Chapter 9
Cryptography Based on Spatiotemporal Chaotic Systems

Ping Li, Zhong Li, Wolfgang A. Halang, and Guanrong Chen

Abstract. Chaos has been applied in cryptography in the past decades since there are tight relationships between chaos and cryptography. Especially, spatiotemporal chaotic systems can be used to design cryptosystems with satisfactory properties. The chapter focuses on applying a typical spatiotemporal chaotic system, i.e., a coupled map lattice (CML) in cryptography. Multiple-output pseudo-random bit generators (PRBGs) based on CMLs with various constructions and parameters values are designed. Their properties are investigated and compared to determine a certain CML with certain parameters from which the resulting PRBG have satisfactory properties. Additionally, a stream cipher based on the CML is designed and analyzed. It is shown that it has high security, high efficiency and low cost. Moreover, a multimedia cryptosystem based on the proposed stream cipher is constructed by using a field programmable gate array (FPGA). The effects of the encryptions of the text file, the audio file and the image file by using the cryptosystem is measured as effective.

9.1 Introduction

Over the past decades, there has been much interest in designing and analyzing chaos-based ciphers [16, 4]. The main reason for it is that chaotic systems are

Ping Li
Department of Electronic Engineering, Shunde Polytechnic, Kanton, P.R. China
e-mail: Kikiliping@hotmail.com

Zhong Li · Wolfgang A. Halang
Faculty of Electrical and Computer Engineering, FernUniversität in Hagen,
58084 Hagen, Germany
e-mail: zhong.li@fern-hagen.de

Guangrong Chen
Department of Electronic Engineering, City University of Hong Kong,
Kowloon, Hong Kong SAR, P.R. China

I. Zelinka et al. (Eds.): Evolutionary Algorithms and Chaotic Systems, SCI 267, pp. 293–328.
springerlink.com

characterized by sensitive dependence on initial conditions and control parameters, random-like behavior and unstable periodic orbits with long periods, which are quite advantageous to ciphers [30, 13]. Till now, lots of chaos-based ciphers have been proposed, moreover, various techniques, such as using multiple chaotic systems [14], high-dimensional chaotic systems [5], multiple iterations of chaotic systems [33], have been proposed to improve chaos-based ciphers.

Among the proposed chaos-based ciphers, many ciphers are not applicable in practice, due to the following reasons. Firstly, for ciphers where the orbits of chaotic systems with simple constructions are directly used to encrypt plaintexts, useful information can be extracted from the chaotic orbits to break the ciphers. Secondly, there exists dynamical degradation of chaotic systems in their realization with digital computers, which threatens the security of the ciphers based on these chaotic systems [15]. Thirdly, some chaos-based ciphers have low implementation speeds [3], which makes the ciphers infeasible in practice. To overcome these drawbacks, multiple chaotic systems [14], high-dimensional chaotic systems [5], multiple iterations of chaotic systems [33], and perturbance-based algorithms [21] have been proposed to improve chaos-based ciphers.

More recently, a one-way coupled logistic-map lattice has been used to design ciphers [17]. Some modifications of these ciphers have been made to improve the security [26], performance [28] and robustness against channel noise [31]. These ciphers have high security and good performance since the following special inherent features of spatiotemporal chaos generated by the coupled logistic-map lattice.

1. The orbit of a spatiotemporal chaotic system has a long period even with dynamical degradation of digital chaos [27];
2. The randomness of the orbit of a spatiotemporal chaotic system is guaranteed by the complex dynamics with a large number of positive Lyapunov exponents;
3. There are multiple sites in a spatiotemporal chaotic system, which can generate independent keystreams simultaneously.

Therefore, using spatiotemporal chaotic systems in cryptography is a significant advance for improving chaos-based ciphers. It is meaningful to study spatiotemporal-chaos-based cryptography.

However, current ciphers based on spatiotemporal chaos adopt some conventional cryptographic techniques (such as S-box), and chaos synchronization. To study the benefit of using spatiotemporal chaos in cryptography, we concern about applying only spatiotemporal chaos into cryptography in the chapter.

In addition, the current research on spatiotemporal-chaos-based cryptography leads to the following problems to be addressed. Firstly, various coupled map lattices (CMLs) except for the one-way and diffusive coupled logistic-map lattice have never been applied in spatiotemporal-chaos-based ciphers. Thus, the issues of how to choose suitable spatiotemporal chaotic systems for cryptography are to be considered. Secondly, in the existing design of spatiotemporal-chaos-based ciphers, or even general chaos-based ciphers, the parameters of the nonlinear dynamical systems are fixed without explanation. Though the chaos-based ciphers with these parameters have acceptable properties from the cryptographic point of view, the question of how to choose the parameters remains to be answered.

This chapter is organized in the way of resolving the above listed problems, involving the following aspects of applying spatiotemporal chaotic systems in cryptography: design and analysis of CML-based pseudo-random bit generators (PRBGs), design and analysis of a CML-based stream cipher, design a multimedia cryptosystem based on the proposed stream cipher. Concretely, the chapter is sketched out in the following:

1. Design of a multiple-output PRBG using a diffusive coupled logistic-map lattice. The statistical properties, such as probability density function (PDF), linear complexity, auto-correlation and cross-correlation of the PRBGs based on various digitization methods are to be investigated. It will be shown that binary-representation method is the best one.
2. Analysis of the properties of the PRBGs based on various CMLs. To determine the CMLs, from which the resulting PRBGs have satisfactory properties, six PRBGs based on six different CMLs are investigated. The six CMLs consist of three simple chaotic systems, i.e., logistic map, skew-tent map and r-adic map, with two simplest coupling methods, i.e., one-way coupling and diffusive coupling, respectively. PDF, auto-correlation, cross-correlation, statistical test and cycle length of the six PRBGs with various parameters are to be investigated so as to determine the parameter intervals within which the PRBGs have satisfactory properties. It will be indicated that a one-way coupled logistic-map lattice with certain parameters has the best properties. This research results in criteria for designing PRBGs with proper performance.
3. Design of a stream cipher employing a one-way coupled logistic-map lattice with certain parameters. Only the last 32 bits are extracted from the chaotic orbit of each site in the CML to guarantee the pseudo-random bit sequences (PRBSs), which consist of the sequences of these 32bits, having perfect statistical properties. The PRBSs as keystreams are used to encrypt plaintexts by the "XOR" operation. The security of the stream cipher is to be tested by attacking it via typical attack methods and analyzing its cryptographic properties. Moreover, the efficiency of the stream cipher is to be analyzed. It will be shown that the cipher has higher security, higher efficiency and lower costs by comparing with Hu's stream cipher of a complicated configuration.
4. Design of a multimedia cryptosystem based on the proposed stream cipher. The cipher is to be implemented in a field programmable gate array (FPGA). The enhanced parallel port (EPP) is used to communicate data between a PC and the FPGA. A user-friendly interface is designed with Visual C++ 6.0, with which text, image and audio can be encrypted and decrypted successfully. The properties of the cryptosystem, such as the sensitivity to the key, speed and efficiency of the FPGA, are to be analyzed.

9.2 CML-Based Pseudo-Random-Bit Generators

A multiple-output PRBG is designed only based on a CML. The CML as a typical spatiotemporal chaotic system is introduced firstly. Digitization methods are used in

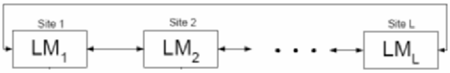


Fig. 9.1 A diffusive coupled map lattice

the PRBG and influence the properties of the PRBG. The statistical properties, such as probability density function (PDF), linear complexity, auto-correlation and cross-correlation of the PRBSs generated from the PRBGs based on various digitization methods are investigated and compared. It is indicated that the digitization method based on binary representation is good for a PRBG. Moreover, the configuration and parameters of the CML have affect on the properties of the PRBG. To determine suitable CMLs and corresponding parameter intervals for the PRBGs, six PRBGs are constructed by using the simplest coupling methods, i.e., one-way coupling and diffusive coupling, and three simple chaotic maps, i.e., logistic map, skew-tent map and *r*-adic map also named as sawtooth map. The statistical properties, periods and efficiency of these PRBGs with various parameters are investigated.

9.2.1 Coupled Map Lattice

CMLs are used as spatiotemporal chaotic systems in the chapter and introduced firstly. A spatiotemporal chaotic system is a spatially extended system, which can exhibit chaos in both space and time. It is often modeled by partial differential equations (PDE), coupled ordinary differential equations (CODE), or CML [23].

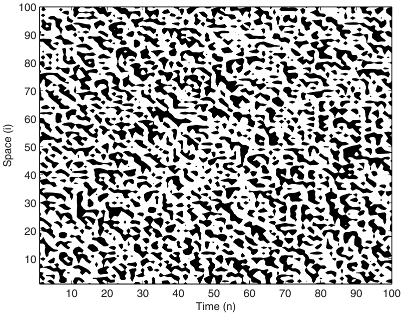
A CML is often adopted as the basic model of a spatiotemporal chaotic system. A CML is a dynamical system with discrete-time, discrete-space and continuous states. It consists of nonlinear maps located on the lattice sites, named as local maps. Each local map is coupled with other local maps in terms of certain coupling rules. Because of the intrinsic nonlinear dynamics of each local map and the diffusion due to the spatial coupling among local maps, the CML can exhibit spatiotemporal chaos. A CML has been extensively studied in the fields of bifurcation and chaos, pattern formation, physical biology and engineering since it was proposed by Kaneko in 1983 [7]. There are two main merits in using a CML as the model of a spatiotemporal chaotic system: one is that a CML captures the most essential features of spatiotemporal chaos; another is that a CML can be easily handled both analytically and numerically [23]. Further, by adopting various local maps and coupling methods [2], various CMLs can be constructed.

A diffusive coupling logistic-map lattice, one of the most popular CMLs, is described as

$$x_{n+1}^{j} = (1-\varepsilon)f(x_n^{i}) + \frac{\varepsilon}{2}[f(x_n^{j+1}) + f(x_n^{j-1})], \tag{9.1}$$

where $f(x) = rx(1-x)$ is the logistic map with $r \in (0,4]$, x_n^j represents the state variable for the jth site ($j = 1,2,...,L$, L is the number of the sites in the CML) at time n ($n = 0,1,2,...$), $\varepsilon \in (0,1)$ is the coupling strength. The periodic boundary condition, $x_n^0 = x_n^L$ for all n, is used in the CML. The CML can be illustrated in Fig. 9.1, where "LM" is the abbreviation of the local map.

Fig. 9.2 The pattern of a CML



Consider the CML (9.1) with $\varepsilon = 0.9$, $r = 4$ and $L = 100$. By starting from random initial conditions and discarding 10^5 initial transients, 100 sequential output of each site of the CML are depicted in Fig. 9.2, where black points stands for ones which values are larger than 0.5 and white points stands for ones which values are smaller than 0.5. It is shown that the CML is chaotic in both time and space; that is, for a certain site, its output is random-like, for certain time, the output of all sites is also random-like.

Similarly, the CMLs in other configurations with certain parameters can exhibit spatiotemporal chaos.

9.2.2 Digitization Method

If a CML exhibits spatiotemporal chaos, then its output, x_n^i, can be regarded as a pseudo-random number, which means that $\{x_n^i\}(n = 1,2,...)$ can be used as a pseudo-random-number sequence (PRNS), denoted by PRNS$_\text{i}$. Therefore, L PRNSs can be simultaneously generated from the CML with L sites.

By digitizing the PRNS of each site of the CML, PRBSs can be generated. There are three general methods to obtain PRBSs from PRNSs generated by chaotic maps as follows,

Method 1: By dividing the interval visited by a chaotic orbit, x_n, into m parts and labelling them with definite integers belonging to $[0, m-1]$, the nth pseudo-random number is assigned with an integer $r \in [0, m-1]$ when x_n enters the rth subinterval [25]. A special case is $m = 2$, that is, the interval $[a,b]$ is divided into two parts $[a,C]$ and $[C,b]$, where $x_n \in [a,b]$ and C is a threshold. This is the so-called threshold method proposed in [9] and applied in many PRBGs [32]. Then, a PRBS is defined as

$$s_n = \begin{cases} 1, \; if & x_n \in [a,C] \\ 0, \; if & x_n \in [C,b]. \end{cases}$$

Method 2: x_n can be represented as a binary sequence

$$x_n = 0.b_{n1}, b_{n2}, \ldots, b_{nP}, \tag{9.2}$$

where P stands for a certain precision. When a double-float precision is used in computer realization, P is equal to 52. Based on the binary representation, the digitization method is shown in the Fig. 9.3. It is seen that the mth bits in the binary representation comprise the mth PRBS. In maximum, P PRBSs can be generated from one PRNS by using this method. This method is proposed in [9] and widely used in PRBGs [32, 8].

Method 3: A modified version of Method 2 is proposed in [8], where a PRBS is generated as follow:

$$s_n = b_{n1} \oplus b_{n2} \oplus, \ldots, \oplus b_{nP}, \tag{9.3}$$

where $\oplus$ means XOR operation.

The digitization methods are applied here to get PRBSs from a PRNS. Thus, 3 PRBGs based on these three digitization methods are constructed, which are called PRBG1–3. In PRBG1 and PRBG3, only one PRBS is generated from one site. In PRBG2, the computation precision is assumed as 52, therefore, 52 PRBSs are generated from one site. Totally, 52L PRBSs can be generated at one time.

9.2.3 *Statistical Properties*

Such statistical properties as a uniform PDF, strong linear complexity, the δ-like auto-correlation and the close-to-zero cross-correlation are desirable for a PRBG to be applicable in cryptography [13].

- PDF
  Denote a PRBS generated from a PRBG as $S_N = \{b_1, b_2, \ldots b_i, \ldots, b_N\}$, where N is the iteration time. The uniform PDF of S_N means $P(b_i = 0) = P(b_i = 1)$. In other words, the ratio between the number of $\{s_i = 0\}$ and that of $\{s_i = 1\}$ is equal to 1.
- Strong linear complexity
  The linear complexity of S_N, denoted by L_n, is the length of the shortest LFSR that generates a sequence having S_N in its first n time. The sequence $\{L_n, n = 1, 2, \ldots, N\}$ is called the linear complexity profile of S_N, which can be computed using the Berlekamp-Massey algorithm [19]. By plotting the points (n, L_n) in the

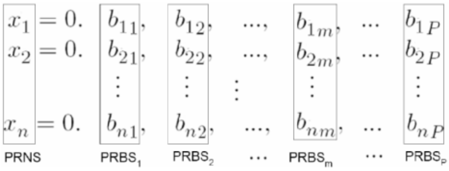

$x_1 = 0.$ $b_{11},$ $b_{12},$ ..., $b_{1m},$ ... b_{1P}
$x_2 = 0.$ $b_{21},$ $b_{22},$ ..., $b_{2m},$ ... b_{2P}
$x_n = 0.$ $b_{n1},$ $b_{n2},$ ..., $b_{nm},$ ... b_{nP}
PRNS
PRBS$_1$
PRBS$_2$
...
PRBS$_m$
...
PRBS$_P$

Fig. 9.3 Digitization method

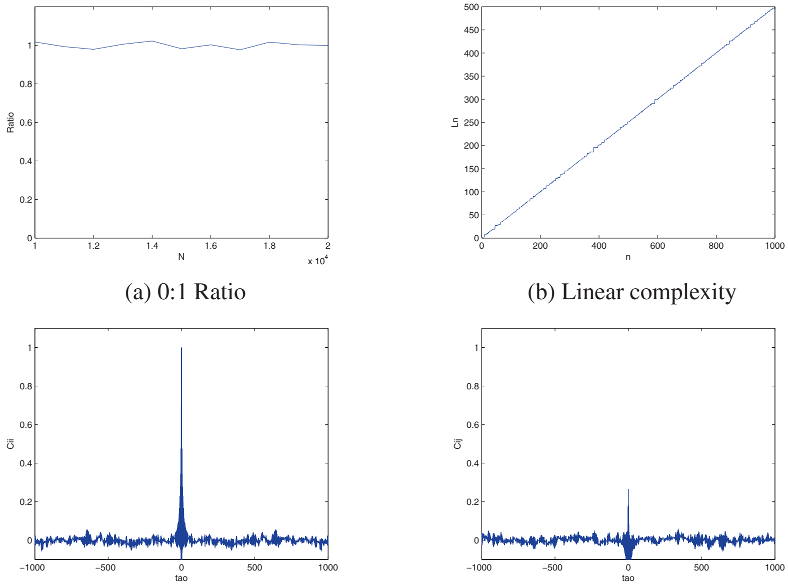
$n \times L$ plane and then joining the successive points by a horizontal line followed by a vertical line, the linear complexity profile of S_N is graphed. The expected linear complexity of a PRBG should closely follow the line $L = N/2$.

- δ-like auto-correlation
  The auto-correlation of S_N measures the extent of similarity between the sequences S_N and a shift of S_N by t positions. The mean-removed auto-correlation, $C_{ii}(\tau)$, of a PRBS is given by

$$\begin{aligned}
&C_{ii}(\tau) = \hat{C}_{ii}(\tau)/\hat{C}_{ii}(0),\\
&\hat{C}_{ii}(\tau) = \tfrac{1}{N}\textstyle\sum_{n=1}^{N}(b_n - \bar{b}_n)(b_{n+|\tau|} - \bar{b}_n),\\
&\bar{b}_n = \tfrac{1}{N}\textstyle\sum_{k=1}^{N} b_k,\\
&|\tau| = 0, 1, \ldots, N-1.
\end{aligned}$$

- Close-to-zero cross-correlation
  It is known that PRBSs can be generated simultaneously from a PRBG. If being independent of each other with zero cross-correlation, they can be used to encrypt multiple plaintexts at one time.

The statistical properties of PRBG1, PRBG2 and PRBG3 are investigated and shown in Figs. 9.4, 9.5 and 9.6, respectively. Here, one of 52 PRBSs is randomly chosen for testing its distribution, linear complexity and auto-correlation. Additionally, two of 52 PRBSs are randomly chosen for testing their cross-correlation.



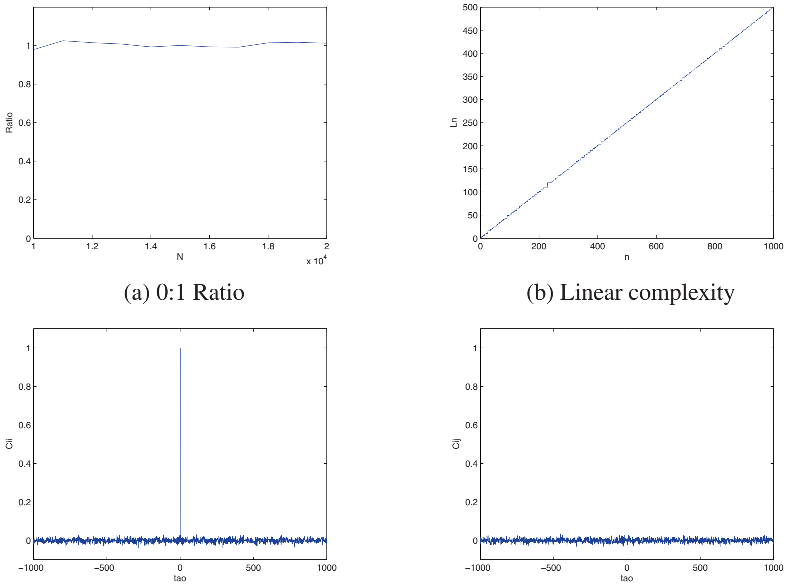
(a) 0:1 Ratio

(b) Linear complexity

(c) Auto-correlation

(d) Cross-Correlation

Fig. 9.4 Statistical properties of PRBG1



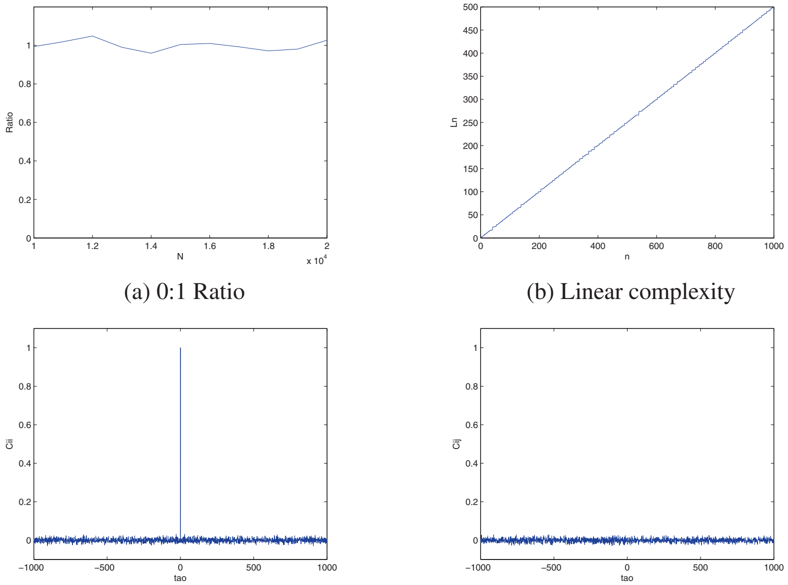
(a) 0:1 Ratio

(b) Linear complexity

(c) Auto-correlation

(d) Cross-Correlation

Fig. 9.5 Statistical properties of PRBG2



(a) 0:1 Ratio

(b) Linear complexity

(c) Auto-correlation

(d) Cross-Correlation

Fig. 9.6 Statistical properties of PRBG3

As we can see, different digitization methods result in different statistical properties of the PRBGs. A comparison of the statistical properties among the three PRBGs is concluded in Table. 9.1. The cross-correlation and the linear complexity of the three PRBGs meet the requirements of cryptography. It is remarked that the auto-correlation of PRBG1 is not δ-like, while the uniform distributions of PRBG1-2 are better than that of PRBG3; therefore, PRBG2 has the best overall statistical properties. The digitization method 2 is thus preferred in designing a PRBG.

Table 9.1 Comparison of statistical properties of Three PRBGs

PRBG	distribution	linear complexity	auto-correlation	cross-correlation
PRBG1	uniform	follows line $L = N/2$	not δ-like	close to zero
PRBG2	uniform	follows line $L = N/2$	δ-like	close to zero
PRBG3	almost uniform	follows line $L = N/2$	δ-like	close to zero

9.2.4 *PRBGs Based on Various CMLs*

A multiple-output PRBG based on a CML with certain parameters has good properties. This subsection concerns how to determine CMLs and their parameters for constructing PRBGs with satisfactory properties.

It is known that the PDF of the logistic map is equal to $\frac{1}{\pi\sqrt{x(1-x)}}$ [10], which is not uniform; whereas, any piecewise linear chaotic map $f : I \mapsto I$, $I = [a,b] \subset R$, have a uniform PDF, namely, $\frac{1}{b-a}$ [1]. The ununiformity of the PDF of the local map may have a negative effect on the PDF of the CML. Therefore, the CMLs based on a piecewise linear chaotic map, i.e., the skew tent map [18], is employed to construct a PRBG. The skew tent map is described as

$$f(x,p) = \begin{cases} x/p & , \quad x \in [0,p) \\ (x-p)/(1-p) & , \quad x \in (p,1] \end{cases} \quad p \in (0.5,1).$$

In the following, the properties of the PRBG based on diffusive coupled skew-tent-map lattice (DCSTML) are analyzed.

9.2.4.1 Statistical Properties of the PRBGs

Since the parameters of the PRBG, p, ε and L, may have effects on the properties of the PRBG, the effect of each parameter on the properties of the PRBG is analyzed. PDF, auto-correlation and cross-correlation as the important statistical properties of the PRBGs with one varying parameter and others fixed are investigated. In the simulation, the lengths of the PRBSs are computed to be 10^4 and the parameters vary in the following way: firstly, increase p from 0.51 to 0.99 by 0.01 each time, while fix ε as 0.9 and L as 8; then increase ε from 0.01 to 0.99 by 0.02 each time, while fix p as 0.51 and L as 8; finally, increase L from 8 to 64 by 1 each time with fixing p as 0.51 and ε as 0.9. Due to the symmetric configuration of the CML, it is reasonable to analyze the statistical properties of the multiple PRBSs generated from an arbitrarily chosen PRNS.

PDF

To analyze the PDF of a PRBG, a scaled difference ΔP between $P\{(b_n)=0\}$ and $P\{(b_n)=1\}$ in each PRBS, i.e., $\Delta P=(N_1-N_0)/(N/2)$, (N_1, N_0, and N are the number of "1" and "0", and the length of the PRBS, respectively) is computed. Fig. 9.7(a) shows ΔP of the 52 PRBSs output from the PRBG with various p, where the x-axis is the index of the 52 PRBS, denoted by i, the y-axis denotes the various p and the z-axis stands for ΔP.

It is shown that ΔP of the first 4 PRBSs are much bigger than zero. Additionally, by setting a threshold of ΔP as 0.07, Figs. 9.7(a) and 9.7(d) are plotted in the following way: if ΔP of the PRBS is smaller than the threshold, the point corresponding to the index of the PRBS and p of the PRBG from which the PRBS output is drawn black, otherwise, the point is drawn white. In the same way, ΔP of the 52 PRBSs from the PRBG with various ε and various L are plotted in Figs. 9.7(b)(e) and in Figs. 9.7(c)(f), respectively. It is shown that the 5th–52nd PRBSs have uniform PDF whatever the parameters are.

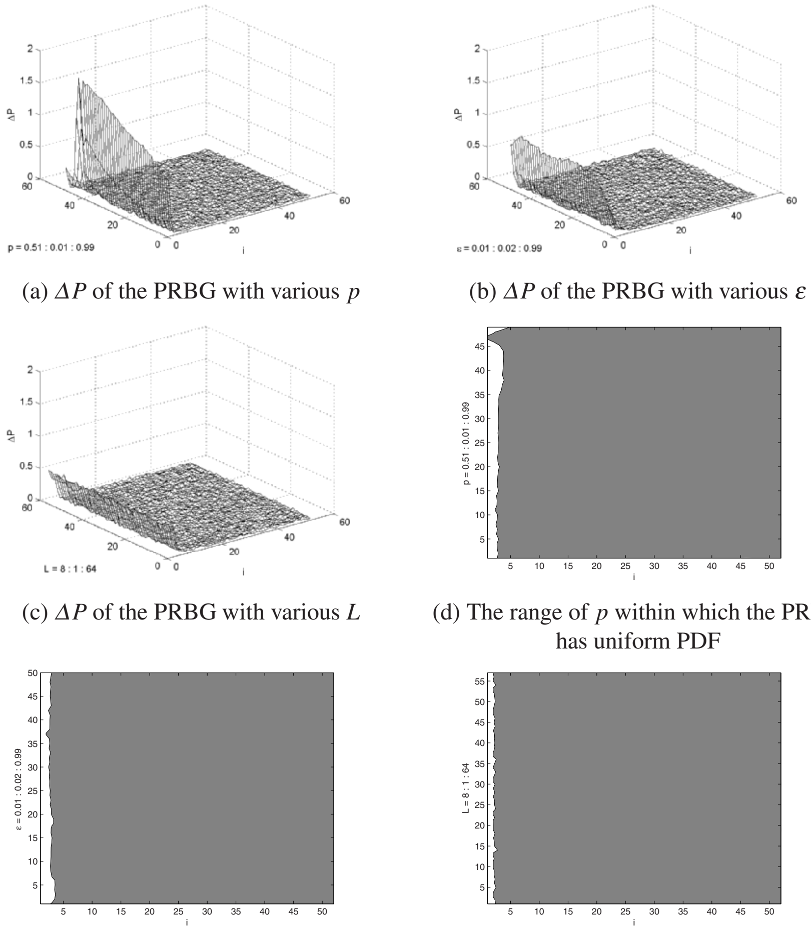
Auto-correlation

Since δ-like auto-correlation means $C_{ii}(0)=1$ and $\{C_{ii}(\tau)\}(|\tau|=1,2,...,N-1)$ or the maximum of $\{C_{ii}(\tau)\}(|\tau|=1,2,...,N-1)$ is close to zero, the close-to-zero maximum auto-correlation of a PRBS is equivalent to the δ-like auto-correlation of a PRBS. The maximum auto-correlations of the 52 PRBSs of the PRBG with various parameters are computed and shown in Fig. 9.8. It is indicated that the maximum auto-correlations of the first 4 PRBSs are much far from zero and the rest except for the PRBSs from the PRBG with p close to 0.99 is close to zero.

Additionally, we set the threshold of the maximums as 0.05, and get Fig. 9.8(d)(e) and (f) in the same way as that of the previous section. It is shown that the maximum auto-correlations of all the 5th–52nd PRBSs output from the PRBG with any parameters values are smaller than 0.05. Therefore, the auto-correlation of the PRBG without the first 4 PRBSs satisfies the requirement of cryptography.

Cross-correlation

In order for the 5th-52nd PRBSs generated from the output of each site of the PRBG to be applicable in parallel, the cross-correlation between arbitrary two PRBSs should be close-to-zero. Maximum cross-correlations, denoted by $C_{i_1 i_2}$, between $\mathrm{PRBS}^i_{m_1}$ and $\mathrm{PRBS}^i_{m_2}$ $(i,m_1,m_2\in N, i\in[1,L], m_1,m_2\in[5,52], m_1\neq m_2)$ output from the PRBG with various parameters are computed and shown in Fig. 9.9. Additionally, other maximum cross-correlations, denoted by $C_{i_1 j_2}$, between $\mathrm{PRBS}^i_{m_1}$ and $\mathrm{PRBS}^j_{m_2}$ $(i,j,m_1,m_2\in N, i,j\in[1,L], i\neq j, m_1,m_2\in[5,52])$ of the PRBG with various parameters are shown in Fig. 9.10.



(a) ΔP of the PRBG with various p

(b) ΔP of the PRBG with various ε

(c) ΔP of the PRBG with various L

(d) The range of p within which the PRBG has uniform PDF

(e) The range of ε within which the PRBG has uniform PDF

(f) The range of L within which the PRBG has uniform PDF

Fig. 9.7 ΔP of the PRBG with various parameters

We set the threshold of the maximum cross-correlation as 0.05 and find that the maximum cross-correlation between all pairs of PRBSs output from the PRBG with all parameters values are smaller than 0.05. Therefore, the cross-correlation of the PRBG is acceptable from the cryptographic point of view.

Statistical test

In practice, a statistical test is employed to investigate the randomness of PRBGs and thus to verify whether PRBGs are acceptable or not from the statistical point of view. There are many statistical test available, such as Diehard Battery of Tests, Knuth's

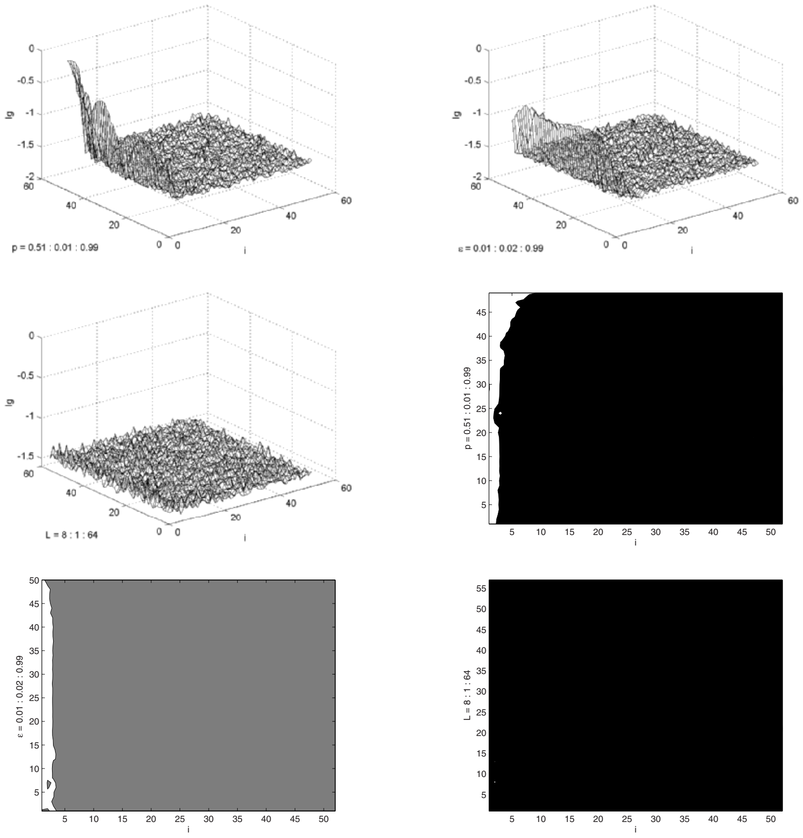


Fig. 9.8 The maximum auto-correlations of the PRBG with various parameters

Collection, FIPS 140-2 Statistical Test Suite, and NIST Statistical Test Suite [24]. Since the computations of the 52 PRBSs from the PRBG with various parameters are time-consuming, and among them, the FIPS 140-2 test takes the least computation time, therefore, in our work, the FIPS 140-2 [20] is used to evaluate the randomness of the proposed PRBG. The FIPS 140-2 specifies four statistical tests, i.e., monobit test, poker test, run test, and long run test, all of which should be passed if a PRBS passes the FIPS 140-2.

The 52 PRBSs generated from arbitrary one PRNS from the PRBG with various parameters are detected by using the FIPS 140-2. The results are shown in Fig. 9.11, where a black point corresponds to the index of the PRBS which passes the test and the parameter of the PRBG from which the PRBS is output. It is shown that the first 4 PRBSs have bad randomness with any parameter values. Therefore, these PRBSs should be discarded for a good PRBG. In addition, since the first 12 PRBSs

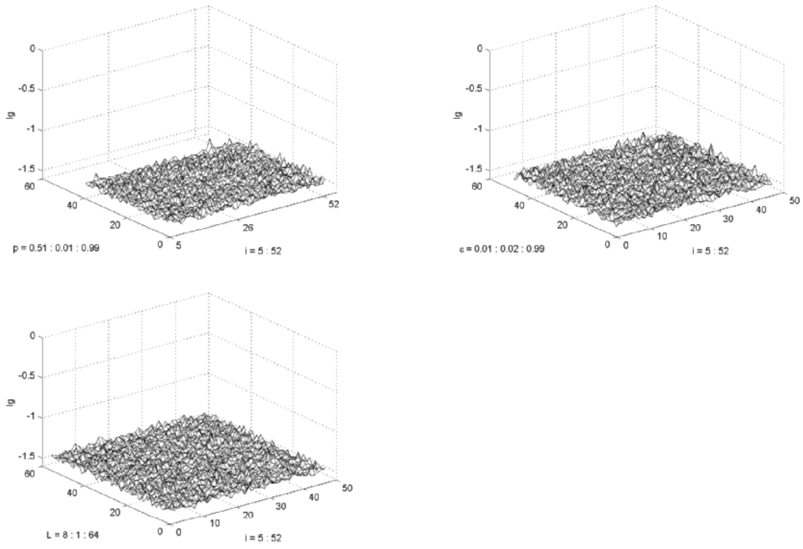


Fig. 9.9 The maximum $C_{i_1 i_2}$ of the PRBG with various parameters

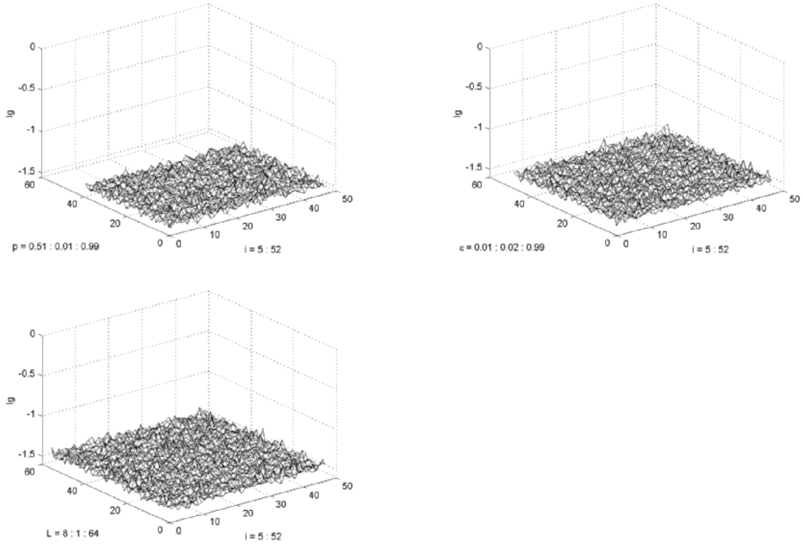
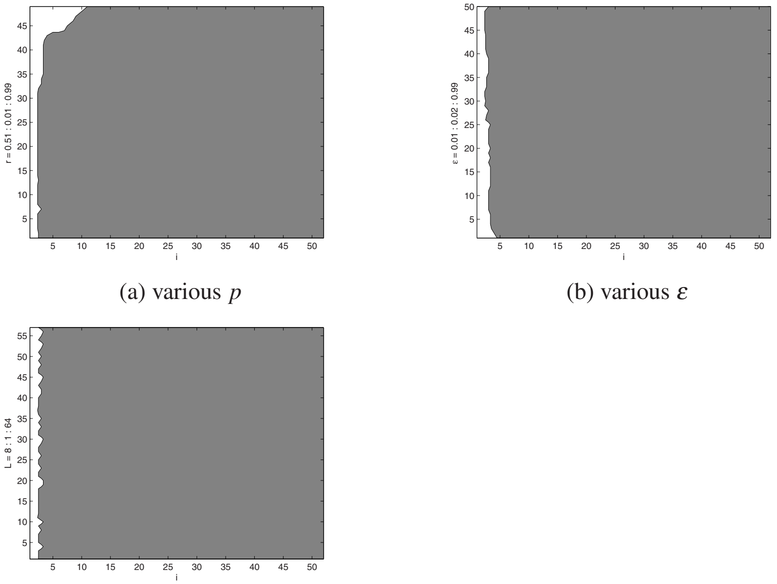


Fig. 9.10 The maximum $C_{i_1 j_2}$ of the PRBG with various parameters

from the PRBG with p close to 0.99 can not pass the FIPS 140-2 test, as shown in Fig. 9.11(a), p close to 0.99 should be avoided. According to Figs. 9.11(b) and 9.11(c), the 5th-52nd PRBSs can pass the FIPS 140-2 whatever ε and L are.



(a) various p

(b) various ε

(c) various L

Fig. 9.11 FIPS 140-2 of the PRBG with various parameters

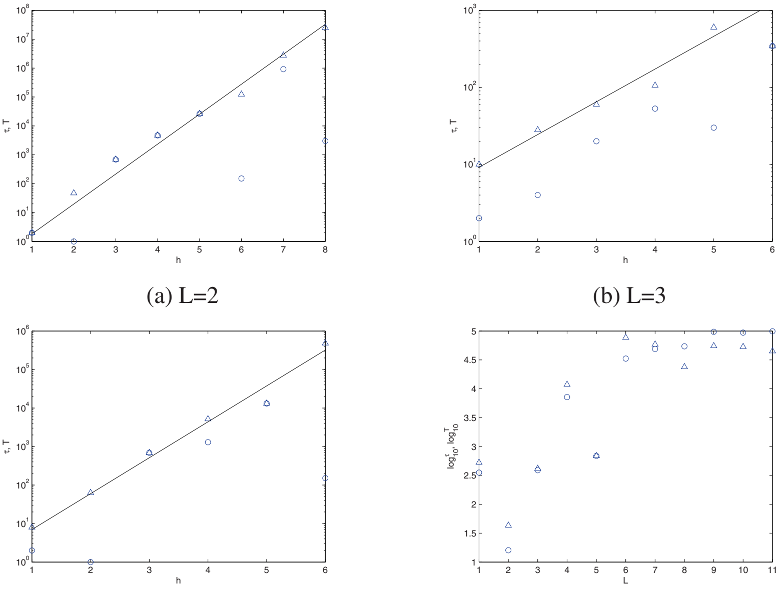
Periodicity

Long period is an important cryptographic requirement to a PRBG. Similar to the method of estimating the periodicity of a PRNS employed in [27], the dependence of the transient time for the trajectory to enter the periodic circle and the period on the precision $\delta = 10^{-h}$ and L is investigated. $\tau = \sum_{i=1}^{N} \tau_i$ and $T = \sum_{i=1}^{N} T_i$ of any PRNS generated from the PRBG with $L = 2$ are derived by arbitrarily choosing N ($N = 10^h$ if h ≤ 6 or $N = 1$ if $h > 6$) different initial conditions of a PRBG and computing transient times τ_i and the periods T_i in the way as [27]. τ and T vs h, respectively, are plotted in Fig. 9.12, where the triangle and circle signs stand for τ and T, respectively.

It is shown that τ follows the solid curves well. Therefore, for a certain L, one has

$$\tau(h,L) \propto 10^{\alpha(L)+\beta(L)h}.$$

For different L, $\alpha(L)$, $\beta(L)$ have different values. According to Fig. 9.12(d), τ and T do not increase as L increases, and in addition, T is always smaller than τ.



(a) L=2

(b) L=3

(c) L=4

(c) h=3

Fig. 9.12 τ and T of the PRBG with various L and h

Since the cryptographic properties of a PRBG are related to $T+\tau$ rather than τ, power law behavior of τ has an more important effect on the properties of the PRBG.

9.2.4.2 Comparison of PRBGs Based on Various CMLs

With different local maps and different coupling methods various CMLs can be constructed. In order to answer the question of how to determine the CMLs with certain parameters for constructing PRBGs with proper statistical properties, various CMLs with various local maps and coupling modes are used to construct PRBGs, and a comparison among the PRBGs is carried out in this section. Logistic map as the most well-known one-dimensional chaotic system, skew-tent map and r-adic map, described as $f(x,r) = rx \bmod 1$ $(r > 1)$ [2] as the simple piecewise linear chaotic maps are employed as the local maps of the CMLs; one-way coupling and diffusive coupling as the two simplest coupling modes are used as the coupling methods; thereby, six CMLs are obtained.

Similar to the analysis of the PRBG based on DCSTML, statistical properties of the PRBGs based on diffusive coupled logistic-map lattice (DCLML), diffusive coupled r-adic-map lattice (DCRML), one-way coupled logistic-map lattice (OWCLML), one-way coupled skew-tent-map lattice (OWCSTML), and one-way coupled r-adic-map lattice (OWCRML) with various parameters are investigated, respectively.

The results show that the PRBG based on one-way coupled logistic-map lattice with certain parameter has the most satisfactory statistical properties, largest period and highest efficiency among the six PRBGs.

9.3 CML-Based Stream Cipher

A one-way coupled logistic-map lattice (OWCLML) with certain parameters has the best cryptographic properties among the six simplest PRBGs, based on which a stream cipher is designed in the section.

9.3.1 Algorithm of the Cipher

Based on the OWCLML, a stream cipher can be constructed. The encryption is described as

$$\begin{aligned}
&x^j_{n+1} = (1-\varepsilon)f(x^j_n, a_j) + \varepsilon f(x^{j-1}_n, a_{j-1}),\\
&f(x^j_n, a_j) = (3.9 + 0.1a_j)x^j_n(1-x^j_n),\\
&K^j_n = \mathrm{int}[x^j_n \times 2^u] \quad \mathrm{mod} \quad 2^v,\\
&C^j_n = M^j_n \bigoplus K^j_n,
\end{aligned} \tag{9.4}$$

where $u, v \in N$, K^j_n, M^j_n, and C^j_n are keystream, plaintext and ciphertext, respectively, and $\bigoplus$ means bitwise XOR. Actually, the CML serves as a PRBG to produce L keystreams by imposing *int* and *mod* algebraic operations on the outputs of the CML. Plaintexts are bitwise XORed with keystreams to produce the ciphertext. Encryption keys are assumed as $a_j \in [0,1]$, denoted as a vector form $\mathbf{a} = \{a_1, a_2, ..., a_L\}$.

The configuration and parameters of the decryption are the same as those of the encryption, which is described as

$$\begin{aligned}
&y^j_{n+1} = (1-\varepsilon)f(y^j_n, a'_j) + \varepsilon f(y^{j-1}_n, a'_{j-1}),\\
&f(y^j_n, a'_j) = (3.9 + 0.1a'_j)y^j_n(1-y^j_n),\\
&K'^j_n = \mathrm{int}[y^j_n \times 2^u]\mathrm{mod}2^v,\\
&M'^j_n = C^j_n \bigoplus K'^j_n,
\end{aligned} \tag{9.5}$$

where $a'_j \in [0,1]$, denoted as $\mathbf{a}' = \{a'_1, a'_2, ..., a'_L\}$, are decryption keys. When $\mathbf{a}' = \mathbf{a}$ and $y^j_0 = x^j_0$, these two CMLs are synchronized, i.e., $y^j_n = x^j_n$, thus producing identical keystreams, $K'^j_n = K^j_n$. As a result, the plaintext is decrypted, $M'^j_n = M^j_n$.

Remarks:

1. Self-synchronous chaotic ciphers have an advantage that they do not need an extra synchronization signal, but also a disadvantage that a ciphertext, which controls a keystream generator in a cipher, is accessible and thus can be used for cryptanalysis [6].

2. In terms of the statistical properties discussed in section 9.2, in order for the keystreams in the proposed cipher to have proper statistical properties [11], a_j $(j = 1,2,...,L)$ are set as keys to guarantee that the parameter of the logistic map falls in the range $[3.9, 4.0]$ and ε is fixed as 0.95.
3. The double floating-point arithmetic is used in the cipher. Since the number of the significant bits of the binary representation of double floating-point number in the computer is 52, u is set as 52.
4. v is assumed as 32 for the following reasons. First, the first 4 bits are discarded for their bad statistical properties. Second, the smaller v is, the harder it is to break the cipher with known-plaintext attack, which will be indicated in subsection 9.3.3. Finally, from the implementation point of view, the larger v is, the more efficient the cipher will be. Therefore, a tradeoff between efficiency and security leads to fix v as 32 by considering that common computers adopt 32 bits or 64 bits CPUs.
5. The determination of L lies in the following considerations. L has no evident influence on the cryptographic properties of the keystream [11] except for its period equal to about 10^{7L} [27]. Meanwhile, it does not influence the encryption speed, too, which will be indicated in subsection 9.3.4. Additionally, the cost of breaking the cipher is about 2^{40L}, which will be analyzed in detail in subsection 9.3.3. Therefore, in investigating a concrete cipher thereafter, L is assumed as 4 in order that the keystream has period of 10^{28} and the cost of breaking the cipher is up to 2^{160}, which are suitable from cryptographic point of view.

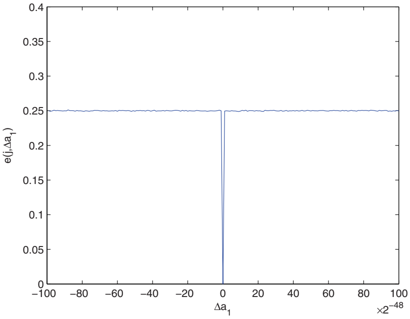
9.3.2 Keyspace

A keyspace is defined as a set of all possible keys [22], which should be studied in depth in designing a cipher. Error function [17] is used here to determine the keyspace of the cipher. When $\mathbf{a}' \neq \mathbf{a}$, the decrypted plaintext, ${M'}_n^j$, can be deviated from the original one, M_n^j. The error function is defined as

$$\begin{aligned} &e(j, \Delta\mathbf{a}_t) = \tfrac{1}{T}\textstyle\sum_{n=1}^{T} |{m'}_n^j - m_n^j|, j = 1,2,...L,\\ &{m'}_n^j = \tfrac{{M'}_n^j}{2^{32}}, m_n^j = \tfrac{M_n^j}{2^{32}}, \end{aligned} \tag{9.6}$$

where $\Delta\mathbf{a}_t = \{\Delta a_1, \Delta a_2, ..., \Delta a_t\}(\Delta a_i = a'_i - a_i, i = 1,2,...,t, t \leq L)$, and T is encryption times. The error function vs Δa_1 with $T = 10^5$ is plotted in Fig. 9.13. It is shown that the error function is not equal to zero but 0.25 even if Δa_1 takes an extremely small value 2^{-47}. In other words, the key a'_1 is sensitive to any differences equal to or larger than 2^{-47}. Similarly, the error function of $\Delta a_i (i = 2,3,...,L)$ are computed, and it is shown that the keys $a'_i (i = 2,3,...,L)$ are also sensitive to any differences equal to or larger than 2^{-47}. Therefore, the keyspace is 2^{47L}.

Fig. 9.13 Error function



9.3.3 Cryptographic Properties of the Keystream

Since the ciphertext is generated by using directly bitwise XOR between the plaintext and the keystream in the cipher, the cryptographic properties of the keystream have significant effects on the security of the cipher. Due to the symmetric configuration of the CML, all keystreams have similar cryptographic properties. Some cryptographic properties of a keystream among the L ones, such as probability distribution, auto-correlation, and run probability, are numerically investigated in this section.

Probability distribution

The order-1 and order-2 probability distributions [17] of the keystreams in the cipher with random initial conditions, arbitrarily chosen plaintext, and $\mathbf{a} = 0.5\mathbf{I}$ ($\mathbf{I}$ is a L-vector with all elements equal to 1) are investigated.

The order-1 probability distribution of the keystream, $\rho(ks_n^j)(=\frac{\rho(K_n^j)}{2^{32}})$, is plotted in Fig. 9.14(a). The order-2 probability distribution of the keystream, $\rho(ks_n^j, ks_{n-1}^j)(=\frac{\rho(K_n^j,K_{n-1}^j)}{2^{32}})$, is plotted in Fig. 9.14(b). The length of the keystream is 10^6. It is shown that the probability distributions are uniform.

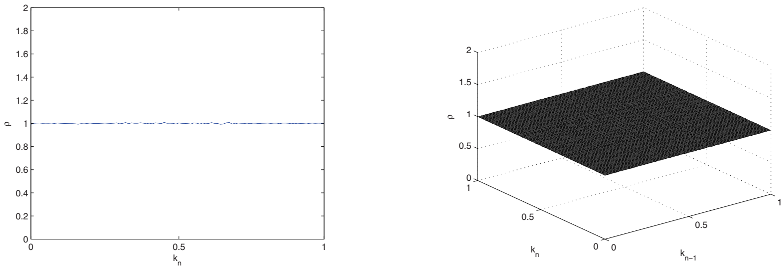


Fig. 9.14 Probability distribution of the keystream

Run

A run of a binary sequence s is another postulate of randomness, and defined as a subsequence of s consisting consecutive 0's or consecutive 1's that is neither preceded nor succeeded by the same symbol [19]. The probabilities of 0/1 runs of length $n(n = 1,2,...,N)$, denoted as $p_0(n)/p_1(n)$ or $p_{0/1}(n)$ of K_j, are investigated, where $p_{0/1}(n) = \frac{R_{0/1}(n)}{R_{0/1}}$ and $R_{0/1} = \sum_{n=1}^{N} R_{0/1}(n)$ with $R_{0/1}(n)$ being the number of 0/1 runs of length n. $p_{0/1}(n)$ vs n is plotted in Fig. 9.15. It is shown that $p_{0/1}(n)$ is directly proportional to n, which is the characteristic of a truly random binary sequence of an infinite length [17].

Auto-correlation

The δ-like auto-correlation is one of cryptographic requirements to keystream in a cipher. The mean-removed auto-correlation of the keystream with length $T = 10^6$ is plotted in Fig. 9.16. It is shown that the keystream has the δ-like auto-correlation.

In summary, according to the analysis above, L keystreams have satisfactory random-like statistic properties.

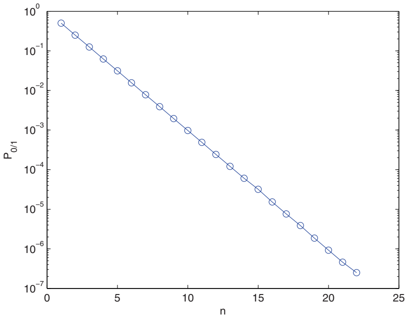


Fig. 9.15 Probability of the run of the keystream

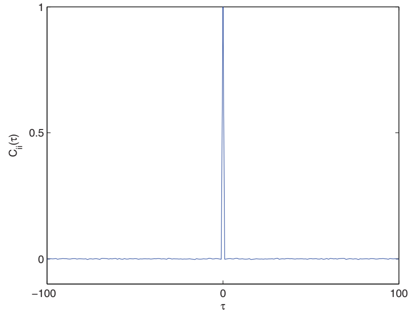


Fig. 9.16 Auto-correlation of the keystream

Security Analysis

In this section, the security of the cipher is evaluated by investigating its confusion and diffusion properties and using various typical attacks, such as the error function attack, the differential attack, the known-plaintext attack, the brute-force attack, and the chosen-plaintext/ciphertext attack.

9.3.3.1 Confusion and Diffusion

To resist common attacks, the cipher should have the following two basic cryptographic properties: confusion and diffusion. Confusion reflects the uniformity of all keys. To evaluate the confusion of the cipher, the independence of the probability distribution of the ciphertext on the exact value of a key is analyzed via $\rho(c|a)$ $(c = \frac{C}{2^{32}},\ C = C_n(j;a),\ \mathbf{a} = a\mathbf{I})$, which is shown in Fig. 9.17. The conditional probability distribution of the ciphertext is uniform for different keys. Therefore, the confusion of the cipher is guaranteed.

Diffusion reflects strong sensitivity of a key to tiny changes. In terms of the analysis of the error function described in subsection 9.3.2, the key of the cipher is even sensitive to a extremely small change 2^{-47}, which verifies the diffusion of the cipher.

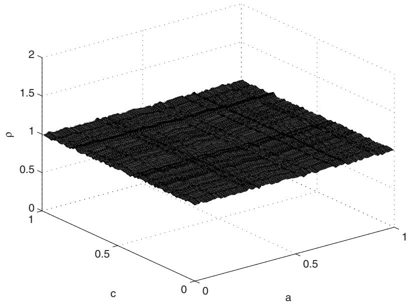


Fig. 9.17 Conditional probability distribution $\rho(c|a)$

9.3.3.2 Error Function Attack

The error function can also be used to break a cipher by an attacker, which is called Error Function Attack (EFA). For the proposed cipher, the cost of EFA is up to 2^{47L}. It is noted that the cost of EFA can be reduced by using some optimal adaptive searching methods if there exists certain tendency toward the key in the error function attack. To check if there is such a tendency, Fig. 9.13 is enlarged to Fig. 9.18. It is indicated that there is no tendency about the location of a_1 even if $a_1' - a_1$ is equal to 2^{-47}. Thus, any adaptive searching can hardly work without any tendencies. Therefore, the cost of EFA of the cipher is exactly equal to 2^{47L}.

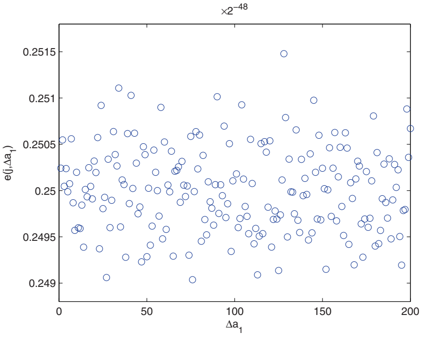


Fig. 9.18 Enlarged error function

9.3.3.3 Differential Attack

Some features of the differential relations between ciphertexts and plaintexts, such as some characteristic differential relations caused by any imperfect statistical properties of keystreams, can be used to break a cipher by a differential attack. To investigate whether there are such differential relations in the cipher, conditional probability of the ciphertext $\rho(\Delta c|\Delta m)(\Delta c = \frac{\Delta C}{2^{32}}, \Delta C = C_n(j;\hat{M}_n^j) - C_n(j;M_n^j))$ under the condition $\Delta m = \frac{\Delta M_n^j}{2^{32}}(\Delta M_n^j = \hat{M}_n^j - M_n^j)$ is shown in Fig. 9.19. The differential probability $\rho(\Delta c)$ is uniform whatever the differential probability $\rho(\Delta m)$ is. Therefore, the cipher is immune to the differential attack.

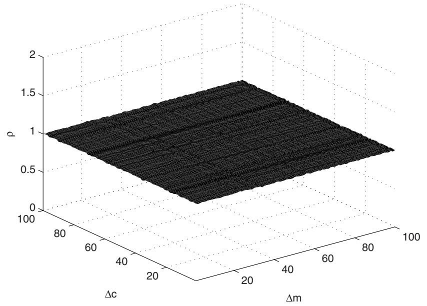


Fig. 9.19 Conditional differential probability of the ciphertext

9.3.3.4 Known-Plaintext Attack

With Kerchoffs' assumption, i.e., an attacker knows complete details of a cipher and implementation except keys, known-plaintext attack is to expose keys with public ciphertexts and known plaintexts. To break the proposed cipher, a known-plaintext attack is applied via an inverse analytical computation with known plaintexts and accessible ciphertexts, i.e., the keystreams K_n^j. The cost of the known-plaintext attack to the cipher can be estimated as follows. To simplify the conduction, $x_{n+1}^j = (1-\varepsilon)f(x_n^j, a_n^j) + \varepsilon f(x_n^{j-1}, a_n^{j-1})$ and $f(x_n^j, a_j) = (3.9 + 0.1a_j)x_n^j(1-x_n^j)$ in (9.4) can be recast as $x_{n+1}^{j}=G(x_n^j,x_n^{j-1},a_j,a_{j-1})$. To obtain keys $\mathbf{a}$, a set of L equations is given by

$$\begin{aligned} x_{n+1}^{1} &= G(x_n^1,x_n^L,a_1,a_L),\\ x_{n+1}^{2} &= G(x_n^2,x_n^1,a_2,a_1),\\ &\cdots\\ x_{n+1}^{L} &= G(x_n^L,x_n^{L-1},a_L,a_{L-1}), \end{aligned} \tag{9.7}$$

where $2L$ variables, x_{n+1}^{j} and x_n^j $(j=1,2,...,L)$, should be known to solve $\mathbf{a}$. In addition, one x_n^j can be obtained from one K_n^j, however, one K_n^j corresponds to 2^{52-32} possible x_n^j. Consequently, the cost of the known-plaintext attack is no less than 2^{40L}.

A typical known-plaintext attack is the brute-force attack, where a cipher is attacked by trying every possible key one by one to decrypt plaintext with public ciphertext and checking whether the resulting plaintext is the original one. Since the keyspace is deduced as 2^{47L}, the cost of the brute-force attack of the cipher is 2^{47L}.

9.3.3.5 Chosen-Plaintext Attack and Chosen-Ciphertext Attack

In applying chosen-plaintext and chosen-ciphertext attacks to the cipher, an attacker chooses some special plaintexts and ciphertexts to capture certain keystreams. In the cases that certain keystreams correspond to certain keys, i.e., there exist some characteristic relations between keystreams and keys, those attacks are effective. The relation between the keystream $k(=\frac{K_n^j}{2^{32}})$ and the key $a(\mathbf{a}=a\mathbf{I})$ is investigated with conditional probability distribution of k under the condition a. $\rho(k|a)$ is plotted in Fig. 9.20. It is indicated that no characteristics of keys can be extracted from keystreams, consequently, no special plaintexts or ciphertexts can be chosen to break keys. In other words, chosen-plaintext/ciphertext attack has the same efficiency as the known-plaintext attack to this cipher.

In summary, the known-plaintext attack is the most effective attack to the cipher and its cost to break the cipher is 2^{40L}. Moreover, the security of the cipher can be increased conveniently by adding one lattice to the CML in the cipher with little more computation, which results in the cost of breaking the cipher rising 2^{40} times.

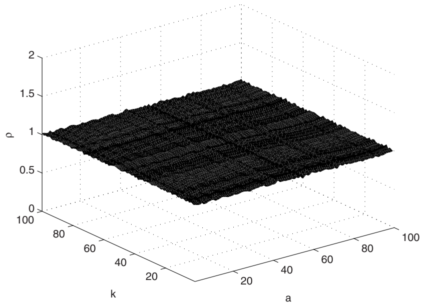


Fig. 9.20 Conditional probability distribution $\rho(k|a)$

9.3.4 High Efficiency

In addition to high security, the cipher is quite efficient. All the coupled maps of the CML in the cipher are used to encrypt plaintexts simultaneously. A close-to-zero cross-correlation among L keystreams guarantees the efficiency of the parallel L encryptions/decryptions in the cipher. The cross-covariance, i.e., mean-moved cross-correlation, is used here to analyze the cross-correlation of any two keystreams among L keystreams with length 10^6, K_n^i and K_n^j, which are described as

$$\begin{aligned}&C_{ij}(\tau) = \hat{C}_{ij}(\tau)/\sqrt{\hat{C}_{ii}(0)\hat{C}_{jj}(0)}, \tau = 0,1,...,T-1,\\ &\hat{C}_{ij}(\tau) = \tfrac{1}{T}\textstyle\sum_{n=1}^{T}(K_n^i - \overline{K}_n^i)(K_{n+\tau}^j - \overline{K}_n^j),\\ &\overline{K}_n^i = \tfrac{1}{T}\textstyle\sum_{n=1}^{T} K_n^i.\end{aligned} \tag{9.8}$$

The result of the computation of the cross-covariance is plotted in Fig 9.21. It is shown that all keystreams are independent. Therefore, the parallel L keystreams can be used to effectively encrypt plaintexts at one time.

Due to the parallel operation, an around 700M bits plaintext can be encrypted per second in our computer with 1.8GHz CPU and 1.5GB RAM. In addition, the encryption speeds of the ciphers based on the CMLs of various sizes are similar, and this is indicated by the relation between the speeds and L as shown in Fig. 9.22. As a comparison, the encryption speed of the cipher proposed in [17] is computed

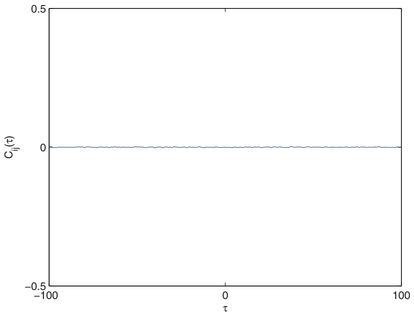


Fig. 9.21 Cross-correlation of any two keystreams

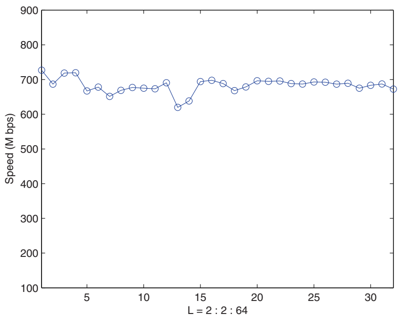


Fig. 9.22 The relation between the encryption speed and L

to be about 300M bits per second by our computer, which is much slower than the proposed cipher.

9.4 CML-Based Multimedia Cryptosystem

In this section, a multimedia cryptosystem based on a spatiotemporal chaotic system is proposed and implemented by a field programmable gate array (FPGA). The modification of the stream cipher proposed in section 9.3 is used in the cryptosystem. Since the generation processes of the multiple keystreams from the sites of the CML by a CPU of a computer are actually not in parallel, the cipher is implemented in an FPGA to generate the keysteams simultaneously. For implementation of the cipher in FPGA, the values of the cipher are digitized. The FPGA adopted in the cryptosystem is one Sparten-3 device produced by Xilinx company, i.e., XC3S400, because of its low cost and high efficiency of resource for implementing the digitized cipher. In designing the FPGA, a pipeline architecture is adopted to improve the usage efficiency of the FPGA. In the cryptosystem, data for encryption are input from a PC and transmitted to the FPGA. Data usually communicates via a serial port, a parallel port, a usual serial bus (USB), a peripheral component interconnect (PCI) bus, etc.. Within these communication modes, using a parallel port has some advantages: simple transfer protocol, high speed and easy implementation. Especially, an enhanced parallel port (EPP), which is a data transfer mode defined in IEEE 1284 standard, has simple transfer protocol and high transfer speed. Therefore, an EPP is applied in this system. Simulation shows that the data are communicated efficiently between a PC and an FPGA via the EPP. In addition, a user-friendly interface of the cryptosystem is designed with Visual C++. With the easy-to-handle interface, a user can encrypt/decryt text, image and audio files, and observe the results.

For testing the performance of the cryptosystem, the keyspace, the statistical properties, the security and the efficiency of the cryptosystem are investigated.

9.4.1 Design of CML-Based Multimedia Cryptosystem

The multimedia cryptosystem adopting a cipher based on a CML, which is implemented in a FPGA, is described in this section. The design of the cryptosystem consists of the FPGA implementation of the cipher, the EPP communication between a PC and the FPGA and a user-friendly interface, which are to be described in the following.

9.4.1.1 FPGA Implementation of the CML-Based Cipher

The multimedia cryptosystem adopts the cipher modified from that presented in section 9.3 [12]. Since the cipher (9.4) adopts double float precision, it should be digitized into integer domain for its implementation in an FPGA. In addition, to make the balance between efficiency and security of the cipher (9.4), the highest 32 bits in binary representations of the values are used as keystream. Similarly, only the highest 32-bit in binary representations of the values are used during iterations of the cipher. The modified cipher, named as digital CML-based cipher, is described as

$$\begin{aligned}
&x_{n+1}^{j} = (((2^{32} - \varepsilon')f(x_n^j))_{h32} + (\varepsilon' f(x_n^{j-1}))_{h32})_{h32},\\
&f(x_n^j) = ((x_n^j << 2)_{h32}(2^{32} - x_n^j))_{h32},\\
&K_n^j = x_n^j\\
&C_n^j = M_n^j \oplus K_n^j,
\end{aligned} \tag{9.9}$$

where "$(\cdot)_{h32}$" means to extract the highest 32 bits of the value in "()"; "$<< 2$" stands for right-shifting 2 bits since the parameter of the logistical map in each site is equal to 4 to achieve the strongest chaos of the CML [12]; $\varepsilon' = (\text{int})\varepsilon \times 2^{32}$; ε is assumed as 0.95 to make the keystream possessing good statistical properties [11].

The FPGA implementation of each site in the cipher is similar. Taking the 1st site as an example, its implementation at one round is realized in the following four steps:

1. compute $1 - x_n^1$ and $1 - x_n^4$,
2. compute $f(x_n^1)$ and $f(x_n^4)$,
3. compute $(1-\varepsilon)f(x_n^1)$ and $\varepsilon f(x_n^4)$,
4. compute x_{n+1}^1,

To achieve the highest speed of the cipher with the limited resource of the FPGA, a pipeline architecture is adopted. Meantime, to use effectively the resource of the FPGA with the pipeline architecture, the CML in the cipher adopts 8 sites. Thus, 16 18 × 18 multipliers of Spaten3 series are needed simultaneously for four multiple arithmetic of two 32-bit integers. Therefore, the FPGA in the cryptosystem adopts the type XC3S400 of Spaten3 series. The simulation done in Modelsim 5.8 environment shows that all multipliers of the FPGA which are the key resource of a FPGA work simultaneously and one datum can be output from the CML at any clock cycle.

Moreover, to evaluate the efficiency of using the FPGA resources, the conceptual VHDL design definition is synthesized to generate the logical or physical representation for the targeted silicon device. Xilinx Synthesis Technology (XST) is used to synthesize the CML-based cipher by choosing the device xc3s400pq208. The result is shown in Table. 9.2, which indicates that all multipliers are used, therefore, the device xc3s400 can execute the algorithm well.

Table 9.2 Performance of the FPGA implementation of the CML-based cipher

CLK	Slices	Flip Flops	LUT-4	GCLK	MULT18X18s
150.060MHz	504 (14%)	545 (7%)	677(9%)	1(12%)	16 (100%)

9.4.1.2 EPP Communication between PC and FPGA

In the cryptosystem, data for encryption are input from a PC and transmitted to the FPGA. Data communication can be usually via a serial port, a parallel port, USB, a PCI bus, etc.. Within these communication modes, since the EPP has relatively simple transfer protocol and high transfer speed, the data between a PC and an FPGA are transferred via EPP in the presented cryptosystem. The implementation of EPP communication includes designing a state machine and optimizing.

Design of state machine: A state machine is used to realize the communication protocols of EPP in the FPGA. The adopted state machine is the so-called mealy state machine consisting of 5 states.

Optimization of communication: Since the transistor-transistor logic (TTL) of an EPP adopts $0 \sim 5v$ voltage standard, which can be easily perturbed, the output signal from a PC, such as a read/write control signal, a data selection signal and 8-bit signal in the data bus, should be filtered before entering into the FPGA. A low-pass filter using two-stage D flip-flops is employed here as filtering function.

The cryptosystem is simulated in Modelsim 5.8. The simulation result indicates that the data are successfully communicated between a PC and the FPGA, and encrypted/decrypted correctly.

9.4.1.3 Implementation of the Cryptosystem

An interface for a user to manipulate the cryptosystem is designed with Visual C++ 6.0, as shown in Fig. 9.23. The interface includes the buttons of inputting encryption/decryption keys and sending the keys. For text encryption/decryption, it supports the windows for inputting plaintexts and displaying ciphertexts and decrypted texts. For image encryption/decryption, plain-image, cipher-image and decrypted-image can be visible by choosing the corresponding check boxes. The cipher-image and decrypted image can also be saved by clicking the **Save** buttons. For audio encryption/decryption, by clicking **Read In Audio** button, one audio file can be chosen or a recorder can be open for recording audio files to be encrypted. Plain-audio, cipher-audio and decrypted-audio can be played by clicking corresponding buttons.

With the user-friendly interface, the sender can input plain-media and encryption key, encrypt the plain-media and display cipher-media in his PC; the receiver can input decryption keys, and display cipher-media sent by the sender and decrypted-media in his PC.

9.4.2 Performance Analysis

For measuring the performance of the cryptosystem, its cryptographic properties, security, speed and its effects of encrypting multimedia is investigated quantitively in this section.

9.4.2.1 Properties of the Cryptosystem

Since the cipher of the cryptosystem is modified from that in [12], the properties of the cryptosystem may be different from those of the CML-based cipher in [12]. Similar to the way in [12], the keyspace, the statistical properties of the keystreams,

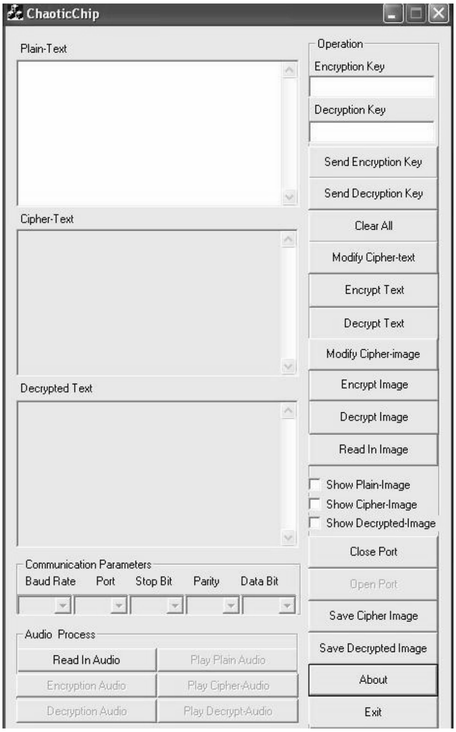


Fig. 9.23 The system interface

the security and the efficiency of the cryptosystem are analyzed, which results are described as follows.

Keyspace: Error function is used to determine the keyspace of the cryptosystem, which is equal to $2^{8\times 32} = 2^{256}$.

Statistical properties of the keystream: The important statistical properties, such as probability distribution, auto-correlation and run probability, of the keystream generated from the cryptosystem are numerically investigated. It is shown that the cryptosystem has uniform 1-order and 2-order probability distribution, δ-like auto-correlation and random-like run-probability. Additionally, NIST test suite is used to measure the randomness of the keystream. 20 keystream sequences of length 10^6 are tested here by the NIST statistical test suite. All passing rate of tests except for AET being 95% are equal to 100%, which indicates that the keystreams pass the NIST test suite.

Security: The security of the cryptosystem is evaluated by investigating its confusion and diffusion properties and using various typical attacks, such as the error function attack, the differential attack, the known-plaintext attack, the brute-force attack and the chosen-plaintext/ciphertext attack. It is shown that the brute force attack is the most efficient attack, which costs 2^{256}.

Efficiency: Since the FPGA adopts the pipeline architecture, the generation speed of one CML output is up to the clock speed in the FPGA. In the cases of adopting an FPGA with the clock frequency of 150.060MHz, a plaintext of 4.8G bits can be encrypted per second.

In addition, the cost of the cryptosystem is nearly equal to that of the adopted FPGA. Since the cost of the adopted FPGA is low, the cryptosystem is low-cost. Therefore, the cryptosystem has been verified to possess satisfactory statistical properties, high security, high encryption speed and low-cost.

9.4.2.2 Effect of Encrypting Multimedia

Various media have own special features, so the effect of encrypting various multimedia may be different. In this section, the effect of the cryptosystem's encrypting text, audio and image files are investigated, respectively.

Text

The effect of encrypting a text file is analyzed by taking a special file with identical characters as a plaintext. A text file consisting of 10000 "1" in byte is used as a plaintext and encrypted by our cryptosystem. To resist statistical attacks, the ciphertext should have random-like properties. The distribution and auto-correlation, as the important statistical properties, of the ciphertext are investigated.

The plain-text and its cipher-text are plotted in Figs. 9.24(a) and 9.24(b), respectively. Their histograms with 256 bins are obtained in Figs. 9.24(c) and 9.24(d), respectively. Since the text file consists of identical characters, its distribution is the worst. However, the ciphertext has nearly uniform distribution. To measure the uniformity of the generated ciphertext, the χ^2 test of the ciphertext's histogram is applied [29].

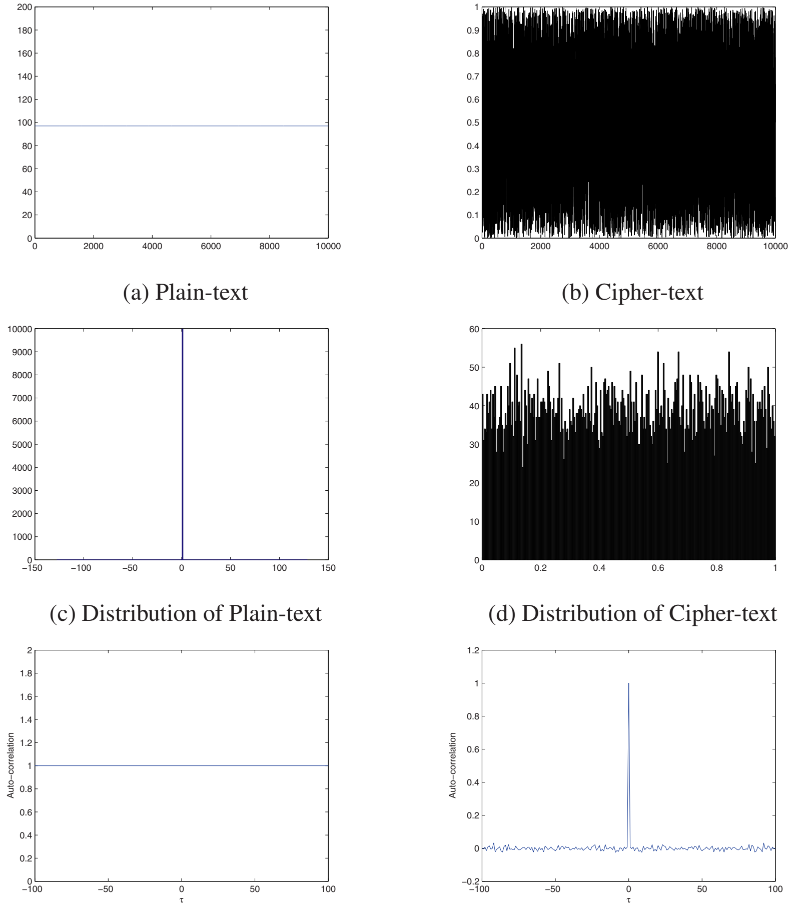
The statistic of the χ^2 test is described as

$$\chi^2 = \sum_{i=1}^{K} \frac{(o_i - e_i)^2}{e_i}, \tag{9.10}$$

where o_i, e_i and K are an observed frequency, an expected (theoretical) frequency and the number of distinct events, respectively. Here, K is equal to 255 due to the histogram having 256 bins, then $e_i = (int)\frac{10000}{256}$ and χ^2 of the histogram is computed as 223.88. With a significance level of 0.05, it is found that $\chi^2_{255;0:05} = 293.2478 > 223.88$, which implies that the distribution of the ciphertext is uniform.

In addition, the auto-correlations of the plaintext and the ciphertext are obtained and plotted in Figs. 9.24(e) and 9.24(f), respectively. It is seen that the ciphertext has δ-like auto-correlation, which is similar to random texts.

To check the diffusion with respect to the plaintext, the cross-correlation of the two ciphertexts from two plain-texts with the difference of only 1 bit is computed and plotted in Fig. 9.25.



(a) Plain-text

(b) Cipher-text

(c) Distribution of Plain-text

(d) Distribution of Cipher-text

(e) Auto-correlation of Plain-text

(f) Auto-correlation of Cipher-text

Fig. 9.24 Plain-text and cipher-text, and their distributions and Auto-correlation

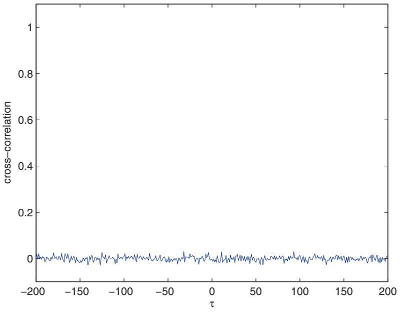
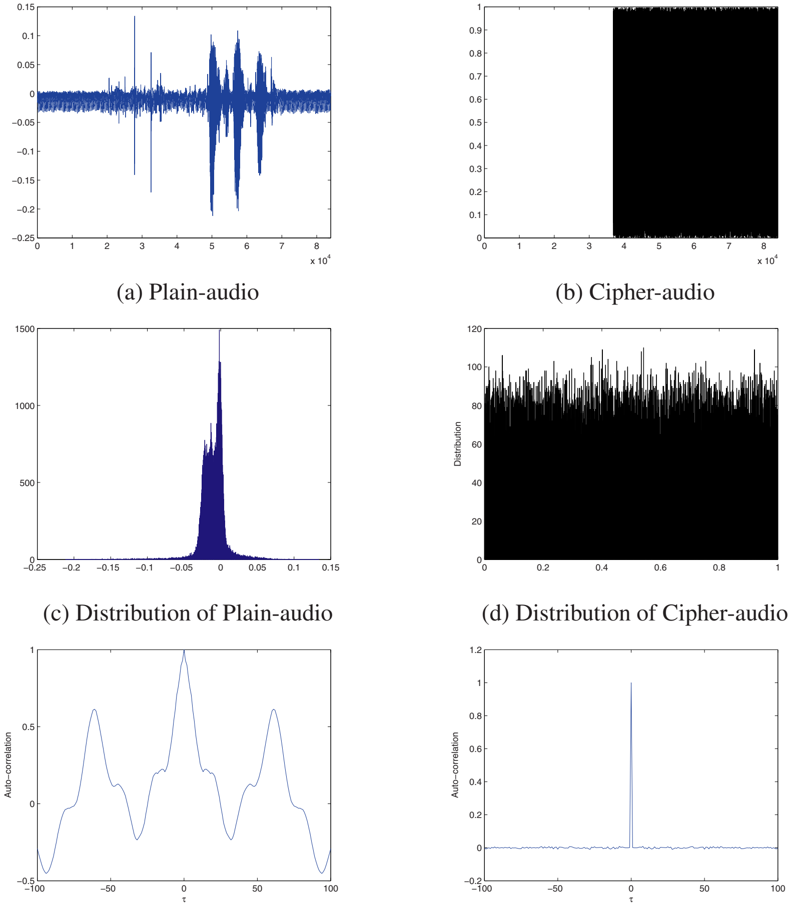


Fig. 9.25 The correlation of two cipher-texts from two plain-texts with tiny difference

It is seen that the cross-correlation is close-to-zero, which indicates that small change in a plain-text can influence over whole the cipher-text. Therefore, the cryptosystem can encrypt texts effectively.

Audio

Generally, the correlation of adjacent data in an audio file is stronger than that of a text file. To check the effect of encrypting an audio file with this cryptosystem, an audio file consisting of a short time of silence and several same words is tested. The plain-audio and its cipher-audio are plotted in Figs. 9.26(a) and 9.26(b), respectively. Their histograms with 1024 bins are obtained in Figs. 9.26(c) and 9.26(d),



(a) Plain-audio

(b) Cipher-audio

(c) Distribution of Plain-audio

(d) Distribution of Cipher-audio

(e) Auto-correlation of Plain-audio

(f) Auto-correlation of Cipher-audio

Fig. 9.26 Plain-audio and cipher-audio, and their distributions and Auto-correlation

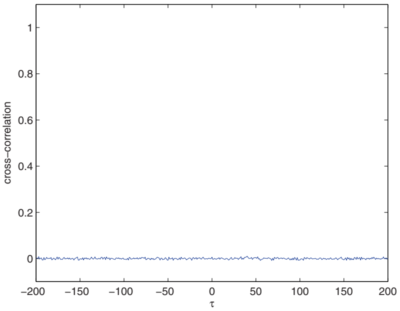


Fig. 9.27 The correlation of two cipher-Audios from two plain-Audios with tiny difference

respectively. It is shown that the cipher-audio has much more uniform distribution than the plain-audio. χ^2 test is also used here to check whether the distribution of the cipher-audio is uniform. χ^2 of the distribution of the cipher-image is computed as 926.54 and smaller than $\chi^2_{1023;0:05} = 1098.5208$ with a significance level of 0.05. Therefore, the distribution of the cipher-audio is regarded as uniform. In addition, the auto-correlations of the plain-audio and cipher-audio are obtained and plotted in Figs. 9.26(e) and 9.26(f), respectively. It is indicated that the cipher-audio has δ-like auto-correlation.

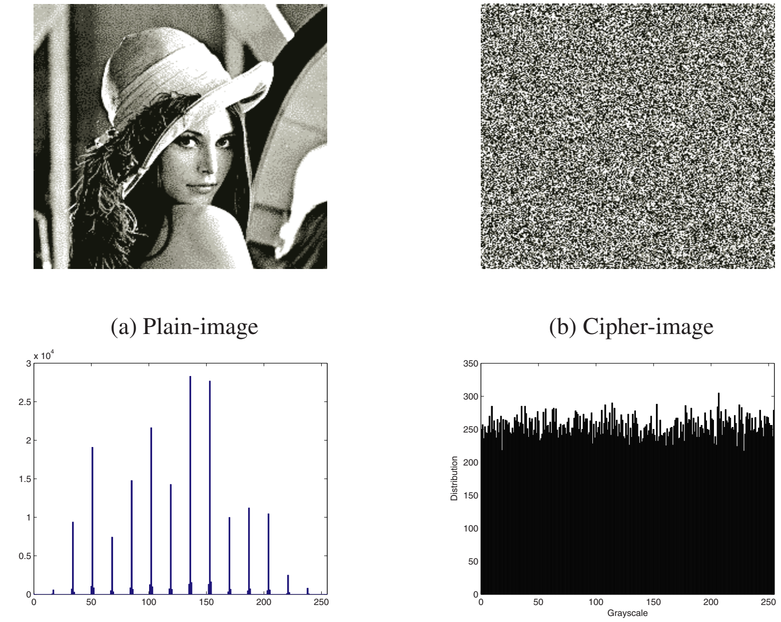
Moreover, the cross-correlation of two cipher-audios from two plain-audios with tiny difference is computed and plotted in Fig. 9.27.

The close-to-zero cross-correlation indicates the complete diffusion with respect to a plain-audio. As a result, the cryptosystem can encrypt an audio file effectively.

Image

It is known that adjacent pixels in an image have high correlation. In order to resist the statistical attacks, a cipher-image should possess uniform distribution and close-to-zero correlation of adjacent pixels. Moreover, to avoid the known-plaintext attack and the chosen-plaintext attack, the changes in the cipher-image should be significant even with a small change in the original plain-image; that is, the influence of one-pixel change in the plain-image should be on whole the cipher-image. Two measures, i.e., the number of pixel change rate (NPCR) and unified average changing intensity (UACI) [4], can be adopted to test the influence. The NPCR is used to measure the number of different pixels between two images. UACI is to measure the average intensity difference between two images. In the following, by taking the "Lena" gray image with 256 × 256 pixels, which is a popular image for general image analysis, as a plain-image, the distribution and correlation of adjacent pixels of the plain-image and its cipher-image, and NPCR and UACI of the cipher-images are analyzed.

Distribution: The Lena image and its histogram are plotted in Figs. 9.28(a) and 9.28(b), respectively.



(a) Plain-image

(b) Cipher-image

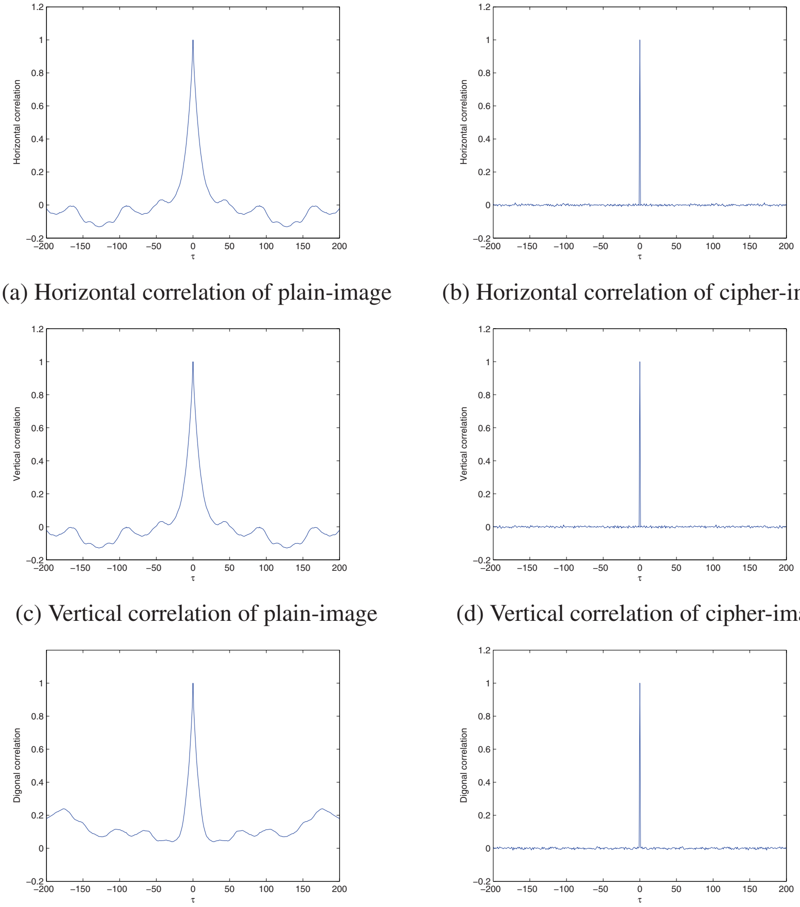
(c) Distribution of Plain-image

(d) Distribution of Cipher-image

Fig. 9.28 Plain-image, cipher-image and their distributions

It is seen that the distribution of the plain-image is not uniform. In addition, the cipher-image and its gray histograms are plotted in Figs. 9.28(c) and 9.28(d), respectively. It is shown that the distribution of the cipher-image becomes uniform. This uniformity is further justified by the χ^2 test. According to Eq. (9.10), o_i is the observed occurrence frequencies of each gray level, which belongs to the range $[0,255]$; e_i is expected occurrence frequencies of each gray level and equal to 256 for the image with 256×256 pixels; k is 255 due to the number of gray levels as 256. χ^2 of the distribution of the cipher-image is computed as 256.09 and smaller than $\chi^2_{255;0:05} = 293.25$ with a significance level of 0.05, which verifies that the distribution of the cipher-image is uniform.

Correlation of adjacent pixels: For an ordinary image, each pixel is usually highly correlated with its adjacent pixels either in horizontal, vertical or diagonal directions. The correlation property in horizontal, vertical and diagonal direction can be quantified by the auto-correlation of the sequence consisting of pixels queuing row by row, column by column and diagonal row by diagonal row, respectively. The three correlation coefficients of the Lena image and those of its cipher-image are plotted in Fig. 9.29. It can be observed that the cipher-image obtained from the cryptosystem retains small correlation coefficients in all directions, which are similar to those of random image.



(a) Horizontal correlation of plain-image (b) Horizontal correlation of cipher-image

(c) Vertical correlation of plain-image (d) Vertical correlation of cipher-image

(e) Diagonal correlation of plain-image (f) Diagonal correlation of cipher-image

Fig. 9.29 Horizontal, vertical and diagonal correlation of plain-image and cipher-image

NPCR: Let $C(i,j)$ and $C'(i,j)$ be the pixel in the ith row and jth column of two images C and C', respectively, the NPCR can be defined as

$$\mathrm{NPCR} = \frac{\sum\limits_{i,j}^{N} D(i,j)}{N} \times 100\%,$$

where N is the total number of pixels in the image and $D(i,j)$ is defined as

$$D(i,j) = \begin{cases} 0, \; C(i,j) = C'(i,j) \\ 1, \; C(i,j) \neq C'(i,j) \end{cases}$$

To check whether the NPCR of a cryptosystem is similar to that of random image, the NPCR of a random image is computed, which is given by

$$NPCR_R = 1 - 2^S,$$

where s is the number of bits representing 256 gray scales, therefore, equal to 8; that is, $NPCR_R = 0.9961$.

The NPCR of the cipher-images encrypted from two images with only one bit difference via the cryptosystem is computed as 0.996048, which is similar to that of random images.

UACI: UACI is defined as

$$\text{UACI} = \frac{1}{N}\left(\sum_{i,j}^{N} \frac{|C(i,j) - C'(i,j)|}{255}\right) \times 100\%$$

Similarly, to check whether the UACI of the cryptosystem is similar to that of random image, the UACI of a random image is computed, which is given by

$$UACI_R = \frac{\frac{1}{2^{2S-1}} \sum_{i=1}^{2^S-1} i(i+1)}{2^S - 1} = 0.3346354,$$

The UACI of the cipher-images encrypted from two images with only one bit difference via the cryptosystem is computed as 0.335383, which is similar to that of random images.

Therefore, the cryptosystem can also encrypt an image well.

In summary, since the plain-media with worse or the worst statistical properties can be encrypted to the cipher-media with random-like statistical properties, the cryptosystem is able to encrypt multimedia effectively.

9.5 Conclusion

Spatiotemporal chaos has advantages to ciphers because of tis inherent characteristics. This chapter has applied a typical spatiotemporal chaotic system, i.e., a CML, in cryptography. Firstly, a multiple-output PRBG using a CML is designed. The statistical properties, such as probability density function (PDF), linear complexity, auto-correlation and cross-correlation of the PRBGs based on various digitization methods have been investigated. It has shown that binary-representation method is the best one. To determine the CMLs, from which the resulting PRBGs have satisfactory properties, six PRBGs based on six different CMLs have been investigated. The six CMLs consist of three simple chaotic systems, i.e., logistic map, skew-tent map and r-adic map, with two simplest coupling methods, i.e., one-way coupling and diffusive coupling, respectively. PDF, auto-correlation, cross-correlation, statistical test and cycle length of the six PRBGs with various parameters have been investigated so as to determine the parameter intervals within which the PRBGs have satisfactory properties. It has been indicated that a one-way coupled logistic-map lattice with certain parameters has the best properties. This research results in criteria for designing PRBGs with proper performance. Secondly, a stream cipher employing a one-way coupled logistic-map lattice with certain parameters has been designed. The security of the stream cipher has been tested by attacking it via typical attack methods and analyzing its cryptographic properties. Moreover, the efficiency of the stream cipher has been analyzed. It has shown that the cipher has higher security, higher efficiency and lower costs by comparing with Hu's stream cipher of a complicated configuration. Finally, a multimedia cryptosystem based on the proposed stream cipher has been designed and implemented in a FPGA. The EPP is used to communicate data between a PC and the FPGA. A user-friendly interface is designed with Visual C++ 6.0, with which text, image and audio can be encrypted and decrypted successfully. The properties of the cryptosystem, such as the sensitivity to the key, speed and efficiency of the FPGA, have been analyzed to be satisfactory.

References

1. Baranovsky, A., Daems, D.: Design of one-dimensional chaotic maps with prescribed statistical properties. Int. J. Bifurcat Chaos Appl. Sci. Eng. 5, 1585–1598 (1995)
2. Batista, A.M., Pinto, S.E., Viana, R.L., Lopes, S.R.: Lyapunov spectrum and synchronization of piecewise linear map lattiecs with power-law coupling. Phys. Rev. E 65 (2002)
3. Baptista, M.S.: Cryptography with chaos. Phys. Lett. A 240, 50–54 (1999)
4. Chen, G., Mao, Y., Chui, C.: A symmetric image encryption scheme based on 3rd chaotic cat maps. Chaos, Solitons & Fractals 21, 749–761 (2003)
5. Garcia, P., Parravano, A., Cosenza, M., Jimenez, J., Marcano, A.: Coupled map networks as communication schemes. Phys. Rev. E 65, 195–201 (2002)
6. Gotz, M., Kelber, K., Schwarz, W.: Discrete-time chaotic encryption systems-part i: Statistical design approach. IEEE Trans. Circ. Syst. Fund. Theor. Appl. 44(10), 963–970 (1997)
7. Kaneko, K.: Theory and Application of Coupled Map Lattices. John Wiley and Sons, New York (1993)
8. Kocarev, L., Jakimoski, G.: Pseudorandom bits generated by chaotic maps. IEEE Trans. Circ. Syst. Fund. Theor. Appl. 50, 123–126 (2003)
9. Kohda, T., Tsuneda, A.: Pseudonoise sequence by chaotic nonlinear maps and their correlation properties. IEICE Trans. Commun. E76-B, 855–862 (1993)
10. Lasota, A., Mackey, M.C.: Chaos, Fractals, and Noise: stochastic aspects of dynamics. Springer, New York (1997)
11. Li, P., Li, Z., Halang, W.A., Chen, G.R.: Analysis of a multiple output pseudo-random-bit generator based on a spatiotemporal chaotic system. Int. J. Bifurcat Chaos Appl. Sci. Eng. 16(10), 2949–2963 (2006)
12. Li, P., Li, Z., Halang, W.A., Chen, G.R.: A stream cipher based on a spatiotemporal chaotic system. Chaos, Solitons & Fractals 32(5), 1867–1876 (2007)
13. Li, S.J.: Analyses and New Designs of Digital Chaotic Ciphers. Ph.D thesis, School of Electronics and Information Engineering, Xi'an Jiaotong University, Xi'an, China (2003)

14. Shujun, L., Xuanqin, M., Yuanlong, C.: Pseudo-random bit generator based on couple chaotic systems and its applications in stream-cipher cryptography. In: Pandu Rangan, C., Ding, C. (eds.) INDOCRYPT 2001. LNCS, vol. 2247, pp. 316–329. Springer, Heidelberg (2001)
15. Li, S.J., Chen, G.R., Qin, M.: On the dynamical degradation of digital piecewise linear chaotic maps. Int. J. Bifurcat Chaos Appl. Sci. Eng. 15(10), 3119–3151 (2005)
16. Li, S., Álvarez, G., Chen, G.R.: Breaking a chaos-based secure communication scheme designed by an improved modulation method. Chaos, Soliton & Fractals 25, 109–120 (2005)
17. Lu, H., Wang, S., Li, X., Tang, G., Kuang, J., Ye, W., Hu, G.: A new spatiotemporally chaotic cryptosystem and its security and performance analyses. Chaos 14(3), 617–629 (2004)
18. Masuda, N., Aihara, K.: Cryptosystems with discretized chaotic maps. IEEE Trans. Circ. Syst. Fund. Theor. Appl. 49(1), 28–40 (2002)
19. Menezes, A., Oorschot, P.V., Vanstone, S.: Handbook of Applied Cryptography. CRC Press, Boca Raton (1997)
20. NIST, Security requirements for cryptographic modules (FIPS pub 140-2) (2001), `http://csrc.nist.gov/publications/fips/fips140-2`
21. Sang, T., Wang, R., Yan, Y.: Clock-controlled chaotic keystream generators. Electronics Letters 34(20), 1932–1934 (1998)
22. Schneier, B.: Applied Cryptography: Protocols, algorithms, and source code in C. John Wiley and Sons, New York (1996)
23. Schuster, H.G.: Handbook of Chaos Control. WILEY-VCH, Weinheim (1999)
24. Soto, J.: Statistical testing of random number generators (1999), `http://csrc.nist.gov/rng/rng5.html`
25. Stojanovski, T., Kocarev, L.: Chaos-based random number generators-part i: Analysis. IEEE Trans. Circ. Syst. Fund. Theor. Appl. 48(3), 281–288 (2001)
26. Tang, G., Wang, S., Lu, H., Hu, G.: Chaos-based cryptograph incorporated with S-box algebraic operation. Phys. Lett. A 318, 388–398 (2003)
27. Wang, S., Liu, W., Lu, H., Kuang, J., Hu, G.: Periodicity of chaotic trajectories in realizations of finite computer precisions and its implication in chaos communications. Int. J. Mod. Phys. B 18(17-19), 2617–2622 (2004)
28. Wang, S., Ye, W., Lu, H., Kuang, J., Li, J., Luo, Y., Hu, G.: A spatiotemporal-chaos-based encryption having overall properties considerably better than advanced encryption standard. Comm. Theor. Phys. 40, 57–60 (2003)
29. Wikipedia (2006) Chi Test, `http://en.wikipedia.org/wiki/Pearson%7s-chi-square-test`
30. Yang, T.: A survey of chaotic secure communication systems. International Journal of Computational Cognition 2(2), 81–130 (2004)
31. Ye, W., Dai, Q., Wang, S., Lu, H., Kuang, J., Zhao, Z., Zhu, X., Tang, G., Huang, R., Hu, G.: Experimental realization of a highly secure chaos communication under strong channel noise. Phys. Lett. A 330, 75–84 (2004)
32. Zhang, H., Wang, H., Chen, W.: Oversampled chaotic binary sequences with good security. J. Circ. Syst. Comput. 11, 173–185 (2002)
33. Zhou, H., Ling, X.: Problems with the chaotic inverse system encryption approach. IEEE Trans. Circ. Syst. Fund. Theor. Appl. 44(3), 268–271 (1997)