo random sequence from a chaotic system. With this new scheme the Chua's system shows better chaotic performance by inheriting the high sensitivity to the initial conditions and expanding the range of parameters. In addition, the generated sequence passes all the NIST statistical tests which confirm its effectiveness for cryptographic issues.

Shannon in his classic 1949 first mathematical paper on Cryptography, proposed chaotic maps as models mechanisms for symmetric key encryption, before the development of Chaos Theory. Chaotic maps are simple unstable dynamical systems with high sensitivity to initial conditions. Small deviations in the initial conditions (due to approximations or numerical calculations) lead to large deviations of the corresponding orbits, rendering the long-term forecast for the chaotic systems intractable [11].

Chaotic PRNGs

For about two decades now, a lot of research has been ongoing in the area of chaotic cryptography particularly chaotic pseudorandom number generators [12], [13], [14], [15], [16], [17]. Simultaneously, many cryptanalytic researchers have analyzed the proposed chaos-based cryptographic algorithms and found that some of them are not secure enough and/or are slow algorithms[12]. Therefore, the main challenge in this research is to look at the relevance/importance of pseudorandom numbers in the design of secure and fast chaos- based cryptographic algorithms.

Since the advent of research in the chaotic systems, a lot of researchers have become interested into the use of chaotic maps to generate pseudorandom numbers using the unpredictable nature of chaotic systems. Chaotic systems like piecewise non-linear chaotic maps, Logistic map, Tent map, the Henon attractor and many more were used. Some proposals were based on multiple chaotic maps to enhance the PRNG security by some researchers.[13]we intend to review chaotic pseudorandom number generators that are cryptographically secure.

The first paper on pseudorandom number generators is due to Wolfram in 1985, where he used cellular automata to design a stream cipher algorithm [13]. The ciphertext is produced by XORing the plaintext with the random bits generated from the cellular automata.

Matthews in 1989 substituted the used of pads with random sequence generated from chaotic functions as system keys in the design of a chaos based stream cipher

INTERNATIONAL JOURNAL – ASD
VOL. 10 NO.2 DEC-2019 ISSN: 2760-4106

algorithm [14].Chaotic credentials of a tent map were utilized by E. alvarez,[15]in a symmetric block cipher to generate a pseudorandom number from its orbits using a certain threshold. A ciphertext is produced as the information on the position of the plaintext in the generated sequence. G Alvarez cryptanalyse E. Alvarez's scheme with four methods among other weaknesses.

The use of coupled chaotic systems to generate pseudorandom sequences was proposed by Shujunaet. al. in 2001.The coupled two chaotic systems to generate pseudorandom binary sequences which was claimed to have higher security than the individual maps[16]. Shortly afterwards, [17] used the trajectories of two logistic maps that are close to each other to generate pseudorandom sequences of high complexity.The cipher XORs the plaintext with the generated sequence to generate the ciphertext.

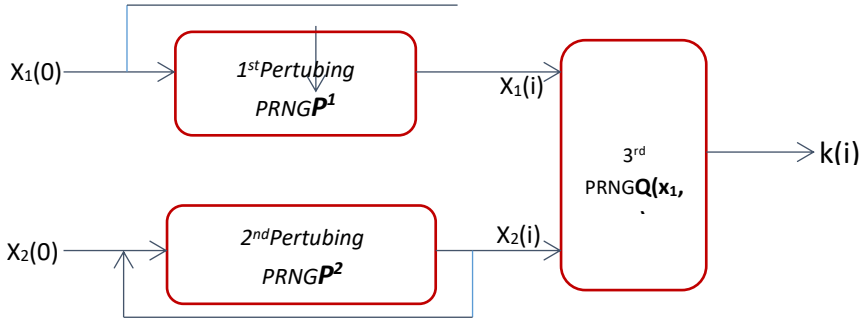

Figure 1.An illustration of coupling Chaotic maps to generate pseudorandom numbers.

In 2003 Lee et. al.[18] considered the composition of multiple chaotic maps to design a chaotic stream cipher. The scheme generates pseudorandom byte sequences and a two dimensional chaotic map is used to permute the generated byte sequence. Thus far, there have been no successful attacks on this algorithm. In 2005, another research group proposed a pseudorandom number generator derived from a discrete chaotic map that is defined over a long interval [19].

Wang et al. used an n-dimensional non-linear digital filter (n-NDF) (to improve randomness and security) and a chaotic system to design a proposed pseudorandom binary generator [20]. The authors used n-NDF_1 and n-NDF_2 as transition functions defined by:

$$y_1(k+1) = z_{11}(k+1) = F_1(z_1(k), \varphi_1, c_1)$$
$$y_2(k+1) = z_{21}(k+1) = F_2(z_2(k), \varphi_2, c_2)$$

With $y_1(k) and\ y_2(k)$ as the output of the two n-NDFs at step k. (Details in [18]).

INTERNATIONAL JOURNAL – ASD
VOL. 10 NO.2 DEC-2019 ISSN: 2760-4106

A cipher encryption algorithm based on the combination of the XOR operation and the logistic map was proposed by Xiang. This scheme is considered as an improvements to the Baptista's algorithm, a combination of XOR and circular bit shift was used in the encryption and the decryption processes[21]. Yu and Cao [22] modified Xiangs scheme by replacing the Logistic map with a chaotic neural network time varying delay.

Yu and Cao's scheme was cryptanalyse by Li et. al. in 2007 by showing that the pseudorandom number generator upon which the security of the system is based, does not have sufficient randomness and is not uniformly distributed.In the same year, high dimensional cat and tent maps were used to generate a pseudorandom key stream with stream cipher architecture for a chaotic image encryption. This scheme is not known to have successfully been broken till now.

The dynamics of chaotic system-based synchronization to generate a pseudorandom sequence as a keystream based on the value of the secret key is utilized. The plaintext is encrypted using the symbolic dynamics of the logistic map or tent map with certain values of its parameters and initial conditions[23]. A research group analyzed the proposed stream cipher encryption scheme In 2011 [24]. They were able to deduce and estimate chaotic systems' parameters with low error rate, and pointed out that a tent map is not a good source for a pseudorandom number generator and that the logistic map key stream has to be generated from a positive Lyapunov exponent.

Intermediate chaotic keystreams are generated based on a logistic map and chaotic standard map to provide high confusion and diffusion properties. This novel scheme was proposed by Patidar et. al. [25] with a mixing operation. they modified the scheme after it was cryptanalyse by Rhouma e. al in 2010 with only a pair of plaintext and ciphertext [26]. In 2011 the modified version was analyzed by Lie et. al and found it not to be secure against known plaintext and known ciphertext attacks. The weakness is based on the generated Logistic map sequence which is weak and non-random.

Since the Tent map is one of the equations that produce pseudorandom numbers that have no stability island [27], The noise function used in the proposed cryptosystem is an approximation to the chaotic tent map, and it is called pseudo chaotic tent map (PCT map)it was used as a chaotic map to generate pseudorandom sequence of bits to be used in an encryption algorithm. PCT with

sub-block size of 16-bits (PCT-16) has better statistical distribution than that of 8-bits sub-block size (PCT-8).

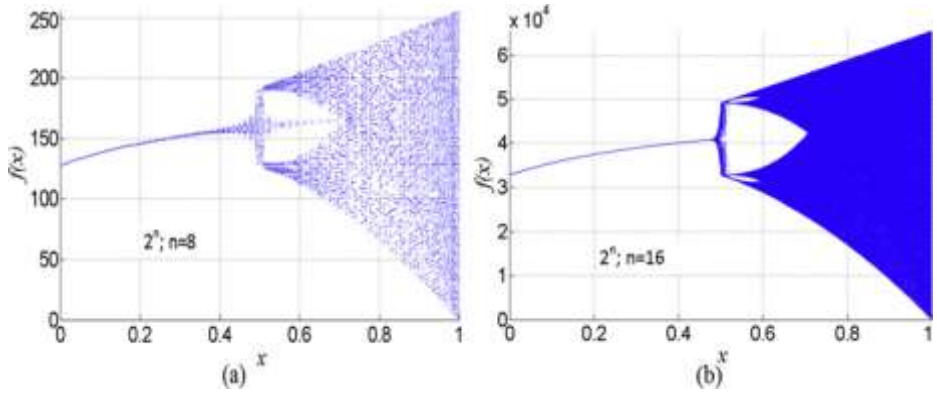

Figure 3.A typical bifurcation diagram for a tent map with varying n.

The encryption process consists of r rounds of PCT-16 map using encryption key.The result of the proposed chaotic pseudorandom number generator was tested using NIST statistical test suite, and it confirmed its randomness by passing all the tests.

A pseudorandom bit generator was proposed using the combination of three standard chaotic maps which generates a 32 random bits at each iteration. The authors noted that the proposed generator has the advantage of high sensitivity to initial seeds, high randomness and resistance to several attacks in addition to speed of the algorithm[28]. They used the map with an initial seed belonging to]0,1[

$$X_{n+1} = 3.9999X_n(1 - X_n)$$

$with\ \lambda = 3.9999$ and all X_n belonging to]0,1[.

The three combined equations are givem by the following in the same algorithm.

$$X_{n+1} = 3.9999X_n(1 - X_n)$$
$$Y_{n+1} = 3.9999Y_n(1 - Y_n)$$
$$Z_{n+1} = 3.9999Z_n(1 - Z_n)$$

Fo each computed value of X_n, Y_n, Z_n binary64 floating point format is used.

The generation of pseudo-random numbers (bits) plays a critical role in a large number of applications such as statistical mechanics, numerical simulations, gaming industry, communication or cryptography.

Chaotic trajectories even look random, and, they pass many classic “tests” ofrandomness. This in fact generates the principle of equivalence between chaotic

and random systems.In there paper,[29], observed that:- chaotic and randomsystems are observationally indistinguishable, thus, one can replace a random system by an equivalent chaotic system, and vice versa, as has been argued in[30].

Pseudorandom number generators' (PRNGs) results are mainly used on stream cipher algorithms as key streams that simply XOR with plaintext to generate the correspondence ciphertexts using any mode of operation.

A PRNG using a standard chaotic function is proposed by [31]. The algorithm uses a degressive modulo to index progressively the positions of an initial vector, before permuting their associated elements through the use of a XOR operator. The chaotic permutations are achieved iteratively on the initial vector in order to produce three chaotic maps. These maps are xored and the resulting sequence is the output of the algorithm. This PRNGhas shown its ability to produce a very large number of pseudo-random sequences which can be useful in several cryptographic applications.

Chaotic functions (Tent and Logistic maps) [32] were used to generate pseudorandom numbers which are then converted to binary numbers to be used as random bits stream. These random bits are therefore used for image encryption by forming a random bits matrix.

Quantum chaos theory seems to be a tool that can be used to improve the quality of pseudorandom number generators. It helps in producing sequence at a speed that cannot be obtained with a true number generator [8]. They proposed a novel pseudo-random number generator based on the quantum chaotic map. A quantum map is the logistic map with additive noise that arises from the very lowest-order quantum corrections. The proposed scheme exploits the interesting properties of three-dimensional quantum logistic map such as statistical complexity. The three different statistical tests, NIST, DIEHARD and ENT test suites are employed to evaluate the randomness and uniformity of the sequence generated.

Conclusion

Chaos based Pseudo-Random number (Bit) Generators (PRBGs) is an algorithm that generates pseudorandom numbers through the use of chaotic maps. The dimension of the map to beused depends on application requirments of the schemes. So many generators were cryptanalysed almost as immediately as they were proposed. The quality of a pseudorandom number generator largely

depends on the choice of seed(s) and the control parameter(s) of the chaotic map(s) to be used. A good number of Pseudorandom Numbers Generators that were considered to be good for some purposes did not find a place with cryptographic applications, thus all random numbers generated must be subjected to a rigourous statistical tests using any of the industry standard statistical suits like NIST, NFIS Diehard Tests,TestU01etc.

Though chaotic cryptography may be considered at present peripheral in circles of conventional cryptography, chaotic number generation may have attractive applications as simulation engines in computational science (Pellicer-Lostao& Lopez-Ruiz, 2011c, 2011d). Chaos based number generators are easy to use and highly configurable. This makes them a valuable tool for the realization of effective and efficient cryptosystems.

References

[1] C. P. Silva and a. M. Young, "Introduction to chaos-based communications and signal processing," *2000 IEEE Aerosp. Conf. Proc. (Cat. No.00TH8484)*, vol. 1, pp. 279–299, 2000.

[2] L. Kocarev, "Cryptography : A Brief Overview," *Circuits Syst. Mag. IEEE*, vol. 1, no. 3, 2001.

[3] S. Juan, "Statistical Testing of Random Number Generators," *Available at http://csrc.nist.gov/rng/ rng5.html.*, 1999.

[4] F. James, "A review of pseudorandom number generators," *Comput. Phys. Commun.*, vol. 60, no. 3, pp. 329–344, Oct. 1990.

[5] Donald E. Knuth, *The Art of Computer Programming*, vol. 3. Addison-Wesley, 1973, p. 829.

[6] F. Özkaynak and S. Yavuz, "Security problems for a pseudorandom sequence generator based on the Chen chaotic system," *Comput. Phys. Commun.*, vol. 184, no. 9, pp. 2178–2181, Sep. 2013.

[7] A. J. Menezes, P. C. Van Oorschot, and S. A. Vanstone, *APPLIED CRYPTOGRAPHY*. 1996, p. 794.

[8] a. Akhshani, a. Akhavan, a. Mobaraki, S.-C. Lim, and Z. Hassan, "Pseudo random number generator based on quantum chaotic map," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 19, no. 1, pp. 101–111, Jan. 2014.

[9] L. Blum, "Pseudo-random number generator*," *SIAM J. Comput.*, vol. 15, no. 2, pp. 364–383, 1986.

[10] L. Merah, A. Ali-pacha, and N. H. Said, "A Pseudo Random Number Generator Based on the Chaotic System of Chua ' s Circuit , and its Real Time FPGA Implementation," vol. 7, no. 55, pp. 2719–2734, 2013.

[11] G. Makris and I. Antoniou, "Cryptography with Chaos," in *5th Chaotic Modelling and Simulation International Conference.*, 2012, no. June, pp. 12–15.

[12] M. M. Maqableh, "Analysis and Design Security Primitives Based on Chaotic Systems for eCommerce," Durham, 2001.

[13] S. Wolfram, "cryptography with cellular-automata," in *Advances in Cryptology–CRYPTO 1985 PROC.*, 1985.

[14] R. Matthews, "on the Derivation of a 'Chaotic' Encryption Algorithm," *Cryptologia*, vol. 13, no. 1, pp. 29–42, Jan. 1989.

[15] A. M. E. Alvarez, A Fernandez, P. Garcia, J. Jimenez, "New approach to chaotic encryption," *Phys. Lett. A*, vol. 263, no. December, pp. 373–375, 1999.

[16] L. Shujun, M. Xuanqin, and C. Yuanlong, "Pseudo-Random Bit Generator Based on Couple Chaotic Systems and its Applications in Stream-Cipher Cryptography," in *Progress in Cryptology:INDOCRYPT 2001, LNCS.*, 2001, vol. 2247, pp. 316–329.

[17] K. B. J. Ninan Sajeeth Philip, "Chaos for stream Cipher," in *proc. of Recent Advances in Computing and Communications.*, 2000, pp. 35–42.

[18] Y.-Y. C. Li, Poh-Han, Soo-Chang Pei, "Gnenerating Chaotic Stream Ciphers using chaotic Systems," *Chinese J. Phys.*, vol. 41, no. 6, 2003.

[19] T. Addabbo, M. Alioto, a. Fort, S. Rocchi, and V. Vignoli, "Long Period Pseudo Random Bit Generators Derived from a Discretized Chaotic Map," *2005 IEEE Int. Symp. Circuits Syst.*, pp. 892–895, 2005.

[20] W. Z. Xiamin Wang, Jianshu Zhang, Yongquan Fan, "Chaotic Pseudorandom Bit Generator Using n-dimensional Nonlinear Digital Filter," *Commun. Nonlinear Sci. Numer. Simulation,*, vol. 2, no. v, pp. 0–3, 2010.

[21] T. Xiang, X. Liao, G. Tang, Y. Chen, and K. Wong, "A novel block cryptosystem based on iterating a chaotic map," *Phys. Lett. A*, vol. 349, no. 1–4, pp. 109–115, Jan. 2006.

[22] W. Yu and J. Cao, "Cryptography based on delayed chaotic neural networks," *Phys. Lett. A*, vol. 356, no. 4–5, pp. 333–338, Aug. 2006.

[23] A. P. Kurian and S. Puthusserypady, "Self-synchronizing chaotic stream ciphers," *Signal Processing*, vol. 88, no. 10, pp. 2442–2452, Oct. 2008.

[24] D. Arroyo, G. Alvarez, and S. Li, "Cryptanalysis of a family of self-synchronizing chaotic stream ciphers," *Commun. Nonlinear Sci. Numer. Simulations.*, vol. 2, no. 16, pp. 805–813, 2011.

[25] V. Patidar, N. K. Pareek, and K. K. Sud, "A new substitution–diffusion based image cipher using chaotic standard and logistic maps," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 14, no. 7, pp. 3056–3075, Jul. 2009.

[26] R. Rhouma, E. Solak, and S. Belghith, "Cryptanalysis of a new substitution–diffusion based image cipher," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 15, no. 7, pp. 1887–1892, Jul. 2010.

[27] J. a. Martínez-Ñonthe, a. Castañeda-Solís, a. Díaz-Méndez, M. Cruz-Irisson, and R. Vázquez-Medina, "Chaotic block cryptosystem using high precision approaches to tent map," *Microelectron. Eng.*, vol. 90, pp. 168–172, Feb. 2012.

[28] M. Franc, "A Pseudo-Random Bit Generator Using Three Chaotic Logistic Maps," *Theory Appl. Model. Comput.*, pp. 229–247, 2013.

[29] M. Francois, T. Grosges, D. Barchiesi, and R. Erra, "A New Pseudo-Random Number Generator Based on Two Chaotic Maps," *INFORMATICA*, vol. 24, no. 2, pp. 181–197, 2013.

[30] C. Werndl and F. Philosophy, "Are Deterministic Descriptions And Indeterministic Descriptions Observationally Equivalent ?," no. 1996, pp. 1–33, 2009.

[31] M. François, T. Grosges, D. Barchiesi, and R. Erra, "Pseudo-random number generator based on mixing of three chaotic maps," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 19, no. 4, pp. 887–895, Apr. 2014.

[32] H. Khanzadi, M. Eshghi, and S. E. Borujeni, "Image Encryption Using Random Bit Sequence Based on Chaotic Maps," *Arab. J. Sci. Eng.*, vol. 39, no. 2, pp. 1039–1047, Sep. 2014.

[33] L. Merah, A. Ali-pacha, and N. H. Said, "A Pseudo Random Number Generator Based on the Chaotic System of Chua ' s Circuit , and its Real Time FPGA Implementation," vol. 7, no. 55, pp. 2719–2734, 2013.