A Generalized Stability Theorem for Discrete Chaos Systems with Application in Avalanche Image Encryption

Xue Wang[a], Lequan Min[b*] and E Chen[c]

Department of Information and Computer Sciences, College of Mathematics and Physics

University of Science and Technology Beijing, Beijing 100083, China

[a]wangxue_20130818@163.com, [b]13501029489@163.com, [c]chene5546@163.com

Keywords: Generalized stability; constructive theorem; 4D chaotic map; pseudorandom number generator; FIPS 140-2 test; avalanche encryption scheme

Abstract. This study first proposes a concept of generalized stability (GST) for discrete chaos system, which is the generalization of chaos generalized synchronization (CGS). Then this study sets up a constructive theorem of GST for discrete chaos system, which provides a general representation of GST in discrete chaos system. Using the theorem designs an 8-dimensional GST system consisting of a driving chaotic system and a driven chaotic system. Numerical simulation verifies the chaotic dynamic behaviors of such GST system, which is used to design a chaotic pseudorandom number generator (CPRNG). Using FIPS 140-2 test suite and G FIPS 140-2 test suite test the randomness of four 1,000-key streams consisting of 20,000 bits generated respectively by the CPRNG, the RC4 algorithm and the ZUC algorithm. The results show that the randomness performances of the CPRNG is promising, and suggest that the statistical properties of the randomness of the sequences generated via the CPRNG and the two algorithms do not have significant differences. As an application, using the sequences generated via the CPRNG and a stream encryption scheme with avalanche effect (SESAE) encrypts an RGB image. The results show that the encrypted RGB image have significant avalanche effects, and suggest the CPRNG is a qualified candidate for the stream encryption scheme with avalanche effect.

Introduction

Chaos is one type of complex dynamic behaviors and generated from determined nonlinear discrete or continuous systems. Chaotic dynamics are highly sensitive to the initial conditions and the parameters of the chaos systems. Chaotic behaviors are unpredictable for long terms ([1, 2]).

Chaos synchronization (CS) have been one of the major issues in many physical, biological and technological fields. Since the seminar paper of Pecora and Carroll [3] on chaos synchronization communications, the research on chaos synchronization-based communications has been attracted much attention ([4-9]). As a generalization, chaos generalized synchronization (CGS) means that the trajectories of two different systems trend to each other with respect to a transformation starting from different initial conditions in a specific domain. The study of generalized synchronization has also got extensive attention ([5-15]). Chaos generalized synchronization may provide some new tools for cryptography and communications ([16-21]).

Pseudorandom number sequences are useful in many fields such as simulations of physical systems and computer simulation, particularly cryptography ([22, 23]). Today algorithmic pseudorandom number generators (PRNGs) have replaced almost all random number tables and hardware random number generators in practical applications ([24-26]).

The main aims of this paper are to extend the concept of the generalized synchronization to generalized stability (GST) for discrete chaos system, and set up the corresponding GST theorem. Based on the GST theorem, this study introduces a novel chaotic discrete map, constructs a chaotic pseudorandom number generator (CPRNG) with a large key space and sound pseudo randomness. The FIPS 140-2 test suite and G FIPS 140-2 test suite [27] are used to test the randomness of the CPRNG, the RC4 algorithm and the ZUC algorithm [28], respectively. As an application, using the CPRNG and the stream encryption scheme with avalanche effect (SESAE) encrypts a RGB image.

Definition and Theorem on GST

A point of view states that two events with relationship of cause and effect might be described via two generalized synchronization systems. Motivated by CGS, we introduce the concept of GST for two systems, which is an extension of the concept of generalized synchronization. The definition of the GST is described as follows.

Definition 1: Consider two systems

$$\mathbf{X}(k+1) = F(\mathbf{X}(k)), \tag{1}$$

$$\mathbf{Y}(k+1) = G(\mathbf{Y}(k), \mathbf{X}(k)), \tag{2}$$

where

$$\mathbf{X}(k) = (x_1(k), \ldots, x_n(k))^T, \tag{3}$$

$$\mathbf{Y}(k) = (y_1(k), \ldots, y_m(k))^T, m \le n, \tag{4}$$

$$F(\mathbf{X}(k)) = (f_1(\mathbf{X}(k)), \ldots,, f_n(\mathbf{X}(k)))^T, \tag{5}$$

$$G(\mathbf{Y}(k), \mathbf{X}(k)) = (g_1(\mathbf{Y}(k), \mathbf{X}(k)) \ldots,, g_m(\mathbf{Y}(k), \mathbf{X}(k)))^T. \tag{6}$$

If there exists a transformation

$$H : \mathbb{R}^n \to \mathbb{R}^m, \tag{7}$$

$$H(\mathbf{X}(k)) = (h_1(\mathbf{X}(k)), \cdots, h_m(\mathbf{X}(k)))^T, \tag{8}$$

and $(\mathbf{X}(0), \mathbf{Y}(0)) \in \mathbb{R}^n \times \mathbb{R}^m$, for $\forall \grave{o} > 0$ there exists $\delta_1 > 0$ and $\delta_2 > 0$ such that all trajectories of (1) and (2) with initial conditions $(\mathbf{X}(0), \mathbf{Y}(0)) \in B(\mathbf{X}_0, \delta_1) \times B(\mathbf{Y}_0, \delta_2) \subset \mathbb{R}^n \times \mathbb{R}^m$ satisfy:

$$\| H(\mathbf{X}_m(k)) - \mathbf{Y}(k) \| < \grave{o}, \quad k \to \infty, \tag{9}$$

where

$$\mathbf{X}_m(k) = (x_1(k), \cdots, x_m(k))^{\mathrm{T}}.$$

Then the systems (1) and (2) are said to be in GST with respect to the transformation H. System (1) is called the driving system, system (2) is said to be the driven system.

Now a general problem is: if two systems can achieve GST, what kinds of representations should these systems have? To answer this question, we propose the following constructive theorem:

Theorem 1: $\mathbf{X}, \mathbf{Y}, \mathbf{X}_m, F(\mathbf{X})$ and $G(\mathbf{Y}, \mathbf{X})$ be defined by (3)-(6), suppose that

$$H(x_1, x_2, \ldots, x_m) = (y_1, y_2, \ldots, y_m),$$

If the two systems (1) and (2) are in GST via the transformation $\mathbf{Y} = H(\mathbf{X}_m)$, if, and only if, the driven system function $G(\mathbf{Y}, \mathbf{X})$ given in (2) has the following form:

$$G(\mathbf{Y}, \mathbf{X}) = H\left[F_m(\mathbf{X})\right] - q(\mathbf{X}_m, \mathbf{Y}),$$

where

$$F_m(\mathbf{X}) = \left(f_1(\mathbf{X}), f_2(\mathbf{X}), \cdots, f_m(\mathbf{X})\right)^T,$$

and the function

$$q(\mathbf{X}_m, \mathbf{Y}) = (q_1(\mathbf{X}_m, \mathbf{Y}), q_2(\mathbf{X}_m, \mathbf{Y}), \ldots, q_m(\mathbf{X}_m, \mathbf{Y}))^T,$$

guarantees that the zero solution of the following error equation is stable on the open set $B(\mathbf{X}_0, \delta_1) \times B(\mathbf{Y}_0, \delta_2)$:

$$\mathbf{e}(k+1) = H\left(\mathbf{X}_m(k)\right) - \mathbf{Y}(k+1) = q(\mathbf{X}_m, \mathbf{Y}). \tag{10}$$

Proof: Denote

$$G(\mathbf{Y}, \mathbf{X}) - H\left[F_m(\mathbf{X})\right] = -q(\mathbf{X}_m, \mathbf{Y}),$$

Then

$$\mathbf{e}(k+1) = H\left(\mathbf{X}_m(k+1)\right) - \mathbf{Y}(k+1) = q(\mathbf{X}_m, \mathbf{Y}).$$

Therefore, two dynamic systems (1) and (2) are in GST via the transformations H, if and only if the function $q(\mathbf{X}_m, \mathbf{Y})$ makes the trajectory in (10) tends to zero solution stable. This completes the proof. ■

Remark 1: This theorem provides a general approach to construct a discrete GST system.

A novel Chaotic Map and GST System

In this section, using the GST theorem constructs a discrete GST system. Firstly, we propose a novel chaotic system:

$$\begin{cases} x_1(k+1) = 3\sin x_1(k) + 2\sin x_2(k)\cos x_3(k) \\ x_2(k+1) = -2\sin x_1(k) + 7\sin x_2(k) \\ x_3(k+1) = 4\sin x_3(k) - x_2(k) \\ x_4(k+1) = 4\sin\left(x_3(k) + x_4(k)\right). \end{cases} \tag{11}$$

The calculated Lyapunov exponents of this new system are {1.2433, 0.8209, 0.6974, 0.2650}. Hence system (11) is a chaitic system. Now, select the following initial conditions:

$$X(0) = (-0.112, 0.245, 1.501, 0.659)^T. \tag{12}$$

The orbits of the state variables x_1, x_2 x_3, x_4 for the first 5000 iterations are shown in Figs. 1(a)-1(d). A systematic computer simulation shows that the orbits of system (11) display chaotic characteristics as theory expects.

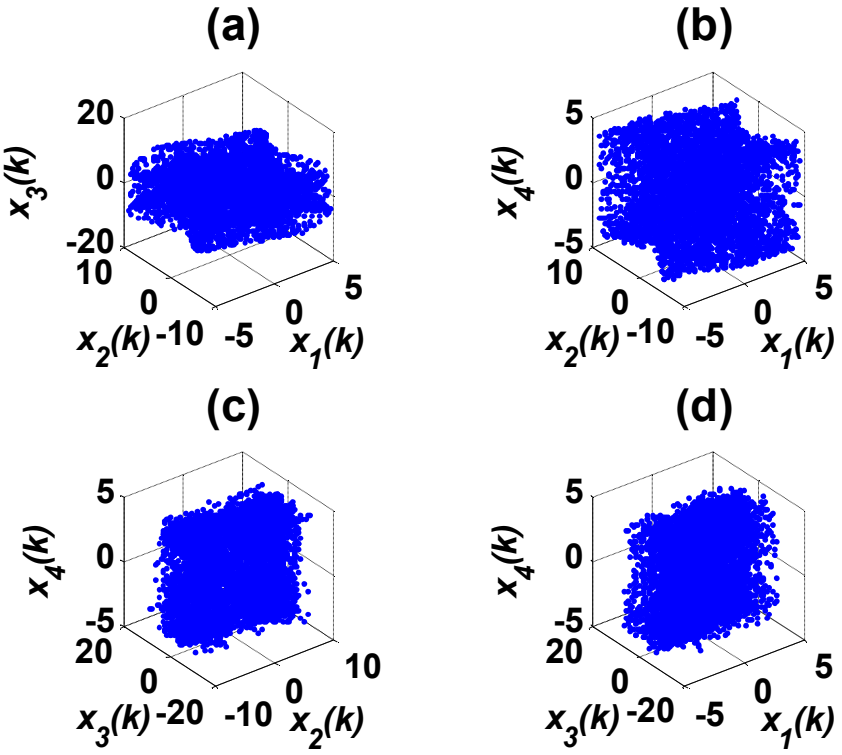


Figure 1. Chaotic trajectories of variables: (a) $x_1(k) - x_2(k) - x_3(k)$, (b) $x_1(k) - x_2(k) - x_4(k)$, (c) $x_2(k) - x_3(k) - x_4(k)$, and (d) $x_1(k) - x_3(k) - x_4(k)$.

Secondly, construct an invertible matrix

$$A = \begin{pmatrix} 9 & 4 & -7 & 2 \\ 4 & -5 & 3 & 1 \\ 5 & 2 & 0 & 3 \\ 3 & 1 & -2 & 7 \end{pmatrix}, \tag{13}$$

define a transformation $H:\mathbb{R}^4 \to \mathbb{R}^4$ as follows

$$H(\mathbf{X}) = A\mathbf{X} \triangleq (h_1(\mathbf{X}), h_2(\mathbf{X}), h_3(\mathbf{X}), h_4(\mathbf{X}))^{\mathrm{T}}. \tag{14}$$

Motivated by the case in [29], we consider the following system

$$\begin{cases} e_1(k+1) = e_2(k) \\ e_2(k+1) = -\dfrac{e_1(k)}{1.01} \\ e_3(k+1) = e_4(k) \\ e_4(k+1) = -\dfrac{e_3(k)}{1.01}. \end{cases} \tag{15}$$

Then we will show that the error equation

$$\mathbf{e}(k+1) \triangleq \left(e_1(k+1), e_2(k+1), e_3(k+1), e_4(k+1)\right),$$

is zero solution stable. In order to prove equation (15) is zero stable, we need the following lemma

Lemma 1: **(**[30]) If there exists a positive definite function $V(e_1, e_2, e_3, e_4)$ such that $\Delta V(e_1, e_2, e_3, e_4)$ is negative semidefinite, then the equation (15) is zero stable.

Proof of the zero solution stable of system (15): according to Lemma 1, we can construct a Lyapunov function

$$V(e_1, e_2, e_3, e_4) = e_1^2 + e_2^2 + e_3^2 + e_4^2.$$

Then

$$\begin{aligned} \Delta V(e_1, e_2, e_3, e_4) &= V\left(e_1(k+1), e_2(k+1), e_3(k+1), e_4(k+1)\right) - V\left(e_1(k), e_2(k), e_3(k), e_4(k)\right) \\ &= e_2^2(k) + \frac{1}{1.0201} e_1^2(k) + e_4^2(k) + \frac{1}{1.0201} e_3^2(k) - e_1^2(k) - e_2^2(k) - e_3^2(k) - e_4^2(k) \\ &= -\frac{0.0201}{1.0201} e_1^2(k) - \frac{0.0201}{1.0201} e_3^2(k) \le 0, \quad for \ \mathbf{e} \ne 0 \end{aligned}$$

Therefore, the system (15) is zero solution stable.

Let

$$q\left(\mathbf{X}(k), \mathbf{Y}(k)\right) = \left(e_2(k), -e_1(k)/1.01, e_4(k), -e_3(k)/1.01\right)^T,$$

where

$$\mathbf{e}(k) = H\left(\mathbf{X}_m(k)\right) - \mathbf{Y}(k).$$

Select

$$\mathbf{Y}(k+1) = A[F\left(\mathbf{X}(k+1)\right)] - q\left(\mathbf{X}(k), \mathbf{Y}(k)\right), \tag{16}$$

as a driven system. Then from Theorem 1, system (11) and (16) are GST with respect to the transformation H.

Now we choose (17) as the initial conditions

$$Y(0) = (-30.456, -40.708, 0.543, -17.467)^T. \tag{17}$$

The chaotic orbits of the state variables y_1, y_2, y_3, y_4 for the first 5000 iterations are shown in Figs. 2(a)-2(d). The simulation results show that the system has chaotic attractor characteristics.

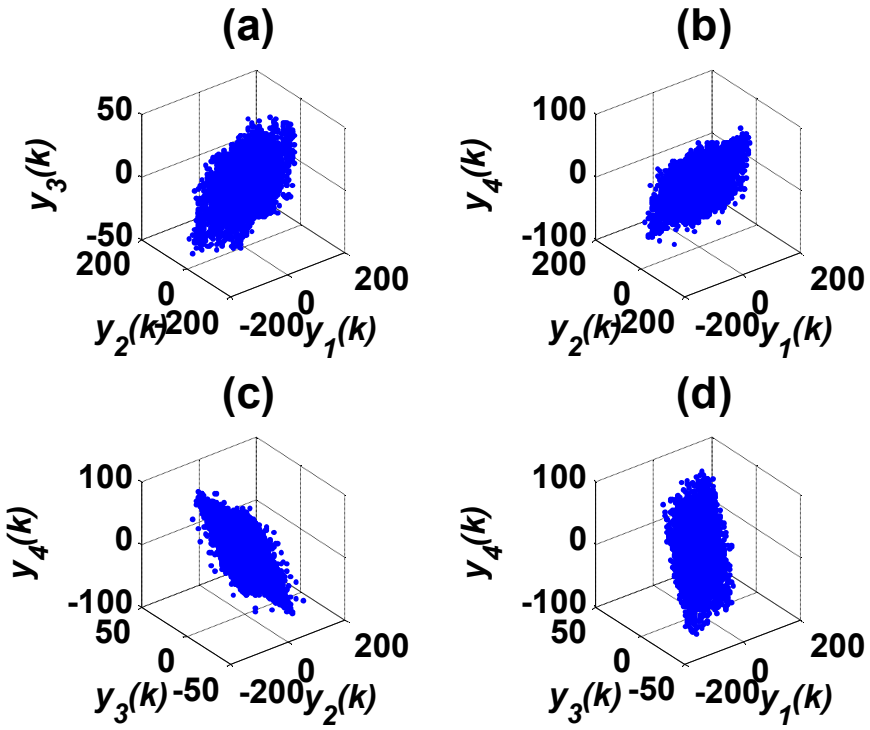


Figure 2. Chaotic trajectories of variables: (a) $y_1(k)-y_2(k)-y_3(k)$, (b) $y_1(k)-y_2(k)-y_4(k)$, (c) $y_2(k)-y_3(k)-y_4(k)$, and (d) $y_1(k)-y_3(k)-y_4(k)$.

Extensive simulations show that the dynamic behaviors of the GST system have chaotic attractor characteristics. Figs. 3(a)-3(d) show that $\mathbf{X}(k)$ and $\mathbf{Y}(k)$ are in GST with respect to transformation $H = A$, as the theory expects.

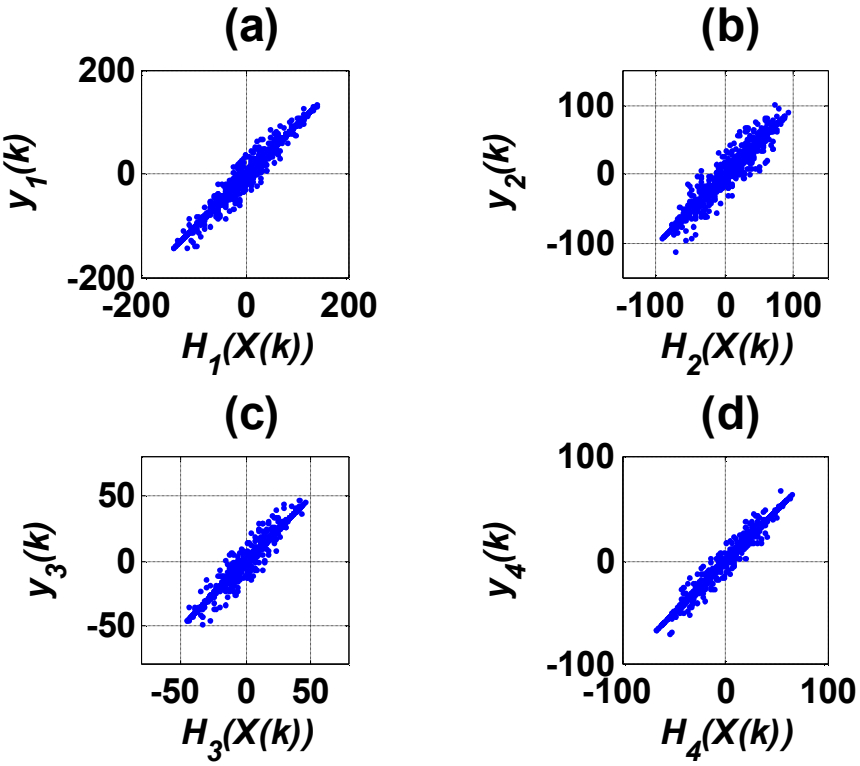


Figure 3. Chaotic trajectories of variables: (a) $h_1(\mathbf{X}(k))-y_1(k)$, (b) $h_2(\mathbf{X}(k))-y_2(k)$, (c) $h_3(\mathbf{X}(k))-y_3(k)$, and (d) $h_4(\mathbf{X}(k))-y_4(k)$.

Chaotic Pseudorandom Number Generator

Pseudorandom Number Generator

Denote

$$\mathbf{X}_i = \{x_i(k) \mid k = 1, 2, \cdots, N\}, \tag{18}$$

$$\mathbf{Y}_i = \{y_i(k) \mid k = 1, 2, \cdots, N\}, \tag{19}$$

where x_i and y_i are defined by (11) and (16). Firstly, introduce a transformation $T_1 : \mathbb{R} \to \{0, 1, \cdots, 2^{16} - 1\}$, which transforms the chaotic streams of GST systems (18) and (19) into key streams.

Let $L = 10^{15}$, then

$$C1 = \mathrm{mod}\left(round\left(L\left(\mathbf{X}_1 - \min\left(\mathbf{X}_1\right)\right) / \left(\max\left(\mathbf{X}_1\right) - \min\left(\mathbf{X}_1\right)\right)\right), 2^{16}\right). \tag{20}$$

$$C2 = \mathrm{mod}\left(round\left(L\left(\mathbf{Y}_1 - \min\left(\mathbf{Y}_1\right)\right) / \left(\max\left(\mathbf{Y}_1\right) - \min\left(\mathbf{Y}_1\right)\right)\right), 2^{16}\right). \tag{21}$$

Therefore, the T_1 is defined by

$$C0 = T_1\left(\mathbf{X}_1, \mathbf{Y}_1\right) = \mathrm{mod}\left(C1 + C2, 2^{16}\right). \tag{22}$$

Now we can design a CPRNG based on the transformation (20)-(22) and the GST systems (11) and (16). The seeds of the CPRNG are the initial conditions of the GST systems, which can be chosen via random number generators. Therefore the output key streams of the CPRNG can be obtained via transformation (22) acting on the chaotic streams of the GST systems (11) and (16).

Pseudorandomness Test

The FIPS 140-2 test consists of four sub-tests: Monobit Test, Poker Test, Runs Test and Long Runs Test. Each test needs a single stream of 20,000 one and zero bits from the keystream generator. Any failure in the first three tests means that the corresponding quantity of the sequences falls out the required intervals listed in the second column in Table 1. The Long Runs test is passed if there are no runs of length 26 or more.

It has been pointed out that the required intervals of the Monobit test, the Porker test and the Runs test correspond to the confident interval with significant levels: $\alpha = 10^{-4}, 10^{-4}$ and 1.6×10^{-7} (approximatly), respectively ([27], [31]). The required intervals of the Runs test with significant levels: $\alpha = 10^{-4}$ are listed in the third column in Table 1. We denote the accepted intervals by G FIPS 140-2 test.

According to Golomb's three postulates on the randomness that ideal pseudorandom sequences should satisfy [32], the ideal values of the first three tests are listed in the 4th column in Table 1.

The FIPS 140-2 test suit and G FIPS 140-2 test suit are used to test 1,000 keystreams randomly generated, respectively by CPRNG with perturbed randomly initial condition $\mathbf{X}(0)$, $\mathbf{Y}(0)$ and the parameters of matrix (13) in the range $|\dot{o}| \in [10^{-16}, 10^{-1}]$.

Table 1 The required intervals of the FIPS 140-2 Monobit Test, Porker Test and Runs Test. Here MT, PT, and LT represent the Monobit Test, the Porker Test and the Long Runs Test. k represents the length of the run of a tested sequence. $\chi^2 DT$ represents χ^2 distribution.

Test	FIPS 140-2 Standand	α =0.0001	Golomb's
Item	Required Intervals	Required Intervals	Postulates
MT	9,725 ~ 10,275	9,725 ~ 10,275	10000
PT	2.16 ~ 46.17	2.41 ~ 44.26	$\chi^2 DT$
LT	<26	<26	—
k	Run Test	Run Test	Run Test
1	2,315 ~ 2,685	2,362 ~ 2,638	2,500
2	1,114 ~ 1,386	1,153 ~ 1,347	1,250
3	527 ~ 723	556 ~ 694	625
4	240 ~ 384	264 ~ 361	313
5	103 ~ 209	122 ~ 191	156
6+	103 ~ 209	122 ~ 191	156

In order to test the pseudorandomness of the CPRNG, we transform the "16-bit" stream defined by (22) to the $\{0,1\}$ bit stream as follows.

Construct a transform $T_2: \{0,1,\cdots,2^{16}-1\} \to \{0,1\}$ which is defined by

$$T_2 = T_{22} \circ T_{21}, \tag{23}$$

$s.t. \forall \mathbf{y} \in \{0,1,\cdots,2^{16}-1\}^N$

$T_{21}(\mathbf{y}) = dec2bin(\mathbf{y})$.

Let $\mathbf{z} = dec2bin(\mathbf{y})$, then

$T_{22}(\mathbf{z}) = \mathbf{z}(:)$,

where dec2bin and $\mathbf{z}(:)$ are both Matlab commands.

Finally the transformation $T: \mathbb{R}^4 \to \{0,1\}$ is defined via

$$T = T_2 \circ T_1. \tag{24}$$

All sequences successfully pass the FIPS 140-2 test and there are 13 sequences failing to pass the G FIPS 140-2 test. The calculated results are listed in the 3rd column in Table 2, in which the statistic results of all tests are described by mean values $\pm$ standard deviation (Mean $\pm$ SD).

The well-known RC4 was designed by Rivest of the RSA Security in 1987, which has been widely used in popular protocols such as Secure Sockets. The RC4 Algorithm as PRNG can be designed via the following Matlab commands:

```
N=20000;
K=randi([0 254],1,255);
S=[0:255-1];j=0;
for i=1:255
    j=mod(j+S(i)+K(i),255);
    Sk=S(j+1);
    S(j+1)=S(i);
    S(i)=Sk;
end
C=zeros(1,N); j=0;i=0; k=1;
for l=1:N/8
    i=mod(i+1,255);
    j=mod(j+S(i+1),255);
    Sk=S(j+1);
    S(j+1)=S(i+1);
    S(i+1)=Sk;
    C(l)=S(mod(S(j+1)+S(i+1),255)+1);
end
C=(dec2bin(C))';
C=C(:);
C=bin2dec(C);
```

Here, "randi([0 254], 1, 255)" generates a vector of uniformly distributed random integers $\{0,1,\cdots,254\}$ of dimension 255; "mod" means taking modulus after division; "zeros(1,N)" is a zero row vector of dimension N. Consequently, the RC4 Algorithm-based PRNG is designed. Then, the FIPS 140-2 test suite and G FIPS 140-2 test suite are used to test 1,000 keystreams randomly generated by the RC4 PRNG. Results show that there is only one sequence failing to pass the FIPS 140-2 test, and there are 12 sequences failing to pass the G FIPS 140-2 test criterions. The statistic test results are shown in the 4th column in Table 2.

Finally, ZUC algorithm is a stream cipher that forms the heart of the third generation partnership project (3GPP) confidentiality algorithm 128-EEA3 and the 3GPP integrity algorithm 128-EIA3. Now, using FIPS 140-2 and G FIPS 140-2 test the 1,000 keystreams randomly generated by the ZUC algorithm program (see Appendix A in [28]). Results show that all of the 1,000 sequences passed the FIPS 140-2 test criterions, and there are 21 sequences failing to pass the G FIPS 140-2 test criterions. The statistic test results are listed in the 5th column in Table 2.

It can be seen that the statistical properties of the randomness of the sequences generated via the CPRNG, the RC4 algorithm and the ZUC algorithm do not have significant differences..

Table 2 The confident intervals of the FIPS 140-2 tested values of 1,000 key streams generated by the new CPRNG, the RC4 PRNG and the ZUC algorithm with significant level $\alpha = 0.0001$. Here, SD represents the standard diviation.

Test item	bits	CPRNG	RC4	ZUC
		Mean ± SD	Mean ± SD	Mean ± SD
MT	0	10000.00 ± 68.210	9999.7 ± 70.092	9998.4 ± 71.843
	1	9999.99 ± 68.210	10000 ± 70.092	10002 ± 71.843
PT	—	14.944 ± 5.3292	14.87 ± 5.433	15.043 ± 5.5491
LT	0	13.738 ± 1.9766	13.6 ± 1.8214	13.488 ± 1.829
	1	13.607 ± 1.8862	13.642 ± 1.9307	13.595 ± 1.9305
1	0	2499.5 ± 47.733	2500.9 ± 45.568	2501.9 ± 45.735
	1	2498.5 ± 47.721	2501.4 ± 46.398	2502.7 ± 46.121
2	0	1248.77 ± 31.167	1250.5 ± 31.372	1252.1 ± 32.606
	1	1250.13 ± 32.143	1249 ± 31.048	1249.5 ± 32.221
3	0	624.99 ± 22.721	624.95 ± 22.964	624.09 ± 22.648
	1	624.98 ± 22.956	625.65 ± 22.93	624.64 ± 23.455
4	0	312.51 ± 16.698	311.71 ± 16.548	312.56 ± 16.748
	1	312.04 ± 16.667	312.17 ± 16.822	312.72 ± 16.506
5	0	155.90 ± 12.208	156.41 ± 12.069	155.65 ± 12.097
	1	155.80 ± 11.900	156.6 ± 11.958	156.66 ± 12.369
6+	0	156.69 ± 11.808	156.15 ± 11.792	155.75 ± 11.719
	1	156.63 ± 11.433	155.79 ± 11.979	155.82 ± 11.497

Key Space

The key set parameters of CPRNG includes the initial condition $X(0), Y(0)$ and the matrix $A = (\alpha_{i,j})$. It can be proved that if the perturbation matrix $\Delta = (\delta_{i,j})$ satisfies

$$|\delta_{i,j}| < 0.5907,$$

the matrix $A + \Delta$ is still invertible. Therefore the CPRNG have 4 + 4 + 16 key parameters denoted by

$$\mathbf{K}_s = \{k_1, k_2, \cdots, k_{24}\}. \tag{25}$$

Let the key set be perturbed by

$$\mathbf{K}_s(\Delta) = \mathbf{K}_s + [\delta_1, \delta_2, \ldots, \delta_{24}], \tag{26}$$

where

$$10^{-16} \le |\delta_i| \le 10^{-1}, \ i = 1, \cdots, 24.$$

Now we compare the difference between the key stream S with 20,000 codes' length generated by the key set (25) and the key streams $S_p's$ generated by the perturbed key set (26), respectively.The comparison results are shown in the third column in Table 3, where SV denotes the statistic values, DC the different codes, and CC the correlation coefficients.

The results show that the average percentage of different codes is 49.9982%, which is very closed to the ideal value of 50%. And the average of the correlation coefficients is 5.6190e-3, also very closed to the ideal value of 0.

Now, compare the same key stream S with the 1000 key streams $S_m's$ generated by the Matlab command randi([0 1], 1, 20000). The comparison results are shown in the fourth column in Table 3. Observed that the average percentage of different codes is 50.0145% and the the average of the correlation coefficients is 5.7424e-3. The results suggest that the key stream S has no significant correlations with the perturbed key streams $S_p's$. The Matlab platform uses double-precision decimal

computations, which means that each computed decimal number has 16 bits' accuracy. In summary, the key space of the CPRNG is larger than $10^{15\times 24} > 2^{1195}$.

Table 3 The statistic data for the percentages and correlation coefficients of the codes of the key stream variations between S and $S_p's$ as well as S and $S_m's$.

Item	SV	$S_p's$	$S_m's$
DC	min mean max	48.9600% 49.9982% 51.1000%	48.9350% 50.0145% 51.2199%
CC	min mean max	2.0800e-6 5.6190e-3 2.1962e-2	4.1600e-6 5.7424e-3 2.4359e-2

Simulation on SESAE

In a recent paper [33], we have introduced a new encryption scheme with avalanche effect:

Definition 2: Let $P=\{p_1,p_2,\ldots,p_n\}$ be a binary key stream with d-bit segments generated by a CPRNG, $M=\{m_1,m_2,\ldots,m_n\}$ be a binary plaintext stream, and $C=\{c_1,c_2,\ldots,c_n\}$ be a ciphertext stream. Then, the stream encryption scheme with avalanche effect (SESAE) is described as follows.

(1) The ciphertext $C=E(M,P)$ is defined by

$$c_i=\begin{cases} p_i & if\ \ m_i=0,\\ \sim p_i & if\ \ m_i=1, \end{cases} \tag{27}$$

where $\sim p_i$ is defined to be the bit string obtained by replacing all $0_s'$ in p_i' with $1_s'$, and all $1_s'$ in p_i' with $0_s'$.

(2) The corresponding decrypted plaintext $M=E^{-1}(C,P)$ is determined by

$$m_i=\begin{cases} 0 & if\ \ c_i=p_i,\\ 1 & if\ \ c_i\neq p_i. \end{cases} \tag{28}$$

Definition 3: ([33]) A CPRNG, S, which generates d-bit key streams, is called an ideal CPRNG, if S has the following properties:

(1) The period of any key stream generated by the PRNG is larger than 2^d. Its seed space and key space are both larger than 2^{512}.

(2) In one period of a pseudorandom key streams generated by the PRNG, the distributions of different d-bit segments in the key stream is homogenous. That is, if the period $p=n\times 2^d$, then the number of each different d-bit segment is equal to n. If the the period p is not an integer multiple of 2^d, then the difference between the numbers of different d-bit segments is at most one.

(3) The two key streams P_1,P_2 generated by any two different seeds have $(2^d-1)/2^d\times 100\%$ different d-bit segments.

Now, we use the CPRNG to investigate SESAE on an RGB image Lotus with 250×140 pixels as shown in Fig. 4(a). The simulation is implemented via the Matlab 2013a platform on a PC computer. The simulations procedures are described below:

(1) Transform the image Lena to a binary plaintext steam $M=\{m_1,m_2,\ldots,m_n\}$, where $n=250\times 140\times 3\times 8$.

(2) Use the CPRNG with initial conditions (12) and (17) to generate a key stream $P = \{p_1, p_2, \ldots, p_{n+1000}\}$.

(3) Drop the first 1000 iterative values, and use formula (27) and the key stream P to encrypt the plaintext steam M, and obtain a ciphertext $C = E(M, P)$.

(4) Using formula (28) and the key stream to decrypt the ciphertext, and obtain a decrypted plaintext image $\bar{M} = E^{-1}(C, P)$ without errors (see Fig. 4(b)).

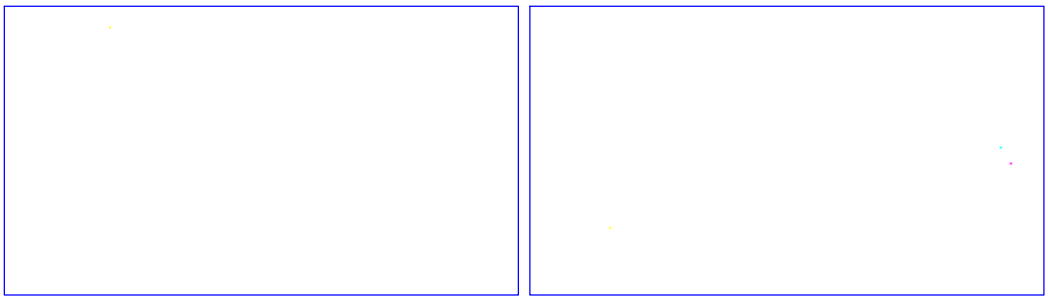
(5) Randomly disturb the initial conditions (12) and (17), and matrix (13) for 1000 times in the range $|\grave{o}| \in [10^{-16}, 10^{-1}]$, and obtain disturbed key streams (Dropping the first 1000 iterative values): $P_i, i = 1, 2, \ldots, 1000$.

(6) Use $\{P_1, \ldots, P_{1000}\}$ to decrypt the ciphertext, and obtain the decrypted plaintext: $\overline{M_i} = E^{-1}(C, P_i), i = 1, 2, \ldots, 1000$.

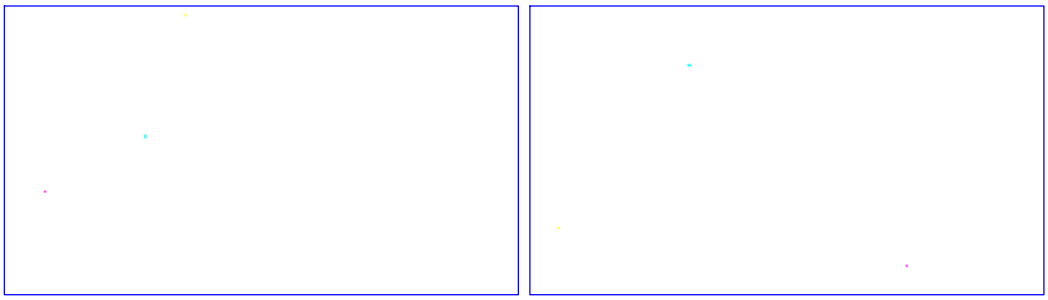
(7) Change $\overline{M_i}$ to RGB images. After changing $\overline{M_i}$ to RGB images, all images become almost pure white images. There are total of 840000 {0, 1} codes in each decrypted image. Among the decrypted images, the minimum number of $0_s^{'}$ is 3, and the maximum of $0_s^{'}$ is 28. Let $I_{i,j}$ denote the jth image having number "i" of zero codes. The first five images with minimum zero codes and the last five images with maximum zero codes are shown in Figs. 4(c)-(l). Observe that the percentages of the number of "1" codes are in the range $[99.9970\%, 99.9998\%]$, which are very closed to the ideal value $100 \times (2^{16} - 1) / 2^{16}\% = 99.9984\%$.

In summary, our experimental results suggest that the CPRNG is a promising candidate for practical applications.

(a) (b)



(c) (d)



(e) (f)

Figure 4. (a) Original image. (b) Decrypted image without error. Ten decrypted images via key streams generated with slightly perturbed initial conditions and system parameters in the range $[10^{-16},10^{-1}]$:(c) $I_{1,1}$, (d) $I_{3,1}$, (e) $I_{3,2}$, (f) $I_{3,3}$, (g) $I_{4,1}$, (h) $I_{23,1}$, (i) $I_{23,2}$, (j) $I_{24,1}$, (k) $I_{24,2}$, and (l) $I_{25,1}$.

Conclusions

(1) It introduces the definition of generalized stability (GST) for discrete system. This definition is a generalization for the definition of generalized synchronization for discrete systems.

(2) It presents a new 4D discrete chaotic map. Using this map and GST theorem designs a 8D GST system.

(3) It stablished a constructive theorem on GST discrete system. This theorem provides a general representation for GST discrete systems.

(4) It constructs an 8D GST-based CPRNG and compares the results tested by the FIPS 140-2 test on the RC4 PRNG and the ZUC algorithm show that the randomness of the sequences generated via the CPRNG is promising. The simulations also suggest that the key space of the CPRNG is larger than 2^{1195}, which is large enough to against brute-force attacks.

(5) It gives an image encryption example by using the CPRNG with the SESAE. Simulations show that the CPRNG is able to generate significant avalanche effects. The results suggest that the CPRNG is a qualified candidate for SESAE.

Acknowledgment

This project is supported by the National Natural Science Foundation of China (Grant Nos. 61074192, 61170037).

References

[1] G. Chen, X. Dong, From Chaos to Order: Methodologies, Perspectives, and Application. Singapore: World Scientific, 1998.
[2] J. G. Sprott. Chaos, Time-Sries Analysis, Oxford: Oxford University Press, 2003.
[3] M. Pecora, L. Carroll, Synchronization in chaotic systems, Physical Review Letters. 64 (1990) 821-825.
[4] T. Sangpet, S. Kuntanapreeda, Adaptive synchronization of hyperchaotic systems via passivity feedback control with time-varying gains, Journal of Sound and Vibration. 329 (2010) 2490-2496.
[5] R. S. Fyath, A. A. Al-mfrji, Investigation of chaos synchronization in photonic crystal lasers, Optics & Laser Technology. 44 (2012) 1406-1419.
[6] S. Kuntanapreeda, T. Sangpet, Synchronization of chaotic systems with unknown parameters using adaptive passivity-based control, Journal of the Franklin Institute. 349 (2012) 2547-2569.
[7] L. Lü, M. Yu, L. Wei, et al., Spatiotemporal chaos synchronization of an uncertain network based on sliding mode control, Chinese Physics B. 21 (2012) 100507-1 to 100507-5.
[8] S. Liu, L. Chen, Chaos synchronization of a chain network based on a sliding mode control, Chinese Physics B. 22 (2013) 100506-1 to 100506-6.
[9] K. Sun, Y. Wang, Y. Wang, Hyperchaos behaviors and chaos synchronization of two unidirectional coupled simplified Lorenz systems, Journal of Central South University. 21 (2014) 948-955.
[10] C. K. AHN, Generalized passivity-based chaos synchronization, Applied Mathematics and Mechanics. 31 (2010) 1009-1018.
[11] L. Lü, G. Li, L. Guo, et al., Generalized chaos synchronization of a weighted complex network with different nodes, Chinese Physics B. 8 (2010) 177-183.
[12] B. Liu, L. Cai, C. Feng, Adaptive generalized synchronization of simple piecewise linear chaotic system and SETMOS chaotic system, Chinese Journal of Quantum Electronics. 29 (2012) 74-79.
[13] L. Min, G. Chen, Generalized synchronization in an array of nonlinear dynamic systems with applications to chaotic cnn, J. Bifurcat. Chaos. 23 (2013) 1350016-1 to 1350016-53.
[13] J. Xu, J. Zhang, M. Pang, Generalized Synchronization Between Two Different Complex Delayed Networks, Journal of Tianjin University (Science and Technology). 47 (2014) 81-85. (In Chinese)
[15] Y. Li, C. Tao, A Class of Adaptive Generalized Projective Synchronization with Unknown Parameter Periods, Journal of Henan University (Natural Science). 45 (2015) 84-89. (In Chinese)
[16] X. Wu, H. Wang, H. Lu, Hyperchaotic secure communication via generalized function projective synchronization, Nonlinear Analysis. 12 (2011) 1288-1299.
[17] F. Min, Analysis of generalized projective synchronization for a chaotic gyroscope with a periodic gyroscope, Communications in Nonlinear Science and Numerical Simulation. 17 (2012) 4917-4929.
[18] X. Wu, C. Bai, H. Kan, A new color image cryptosystem via hyperchaos synchronization, Communications in Nonlinear Science and Numerical Simulation. 19 (2014) 1884-1897.
[19] P. Balasubramaniam, P. Muthukumar, Synchronization of chaotic systems using feedback controller: An application to Diffie-Hellman key exchange protocol and ElGamal public key cryptosystem, Journal of the Egyptian Mathematical Society. 22 (2014) 365-372.
[20] I. Pan, S. Das, A. Routh, Towards a global controller design for guaranteed synchronization of switched chaotic systems, Applied Mathematical Modelling. 39 (2015) 2311-2331.
[21] S. Hammami, State feedback-based secure image cryptosystem using hyperchaotic synchronization, ISA Transactions. 54 (2015) 52-59.
[22] F. Wang, A new pseudo-random number generator and application to digital secure communication scheme based on compound symbolic chaos, Acta Physica Sinica. 60 (2011) 110517-1 to 110517-7.

[23] L. Hao, L. Min, Statistical tests and chaotic synchronization based pseudorandom number generator for string bit sequences with application to image encryption, European Physical Journal Special Topics. 223 (2014) 1679-1697.
[24] M. Francois, T. Grosges, D. Barchiese, et al., Pseudo-random number generator based on mixing of three chaotic maps, Communications in Nonlinear Science and Numerical Simulation. 19 (2014) 887-895.
[25] A. Akhshani, A. Akhavan, A. Mobaraki, et al., Pseudo random number generator based on quantum chaotic map, Communications in Nonlinear Science and Numerical Simulation. 19 (2014) 101-111.
[26] Y. Yu, X. Li, J. Weng, Pseudorandom generators from regular one-way functions: New constructions with improved parameters, Theoretical Computer Science. 569 (2015) 58-69.
[27] L. Min, T, Chen, H, Zang, Analysis of fips 140-2 test and chaos-based pseudorandom number generator, Chaotic Modeling and Simulation, 11 (2013) 273-280.
[28] ETSI/SAGE Specification, Specification of the 3GPP Confidentiality and Integrity Algorithms 128-EEA3 & 128-EIA3. Document 2: ZUC Specification; Version: 1.5, Date: 4 th January 2011.
[29] C. Cui, H. Wang, S. Wen, et al., Using Liapanov Direct Method to Determine the Stability of Differential Equations, Journal of Jiamusi University (Natural Science Edition). 27 (2009) 288-300. (In Chinese)
[30] M. Wang, L. Wang, On stability of discrete dynamical system, Chinese Quarterly Journal of Mathematics. 2 (3) (1987) 12-30.
[31] L. Min, L. Hao, L. Zhang, Statistical test for string pseudorandom number generators, Lecture Notes Artificial Intelligence. 7888 (2013) 278-287.
[32] S. W. Golomb, Shift Register Sequence. Aegean Park, CA: Laguna Hills. 1982.
[33] L. Min, G. Chen, A novel stream encryption scheme with avalanche effect, European Physics Journal B. 86 (11) (2013) 459-1 to 459-7.