Signal Processing 88 (2008) 2442–2452

Contents lists available at ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

Self-synchronizing chaotic stream ciphers

Ajeesh P. Kurian, Sadasivan Puthusserypady *

Department of Electrical and Computer Engineering, National University of Singapore, 4 Engineering Drive 3, Singapore 117576, Singapore

ARTICLE INFO

Article history:
Received 5 November 2007
Received in revised form
4 April 2008
Accepted 10 April 2008
Available online 18 April 2008

Keywords:
Chaotic systems
Synchronization
Symbolic dynamics
Secure communication
Stream ciphers

ABSTRACT

Chaotic communication schemes aim to provide security over the conventional communication schemes. The sensitivity of chaotic systems/maps to their initial conditions and the parameters is used to introduce the security, where the latter is used as the secret key. The applicability of conventional chaotic systems/maps in communication channels with significant noise and multi-path is limited. Symbolic dynamics (SD) based methods have been shown to provide high quality synchronization (HQS). In this paper, a new digital chaotic communication scheme, which utilizes the SD based synchronization, is proposed. This is similar to a self-synchronizing stream cipher where synchronization information is provided periodically. For the proposed scheme, a theoretical expression for the upper bound of the bit error rate (BER) is derived. Numerical simulations are carried out to assess the BER performances of the system in AWGN and multi-path channels. Some security aspects of the proposed system are also studied.

1. Introduction

Noise like appearance and broadband spectrum of the time series generated by chaotic systems/maps have attracted the researchers' interest in applying such systems for secure communication applications [1]. The sensitivity of the chaotic systems/maps to their initial conditions and control parameters are exploited in chaos based cryptography [2]. Although, the security aspects of the chaotic communication systems are not fully understood [3], it is generally believed that these class of systems can be used in applications which do not require a high level of security [4].

Application of chaotic systems/maps in secure communication can be classified broadly into two. In the first class of systems, the analog chaotic signals, which have noise like appearance, are used as the carrier of information [5–7]. Thus, the encryption and modulation are done at the same time. With a synchronized chaotic system/map at the receiver, the information is decoded. Chaotic masking, chaotic shift keying (CSK) and chaotic parameter modulations are examples of such chaotic communication schemes. The second approach is to use discrete chaotic maps for the encryption of the information signal [8]. These types of communication systems can again be classified into two subclasses. The first subclass is the chaotic stream ciphers, where the binary chaotic sequences generated from chaotic symbolic dynamics (SD) is XORed with the information signal in order to encrypt as in a conventional stream cipher [9]. At the receiver, a synchronized chaotic map is used to generate the same encryption key and the received signal is XORed to decrypt the message. The second subclass refers to the block encryption schemes using iterated chaotic maps [2,8,10].

As can be clearly seen, the synchronization between the transmitter and the receiver is a requirement for successful decryption of message at the receiver. Following the seminal paper by Pecora and Carrol [11], researchers have come up with a multitude of approaches for the synchronization of chaotic systems/maps. For a comprehensive survey on this topic, the reader may refer [12] and the references therein. It has been shown that intervals of desynchronization bursts can appear in synchronization when noise is present in the system

* Corresponding author. Tel.: +65 6516 2262; fax: +65 6779 1103.
E-mail address: elespk@nus.edu.sg (S. Puthusserypady).

0165-1684/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.sigpro.2008.04.003

[13]. A high quality synchronization (HQS) is said to have achieved when the transmitter and the receiver synchronizes with an error below a certain threshold [14]. HQS is ideal for setting up a reliable secure communication.

SD is the coarse–grain description of the chaotic dynamics and has been used for the analysis of chaotic systems/maps [15]. In [16], HQS is achieved using the SD based methods. Reformulation of the SD based synchronization from an information theoretic point of view is detailed in [17]. Recently, SD is being proposed for secure communication applications [18–21]. In [18], chaotic communication using the feedback of SD is proposed. Application of SD for the differential chaotic shift keying (DCSK) scheme is discussed in [19]. SD based noise reduction and coding is proposed in [20,21].

Dynamical degradation[1] is one of the main concerns when a stream cipher is implemented on the digital computer [22]. In this paper, a new self-synchronizing chaotic stream cipher is proposed using the SD based synchronization. In the proposed system, the synchronization information is provided periodically. The theoretical and numerical bit error rate (BER) performances for the new system are obtained. These results are compared with those of the binary phase shift keying (BPSK) and the CSK systems. Statistical tests are conducted to asses the security aspects of the proposed system. These test results show that the proposed system has good statistical properties to qualify as a random bit generator which in turn emphasizes the system security. The system's sensitivity to the changes in parameters is also studied.

This paper is organized as follows. A brief overview of SD and synchronization of chaotic maps using SD is given in Section 2. In Section 3, the proposed communication scheme is explained in detail. A theoretical expression for the upper bound of the BER is derived in this section. Numerical results are discussed in Section 4 and the paper is concluded with some remarks in Section 5.

2. Symbolic dynamics

SD is the coarse–grain description of the actual system dynamics [15]. It is being widely applied for the analysis of chaotic systems/maps. By partitioning a chaotic state-space to arbitrary regions, and labeling each region with a specific symbol, the trajectories can be translated to a sequence of symbols. This coarse–grain formulation of the system makes the deterministic nature of the dynamical system to a stochastic one. Hence such systems can be treated as Markov systems which have finite topological entropies.

Let the state-space ($\mathscr{S}$) of the iterated chaotic map[2] be partitioned to m disjoint regions, $\beta = \{\mathscr{C}\}_{i=1}^{m}$, such that $\mathscr{C}_i \cap \mathscr{C}_j = \emptyset$ for $i \neq j$ and $\cup_{i=1}^{m} \mathscr{C}_i = \mathscr{S}$. If one can assign a letter each to each of the disjoint regions, the dynamics of the system can be represented by a sequence of finite alphabet $\mathbf{X} = [X_1, \ldots, X_m]$. This sequence is called the SD of the system/map. The entropy of the new information source is given by

$$H_n^{\beta} = -\sum_{\mathbf{Y}_n} P(\mathbf{Y}_n^i) \log P(\mathbf{Y}_n^i), \tag{1}$$

where $P(\mathbf{Y}_n^i)$ is the probability to find a code word $\mathbf{Y}_n^i$ of length n. The superscript i in Eq. (1) represents a specific combination of symbolic sequence. The summation is taken over all such possible sequences. The source entropy of a dynamical system is

$$h^{\beta} = \lim_{n\to\infty} h_n^{\beta} = \lim_{n\to\infty} \frac{1}{n} H_n^{\beta}. \tag{2}$$

The Kolmogorov–Sinai entropy of the system is defined as

$$h_{\mathrm{KS}} = \sup_{\beta} h^{\beta}. \tag{3}$$

From the above discussions, it is clear that an iterated chaotic map is an information source with entropy h_{KS}. In the next subsections, the SD of two chaotic maps used for this study, namely the tent map and logistic map, are defined.

2.1. SD of the tent map

Tent map is a special case of *skewed tent map*. It is a piecewise linear 1-D map. The dynamics of the skewed tent map is given in [20],

$$x_{k+1} = \begin{cases} \dfrac{x_k}{A}, & 0 < x_k \leqslant A, \\ \dfrac{1 - x_k}{A}, & A \leqslant x_k < 1, \end{cases} \tag{4}$$

where A $(0 < A < 1)$ is the parameter which controls the skewness of the tent map. The case $A = 0.5$ is the normal tent map and the corresponding phase-space representation is shown in Fig. 1. Binary partition (i.e., assigning 0 or 1 to regions of x_k) for generating SD is also shown in Fig. 1. Here, symbol '0' is assigned if $0 < x_k \leqslant A$ and '1' is assigned if $A < x_k < 1$. In this way, the binary sequence for the entire trajectory can be obtained.

2.2. SD of the logistic map

Logistic map [23] is one of the most widely studied 1-D maps; the dynamics of which is governed by

$$x_{k+1} = \mu x_k (1 - x_k), \tag{5}$$

where μ is a constant. For a range of values of μ, logistic map has chaotic dynamics. The dynamics of logistic map is defined in $(0, 1)$. For $\mu = 4$, the SD of the map is given in Fig. 2. Symbol 0 is assigned to the region $0 < x_k \leqslant 0.5$ and 1 is assigned to the region $0.5 < x_k < 1$.

2.3. Synchronization using SD

Using the tent map with binary partition, the SD based synchronization is explained in [17]. Consider the chaotic map described by Eq. (4). Assume that there are no messages transmitted and there is no channel noise. For an initial condition x_0, let $\mathscr{X} = [x_0, \ldots, x_{m-1}]$ be the fiducial

[1] When chaotic maps are implemented in digital computers, eventually all the trajectories become periodic due to the finite precision computations.

[2] For chaotic systems, corresponding discrete-time map can be obtained by constructing the Poincare return map.

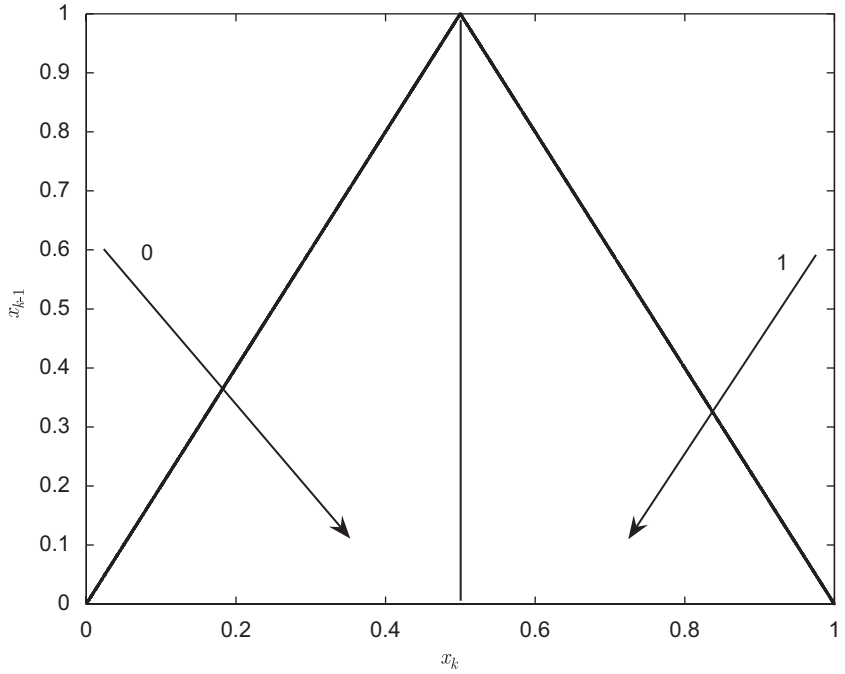

Fig. 1. Generating partition of the tent map.

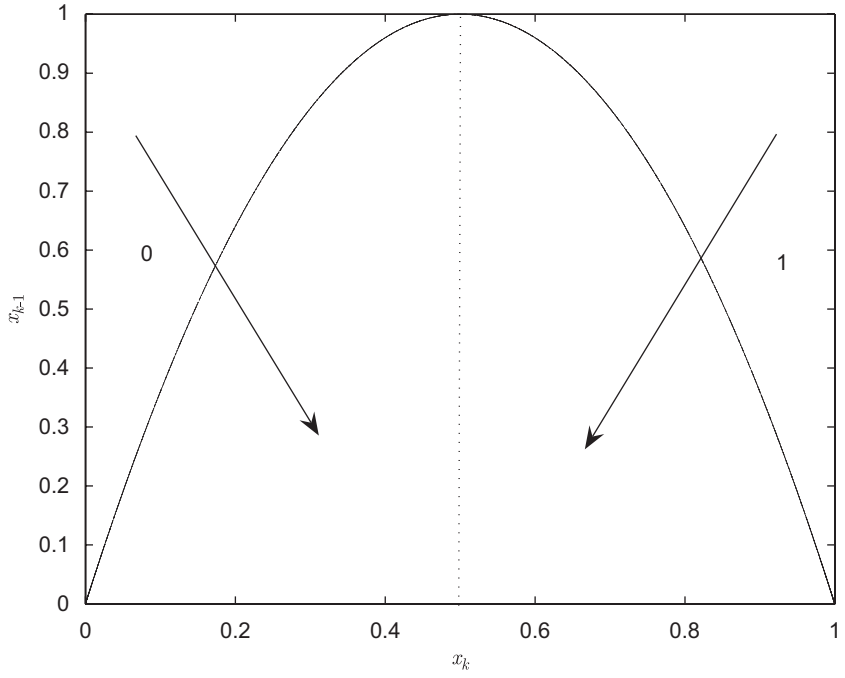

Fig. 2. Generating partition of the logistic map.

trajectory generated by Eq. (4). In this paper, a finite length trajectory is considered for simplicity. Let the corresponding binary trajectory be $\mathscr{X}_b$. As can be readily observed, $\mathscr{X}_b$ will be the bitstream that will be transmitted from the transmitter to the receiver. At the receiver, an exact copy of the same map is available but the initial condition is unknown. The task is, hence, to estimate the initial condition, x_0, using $\mathscr{X}_b$.

If $\mathscr{X}_b(0)$, the first element of $\mathscr{X}_b$ is available at the receiver and if it is '0', then the initial condition x_0 is between 0 and 0.5. If $\mathscr{X}_b(0) =$ '1', then x_0 is in between 0.5 and 1. By considering the next symbol of the sequence,

$\mathscr{X}_b(1)$, a more accurate estimate of the initial condition can be obtained by taking the pre-image of the map, i.e., if $\mathscr{X}_b(0)$ is '1' and $\mathscr{X}_b(2)$ is '0' then the initial condition must be between 0.25 and 0.5. If this procedure is continued, the initial condition can be estimated with an accuracy of the order of $1/2^m$, if m symbols are considered. This means that, with a finite precision computing, m has to be selected such that $1/2^m$ is less than the precision of computing. This in turn would imply $\|x_0 - \hat{x}_0\| = 0$, where $\hat{x}_0$ is the estimated initial condition at the receiver. This idea is used for the synchronization of chaotic maps.

3. Dynamic encoding

The schematic of the baseband representation of the proposed scheme is shown in Fig. 3. The system has identical chaotic maps at the transmitter and the receiver. At the transmitter, using the initial condition x_0, the chaotic time series of length N is generated. The time sequence is then converted to the corresponding symbolic sequence $\mathscr{X}_b$. The filter block suppresses out the first m bits for representing the initial condition. The remaining $N-m$ bits are used to code the binary information signal $\mathscr{B}$ of length $N-m$. Using the cryptographic terminology, let $\mathscr{B}(k)$ be the plain text (the information that the transmitter wants to send), $\mathscr{X}_b(m+k)$ be the key (sequence generated from the chaotic map) and $\mathscr{Y}_{msg}(k)$ (subscript msg means it contains the message) be the cypher text. The encryption can be done using the following operations:

$$\mathscr{Y}_{msg}(k) = \mathscr{B}(k) \oplus \mathscr{X}_b(m+k), \quad \text{for } k = 0, \ldots, N-1-m. \tag{6}$$

where, $\oplus$ is the XOR operation. The purpose of the shuffler block is to hide the bits conveying the initial condition $\mathscr{Y}_{init} = [\mathscr{X}_b(0), \ldots, \mathscr{X}_b(m-1)]$. The output is a binary sequence $\mathscr{Y}$ of length N. The format of the transmitted sequence is shown in Fig. 4. The resultant sequence can be transmitted using conventional digital communication techniques such as the BPSK or quadrature phase shift keying (QPSK).

At the receiver, signal corrupted by AWGN (ν) is available. Using conventional matched filter receiver, the transmitted sequence can be estimated as $\hat{\mathscr{Y}}$. The filter block at the receiver uses the apriori information about the positioning of $\mathscr{Y}_{msg}$ in the received sequence to separate $\hat{\mathscr{Y}}_{msg}$ and $\hat{\mathscr{Y}}_{init}$ from the output of the matched filter. Using the synchronization method described in Section 2.3, the synchronizer estimates the initial conditions, $\hat{x}_0$'s. These initial conditions are used to reconstruct the symbolic sequence $\hat{\mathscr{X}}_b$. The information signal is then retrieved using the following equation:

$$\hat{\mathscr{B}}(k) = \hat{\mathscr{Y}}_{msg}(k) \oplus \hat{\mathscr{X}}_b(m+k), \quad \text{for } k = 0, \ldots, N-1-m. \tag{7}$$

In order to decode these signals, the receiver should know three things—the chaotic map employed, the initial condition and the control parameter of the chaotic map. By selecting different initial conditions, different codes can be derived and thus multiple simultaneous block transmissions can be achieved using the same map. It is worth noting that chaotic maps are capable of generating i.i.d binary sequences [24] and hence the output sequence from the transmitter possess randomness.

3.1. Theoretical upper bound of the BER

It is clear from Fig. 4 that there are two possibilities for the bit error to occur: (i) the decoding information may be wrong thus causing a wrong estimation of the initial conditions or (ii) the detection of the message itself may be wrong due to the presence of noise in the message.

To decode the message completely, all the m bits (i.e., the bits representing the initial condition, x_0) should be detected correctly. Let the BER of the BPSK system, p_b, be

$$p_b = Q\left(\sqrt{\frac{2E_b}{N_0}}\right), \tag{8}$$

where N_0 is the noise power and E_b is the bit energy. Hence, the probability of wrongly detecting the

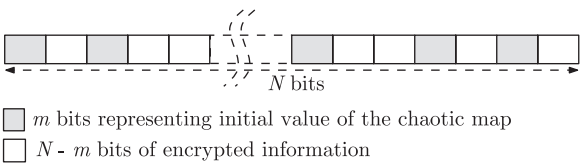

Fig. 4. Format of the transmission sequence with interleaved initial condition.

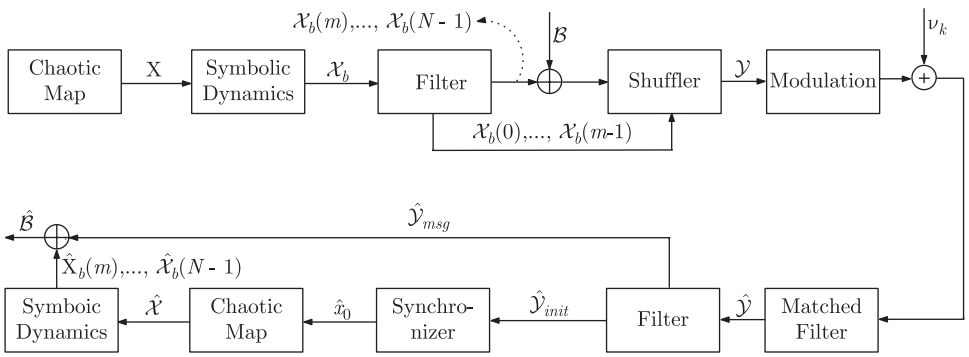

Fig. 3. Proposed communication system.

sequence, p_s, is

$$p_s = 1 - (1 - p_b)^m. \quad (9)$$

Here, it is assumed that the letters are equiprobable (i.e., the probability is 0.5). If a sequence is wrongly detected, the probability of wrong decision about the transmitted message, p_d, is given by

$$p_d = 0.5(1 - p_b). \quad (10)$$

The probability of error when the decoding information is wrong is given by

$$p_1 = p_d p_s = (1 - (1 - p_b)^m) p_d. \quad (11)$$

Considering the second situation, where the first m bits are decoded correctly and the message decoding is incorrect, the bit error probability is given by

$$p_2 = (1 - p_s) p_b. \quad (12)$$

Hence the total probability of error (BER) is given by

$$\begin{aligned} p &= p_1 + p_2 = (1 - (1 - p_b)^m) p_d + (1 - p_s) p_b \\ &= (1 - (1 - p_b)^m) 0.5(1 - p_b) + (1 - p_b)^m p_b \\ &= 0.5(1 - p_b) + (1 - p_b)^m (1.5 p_b - 0.5). \end{aligned} \quad (13)$$

When the SNR is high, p_b is close to zero and hence $(1 - p_b)^m \approx 1$. Then from Eq. (13), it can be clearly seen that the proposed system has a BER performance similar to that of the BPSK communication system.

4. Results and discussions

4.1. BER analysis

Extensive numerical simulations have been carried out to assess the performance of the proposed secure communication system. Tent map, skewed tent map and logistic map have been used for the generation of chaotic sequences. The number of bits that has been transmitted for each of the signal to noise ratio (SNR) values is 10^5 and the BER for each SNR has thus been calculated. The experiments have been carried out for simple AWGN and frequency selective channels. The values of N and m that have been used for the simulations are 1000 and 64, respectively.

The BER performance of the proposed system for the AWGN channel is presented in Fig. 5. Its performance is compared with that of the CSK, conventional stream cyphers (based on pseudo-random noise sequence) and conventional BPSK schemes. Since the performance of the conventional stream cyphers scheme is similar to that of BPSK, this method has not been considered for later analysis. As expected, at lower SNRs, the BER of the proposed scheme is relatively high. For instance, at SNR value of 4 dB, the proposed method has a BER of 0.31, while the corresponding values for CSK and BPSK are about 0.2 and 0.06, respectively. In order to estimate the initial condition accurately, all the m symbols should be detected correctly which is very unlikely for the cases of systems at lower SNR values. However, when the BPSK achieves a BER of 10^{-3}, the BER performance curve of the proposed system starts following that of the BPSK. For example, at 12 dB SNR, the proposed system has a BER of 4×10^{-4}. The corresponding BER values for BPSK and CSK systems are 10^{-4} and 8.2×10^{-3}, respectively. This trend is observed for all the maps used in the simulations. It is also interesting to note that even though the CSK based communication scheme has a slight performance advantage over the proposed system at low SNR regions, at high SNR values CSK systems do not have fast BER decay.

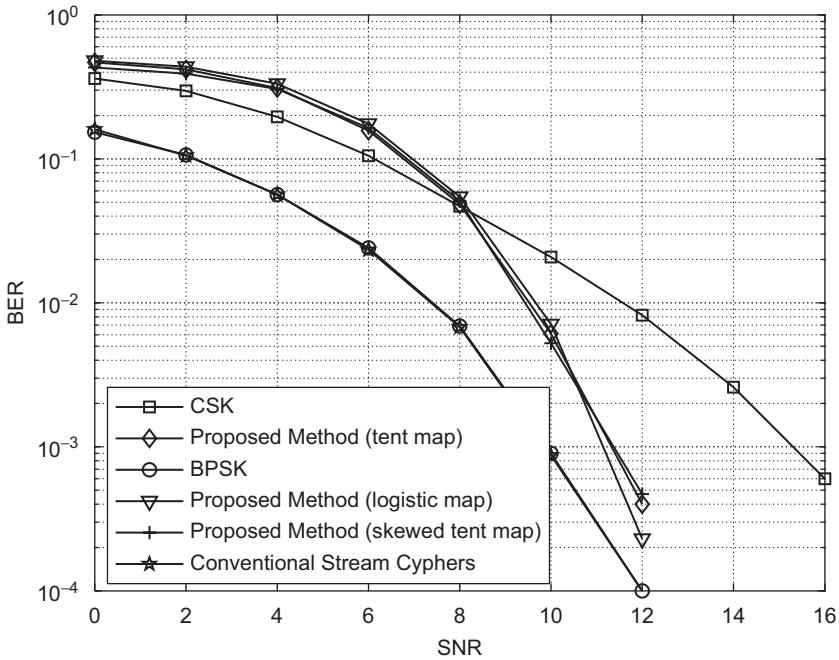

Fig. 5. BER performance under AWGN channel.

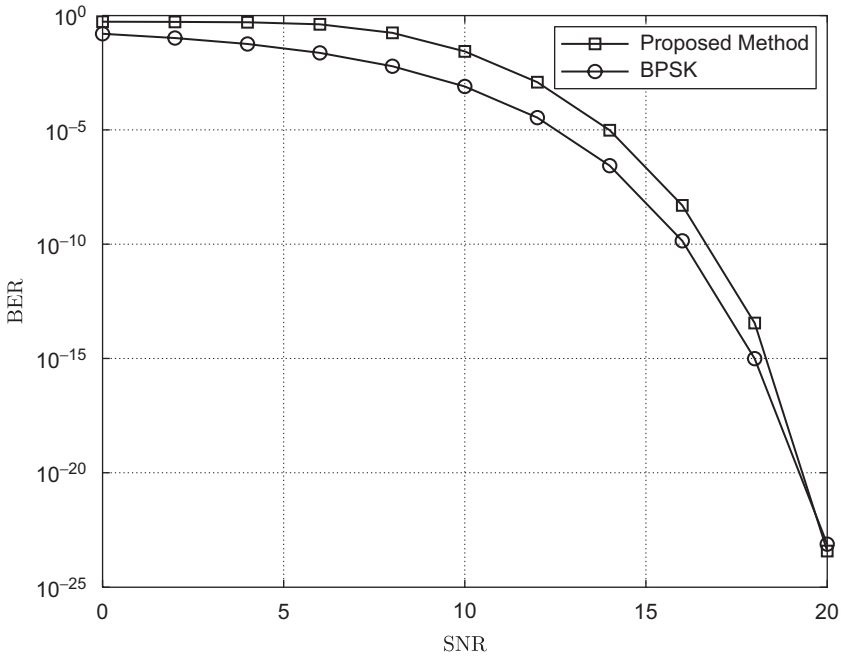

Fig. 6. Theoretical BER curves of BPSK and the proposed method (AWGN channel).

To compare the asymptotic BER performances, the upper bound of BER [derived in Eq. (13)] is plotted in Fig. 6 along with that of the BPSK system. It can be seen that when the SNR increases, the BER performance of the proposed method closely follows that of the BPSK system. Although such high SNRs are possible only in theory, it can be seen that the upper bound of BER of the proposed system can be brought to an arbitrarily small value by appropriate coding schemes.

To test the applicability of the proposed communication scheme to other maps, the experiment has been carried out for the skewed tent map and the logistic map. It can be observed that these maps also exhibit similar behavior. For low SNR cases, the BER is relatively high. As the SNR increases, the BER curve of the proposed system closely follows that of the BPSK system. As the inverse of the logistic map requires a square root, finite precision algorithms can introduce approximation errors. In this experiment, initial conditions have less precision compared with the computer and hence after computing the estimate of the initial condition at the receiver, the digits of higher precision are discarded.

Most of the communication channels encountered in practice are band-limited and frequency selective. To study the performance of the proposed system in such channels, another set of simulations have been carried out. Two different channel models discussed in [25, Chapter 10] have been considered for this study. The first one used is a three-ray channel model with tap weights $[0.474, 0.815, 0.474]$ and the second model is a five-ray channel with tap weights $[0.227, 0.460, 0.688, 0.460, 0.227]$. At the receiver end, the maximum likelihood sequence estimation is used to remove the inter-symbol interference caused by the channel. Simulation results are presented in Figs. 7 and 8. The proposed system behaves exactly as in the previous situation; at low SNRs it exhibits high BER and as the SNR increases, the BER curve closely follows that of the BPSK system. In the second channel condition as well, the proposed system has a fast BER decay which can be observed from Fig. 8. In all the simulations, irrespective of the map used, BER curves show similar characteristics. It is interesting to compare the overhead of this communication scheme. By overhead we mean the percentage of total bits that are not used for carrying information and is defined as

$$\text{overhead} = 100 \times \frac{m}{N}\%, \tag{14}$$

where N is the block length and m is the precision used to specify the initial condition. When we set $N = 1000$ and $m = 64$, the associated overhead is 6.5%. Definitely, this is much lower than that of the CSK scheme, but it is slightly higher than the other modern chaotic encryption systems [27] where the overhead is reported to be 1%. However, as per Eq. (14), the overhead in the proposed system can be varied by changing the values of N and m.

4.2. Security analysis

Here, the transmitter is first verified as source of the random bit sequence. Then a possible way to hide the initial condition is discussed. To study the security of the proposed scheme, its sensitivity to the control parameter variations is analyzed.

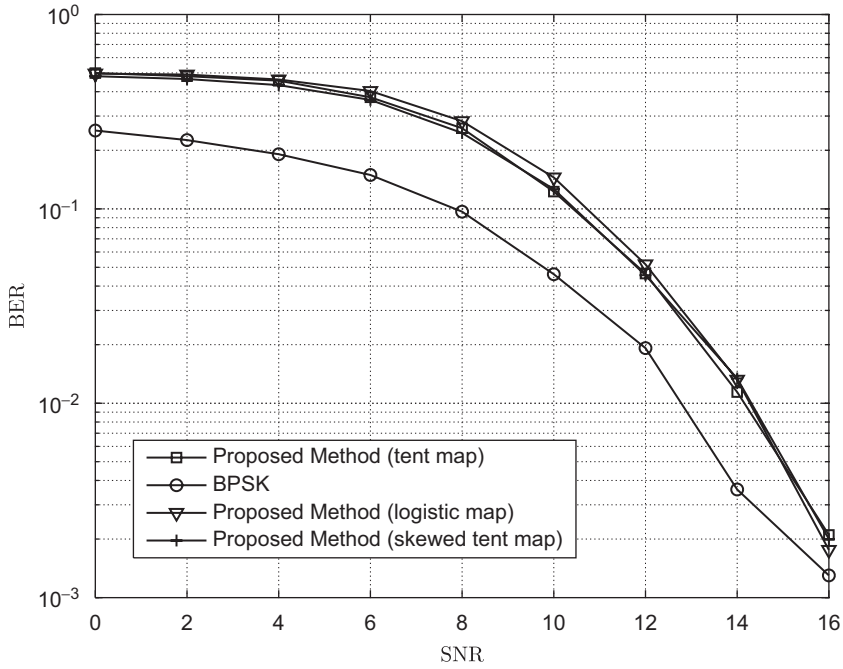

Fig. 7. BER performance under band-limited channel (Channel model-I).

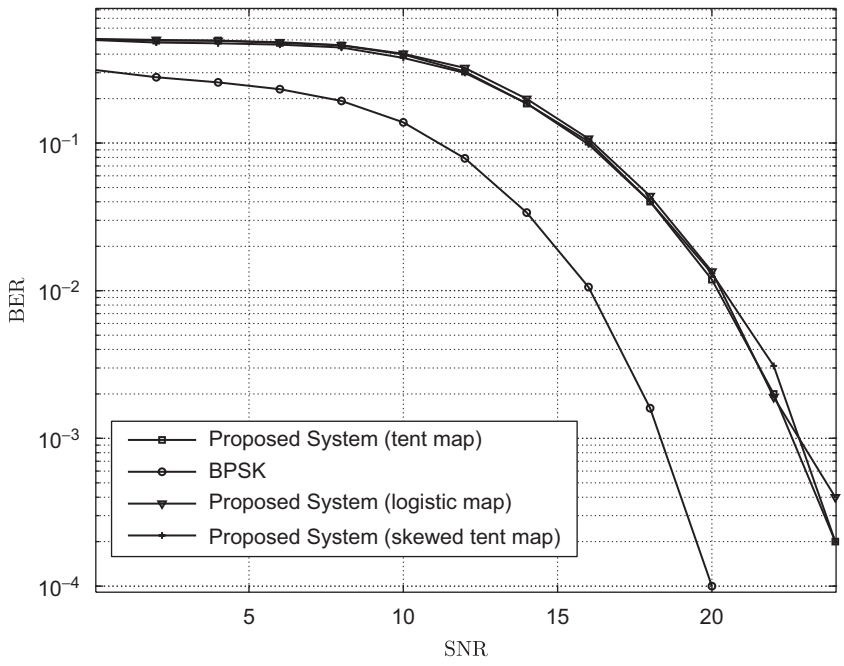

Fig. 8. BER performance under band-limited channel (Channel model-II).

4.2.1. Randomness of the output generated by the transmitter

For this method to work, the encrypted message should not posses any specific pattern. In other words, it should act like pseudo-random sequences. Statistical tests are conducted to asses the randomness of such an encryption method using the statistical testing suite developed in [28,29]. This suit offers many tests (listed in Table 1). Each test results in statistical quantity called the *P*-value. If the *P*-value is greater than $\alpha = 0.01$, the sequence is said to pass the randomness test. To assess the performance of a random number generator, a number of sequences (usually 1000) are tested and corresponding *P*-values are noted. Yield is defined as the fraction of

Table 1
Statistical test results (logistic map, $\mu = 4.0$)

Number	Test	Yield	*U*-value
1	Frequency	0.992	0.2077
2	Block frequency	0.993	9.680e − 011
3	Cumulative sum (forward)	0.992	0.4865
4	Cumulative sum (reverse)	0.995	0.4711
5	Longest run	0.985	0.7398
6	Rank	0.993	0.1663
7	FFT	0.989	0.1786
8	Non-periodic templates	0.996	0.4256
9	Overlapping templates	0.982	0.3026
11	Universal	0.983	0.2167
12	Approximate entropy	0.997	0.5605
13	Random-excursions	0.989	0.7481
14	Random-excursions (variant)	0.990	0.4387
15	Serial-I	0.993	0.6870
16	Serial-II	0.992	0.2392
17	Linear complexity	0.990	0.7279

sequences that pass the test (i.e., *P*-value $\geqslant\alpha$). Another aspect of the test is that the *P*-values should be uniformly distributed. A χ^2 test performed on the *P*-values results in *U*-values which measure the probability that *P*-values come from a uniform distribution. If *U*-value $\geqslant 10^{-4}$, the uniformity of the distribution is asserted.

To conduct the statistical tests, the following steps are performed. It is assumed that the task is to send an uncompressed bitmap image. Many 1000 bit blocks, constructed according to Section 3, are concatenated to get a sequence of length 10^6. One thousand of such sequences have been tested using the statistical test suit version 1.7 [28]. The results are given in Table 1. According to [28], a yield value more than 0.981 confirms the randomness of the generator. Here in all the tests, the yield values are very high. It falls within the three-σ range $1-\alpha\pm\sqrt{\alpha(1-\alpha)/1000}$ which is $[0.981, 0.999]$. The *U*-values of all the tests except the block frequency test confirm that the distribution *P* is uniform. Since the *P*-value for block frequency test is slightly biased toward 0.5, the corresponding *U*-value is close to zero.

It should be noted that not all the parameters of a chaotic map will result in purely random sequence. However, there are many instances in the literature where pseudo-random sequences are generated from piece-wise affine map (PWAM) [30].

4.2.2. Distribution of the bits carrying initial conditions

It is clear from the previous discussions that the initial condition of the chaotic map should be available at the receiver in order to decode the message. One method to improve the security of the system is to diffuse the information about the initial conditions in a random-like manner and make the positioning of the information known only to the receiver. This random pattern can be conveyed to the receiver *a priori* or can be transmitted through a dedicated (secure) channel. For example, this pattern may be based on a linear feedback shift register (LFSR) where the initial state and the feedback connection can be conveyed to the receiver in a highly secure connection.

4.2.3. Precision

Other important parameter that can be used as the security feature is the precision at which the chaotic generators are operating. It also includes the number of bits (m) used to convey the initial conditions. In simulation studies, it is possible to adjust the precision to any required value. If there is a slight mismatch in the precision used by the receiver, the resulting trajectory would become completely uncorrelated and produce an entirely different sequence [23], thus prohibiting an intruder from accessing the message.

4.2.4. Knowledge of the chaotic map

In this section, sensitivity of the proposed system to the parameter mismatch is discussed. Assume that the intruder somehow manages to extract the bits representing the initial condition and the precision with which it is specified. Now the question that follows would be whether the intruder can decode the message using just these information. In order to decode the message, the

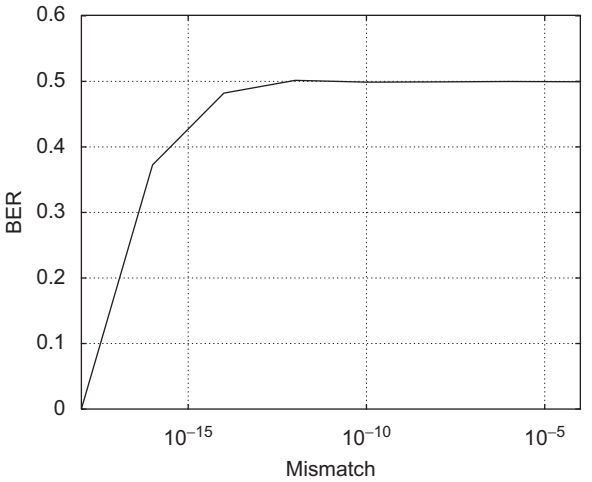

Fig. 9. Parameter mismatch vs. BER (15 dB).

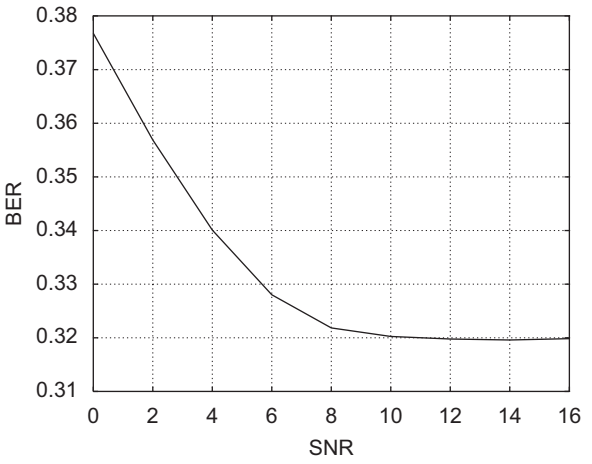

Fig. 10. BER performance under parameter mismatch.

Fig. 11. (a) Original image; (b) receiver uses $A = 0.8$; (c) receiver uses $A = 0.8 + 10^{-16}$.

knowledge of the chaotic map used at the transmitter is essential. In Fig. 9, the parameter mismatch (i.e., the difference between the transmitter and the receiver control parameters) vs. the BER performance is plotted. Clearly, if the mismatch is greater than or equal to 10^{-16}, the BER is unacceptably high. However, if the parameter mismatch is below 10^{-16}, the intruder can decode the information easily since the BER is close to zero.

Fig. 10 shows the performance of the proposed scheme under parameter mismatch. The parameter, A, of the transmitter tent map is set to 0.8. Assuming that the receiver guessed this value approximately (say, $A = 0.8 + 10^{-16}$), it is desirable to know if the intruder is able to decode the message. This slight parameter mismatch makes the receiver incapable of reconstructing the chaotic trajectory generated at the transmitter causing the receiver BER to remain at a high value. For a visual demonstration of this effect, a picture[3] (shown in Fig. 11(a)) is transmitted from the transmitter after encryption. Fig. 11(b) is the decrypted signal when the receiver has the correct knowledge of the parameter. Fig. 11(c) is the decrypted message corresponding to the use of wrongly guessed encryption key. It is clearly seen that the slight parameter mismatch makes the receiver incapable of decoding the information correctly.

If one tries to break the proposed system with brute force attack (i.e., trying each parameter), he has to experiment with only 10^{16} values (since, $0 < A < 1$). This in fact can lead to a low level of security. To improve this, the method suggested in [26] can be used. Here, multiple chaotic systems/maps are used for the generation of the sequence used for encryption. The transmitter and receiver schematic of the improved system is shown in Figs. 12(a) and (b), respectively. At the transmitter, there are L number of chaotic maps. Each of the maps is initialized with initial conditions $x_0^1, \ldots, x_0^L$ and has control parameters $A^1, \ldots, A^L$, respectively. Initial conditions $x_0^2, \ldots, x_0^L$ hold some deterministic relationships with x_0^1. After converting to its corresponding symbolic sequences, first m bits of each sequences are discarded. A bit-wise XOR is done on the resultant sequences to produce a single stream of length $N - m$. This sequence is used to encrypt the message $\mathbf{B}$. Shuffler mixes first m bits generated by the first chaotic map to get $\mathscr{Y}$. Since there is a relationship between x_0^1 and the other initial conditions, they can be computed at the receiver.

Since the current key space is L dimensional ($[A^1, \ldots, A^L]$), to do a brute force attack, the intruder has to search for 10^{16L} values. By adjusting the value of L, a balance can be achieved between the required computational complexity and the security. It may be noted that the synchronizer needs to estimate only one initial condition and hence the computational burden does not increase significantly at the receiver when L becomes larger. In addition to this, the relationship between the initial conditions can also be used as a secret key to improve the security. Moreover, it is observed that the BER performance of the improved system is not affected.

From the above discussion it is clear that the proposed system assures a certain level of security which is ideal for places where moderate security is needed. This system can be used in applications such as remote keyless entry system, video phone, wireless telephone, etc. [4].

5. Conclusion

Synchronization of chaotic systems is an important step in implementing chaotic communication schemes.

[3] This particular portrait of Sir Isaac Newton is taken from http://en.wikipedia.org/wiki/Isaac_Newton

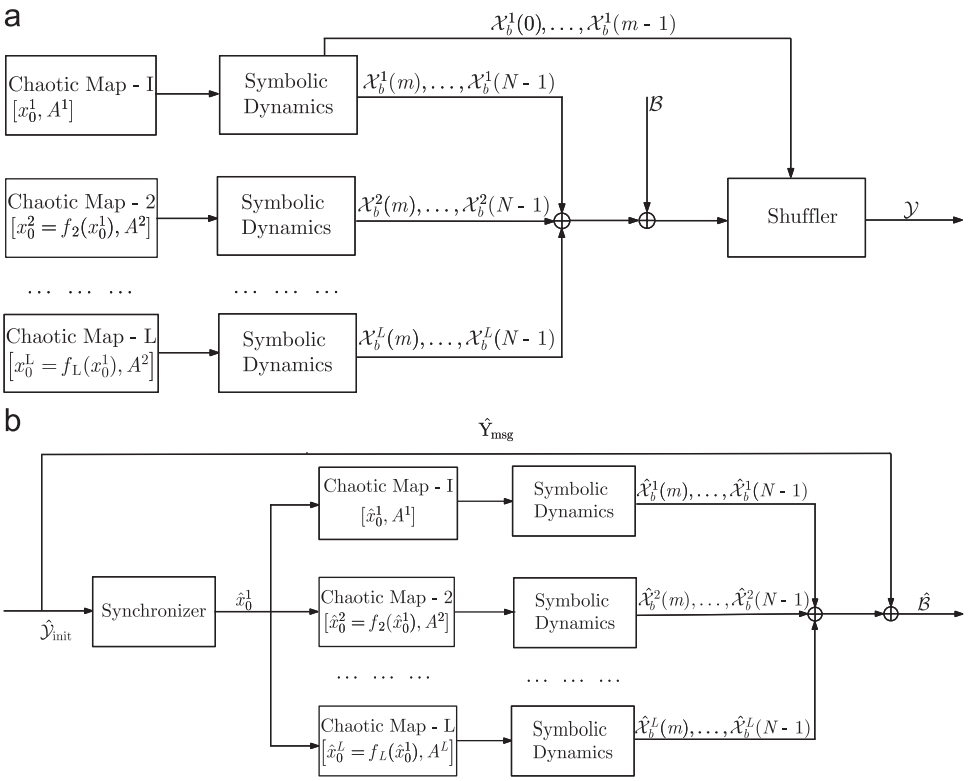

Fig. 12. Schematic of the improved system: (a) transmitter, (b) receiver.

Especially in noisy environment, the application of SD to synchronize chaotic systems has been proven to be a good choice. In this work, using SD of the chaotic maps, a new chaotic communication scheme is proposed. The information is dynamically encoded using 1-D iterated chaotic maps. The proposed method is tested for different maps like tent map, skewed tent map and logistic map. BER performance of the proposed scheme is analyzed analytically and numerically. It is found that the BER of the proposed communication scheme is comparable to that of the BPSK at moderately high SNR. Overhead needed for the proposed communication scheme is very minimal. Statistical tests reveal that the proposed system qualifies as random binary source. The sensitivity of the proposed system to various parameters is then analyzed in detail and possible improvements to the security of the code, based on the system's high sensitivity to certain parameters, are suggested. This in effect emphasizes the security of the proposed communication system.

References

[1] M.P. Kennedy, R. Rovatti, G. Setti (Eds.), Chaotic Electronics in Telecommunications, CRC Press, Boca Raton, 2000.
[2] G. Jakimoski, L. Kocarev, Chaos and cryptography: block encryption ciphers based on chaotic maps, IEEE Trans. Circuits Syst.—I 48 (2001) 163–169.
[3] F. Dachselt, W. Schwarz, Chaos and cryptography, IEEE Trans. Circuits Syst.—I 48 (2002) 1498–1509.
[4] O. Gonzales, G. Han, J. Gyvez, E. Sanchez-Sinencio, Lorenz-based chaotic cryptosystem: a monolithic implementation, IEEE Trans. Circuits Syst.—I 47 (2000) 1243–1247.
[5] K. Cuomo, A.V. Oppenheim, S.H. Strogatz, Synchronization of Lorenz-based chaotic circuits with applications to communications, IEEE Trans. Circuits Syst.—II 40 (10) (1993) 626–633.
[6] M.P. Kennedy, G. Kolumban, Digital communications using chaos, Signal Process. 80 (2000) 1307–1320.
[7] F.C.M. Lau, C.K. Tse, Chaos-based Digital Communication Systems, Springer, Berlin, 2003.
[8] J. Fridrich, Symmetric ciphers based on two dimensional chaotic map, Int. J. Bifurcation Chaos 8 (1999) 1259–1284.
[9] R. Matthews, On the derivation of the chaotic encryption algorithm, Cryptologia XIII (1989) 29–42.
[10] N. Masuda, K. Aihara, Crypto system with dicretized chaotic map, IEEE Trans. Circuits Syst.—I 49 (2002) 28–40.
[11] L.M. Pecora, T.L. Carroll, Synchronization in chaotic systems, Phys. Rev. Lett. 64 (1990) 821–824.
[12] L.M. Pecora, T.L. Carroll, G.A. Johnson, D.J. Mar, J.F. Heagy, Fundamentals of synchronization in chaotic systems, concepts, and applications, Chaos 7 (1997) 520–543.
[13] J.F. Heagy, T.L. Carroll, L.M. Pecora, Desynchronization by periodic orbits, Phys. Rev. E 52 (1995) R1253–R1256.
[14] D.J. Gauthier, J.C. Bienfang, Intermittent loss of synchronization in coupled chaotic oscillators: towards a new criterion for high-quality synchronization, Phys. Rev. Lett. 77 (1996) 1751–1754.
[15] H. Bai-lin, Elementary Symbolic Dynamics and Chaos in Dissipative Systems, World Scientific, Singapore, 1989.
[16] T. Stojanovski, L. Kocarev, R. Harris, Application of symbolic dynamics in chaos synchronization, IEEE Trans. Circuits Syst.—I 44 (1997) 1014–1018.
[17] A.S. Dimitriev, G.A. Kassian, A.D. Khilinsky, Information viewpoint on chaotic synchronization, Int. J. Bifurcation Chaos 10 (2000) 749–761.
[18] E.R. Bollt, Review of chaos communication by feedback control of symbolic dynamics, Int. J. Bifurcation Chaos 13 (2003) 269–285.

[19] G.M. Maggio, G. Galias, Application of symbolic dynamics to differential chaotic shift keying, IEEE Trans. Circuits Syst.—I 49 (2002) 1729–1735.
[20] J. Schweizer, T. Schimming, Symbolic dynamics for processing chaotic signals—I: noise reduction of chaotic sequences, IEEE Trans. Circuits Syst.—I 48 (2001) 1269–1282.
[21] J. Schweizer, T. Schimming, Symbolic dynamics for processing chaotic signals—II: communication and coding, IEEE Trans. Circuits Syst.—I 48 (2001) 1283–1295.
[22] D. Wheeler, Problems with chaotic cryptosystems, Cryptologia XIII (1989) 243–250.
[23] R.L. Devaney, An Introduction to Chaotic Dynamical Systems, The Benjamin/Cummings Publishing Company Inc., 1985.
[24] T. Kohda, Sequences of I.I.D binary random variables using chaotic dynamics, in: C. Ding, T. Helleseth, H. Niederreiter (Eds.), Sequences and Their Applications: Proceedings of SETA'98, Springer, Berlin, 1999, pp. 297–307.
[25] J.G. Proakis, Digital Communications, 4th ed., McGraw-Hill, New York, 2001.
[26] V.A. Protopopescu, R.T. Santoro, J.S. Tollover, Fast and secure encryption–decryption method based on chaotic dynamics, US Patent No. 5479513, 1995.
[27] T. Yang, A survey of chaotic secure communication systems, Int. J. Comput. Cognit. 2 (6) (2004) 81–130.
[28] A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications, National Institute for Standards and Technology, Special publication 800-22, 2001.
[29] S. Callegari, R. Rovatti, G. Setti, Embeddable ADC-based true random number generator for cryptographic applications exploiting nonlinear signal processing and chaos, IEEE Trans. Signal Process. 53 (2) (2005) 793–805.
[30] T. Kohda, Information sources using chaotic dynamics, Proc. IEEE 90 (5) (2002) 641–661.