Chapter

Chaotic Systems with Hyperbolic Sine Nonlinearity

Jizhao Liu, Yide Ma, Jing Lian and Xinguo Zhang

Abstract

In recent years, exploring and investigating chaotic systems with hyperbolic sine nonlinearity has gained the interest of many researchers. With two back-to-back diodes to approximate the hyperbolic sine nonlinearity, these chaotic systems can achieve simplicity of the electrical circuit without any multiplier or sub-circuits. In this chapter, the genesis of chaotic systems with hyperbolic sine nonlinearity is introduced, followed by the general method of generating nth-order (n > 3) chaotic systems. Then some derived chaotic systems/torus-chaotic system with hyperbolic sine nonlinearity is discussed. Finally, the applications such as random number generator algorithm, spread spectrum communication and image encryption schemes are introduced. The contribution of this chapter is that it systematically summarizes the design methods, the dynamic behavior and typical engineering applications of chaotic systems with hyperbolic sine nonlinearity, which may widen the current knowledge of chaos theory and engineering applications based on chaotic systems.

Keywords: chaotic systems, torus chaos, hyperbolic sine nonlinearity, spread spectrum communication, image encryption

1. Introduction

Since Lorenz discovered chaos in a third-order ordinary differential equations, a new field of science has been launched [1]. The fact that simple equations can exhibit incredible complex behavior continues enthrall engineers to apply chaotic systems to cryptosystem, secure communication, spread spectrum communication, etc. [2].

There is no doubt that nonlinear term is very important to design chaotic systems, which has peculiar complex properties such as ergodicity, highly initial value sensitivity, non-periodicity and long-term unpredictability. According to the literature, the nonlinearities can be piecewise nonlinear function [3], trigonometric function [4], absolute value function [5], or power function [6]. With different nonlinearities, the chaotic system can have various strange attractors as single-scroll [7], double-scroll [8], multi-scroll [9], etc. The majority of such chaotic systems are known for many years, and some chaotic systems with hidden attractors are derived from them [10–12].

In recent years, chaotic systems with hyperbolic sine nonlinearities have gained the interest of many researchers. With two back-to-back diodes to approximate the hyperbolic sine nonlinearity, these chaotic systems can achieve simplicity of the electrical circuit without any multiplier or sub-circuits. Compared to single-scroll chaotic systems, the chaotic system with hyperbolic sine nonlinearity has richer dynamic behavior because it is symmetrical and can exhibit symmetry breaking, and offers the possibility that attractors will split or merge as some bifurcation parameter is changed [13].

In this chapter, we will systematically summarize the design method, the dynamic behavior and typical engineering applications of chaotic systems with hyperbolic sine nonlinearity. The genesis and general method of generating nth-order (n > 3) chaotic systems with hyperbolic sine nonlinearity are introduced in Section II. Some derived chaotic systems/torus-chaotic system with hyperbolic sine nonlinearity is discussed in Section III. The application such as random number generator algorithm, spread spectrum communication and image encryption schemes are introduced in Section IV. Conclusions are finally drawn in Section V.

2. General chaotic systems with hyperbolic sine nonlinearity

2.1 The genesis of chaotic systems with hyperbolic sine nonlinearity

In 2011, Sprott and Munmuangsaen proposed an exponential chaotic system [14], which happens to be an example of the simplest chaotic system [15]. In the same year, Sprott used common resistors, capacitors, operational amplifiers, and a diode to successfully implement this system in a circuit [16]. Few years later, the simplest hyperbolic sine chaotic system is proposed [17]. Compared to the exponential chaotic system, the hyperbolic sine chaotic system changed the nonlinearity from exponential function (asymmetric function) to hyperbolic sine function (symmetric function), which can exhibit symmetry breaking, and offers the possibility that attractors will split or merge as some bifurcation parameter is changed [18].

The simplest chaotic system with a hyperbolic sine is described as follows:

$$\dddot{x} + c\ddot{x} + x + \rho * \sinh(\varphi\dot{x}) = 0 \tag{1}$$

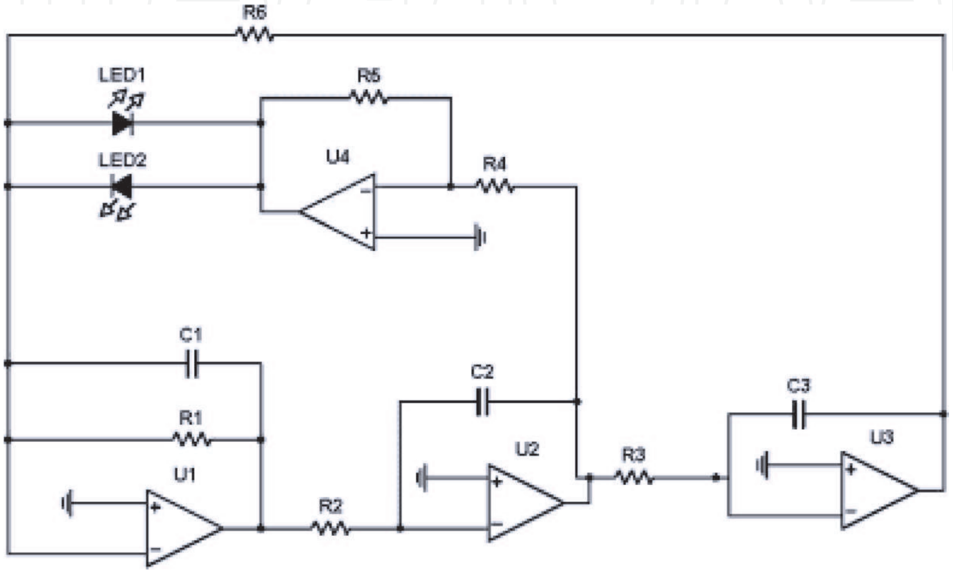

Figure 1.
The corresponding circuit schematic diagram of Eq. (1).

Where c is considered as the bifurcation parameter, $\sinh(\varphi\dot{x}) = \frac{e^{\varphi\dot{x}} - e^{-\varphi\dot{x}}}{2}$, $\rho = 1.2 * 10^{-6}$ and $\varphi = \frac{1}{0.026}$, which have been chosen to facilitate circuit implementation using diodes. The corresponding circuit schematic diagram of Eq. (1) is shown as **Figure 1**.

When $c = 0.75$, the Eq. (1) can exhibit chaotic behavior, which is shown as **Figure 2**.

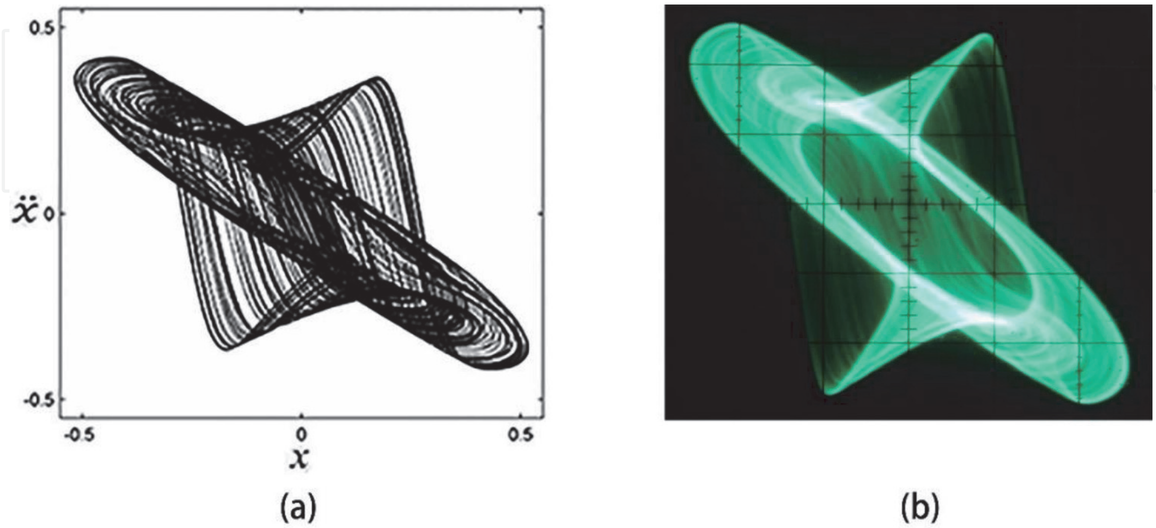

Figure 2.
Numerical and actual circuit state space plot in $x - \ddot{x}$ plane.

2.2 The general equations of generating chaotic systems with hyperbolic sine nonlinearity

It is obvious that Eq. (1) can be written in the form with jerk equations:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = x_3 \\ \dot{x}_3 = -cx_3 - f(x_2) - x_1 \end{cases} \tag{2}$$

where $f(x_2) = \rho * \sinh(\varphi x_2)$. Therefore, the higher order chaotic systems with hyperbolic sine nonlinearity can be generated by adding jerk cabins, which is described by:

$$\begin{cases} \dot{x}_1 = x_2 - x_1 \\ \dot{x}_2 = x_3 - x_2 \\ \cdots \\ \dot{x}_{n-3} = x_{n-2} - x_{n-3} \\ \dot{x}_{n-2} = x_{n-1} \\ \dot{x}_{n-1} = x_n \\ \dot{x}_n = -cx_n - f(x_{n-1}) - nx_{n-2} - nx_{n-3} - \cdots - \dfrac{1}{2n}x_1 \end{cases} \tag{3}$$

where $\dot{x}_{k-1} = x_k - x_{k-1}$ is the jerk cabin. With Eq. (3), we can construct nth-order (n > 3) chaotic systems with hyperbolic sine nonlinearity.

When n = 4, the equations of fourth-order chaotic systems will be:

$$\begin{cases} \dot{x}_1 = x_2 - x_1 \\ \dot{x}_2 = x_3 \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = -x_4 - f(x_3) - 5x_2 - 0.125x_1 \end{cases} \tag{4}$$

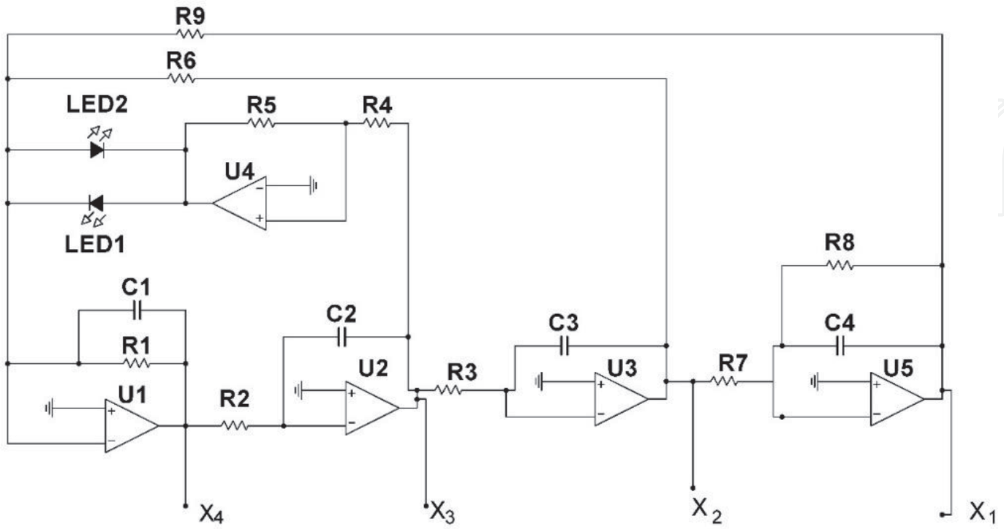

Figure 3.
The corresponding circuit schematic diagram of Eq. (4).

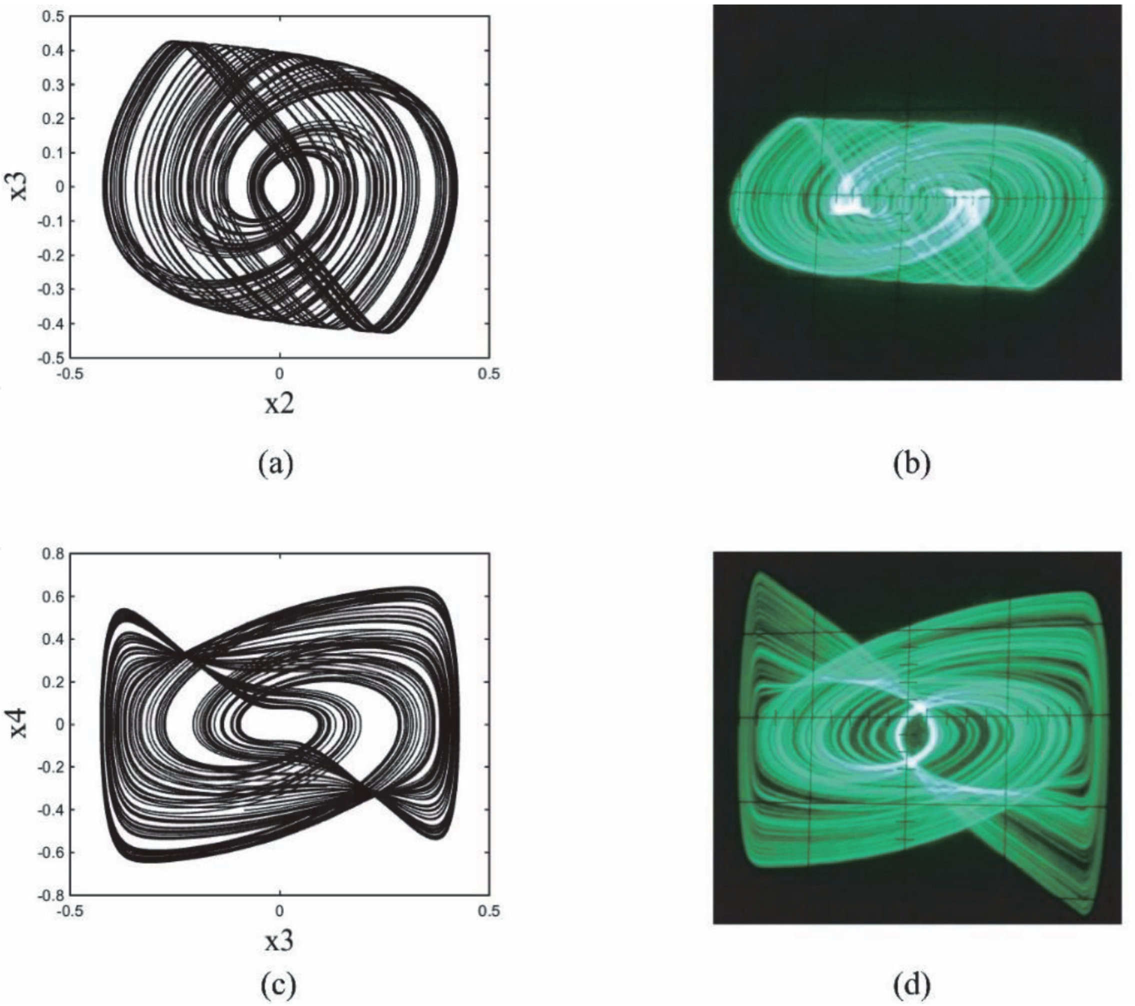

Figure 4.
Numerical and actual circuit state space plot in $x_2 - x_3$ plane and $x_3 - x_4$ plane.

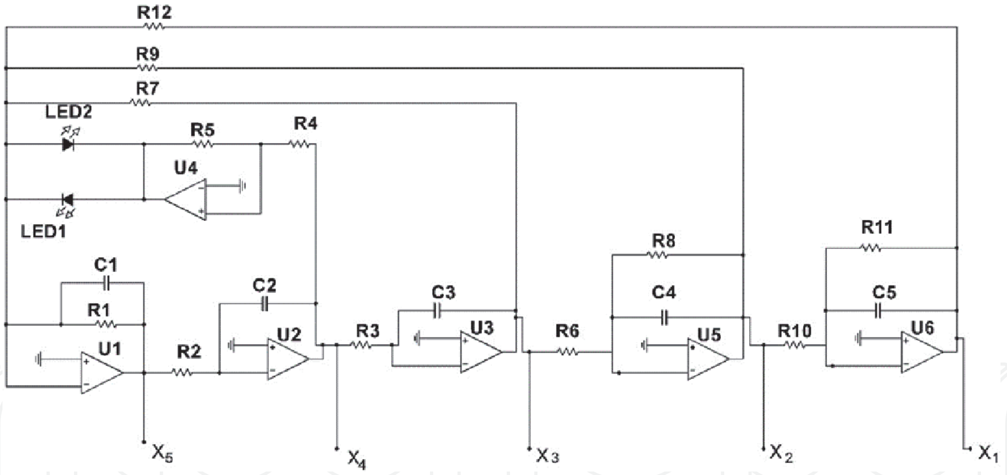

Figure 5.
The corresponding circuit schematic diagram of Eq. (5).

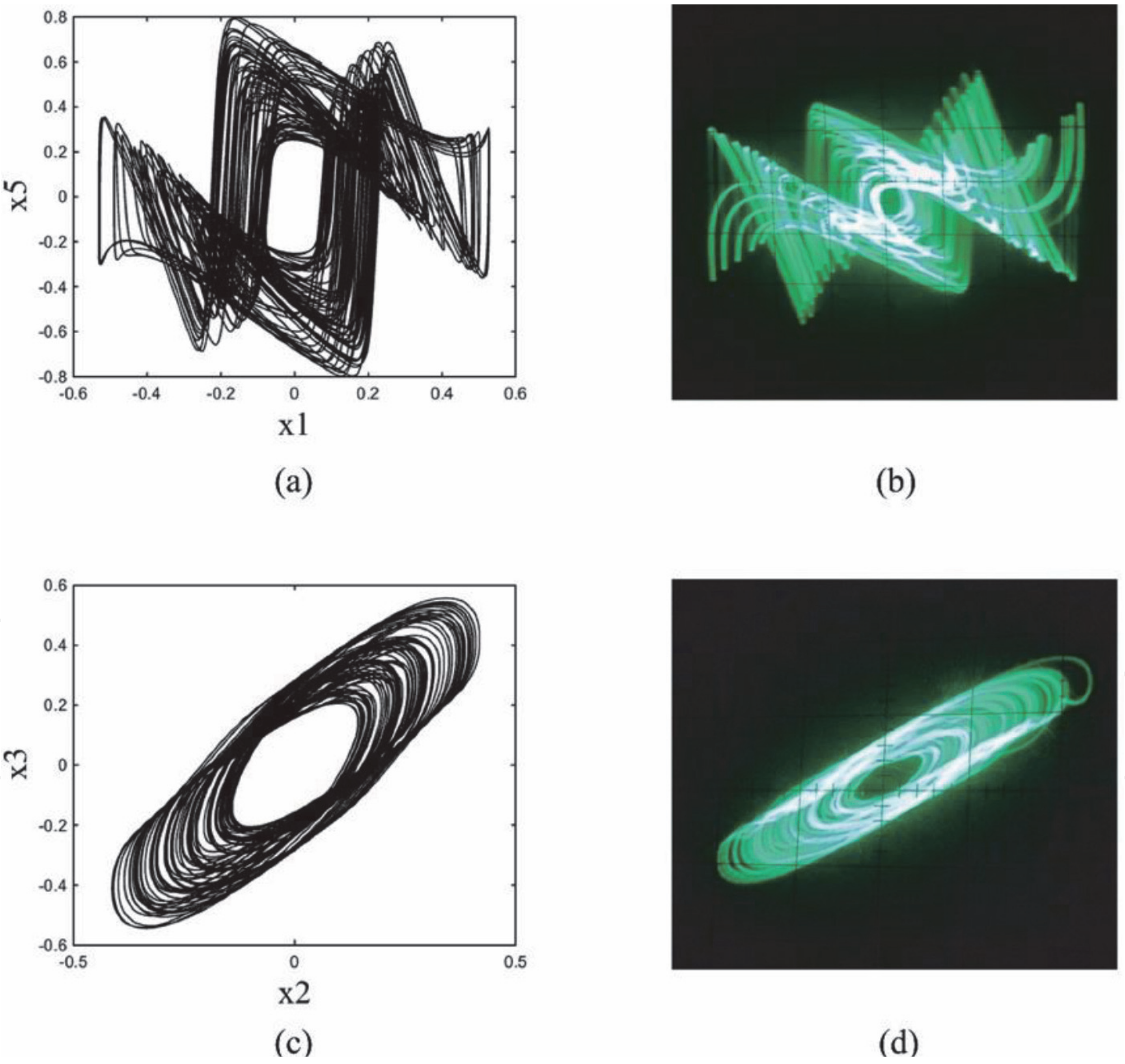

Figure 6.
Numerical and actual circuit state space plot in $x_1 - x_5$ plane and $x_2 - x_3$ plane.

The corresponding circuit schematic diagram of Eq. (4) is shown as **Figure 3**. Its numerical and actual circuit state space plot is shown as **Figure 4**.

When n = 5, the equations of fifth-order chaotic systems will be:

$$\begin{cases} \dot{x}_1 = x_2 - x_1 \\ \dot{x}_2 = x_3 - x_2 \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = x_5 \\ \dot{x}_5 = -x_5 - f(x_4) - 5x_3 - 5x_2 - 0.1x_1 \end{cases} \tag{5}$$

The corresponding circuit schematic diagram of Eq. (5) is shown as **Figure 5**. Its numerical and actual circuit state space plot is shown as **Figure 6**.

3. Derived chaotic systems/torus-chaotic system with hyperbolic sine nonlinearity

3.1 Multi-nonlinearities hyperbolic sine chaotic system

One way to construct the derived chaotic systems is to add more nonlinear terms of the equations. For example, the new chaotic system can be constructed by Eq. (4), which is described as follows:

$$\begin{cases} \dot{x}_1 = x_2 - \rho sinh(\varphi x_1) \\ \dot{x}_2 = x_3 - 0.3x_2 - \rho sinh(\varphi x_2) \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = -0.25x_4 - \rho sinh(\varphi x_3) - 0.5x_2 - 4x_1 \end{cases} \tag{6}$$

Where $\rho = 1.2 * 10^{-6}$, $\varphi = \frac{1}{0.026}$. These equations can exhibit chaotic behavior as shown in **Figure 7**.

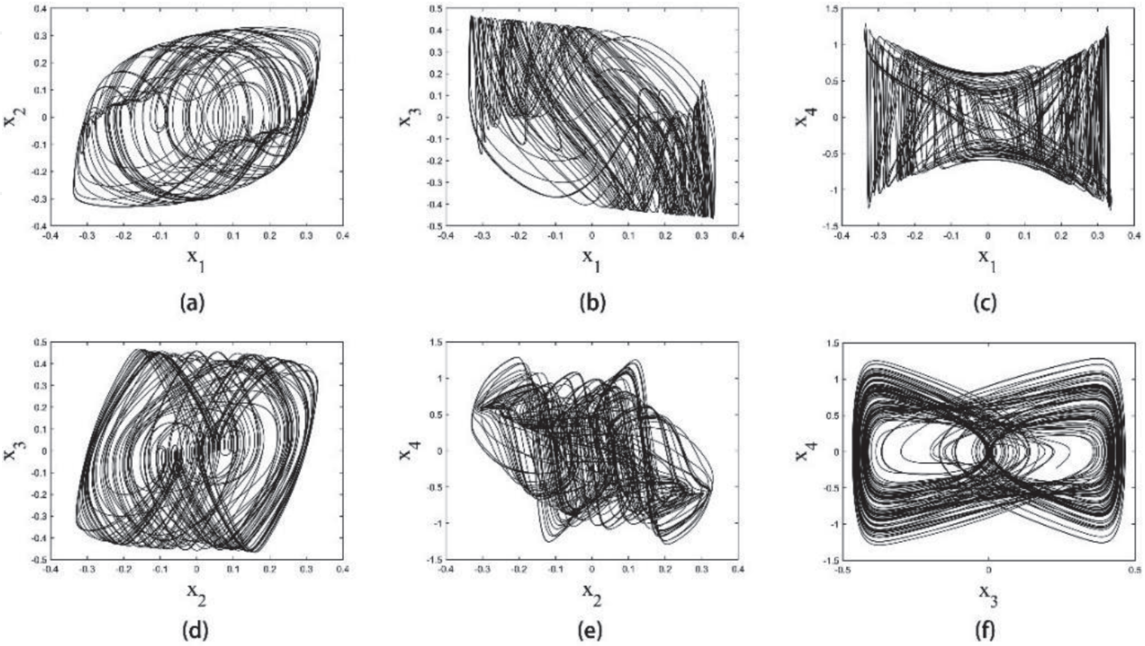

Figure 7.
Numerical phase space plot of Eq. (6).

3.2 Simple chaotic system with hyperbolic sine nonlinearity

The other way to construct the derived chaotic systems is to simplify the known chaotic systems. For example, if we remove the parameter ρ and φ, search the parameter space, we will have the following chaotic system:

$$\begin{cases} \dot{x}_1 = 6x_2 - x_1 \\ \dot{x}_2 = x_3 \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = -x_4 - sinh\,(x_3) - x_1 \end{cases} \tag{7}$$

When initial conditions are set to be $(x_1, x_2, x_3, x_4) = (0.7, 0.9, 1.0, 1.3)$, or $(x_1, x_2, x_3, x_4) = (-0.7, -0.9, -1.0, -1.3)$, the system exhibits period behavior. When the initial conditions are set to be $(x_1, x_2, x_3, x_4) = (7, 9, 10, 13)$ and $(x_1, x_2, x_3, x_4) = (-7, -9, -10, -13)$, the system exhibits chaotic behavior. Therefore, this system has four coexistence attractors [19], as shown in **Figure 8**.

3.3 Torus-chaotic system with hyperbolic sine nonlinearity

By introducing a nonlinear feedback controller to system Eq. (5), the following system is obtained:

$$\begin{cases} \dot{x}_1 = x_2 - \rho \sinh\,(\varphi x_3) \\ \dot{x}_2 = x_3 - x_2 \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = x_5 \\ \dot{x}_5 = -cx_5 - \rho \sinh\,(\varphi x_4) - 5x_3 - 5x_2 - 0.1x_1 \end{cases} \tag{8}$$

When c = 1, the Lyapunov exponents are $(\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5) = (0.47, 0, 0, -1.10, -1.37)$, which suggests Eq. (8) is exhibiting torus-chaos behavior [20].

When c = 1.55 and the initial conditions are set to be $(x_1, x_2, x_3, x_4, x_5) = (-0.1, -0.1, -0.1, -0.1, -0.1)$ and $(x_1, x_2, x_3, x_4, x_5) = (0.1, 0.1, 0.1, 0.1, 0.1)$, the system has two coexisting attractors as shown in **Figure 9**.

Figure 10 shows the Lyapunov exponent spectrum, Kaplan–Yorke dimension spectrum and bifurcations of Eq. (8) as the coefficient c is varied over the range c ∈ [0.3, 2]. Those figures suggest there is an interesting route leading to chaos [21].

1. When c ∈ [0.3, 0.4639], there exists a period-doubling behavior along with $\dot{x}_2$ and $\dot{x}_3$ subspace. However, the system shows torus behavior along with $\dot{x}_2$ and $\dot{x}_3$ subspace. It is like saddle point: the system is stable in one direction but unstable in the other direction.

2. When c ∈ [0.4640, 0.5574], the system exhibits two-torus-chaos behavior except for some 2-torus windows. When the parameter passed c = 0.4639 to c = 0.4640, two-torus-chaos is born by replacing the 2-torus behavior. The Lyapunov exponents at these two critical values are $(\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5) = (0, 0, -0.01, -0.57, -0.88)$ for c = 0.4639 and $(\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5) = (0.02, 0, 0, -0.60, -0.88)$ for c = 0.4640. This may cause by the period-doubling route along with $\dot{x}_2$ and $\dot{x}_3$ subspace.

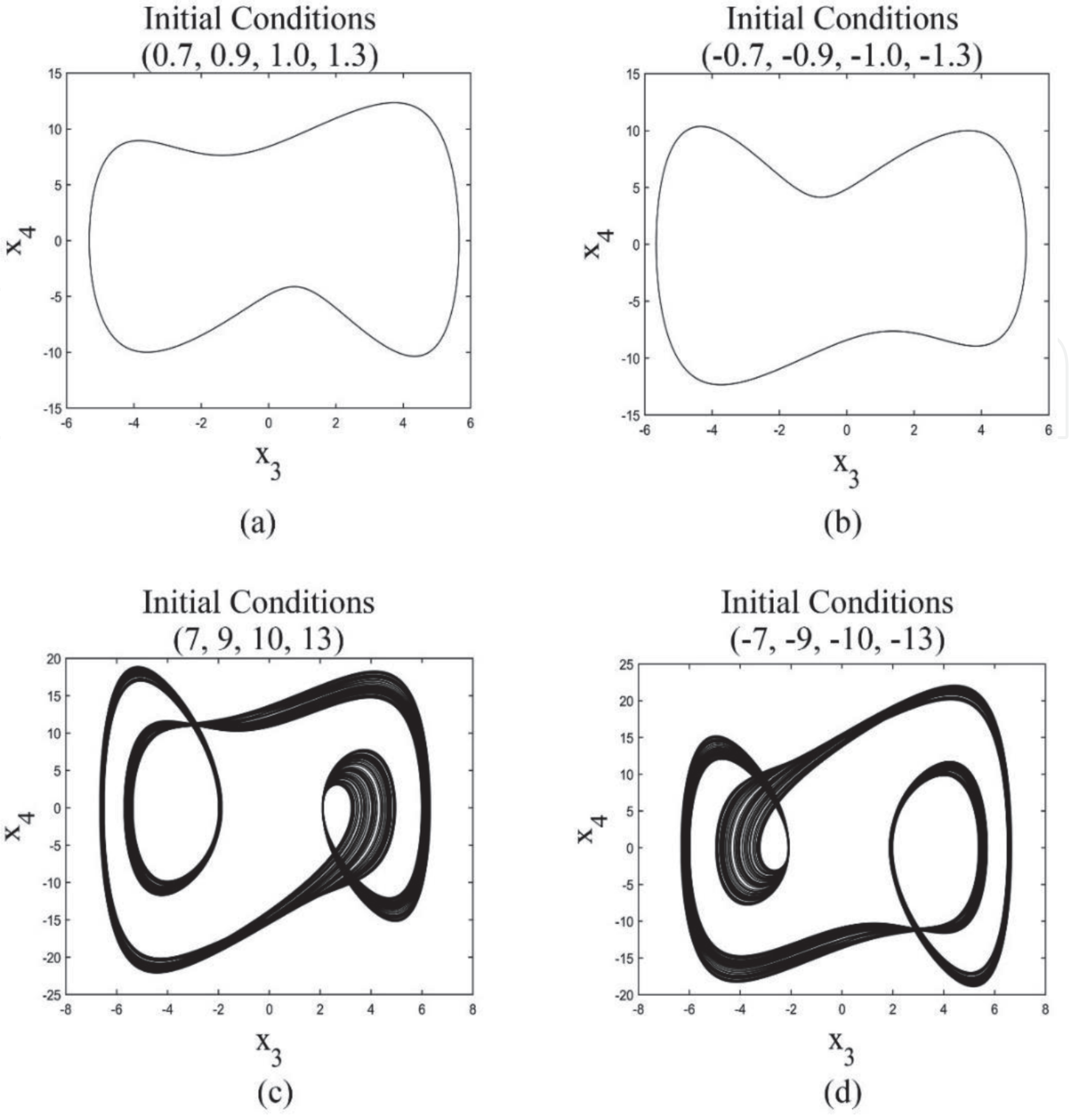

Figure 8.
Coexistence attractors of Eq. (7).

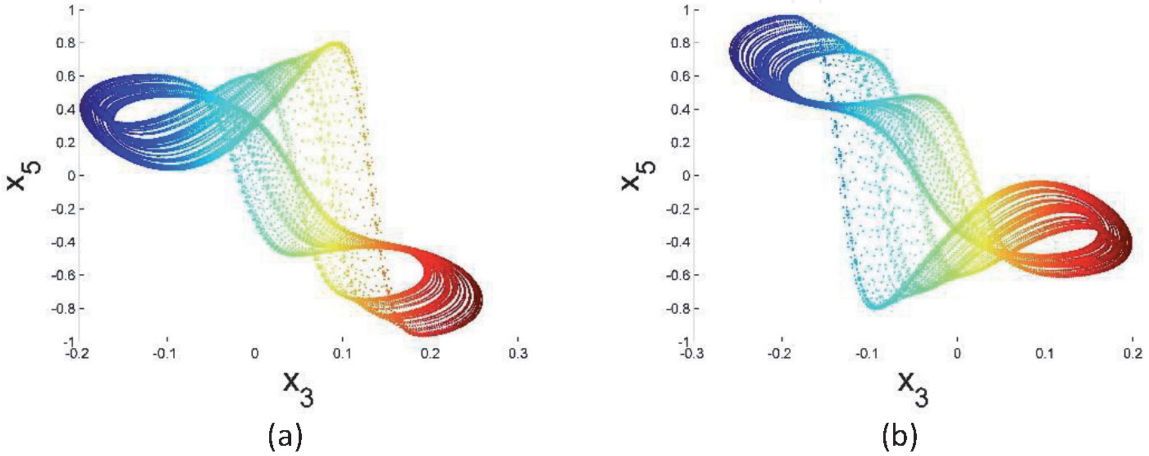

Figure 9.
Coexistence attractors of Eq. (8).

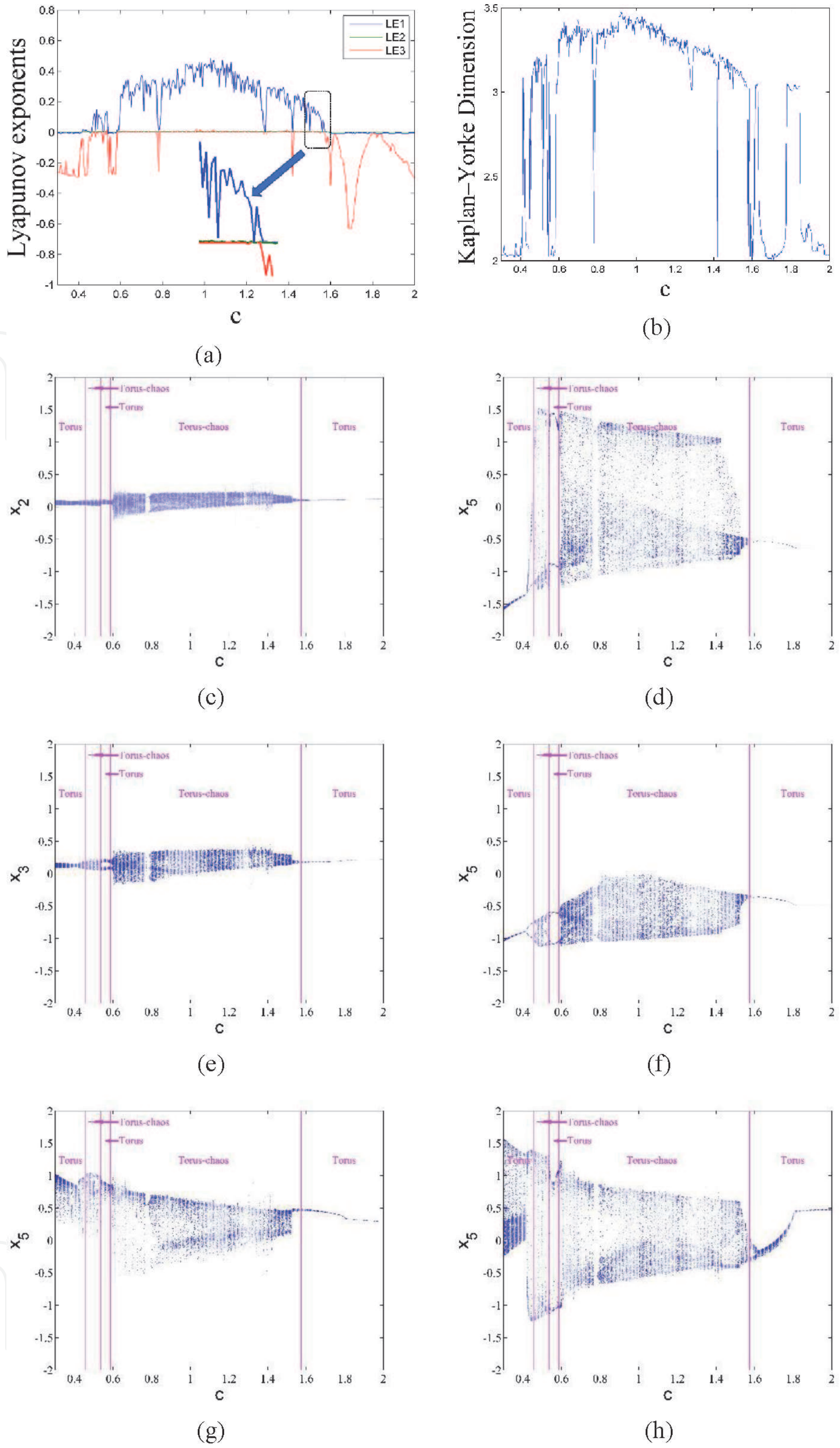

Figure 10.
LEs spectrum, Kaplan–Yorke dimension spectrum and bifurcations of Eq. (8) as the coefficient c is varied over the range c ∈ [0.3, 2].

3. When c ∈ [0.5575, 0.5901], the system exhibits 2-torus behavior.

4. When c ∈ [0.5902, 1.5575], the system exhibits 2-torus-chaos behavior except for 2-torus windows. The route leading to chaos is same to point 3.

5. When c ∈ [1.5575, 2] the system exhibits 2-torus behavior, except for some 3-torus windows like c = 1.6157.

4. Engineering applications with chaotic systems with hyperbolic sine nonlinearity

4.1 Random number generator

Sensitivity to initial conditions is one of the most important property of chaotic systems. Therefore, chaotic systems are very suitable for the cryptography purpose. But before that, it should be noticed that the probability density distributions (PDD) of chaotic systems are not uniform distribution. **Figure 11(a)** and **11(b)** are the waveform and PDD of x_4 of Eq. (4). It shows that PDD of the output sequences has physical characteristic. The cryptosystem with these sequences cannot resist side channel attack.

To remove physical characteristic, one can use the following de-correlation operation:

$$S_{out} = S_{in} * 10^6 - floor\left(S_{in} * 10^6\right) \tag{9}$$

In fact, Eq. (9) can be applied in all chaotic/torus-chaotic/hyperchaotic systems. The output sequences can pass fifteen random tests of NIST 800-22, as shown as in **Table 1**, which indicated the proposed method can provide high security Level. This proposed method can be used as a part of some cyber security systems such as the verification code, secure QR code and some secure communication protocols.

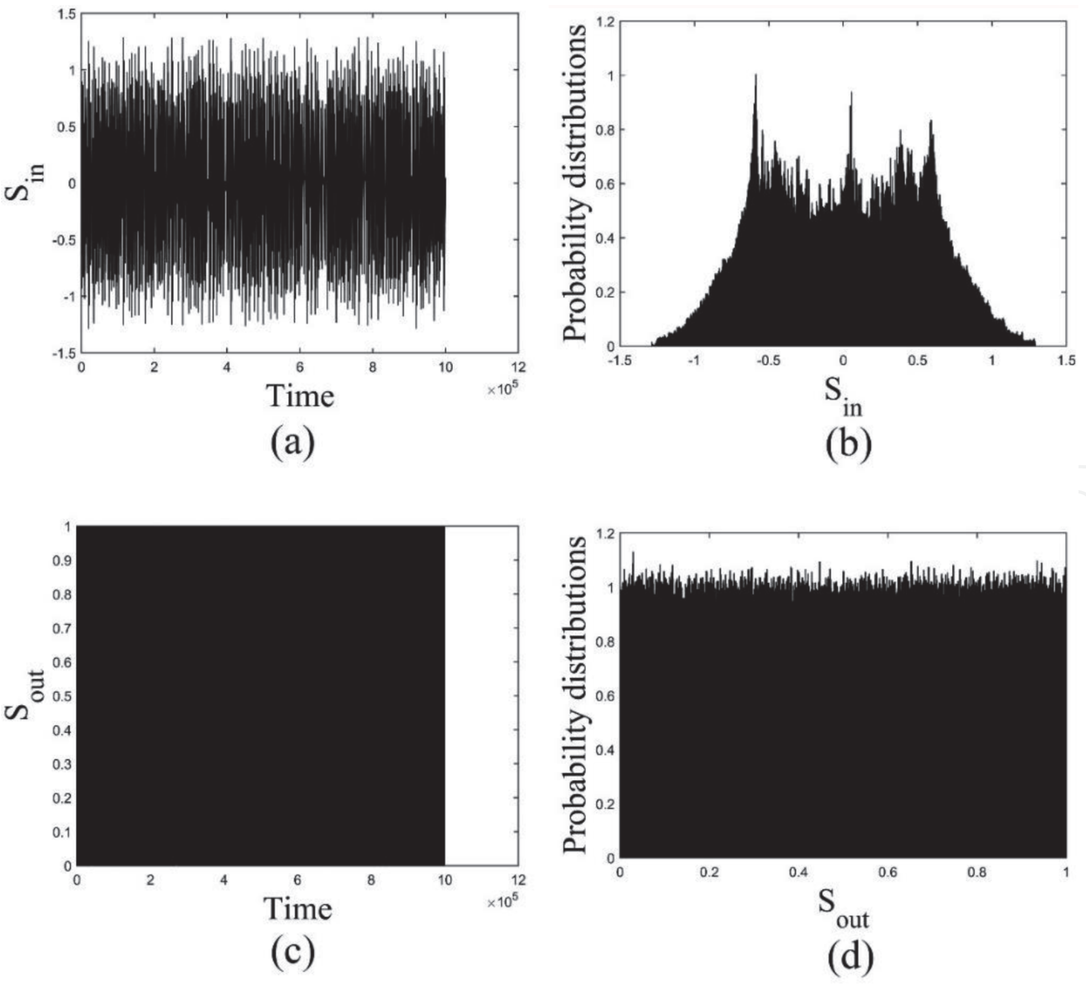

Figure 11.
Waveform and PDD before and after de-correlation operation of x_4 of Eq. (4): (a) is the waveform of x_4 before de-correlation operation; (b) is the PDD of x_4 before de-correlation operation; (c) is the waveform of x_4 after de-correlation operation; (b) is the PDD of x_4 after de-correlation operation.

Test	P-value	Result
Frequency	0.841481	Success
Block frequency	0.900704	Success
Runs	0.744455	Success
Longest run	0.172897	Success
Rank	0.368065	Success
FFT	0.762020	Success
Non-overlapping template	0.813121	Success
Overlapping template	0.532736	Success
Universal	0.856573	Success
Linear complexity	0.408679	Success
Serial	0.967366	Success
Approximate entropy	0.433157	Success
Cumulative sums	0.688582	Success
Random excursions	0.075229	Success
Random excursions variant	0.102049	Success

Table 1.
Pseudo-random properties of x_3 of Eq. (8) after de-correlation operation.

4.2 Image encryption

Image encryption is another widely used engineering application of chaotic system. In this section, we will use Eq. (7) for image encryption purpose.

A flowchart of the encryption scheme is shown in **Figure 12**.

The detailed encryption process includes the following steps.

Input: Plain image; Initial conditions for the chaotic system; Control parameters of the chaotic system.

Output: Ciphered image.

Step 1: Calculate the average pixel value of the plain image and generate the pseudorandom sequence.

Step 2: Transform the pseudorandom sequence and change pixel value of the image via XOR.

Step 3: Sort the pseudorandom sequence for permutation.

Step 4: Shift the pixel positions by column using the sorted elements.

Step 5: Shift the pixel positions by row using the sorted elements.

To provide a better understanding of this scheme, the pseudocode is provided in **Table 2**.

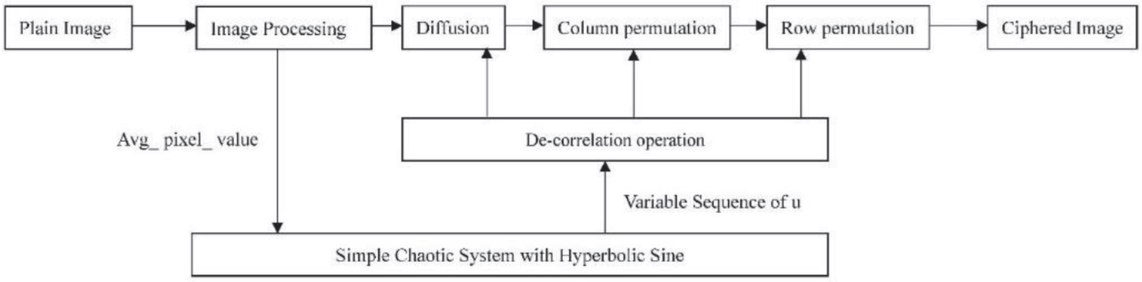

Figure 12.
A flowchart of the encryption scheme.

```
Input: Plain image Org_Img, Initial conditions for the chaotic system, Control parameter for the chaotic
system,
Output: Ciphered Image En_Img
-------------------------------------------------------------------------------------------------
[m,n] ← size(Org_Img);

Avg_pixel_value ← mean2(Org_Img)*10^(-5) % mean2 is a function that
returns the

                        % average value of a matrix

x(1) ← x(1) + Avg_pixel_value
y(1) ← y(1)
z(1) ← z(1)
u(1) ← u(1)
s(1) ← u(1)*10^4 – floor(u(1)*10^4)

For i=1:1:m*n             % Generate pseudorandom sequence that will
                                      % be used for diffusion and permutation
    [dx, dy, dz, du] ← Runge-Kutta (x(i), y(i), z(i), u(i))
    x(i+1) ← x(i) +dx
    y(i+1) ← y(i) +dy
    z(i+1) ← z(i) +dz
    u(i+1) ← u(i) +du
s(i+1) ← u(i+1)*10^4 – floor(u(i+1)*10^4)
End

Count=1         % Count flag
For i=1:m        % Diffusion Operation
   For j=i:n
    diff(Count) ← mod (s(Count)*10^14, 256) % transform s, which could be used for XOR
   En_Dif(i,j)=bitxor(Org_Img(i,j), diff (Count)); % Bitwise exclusive OR
      Count= Count+1;
   End
End

S_index ← Sort(s)
For i=1:n           % Column-wise permutation
 For j=1:m
 En_per_col (i,j) ← Sort (En_Dif, S_index)
 End
End
For i=1:m           % Row-wise permutation
 For j=1:n
 En_Img (i,j) ← Sort (En_per_col, S_index)
 End
End
```

Table 2.
Image encryption scheme.

The decryption process of the proposed algorithm is the reverse process of the encryption algorithm. A flowchart of the decryption process is shown in **Figure 13**.

The detailed decryption process includes the following steps.

Input: Plain image; Initial conditions for the chaotic system; Control parameter of the chaotic system; Average pixel value of the plain image

Output: Decrypted image

Step 1: Generate the pseudorandom sequence via the initial conditions and the average pixel values of the plain image

Step 2: Sort the pseudorandom sequence for row and column recovery.

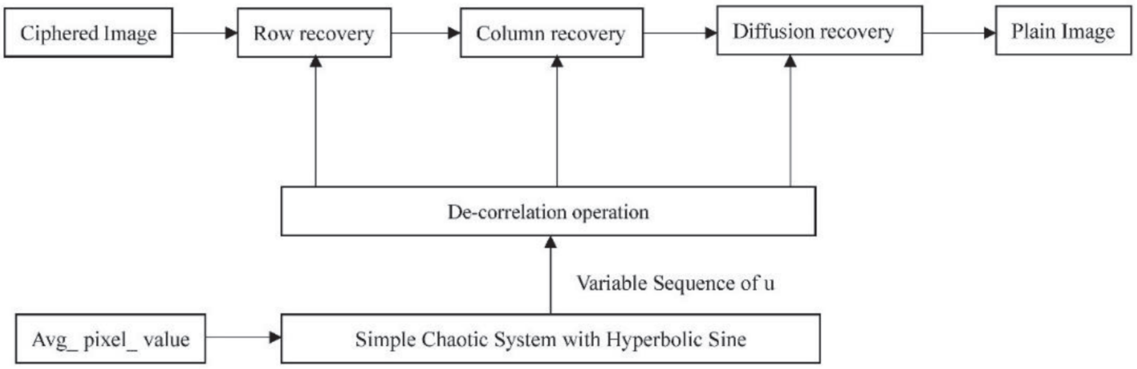

Figure 13.
A flowchart of the decryption scheme.

```
    Input: Ciphered image En_Img, Initial conditions for the chaotic system, control parameter for the
  chaotic system, Avg_pixel_value of Org_Img
    Output: Plain Image Org_Img
---------------------------------------------------------------------------------------------------
    [m,n] ← size(En_Img);
    x(1) ← x(1) + Avg_pixel_value
    y(1)← y(1)
    z(1) ← z(1)
    u(1) ← u(1)
    s(1) ← u(1)*10^4 – floor(u(1)*10^4)

    For i=1:1:m*n              % Generate a pseudorandom sequence that will
    % be used for decryption
    [dx, dy, dz, du] ← Runge-Kutta (x(i), y(i), z(i), u(i))
    x(i+1) ← x(i) +dx
    y(i+1) ← y(i) +dy
    z(i+1) ← z(i) +dz
    u(i+1) ← u(i) +du
      s(i+1) ← u(i+1)*10^4 – floor(u(i+1)*10^4)
    End

    S_index ← Sort(s)
    For i=1:m                  % Row-wise permutation recovery
     For j=1:n
     De_per_row (i,j) ← Sort (En_Img, S_index)
     End
    End

   For i=1:n                     % Column-wise permutation recovery
    For j=1:m
    De_per_col (i,j) ← Sort (De_per_row, S_index)
    End
   End

  Count=1                 % Count flag
  For i=1:m               % Diffusion recovery
    For j=i:n
    diff(Count) ← mod (s(Count)*10^14, 256) % transform s, which could be used for XOR
    Org_Img (i,j)=bitxor(De_per_col (i,j), diff (Count)); % Bitwise exclusive OR
      Count= Count+1;
    End
   End
```

Table 3.
Image decryption scheme.

Step 3: Shift the pixel positions by row

Step 4: Shift the pixel positions by column

Step 5: Transform the pseudorandom sequence and recover the pixel values of the image via XOR

To provide a better understanding of this scheme, the pseudo-code is provided in **Table 3**

The testing results of encryption and decryption are shown in **Figure 14**.

In this system, all the initial conditions and control parameters can be considered as secret keys. Because the basin of attraction of each initial condition is greater than 1, it could have more than 10^{15*4} =10^{60} choices via a resolution of 10^{-15}, in terms of a numeric calculation. Moreover, if a range of control parameters are considered for the key space, the key space of this system would far exceed 10^{90}. Such a large key space provides sufficient security against brute-force attacks.

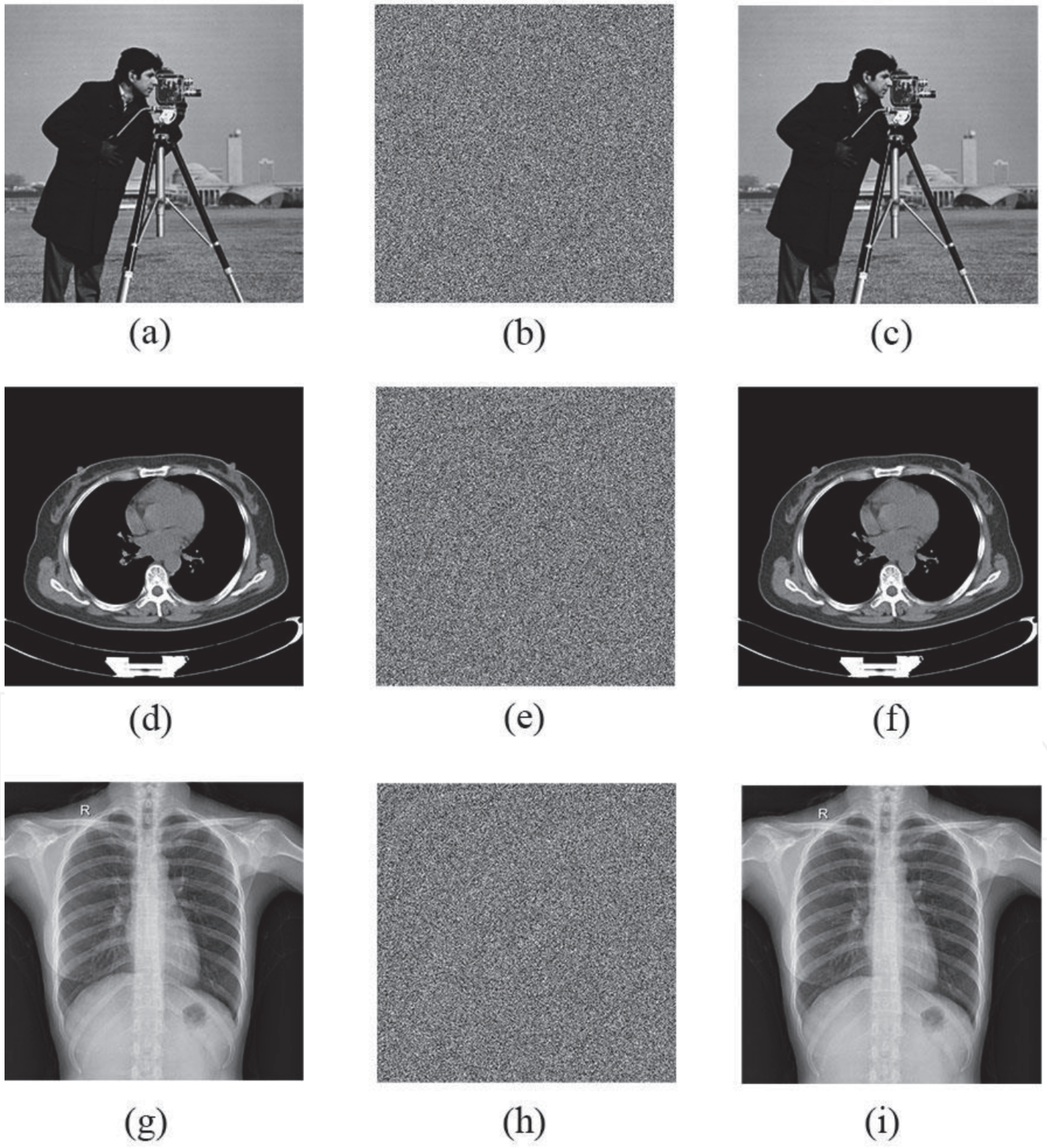

Figure 14.
The testing results of encryption and decryption: (a) is the plain image of cameraman; (b) is the encrypted image of cameraman; (c) is the decrypted image of cameraman; (d) is the plain image of breast CT image; (e) is the encrypted image of breast CT image; (f) is the decrypted image of breast CT image; (g) is the plain image of thorax CT image; (h) is the encrypted image of thorax CT image; (i) is the decrypted image of thorax CT image.

Correlation coefficients of adjacent pixels in the plain and encrypted image are shown in **Table 4**.

The NPCR and UACI score of CT image are 99.5804% and 33.3227%.

From the above security analysis, the proposed scheme can provide high security for cryptographic applications.

4.3. Spread spectrum communication

Chaotic systems can also use for spread spectrum communication propose. Different chaos shift keying (DCSK) technology employs nonperiodic and wideband chaotic signals as carriers so as to achieve the effect of spectrum spreading in the process of digital modulation. **Figure 15** shows the scheme of modulation for DCSK.

In this scheme, every bit has two time slots. The first time slot is used for transmission of a chaotic sequence for the reference signal. The second time slot is used for transmission of another chaotic sequence for the reference signal which has the same length as the first time slot. If the information bit is +1, then the information signal is exactly the same as the reference signal. If the information signal bit is −1, then the information signal is the negative of the reference signal. For bits b_k, the signal at time k is:

$$s_i = \begin{cases} x_i & 2k\beta < i \leq (2k+1)\beta \\ b_k x_{i-\beta} & (2k+1)\beta < i \leq 2(k+1)\beta \end{cases} \tag{10}$$

Where β is the number of sampling points. The spreading factor (SF) in the DCSK system is $SF = 2\beta$.

Figure name	Direction	Plain-image	Ciphered image
Cameraman Image	Horizontal	0.983146	0.001731
Cameraman Image	Vertical	0.990025	0.004141
Cameraman Image	Diagonal	0.973249	0.000324
Breast CT image	Horizontal	0.978292	0.002500
Breast CT image	Vertical	0.955481	0.006207
Breast CT image	Diagonal	0.940737	0.003071
Thorax CT image	Horizontal	0.994585	0.001267
Thorax CT image	Vertical	0.994761	0.001267
Thorax CT image	Diagonal	0.991973	0.001558

Table 4.
Correlation coefficients of adjacent pixels in the plain and encrypted image.

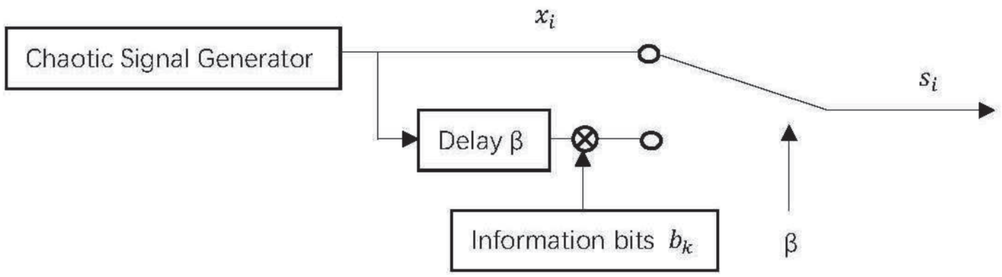

Figure 15.
Scheme of DCSK modulation.

For demodulation as shown in **Figure 16**, the receiver calculates the correlation between the received signal r_i and the signal $r_{i-\beta}$, which is r_i delayed by β . After a time k, the output of the correlator is:

$$Z_k = \sum_{2(k+1)\beta}^{i=(2k+1)\beta+1} r_i r_{i-\beta} \tag{11}$$

Thus, the information bit b_k can be restored by the sign of the decision variable:

$$\hat{b}_k = sgn\,[Z_k] \tag{12}$$

The obtained BER performance under additive white Gaussian noise (AWGN) channels for spreading factor $2\beta = 200$ is shown in **Figure 17**. From the comparison results, DCSK can have a lower BER when using this system as a carrier signal in the presence of noise.

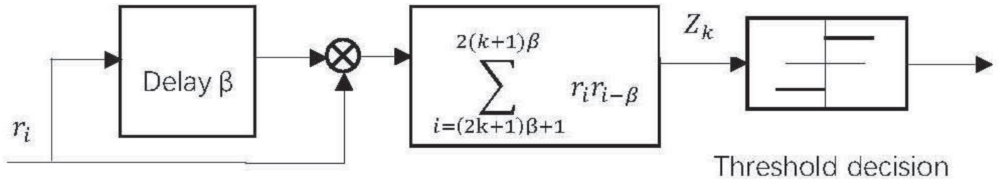

Figure 16.
Scheme of the DCSK demodulation.

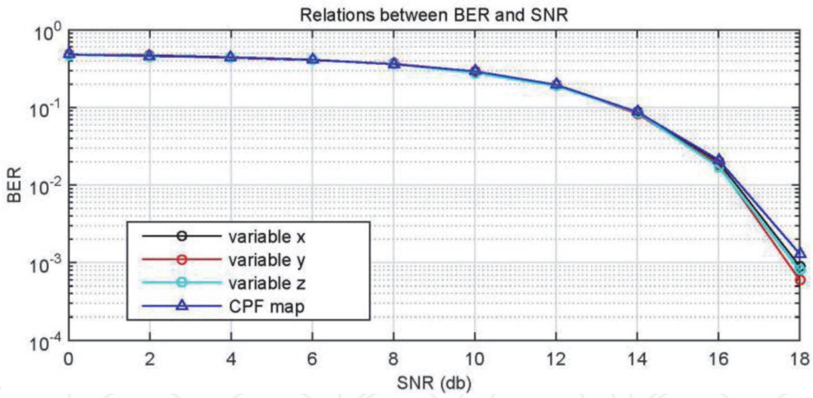

Figure 17.
Comparison of the bit error rate for a Chebyshev sequence and the hyperbolic sine system with DCSK.

5. Conclusions

In this chapter, we first described a third order chaotic system with hyperbolic sine nonlinearity, then we introduced the method to expend this chaotic system to high order chaotic systems. After that, we introduced the method to construct the derived chaotic torus-chaotic systems. Finally, we introduced some applications such as random number generator algorithm, spread spectrum communication and image encryption schemes. The contribution of this chapter is that it systematically summarizes the design method, the dynamic behavior and typical engineering application of chaotic systems with hyperbolic sine nonlinearity, which may widen the current knowledge of chaos theory and engineering applications based on chaotic systems.

Acknowledgements

Jizhao Liu has received research grants from Sun Yat-sen University.

This study was supported by the Fundamental Research Funds for the Central Universities. No. 19lgpy230.

Conflict of interest

The authors declare that they have no conflict of interest.

Notes/thanks/other declarations

The authors would like to thank professor Julien Clinton Sprott for helpful discussion.

Author details

Jizhao Liu[1,2*], Yide Ma[2], Jing Lian[3] and Xinguo Zhang[2]

1 School of Data and Computer Science, Sun Yat-sen University, Guangzhou, Guangdong, China

2 School of Information Science and Engineering, Lanzhou University, Lanzhou, Gansu, China

3 School of Electronics and Information Engineering, Lanzhou Jiaotong University, Lanzhou, Gansu, China

*Address all correspondence to: liujizhao@mail.sysu.edu.cn

References

[1] Sprott, Julien C. "Elegant chaos: algebraically simple chaotic flows." World Scientific, 2010.

[2] Zhang Xinguo and Li Yide, Ma. Shouliang. "Nonlinear Circuit-Based Analysis and Design". Higher Education Press, 2011.

[3] Li, Shujun, Guanrong Chen, and Xuanqin Mou. "On the dynamical degradation of digital piecewise linear chaotic maps." International journal of Bifurcation and Chaos 15.10 (2005): 3119-3151.

[4] Zhu, Hegui, et al. "Analyzing Devaney chaos of a sine–cosine compound function system." International Journal of Bifurcation and Chaos 28.14 (2018): 1850176.

[5] Sprott, J. C. "A new class of chaotic circuit." Physics Letters A 266.1 (2000): 19-23.

[6] Sprott, J. Clint. "Some simple chaotic flows." Physical review E 50.2 (1994): R647.

[7] Wang, Chunhua, Hu Xia, and Ling Zhou. "A memristive hyperchaotic multiscroll jerk system with controllable scroll numbers." International Journal of Bifurcation and Chaos 27.06 (2017): 1750091.

[8] Xiong, Li, et al. "Design and hardware implementation of a new chaotic secure communication technique." PloS one 11.8 (2016): e0158348.

[9] Yu, Simin, et al. "Design and implementation of n-scroll chaotic attractors from a general jerk circuit." IEEE Transactions on Circuits and Systems I: Regular Papers 52.7 (2005): 1459-1476.

[10] Li, Chunbiao, and Julien Clinton Sprott. "Coexisting hidden attractors in a 4-D simplified Lorenz system." International Journal of Bifurcation and Chaos 24.03 (2014): 1450034.

[11] Zaamoune, Faiza, et al. "Symmetries in Hidden Bifurcation Routes to Multiscroll Chaotic Attractors Generated by Saturated Function Series." Journal of Advanced Engineering and Computation 3.4 (2019): 511-522.

[12] Tlelo-Cuautle, Esteban, et al. "Dynamics, FPGA realization and application of a chaotic system with an infinite number of equilibrium points." Nonlinear Dynamics 89.2 (2017): 1129-1139.

[13] Liu, Jizhao, et al. "An approach for the generation of an nth-order chaotic system with hyperbolic sine." Entropy 20.4 (2018): 230.

[14] Munmuangsaen, Buncha, Banlue Srisuchinwong, and Julien Clinton Sprott. "Generalization of the simplest autonomous chaotic system." Physics Letters A 375.12 (2011): 1445-1450.

[15] Piper, Jessica R., and Julien Clinton Sprott. "Simple autonomous chaotic circuits." IEEE Transactions on Circuits and Systems II: Express Briefs 57.9 (2010): 730-734.

[16] Sprott, Julien Clinton. "A new chaotic jerk circuit." IEEE Transactions on Circuits and Systems II: Express Briefs 58.4 (2011): 240-243.

[17] Liu, Jizhao, et al. "Simplest chaotic system with a hyperbolic sine and its applications in DCSK scheme." IET Communications 12.7 (2018): 809-815.

[18] Liu, Jizhao, et al. "An approach for the generation of an nth-order chaotic system with hyperbolic sine." Entropy 20.4 (2018): 230.

[19] Sprott, Julien Clinton, Xiong Wang, and Guanrong Chen. "Coexistence of point, periodic and strange attractors." International Journal of Bifurcation and Chaos 23.05 (2013): 1350093.

[20] Kinsner, Witold. "Characterizing chaos through Lyapunov metrics." IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 36.2 (2006): 141-151.

[21] Liu, Jizhao, et al. "A Torus-Chaotic System and Its Pseudorandom Properties." Complexity 2020 (2020).