Hindawi
Complexity
Volume 2018, Article ID 9818520, 11 pages
https://doi.org/10.1155/2018/9818520

Research Article

Analysis of the Time Series Generated by a New High-Dimensional Discrete Chaotic System

Chuanfu Wang, Chunlei Fan, Kai Feng, Xin Huang, and Qun Ding

Electronic Engineering College, Heilongjiang University, Harbin 150080, China

Correspondence should be addressed to Qun Ding; qunding@aliyun.com

Received 19 December 2017; Revised 14 June 2018; Accepted 4 July 2018; Published 27 August 2018

Academic Editor: Christos Volos

The chaotic behavior of low-dimensional digital chaotic systems is seriously degraded, and the output sequence has a short period. In this study, a digital sequence generator based on a high-dimensional chaotic system is proposed to ensure performance and security. The proposed generator has low resource consumption, and the digital pseudo-random output sequence has a large period. To avoid the nonchaotic state, the multistability in the high-dimensional discrete chaotic system is analyzed. The statistical performance of the output sequence of the proposed digital high-dimensional chaotic system is evaluated, and the results demonstrate that it is a suitable candidate for a long-period pseudo-random sequence generator.

1. Introduction

Random sequence generators are used in several engineering applications including compressed sensing, image encryption, secure communication, spread-spectrum communication, and distance measurement [1–7]. They can be classified into true random sequence generators and pseudo-random sequence generators. True random sequence generators are based on physical sources, such as resistor thermal noise, atmospheric noise, and race hazard circuits. Although true random sequence generators are highly secure, their implementation is overly complex. Moreover, they are difficult to control. By contrast, pseudo-random sequence generators are based on seeds (initial values). A given seed completely determines the behavior of the pseudo-random sequence generator. Pseudo-random sequence generators are designed by using certain mathematical algorithms, such as linear feedback shift register (LFSR), nonlinear feedback shift register (NLFSR), linear congruence, nonlinear congruence, and BBS (Blum Blum Shub). These design methods are quite limited because they depend on the corresponding algorithm. LFSR is a linear function. It can be quickly reconstructed by the Berlekamp–Massey algorithm without prior knowledge of the seed. NLFSR, linear congruence, nonlinear congruence, and BBS are one-dimensional discrete maps. They generate a large-period pseudo-random sequence at high computational cost. With the rapid development of networks, the speed and period of pseudo-random sequence generators have attracted increasing attention. Therefore, the design method of pseudo-random sequence generators should be improved to meet the requirements of fast big data processing.

Chaos is a universal phenomenon in nonlinear systems. Chaotic systems exhibit a large number of special behaviors, such as initial value sensitivity, orbital ergodicity, and aperiodicity [8]. These behaviors are in accordance with the confusion and diffusion proposed by Shannon [9]. Therefore, chaotic systems are considered a new method for constructing pseudo-random sequence generators. A large number of chaotic systems have been discovered, and several chaos control methods have been proposed. Compared with other pseudo-random sequence generators, chaotic pseudo-random sequence generators allow a large variety of design choices. Therefore, the standards can be focused not only on the randomness of the output sequence but also on speed, period, and resource consumption. A chaotic pseudo-random sequence generator should be appropriately optimized to meet the requirements of different engineering applications, particularly big data processing.

Chaotic systems can be classified into continuous and discrete systems. For digital applications, continuous chaotic systems should first be discretized and then digitized. Discretization methods include the Euler method [10] and the Runge–Kutta method [11]. By contrast, discrete chaotic systems require digitizing only and are thus more appealing in digital applications. However, the chaotic behavior of digital chaotic systems gradually degenerates owing to the finite precision effect. Digital chaotic systems are not aperiodic but periodic [12–15].

Chaotic systems can also be classified into high-dimensional and low-dimensional systems. Low-dimensional chaotic systems have high efficiency and low resource consumption. The most commonly used low-dimensional systems are the logistic map [16], the Henon map [17], and the Sawtooth chaotic map [18–21]. The chaotic behavior of these systems is highly degenerate. It is difficult to ensure that the output sequence has a large period. By contrast, high-dimensional chaotic systems have a more complex nonlinear dynamic behavior. However, they have the disadvantages of high resource consumption and low-speed performance. Therefore, it is very necessary to design a large-period high-dimensional digital chaotic system with high-speed performance and low resource consumption.

Compared with other low-dimensional discrete chaotic systems, the Sawtooth chaotic map has a particularly simple form. It is easy to be digitized, as its output consists of positive decimals. There are several pseudo-random sequence generators based on the Sawtooth chaotic map. When the parameter satisfies a certain condition, the period of the output sequence can reach $2^{N-1}(2^n - 1)$ [22]. With parameter perturbations, the period can reach $2^{2N-1}(2^n - 1)$ [23]. The variables N and n represent the precision length and the dimension, respectively. Although the period of the output sequence is large at the same precision, the operating efficiency is not satisfactory.

In this study, a digital pseudo-random sequence generator based on a high-dimensional discrete chaos map is proposed. For low computing complexity, the values of all the parameters of the high-dimensional discrete chaos are set as powers of two. Compared with the period of other digital chaotic pseudo-random sequence generators, the period of the output sequence of the proposed generator is closer to the upper limit of the maximum period.

2. Sawtooth Chaotic Map

The Sawtooth chaotic map is also called Bernoulli shift or Renyi map. It is defined as

$$x_{n+1} = \beta x_n \bmod 1, \quad n = 0, 1, 2, 3, \dots, \tag{1}$$

where $1 < \beta \in R$ and $x_n \in [0, 1)$. It can be digitized by either fixed-point or float-point representation. Compared with floating-point computing, fixed-point computing is faster, and hardware implementation is smaller. Thus, for more efficient hardware implementation, fixed-point representation is chosen. For the decimal, the fixed-point representation of x_n is

$$\bar{x}_n = \lfloor 2^N x_n \rfloor, \quad \bar{x}_n \in N, \tag{2}$$

where $\lfloor 2^N x_n \rfloor$ represents the integer part of $2^N x_n$. Therefore, (1) can be transformed into

$$x_{n+1} = \lfloor 2^N \beta x_n \bmod 2^N \rfloor 2^{-N}, \quad n = 0, 1, 2, 3, \dots, \tag{3}$$

where β is an integer. Multiplying both sides by 2^N, (3) is transformed into

$$\bar{x}_{n+1} = \beta \bar{x}_n \bmod 2^N, \quad n = 0, 1, 2, 3, \dots . \tag{4}$$

The maximum period of the digital Sawtooth chaotic map (4) can reach 2^{N-2} for a certain parameter β. There are two most commonly used parameters: one is 30517578125 ($N = 35$), and the other is 1220703125 ($N = 31$). The output sequence of the digital Sawtooth chaotic map consists of integers. They should be quantified as binary sequences before being tested by the NIST SP800-22 test suite. To ensure large periodicity, the quantified binary output sequences are the most significant bits (MSB) of the digital Sawtooth chaotic map output. When the parameter β is 30517578125, the period of the output sequence can reach 2^{33}. The NIST SP800 test suite can reflect the statistical properties of the randomness of the pseudo-random sequence [24]. It consists of 15 different tests. Each test accepts as input a sequence of L bits and returns a P value. A sequence will pass the test if the corresponding P value is greater than 0.01. The NIST SP800 test suite may also be used as follows: each test accepts as input K sequences of L bits. The test results contain two indicators: U value and ratio. The ratio changes with K. The U value is the uniformity indicator of the P value. If the U value is greater than 10^{-4}, the P values are uniformly distributed. 100 output sequences of the digital Sawtooth chaotic map of length $L = 2 \times 10^6$ bits were tested by the NIST SP800 test suite, and the results are shown in Table 1.

The digital Sawtooth chaotic map fails the randomness test because the U value of the nonoverlapping template is significantly less than 10^{-4}. The resource consumption of the hardware implementation by FPGA (field-programmable gate array) is shown in Table 2, and the block diagram of the digital Sawtooth chaotic map in FPGA is shown in Figure 1.

3. Digital Pseudo-Random Sequence Generator

Although the randomness of the digital Sawtooth chaotic map is not satisfactory, its form is quite simple. Accordingly, a high-dimensional discrete chaotic system based on the Sawtooth chaotic map is proposed, and its digital model is analyzed in detail.

Table 1: Results of the NIST SP800-22 test suite.

Test	U value	Ratio	Result
Frequency	0.534146	99/96	Success
Block frequency	0.102526	98/96	Success
Cumulative sums	0.816537	98/96	Success
Runs	0.911413	99/96	Success
Longest run	0.935716	99/96	Success
Rank	0.080519	99/96	Success
FFT	0.213309	100/96	Success
Nonoverlapping template	0.000005	100/96	Failure
Overlapping template	0.996335	98/96	Success
Universal	0.181557	97/96	Success
Approximate entropy	0.739918	100/96	Success
Random excursions	0.103401	72/70	Success
Random excursions variant	0.032000	74/70	Success
Serial	0.834308	100/96	Success
Linear complexity	0.759756	99/96	Success

Table 2: Resource consumption of the hardware implementation by FPGA.

Logic elements	Memory bits	Multiplier 9-bit elements	Max frequency	Throughput
52	0	6	72.97 MHz	72.97 M/s

3.1. Fast Arithmetic Operation on Fixed-Point Computing. The arithmetic operations are multiplication, division, addition, and subtraction. Division is difficult to implement on FPGA; thus, it should be avoided in the formulation of the chaotic equation. The maximum frequency of FPGA is seriously affected by multiplication. In Table 2, the maximum frequency of the hardware implementation of the digital Sawtooth chaotic map by FPGA is not high for multiplication. For fast arithmetic operations in fixed-point computations, the values of all parameters are set as powers of two in this study. Therefore, division and multiplication are easier to implement. When the multiplier is a power of two, the function of 2^n is to shift n bits of the multiplicand to the right, and the missing low position bits are filled by 0. When the divisor is a power of two, the function of 2^n is to shift n bits of the dividend to the left, and the missing high position bits are filled by 0. The complex multiplication and division operations are thus reduced to the right and left shift operations, respectively. The shift operation is easier to implement on FPGA. Compared with subtraction, addition is easier to implement. Therefore, optimized multiplication, optimized division, and addition are selected in the design of the high-dimensional discrete chaotic system.

3.2. High-Dimensional Discrete Chaotic Map Modeling. By taking into account the form of the Sawtooth chaotic map, a high-dimensional discrete chaotic is proposed as follows:

$$S_{n+1} = AS_n (\mathrm{mod}\ 1), \quad n = 0, 1, 2, 3, \dots, \tag{5}$$

where S_n is defined as a state vector $(x_1(n), x_2(n), x_3(n), \dots, x_m(n))^T$, and A is defined by

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdot & \cdot & \cdot & a_{1,m-2} & a_{1,m-1} & a_{1,m} \\ a_{2,1} & a_{2,2} & \cdot & \cdot & \cdot & a_{2,m-1} & a_{2,m-2} & a_{2,m} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & & \cdot & & \cdot & \cdot & \cdot \\ \cdot & \cdot & & & \cdot & \cdot & \cdot & \cdot \\ a_{m,1} & a_{m,2} & \cdot & \cdot & \cdot & a_{m,m-2} & a_{m,m-1} & a_{m,m} \end{bmatrix}. \tag{6}$$

In (6), there are at most two nonzero elements per row to further reduce complexity and improve parallel computing efficiency. Therefore, the following parameter matrix is proposed:

$$A = \begin{bmatrix} 2^0 & 0 & \cdot & \cdot & \cdot & 0 & 0 & 2^0 \\ 2^0 & 0 & \cdot & \cdot & \cdot & 0 & 0 & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & & \cdot & & \cdot & \cdot & \cdot \\ \cdot & \cdot & & & \cdot & \cdot & \cdot & \cdot \\ 2^{-1} & 0 & \cdot & \cdot & \cdot & 0 & 2^0 & 0 \end{bmatrix}, \quad m \geq 3, \tag{7}$$

where $a_{i,i-1} = 2^0 (i = 2, 3, 4, \dots)$, $a_{1,1} = a_{1,m} = a_{m,m-1} = 2^0$, and $a_{m,1} = 2^{-1}$.

Proposition 1. *When m is odd, the value of the determinant of A (*det (A)*) is 1; when m is even,* det $(A) = -1$.

Proof. Using the expansion theorem for determinants, det $(A) = a_{1,m}(-1)^{1+m}\prod_{i=4}^{m-1} a_{i,i-1}(-1)^{i+i}$. For $a_{i,j} = 2^0\ (j = i - 1, i = 2, 3, 4, \dots)$ and $a_{1,m} = 2^0$, det $(A) = (-1)^{1+m}$. When m is odd, det $(A) = 1$; when m is even, det $(A) = -1$.

Proposition 2. *For the high-dimensional discrete map (5) and the parameter matrix A (7), there exists at least one positive Lyapunov exponent in (5).*

Proof. The Jacobian matrix of (5) is the parameter matrix A (7). It is assumed that the eigenvalues of the parameter matrix A are $\lambda_i\ (i = 1, 2, 3, \dots, m)$. By the fundamental property of the eigenvalues, det $(A) = \prod_{i=1}^{m} \lambda_i$. By Proposition 1,

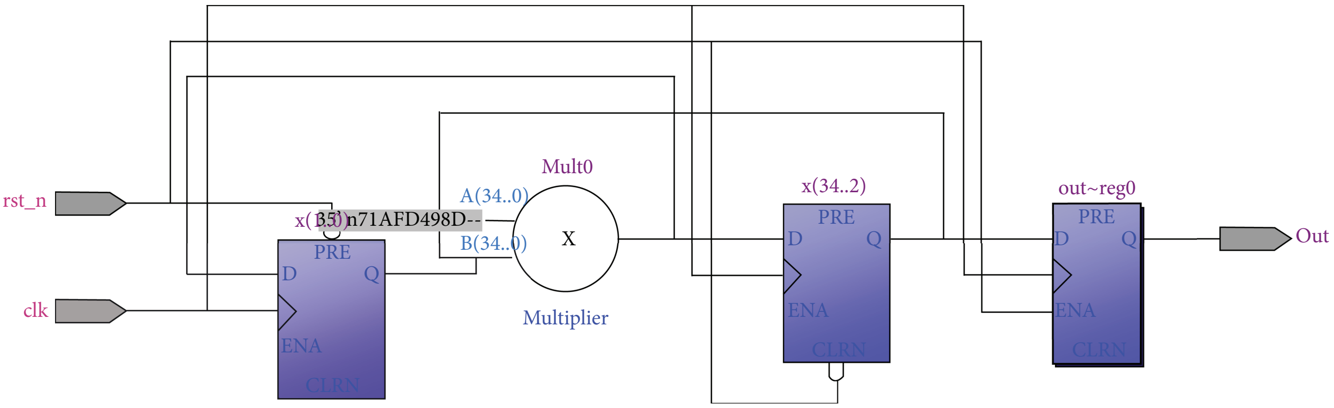


Figure 1: Block diagram of the digital Sawtooth chaotic map in FPGA.

$\det(A) = (-1)^{1+m}$. There are two cases for the eigenvalues of the parameter matrix A: either there exists at least a λ_k whose absolute value is greater than 1 or all eigenvalues are equal to 1. The latter is obviously impossible. Therefore, there exists at least one positive Lyapunov exponent in (5).

For at least one positive Lyapunov exponent in (5), the high-dimensional discrete map must be a chaotic system. In practical engineering, the dimension of the chaotic system need not be high. Therefore, a 6-dimensional discrete chaotic system is proposed in this study. The parameter matrix A is defined by

$$A = \begin{bmatrix} 2^0 & 0 & 0 & 0 & 0 & 2^0 \\ 2^0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2^0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2^0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2^0 & 0 & 0 \\ 2^{-1} & 0 & 0 & 0 & 2^0 & 0 \end{bmatrix}. \tag{8}$$

Combined with (5), the 6-dimensional discrete chaotic system is represented by

$$\begin{aligned} x_1(n) &= x_1(n-1) + x_6(n-1) (\operatorname{mod} 1), \\ x_2(n) &= x_1(n-1) (\operatorname{mod} 1), \\ x_3(n) &= x_2(n-1) (\operatorname{mod} 1), \\ x_4(n) &= x_3(n-1) (\operatorname{mod} 1), \\ x_5(n) &= x_4(n-1) (\operatorname{mod} 1), \\ x_6(n) &= 2^{-1} x_1(n-1) + x_5(n-1) (\operatorname{mod} 1). \end{aligned} \tag{9}$$

Then, the six Lyapunov exponents are $\mathrm{LE1} = 0.3915$, $\mathrm{LE2} = -0.0523$, $\mathrm{LE3} = -0.0523$, $\mathrm{LE4} = -0.0673$, $\mathrm{LE5} = -0.1098$, and $\mathrm{LE6} = -0.1098$. Although (9) has only one positive Lyapunov exponent, the output sequence of its digital model has a large period. A discrete chaotic system with a large number of positive Lyapunov exponents does not ensure that the output sequence generated by its digital model will have a larger period. The phase diagram of the chaotic attractors is shown in Figure 2.

The variables $x_2(n)$, $x_3(n)$, $x_4(n)$, and $x_5(n)$ are the delay signal of $x_1(n)$. Therefore, the phase diagrams (a), (b), and (c) are similar. Figure 3 shows the time series plots of the two sequences $x_1(n)$ and $x_6(n)$ generated by the map in (9), where $n = 0, 1, 2, \ldots, 10000$, $x_1(0) = 0.2$, $x_2(0) = 0.3$, $x_3(0) = 0.1$, $x_4(0) = 0.3$, $x_5(0) = 0.2$, and $x_6(0) = 0.2$.

An autocorrelation algorithm can be used to detect the periodicity of the time series. It is defined as follows:

$$r_{xx}(t) = \sum_{n=1}^{T} x_n x_{n+t}, \qquad t = 0, 1, 2, 3, \ldots, 2n. \tag{10}$$

From Figure 3, it can be seen that the triangular wave is considerably smooth. This indicates that the output sequence is aperiodic.

Multistability is present in various chaotic systems. The parameter and the initial value seriously affect the stability of the chaotic system. In (5) and (9), the fixed parameter has no effect on stability. The Jacobian matrices of (5) and (9) are (7) and (8), respectively, which are constant matrices. Therefore, the Lyapunov exponents depend only on the constant matrices (7) and (8) and are the invariant constants. In (9), the six Lyapunov exponents are 0.3915, −0.0523, −0.0523, −0.0673, −0.1098, and −0.1098 for the initial value S_0. For the positive Lyapunov exponent 0.3915, the discrete map (9) is a chaotic system. Propositions 1 and 2 prove that in (5), there exists at least one positive Lyapunov exponent. Therefore, (5) is also a chaotic system for all initial values.

3.3. High-Dimensional Digital Chaotic Map. Using (2) and (4), the high-dimensional digital chaotic map is defined as follows:

$$S_{n+1} = \lfloor A S_n \rfloor \left(\operatorname{mod} 2^N\right). \tag{11}$$

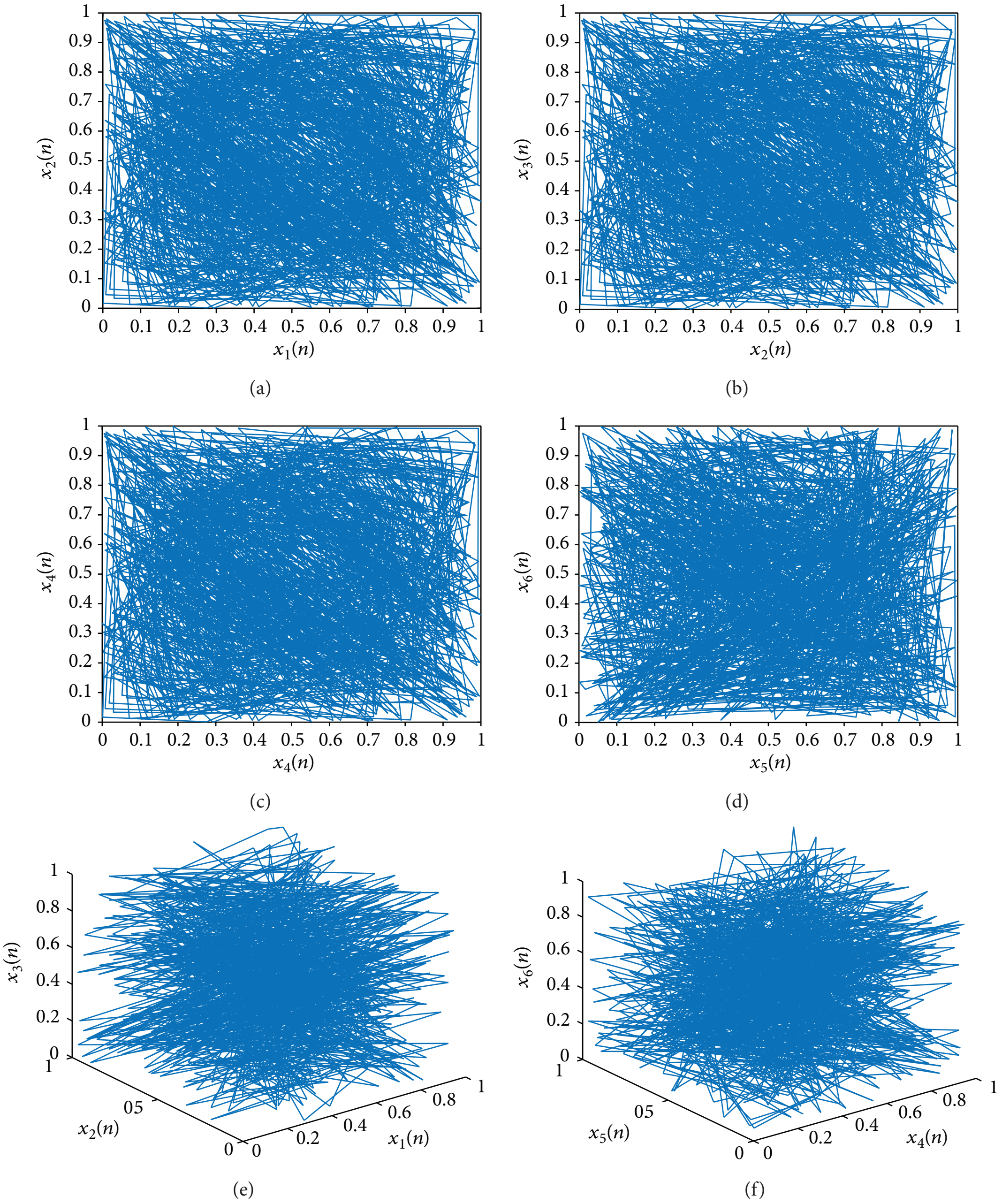


Figure 2: Phase diagram of the chaotic attractors: (a) $x_1(n) - x_2(n)$ plane, (b) $x_2(n) - x_3(n)$ plane, (c) $x_3(n) - x_4(n)$ plane, (d) $x_5(n) - x_6(n)$ plane, (e) $x_1(n) - x_2(n) - x_3(n)$ plane, and (f) $x_4(n) - x_5(n) - x_6(n)$ plane.

Equation (9) is transformed into

$$\begin{aligned}
x_1(n) &= x_1(n-1) + x_6(n-1) \bmod \left(2^N\right),\\
x_2(n) &= x_1(n-1) \bmod \left(2^N\right),\\
x_3(n) &= x_2(n-1) \bmod \left(2^N\right),\\
x_4(n) &= x_3(n-1) \bmod \left(2^N\right),\\
x_5(n) &= x_4(n-1) \bmod \left(2^N\right),\\
x_6(n) &= \left\lfloor 2^{-1} x_1(n-1) \right\rfloor + x_5(n-1) \bmod \left(2^N\right).
\end{aligned} \tag{12}$$

where $x_i \in \mathbb{N}\,(i = 1, 2, 3, 4, 5, 6)$ is in the interval $[0, 2^N - 1]$. $\lfloor 2^{-1} x_1(n-1) \rfloor$ represents the integer part of $2^{-1} x_1(n-1)$. Although the Jacobian matrix of (12) is also (8), the value spaces of $x_i \in \mathbb{N}\,(i = 1, 2, 3, 4, 5, 6)$ are limited. Therefore, (12) is periodic and can be described by the finite state machine in Figure 4.

The phase diagram of the attractors is also shown in Figure 5.

Compared with Figure 2, the phase diagram of the attractors shows a significant change: it is sparse because the value space of the variables $x_i\,(i = 1, 2, 3, 4, 5, 6)$ is limited. In Figure 5, the attractors are obviously periodic; thus, the chaotic attractors degenerate into periodic attractors.

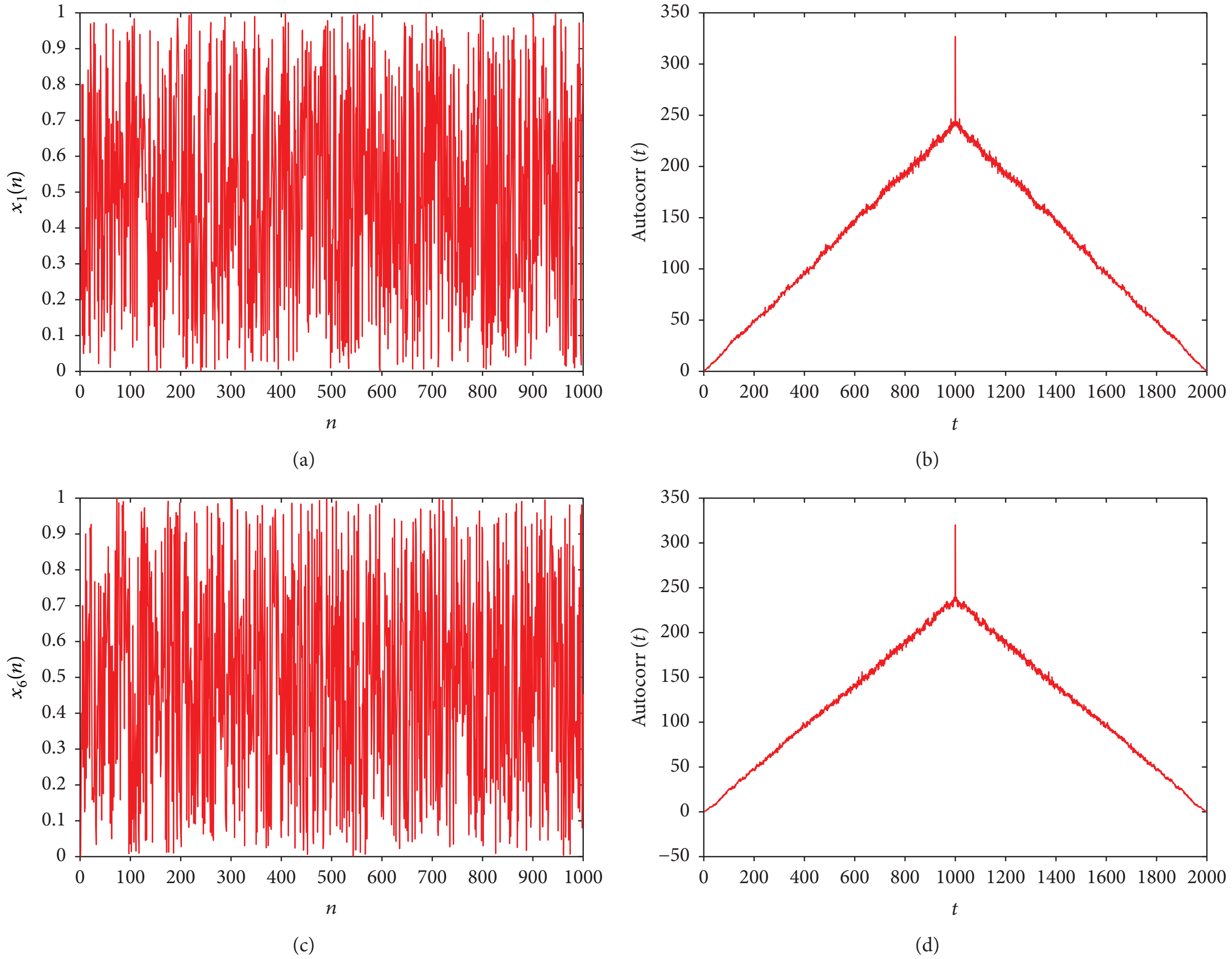


Figure 3: Time series plots of $x_1(n)$, $x_6(n)$, and their autocorrelation: (a) $x_1(n)$, (b) autocorrelation of $x_1(n)$, (c) $x_6(n)$, and (d) autocorrelation of $x_6(n)$.

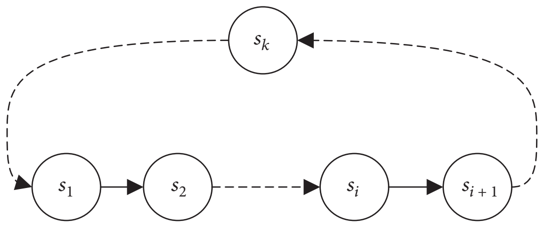


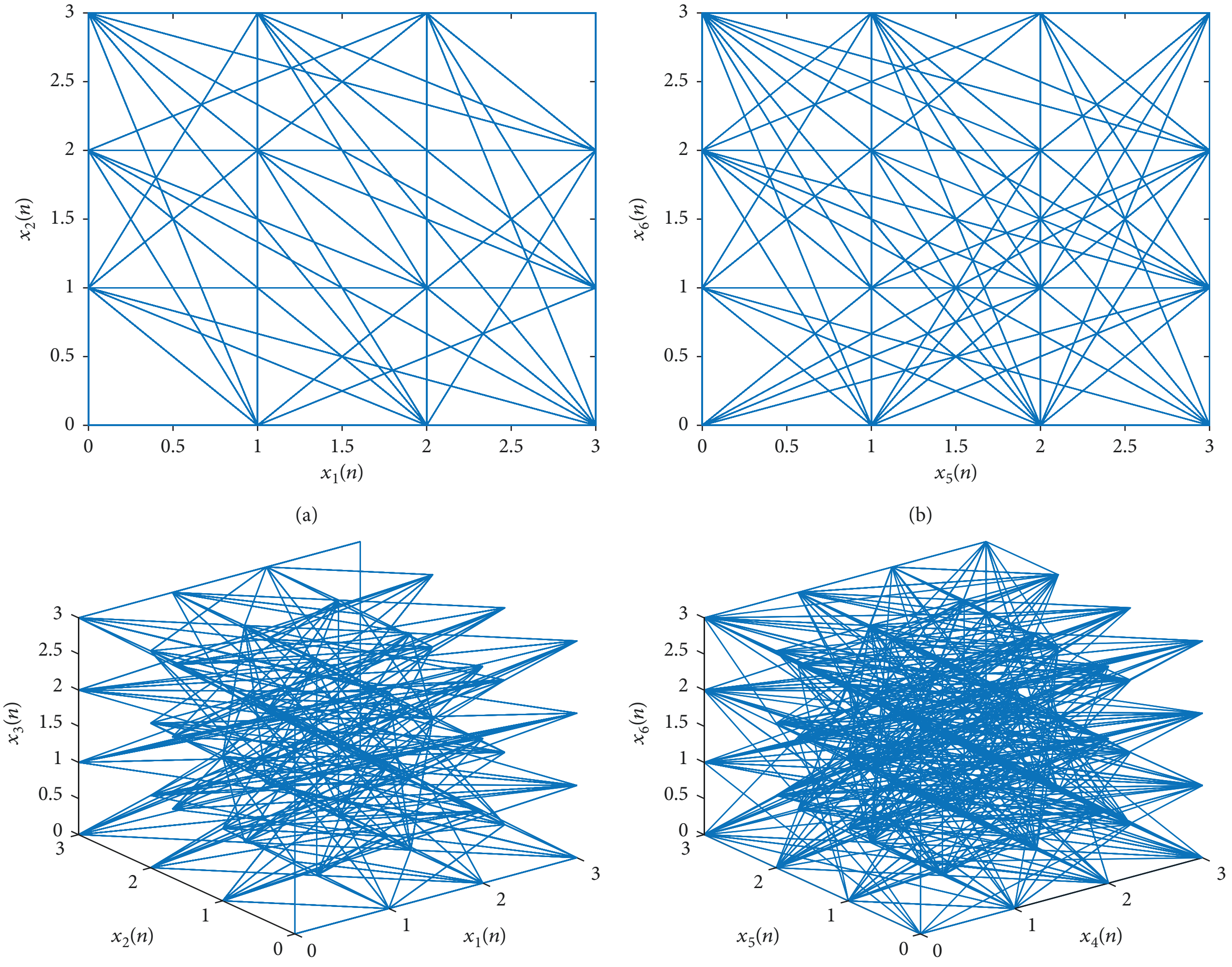
Figure 4: Finite state machine.

Figure 6 shows the time series plots of the two sequences $x_1(n)$ and $x_6(n)$ generated by the map in (11), where $n = 0, 1, 2, \ldots, 5000$, $x_1(0) = 2$, $x_2(0) = 3$, $x_3(0) = 1$, $x_4(0) = 3$, $x_5(0) = 2$, and $x_6(0) = 2$.

Compared with Figure 3, the triangular wave is not quite smooth and has a large number of sharp peaks. This indicates that the output sequence is not aperiodic. From Figures 6(a) and 6(c), it is obvious that the output sequence is periodic.

3.3.1. Period Analysis. Owing to the finite precision effect in the physical device, the chaotic behavior of the digital chaotic system gradually degenerates. The output sequences of the digital chaotic systems are all periodic. Therefore, a large period is an important indicator. For precision length N, the maximum period of the output sequence of the 6-dimensional digital map is $(2^6)^N$. The period of the output sequence of the digital chaotic system (12) for various values of N is shown in Table 3.

From Table 3, it can be seen that the period T of the output sequence generated by (12) increases sharply as N increases. Compared with other high-dimensional map periods (T_1 in [22] and T_2 in [23]), T is significantly larger and closer to the maximum period. The initial values are selected as follows: $x_1(0) = 2$, $x_2(0) = 3$, $x_3(0) = 1$, $x_4(0) = 3$, $x_5(0) = 2$, and $x_6(0) = 2$. With computation precision 5, the period of the output sequence can reach 594621509 by using a 5-bit addition operation. It is efficient



(a)

(b)

(c)

(d)

Figure 5: Phase diagram of the attractors: (a) $x_1(n) - x_2(n)$ plane, $N = 2$; (b) $x_5(n) - x_6(n)$ plane, $N = 2$; (c) $x_1(n) - x_2(n)$ plane, $N = 2$; and (d) $x_5(n) - x_6(n)$ plane, $N = 2$.

to generate a large period pseudo-random sequence with low resource consumption.

3.3.2. Quantification Analysis. The N-bit fixed-point representation of x_i is $x_i = b(x_i)b_{N-1}(x_i) \cdots b_1(x_i)$, $b_j(x_i) \in \{0, 1\}$, $i = 1, 2, 3, 4, 5, 6$, and $j = 1, 2, 3, \ldots, N$. The low position bits of x_i, $i = 1, 2, 3, 4, 5, 6$, are quantified as the output binary sequences. Therefore, more than one bit can be generated at a time. The throughput of the chaotic pseudo-random sequence generator can be significantly improved. Currently, SCMs (single-chip micyocos), ARMs (advanced RISC machines), CPUs, and FPGAs can process several bytes in one clock cycle, that is, 8 bits, 16 bits, 32 bits, and 64 bits. Therefore, quantification with several output bits is beneficial to information processing. However, it is unsafe to quantify all bits of $x_i\,(i = 1, 2, 3, 4, 5, 6)$ as the output binary sequences. If the output sequences contain all the information of $x_i\,(i = 1, 2, 3, 4, 5, 6)$, $x_i\,(i = 1, 2, 3, 4, 5, 6)$ can be easily predicted and reconstructed without prior knowledge of the initial seeds.

3.3.3. Quantity Analysis. The output sequences of (12) are $x_1(n)$, $x_2(n)$, $x_3(n)$, $x_4(n)$, $x_5(n)$, and $x_6(n)$. $x_1(n)$, $x_2(n)$, $x_3(n)$, $x_4(n)$, and $x_5(n)$ are similar. Therefore, two different sequences can be generated by (12). A large number of new sequences can be generated by operations between $x_6(n)$ and $x_1(n)\,(i = 1, 2, 3, 4, 5)$. In (12), there are three types of schemes for the output sequence:

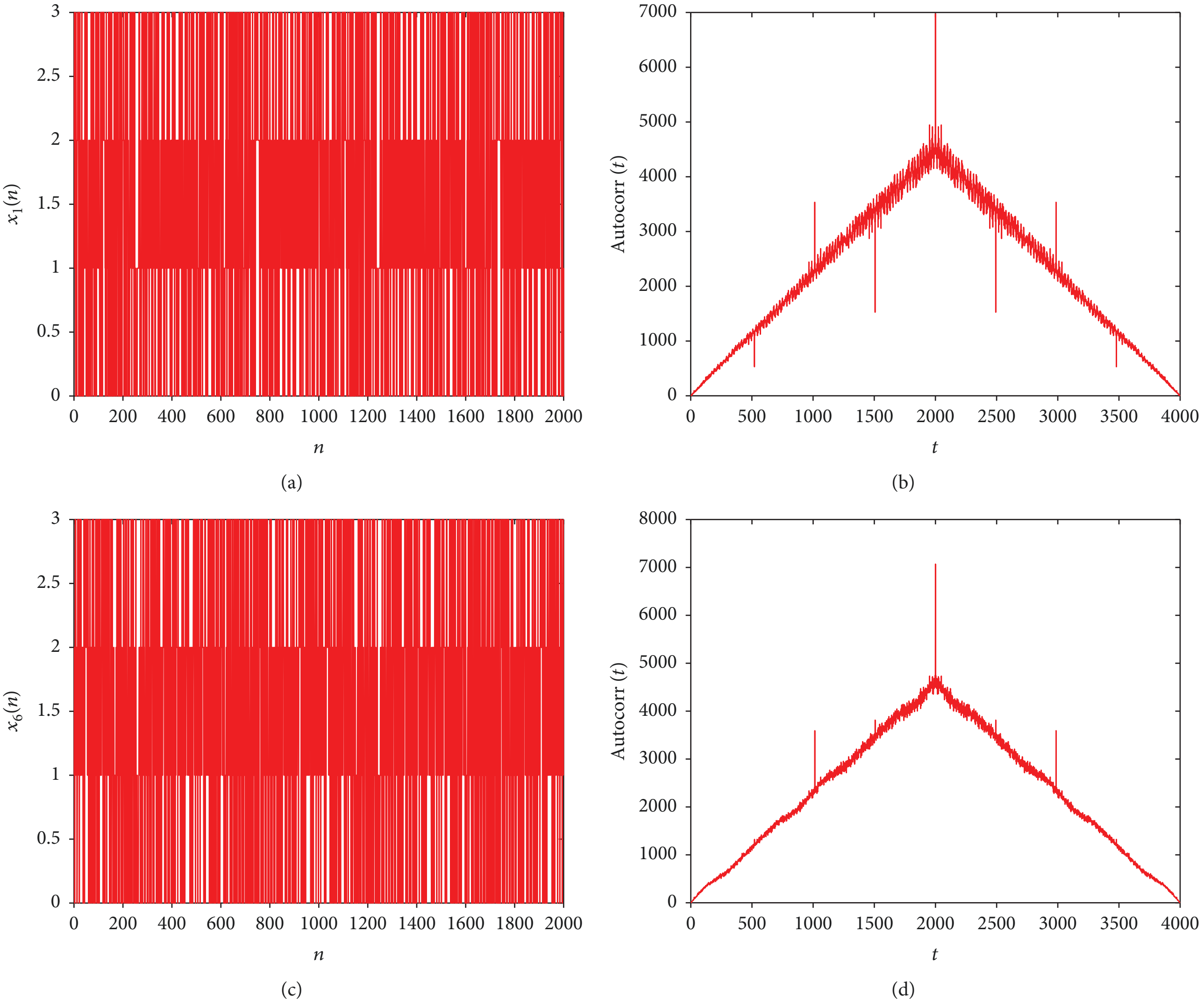
(a) The output sequence generated by $x_1(n)$.

(b) The output sequence generated by $x_2(n)$.

(c) The output sequence generated by operations between $x_6(n)$ and $x_i(n) (i = 1, 2, 3, 4, 5)$, namely, $x_6(n) + x_i(n)$ and $x_6(n) \oplus x_i(n)$.

3.3.4. Randomness Analysis. 100 sequences of length $L = 2 \times 10^6$ bits were tested by the NIST SP800 test suite, and the results are shown in Table 4. For comparison with Table 1, N was set to 35.

The asterisk "*" indicates that the corresponding test failed. From Tables 4 and 5, the randomness of the low



(a)

(b)

(c)

(d)

Figure 6: Time series plots of $x_1(n)$, $x_6(n)$, and their autocorrelation: (a) $x_1(n)$, (b) autocorrelation of $x_1(n)$, (c) $x_6(n)$, and (d) autocorrelation of $x_6(n)$.

Table 3: Period analysis.

Precision	T	T_1	T_2	$T_{\max}$	$T/T_{\max}$
2	986	126	504	4096	24.07%
3	50160	252	2016	262144	19.13%
4	15085157	504	8064	16777216	89.91%
5	594621509	1008	32256	1073741824	55.37%

position bits is better than that of high position bits. The randomness of the 8-bit outputs in $x_1(n)$ and $x_2(n)$ all passed the NIST SP800 test suit. This is due to the fact that the matrix A involves the division 2^{-1}. In fixed-point representations, when the divisor is a power of two, the function of 2^n is to shift n bits of the dividend to the left, and the missing high position bits are filled by 0. Therefore, the low position bits are disturbed by the high position bits. As the missing high position bits are filled by 0, the disturbance of the high position bits is not significant. The test results for the output sequence generated by operations between $x_6(n)$ and $x_i(n), i = 1, 2, 3, 4, 5,$ are shown in Table 6.

By contrast, the randomness of the high position bits is better. The randomness of the output sequence generated by the addition operation "+" is better compared with that by the xor operation "$\oplus$," There are 27 different sequences in Table 6, and 17 different sequences passed the NIST SP800 test suit.

3.3.5. Performance Analysis. With the same N and m, the proposed pseudo-random sequence generator is faster than others because there are at most two nonzero elements per row in the parameter matrix A, which ensures higher parallel performance in the hardware implementation. The resource consumption is shown in Table 7.

The consumption of memory bits and multiplier 9-bit elements is smaller, and the max frequency and throughput

Table 4: Randomness test of x_1 by NIST SP800.

Test	8		16		32	
	U value	Ratio	U value	Ratio	U value	Ratio
Frequency	0.455937	100/96	0.181557	99/96	0.055361	99/96
Block frequency	0.129620	100/96	0.224821	99/96	0.012650	99/96
Cumulative sums	0.401199	100/96	0.262249	98/96	0.798139	98/96
Runs	0.020548	98/96	0.867692	96/96	0.897763	98/96
Longest run	0.911413	99/96	0.153763	99/96	0.816537	99/96
Rank	0.851383	99/96	0.401199	100/96	0.924076	100/96
FFT	0.004981	98/96	0.699313	99/96	0.657933	98/96
Nonoverlapping template	0.007694	99/96	0.014550	96/96	0.004981	98/96
Overlapping template	0.455937	98/96	0.779188	100/96	0.574903	100/96
Universal	0.383827	99/96	0.779188	100/96	0.867692	99/96
Approximate entropy	0.419021	100/96	0.366918	98/96	0.867692	99/96
Random excursions	0.03200	75/72	0.001254	75/71	0.076389	69/67
Random excursions variant	0.007234	75/72	0.069925	74/71	0.063958	71/67
Serial	0.334538	99/96	0.115387	95/96*	0.437274	98/96
Linear complexity	0.678686	96/96	0.534146	99/96	0.304126	95/96*

The asterisk "*" in Table 4 indicate that the corresponding test failed.

Table 5: Randomness test of x_6 by NIST SP800.

Test	8		16		32	
	U value	Ratio	U value	Ratio	U value	Ratio
Frequency	0.779188	98/96	0.897763	99/96	0.055361	99/96
Block frequency	0.213309	99/96	0.350485	100/96	0.012650	99/96
Cumulative sums	0.494392	99/96	0.739918	98/96	0.798139	98/96
Runs	0.045675	97/96	0.494392	99/96	0.897763	98/96
Longest run	0.419021	100/96	0.181557	99/96	0.816537	99/96
Rank	0.037566	98/96	0.366918	99/96	0.924076	100/96
FFT	0.964295	98/96	0.554420	98/96	0.657933	98/96
Nonoverlapping template	0.037566	98/96	0.019188	99/96	0.004981	98/96
Overlapping template	0.455937	99/96	0.334538	97/96	0.574903	100/96
Universal	0.224821	99/96	0.574903	100/96	0.867692	99/96
Approximate entropy	0.935716	97/96	0.437274	99/96	0.867692	99/96
Random excursions	0.063958	76/73	0.197677	72/70	0.076389	69/67
Random excursions variant	0.036868	74/73	0.000854	74/70	0.063958	70/67
Serial	0.798139	98/96	0.494392	98/96	0.437274	99/96
Linear complexity	0.494392	99/96	0.350485	99/196	0.304126	95/96*

The asterisk "*" in Table 5 indicate that the corresponding test failed.

Table 6: Randomness test of the output sequence generated by operations between $x_6(n)$ and $x_i(n)$ $(i = 1, 2, 3, 4, 5)$.

Bits	$x_6 \oplus x_1$	$x_6 \oplus x_2$	$x_6 \oplus x_3$	$x_6 \oplus x_4$	$x_6 \oplus x_5$	$x_6 + x_2$	$x_6 + x_3$	$x_6 + x_4$	$x_6 + x_5$
8	Failure	Failure	Failure	Failure	Success	Success	Failure	Success	Success
16	Failure	Success	Failure	Failure	Success	Success	Success	Success	Failure
32	Success	Success	Failure	Success	Failure	Success	Success	Success	Success

are higher compared with the corresponding values for the Sawtooth chaotic map. The block diagram of the 6-dimensional digital chaotic map in FPGA is shown in Figure 7.

3.3.6. Key Space Analysis. It is proved that the 2^{100} key space is sufficiently large to resist attacks by existing computers [25]. Therefore, the precision length N need only be greater than 17 for (12). The key space changes with the precision

Table 7: Performance of hardware implementation on FPGA.

Logic elements	Memory bits	Multiplier 9-bit elements	Max frequency	Throughput		
				8 bits	16 bits	32 bits
119	66	0	177.62 MHz	1.387 G/s	2.775 G/s	5.550 G/s

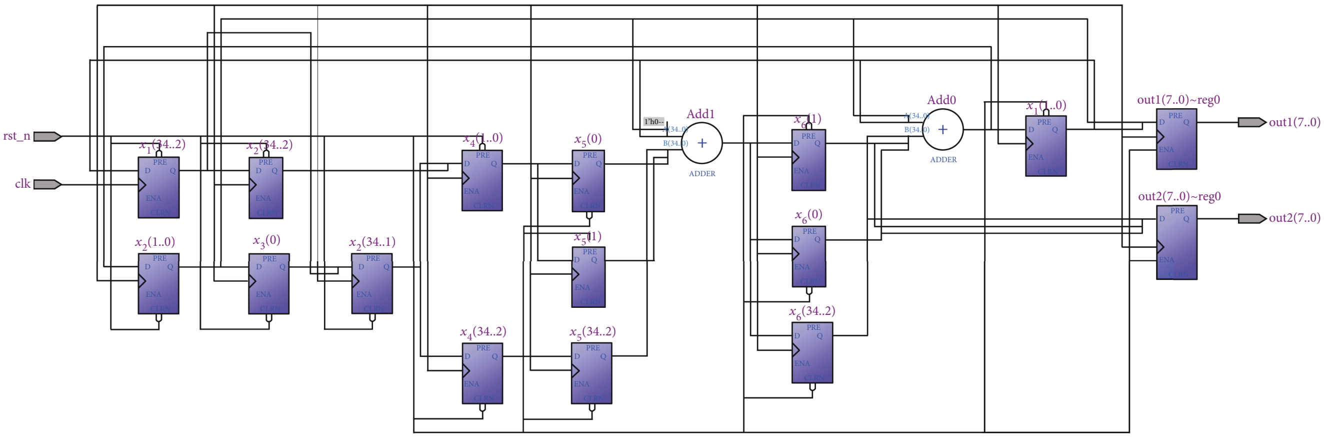


Figure 7: Block diagram of a 6-dimensional digital chaotic map in FPGA.

length N and the number of variables m. The key space of (11) is $2^{N \cdot m}$.

3.3.7. Hardware and Software Parameter Selection. All the logic circuits in the hardware implementation used a single Altera Cyclone II family chip. The statistical analysis of the pseudo-random binary sequence was performed by the NIST SP800 test suite version 2.1.2 software package. In the parameter setting of NIST SP800, the block length for the block frequency test was 128, the block length for the nonoverlapping template test was 9, the block length of the overlapping template test was 9, the block length of the approximate rntropy test was 9, and the block length of the linear complexity test was 500.

4. Conclusion

The periodicity of the output sequence of a high-dimensional digital chaotic map is obviously larger than that of the output sequence of a low-dimensional digital chaotic map, and its randomness is also better. The proposed pseudo-random sequence generator based on a high-dimensional discrete chaotic map has parallel structure and lower hardware resource consumption, and its output sequence has a considerably large period. Moreover, the statistical performance of the proposed pseudo-random sequence generator was evaluated, and it was shown that it can pass all the tests in NIST SP8000 test suit.

Conflicts of Interest

The authors declare that there are no competing interests regarding the publication of this article.

Acknowledgments

This work is supported financially by National Natural Science Foundation of China (no. 61471158) and the Innovative Team of the Heilongjiang Province (no. 2012TD007).

References

[1] P. L'Ecuyer, "Uniform random number generation," *Annals of Operations Research*, vol. 53, no. 1, pp. 77–120, 1994.

[2] S. K. Park and K. W. Miller, "Random number generators: good ones are hard to find," *Communications of the ACM*, vol. 31, no. 10, pp. 1192–1201, 1988.

[3] K. Entacher, "Bad subsequences of well-known linear congruential pseudorandom number generators," *ACM Transactions on Modeling and Computer Simulation*, vol. 8, no. 1, pp. 61–70, 1998.

[4] B. D. Ripley, "Thoughts on pseudorandom number generators," *Journal of Computational and Applied Mathematics*, vol. 31, no. 1, pp. 153–163, 1990.

[5] C. Li, G. Luo, K. Qin, and C. Li, "An image encryption scheme based on chaotic tent map," *Nonlinear Dynamics*, vol. 87, no. 1, pp. 127–133, 2017.

[6] G. Mazzini, "DS-CDMA systems using q-level m sequences: coding map theory," *IEEE Transactions on Communications*, vol. 45, no. 10, pp. 1304–1313, 1997.

[7] D. H. Lehmer, "Mathematical methods in large-scale computing units," in *Proceedings of Second Symposium on Large-Scale Digital Calculating Machinery*, pp. 141–146, Cambridge, MA, USA, 1949.

[8] E. N. Lorenz, "Deterministic nonperiodic flow," *Journal of the Atmospheric Sciences*, vol. 20, no. 2, pp. 130–141, 1963.

[9] C. E. Shannon, "Communication theory of secrecy systems," *Bell Labs Technical Journal*, vol. 28, no. 4, pp. 656–715, 1949.

[10] A. C. Hindmarsh and L. R. Petzold, "Algorithms and software for ordinary differential equations and differential-algebraic equations, part I: Euler methods and error estimation," *Computers in Physics*, vol. 9, no. 1, pp. 34–41, 1995.

[11] E. Hairer, M. Roche, C. Lubich, E. Hairer, M. Roche, and C. Lubich, "Description of differential-algebraic problems," in *The Numerical Solution of Differential-Algebraic Systems by Runge-Kutta Methods*, vol. 1409 of Lecture Notes in Mathematics, pp. 1–13, 1989.

[12] T. Miyazaki, S. Araki, and S. Uehara, "Some properties of logistic maps over integers," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E93-A, no. 11, pp. 2258–2265, 2010.

[13] Y. Wang, Z. Liu, J. Ma, and H. He, "A pseudorandom number generator based on piecewise logistic map," *Nonlinear Dynamics*, vol. 83, no. 4, pp. 2373–2391, 2016.

[14] B. Yang and X. Liao, "Period analysis of the logistic map for the finite field," *Science China Information Sciences*, vol. 60, no. 2, 2017.

[15] K. J. Persohn and R. J. Povinelli, "Analyzing logistic map pseudorandom number generators for periodicity induced by finite precision floating-point representation," *Chaos, Solitons & Fractals*, vol. 45, no. 3, pp. 238–245, 2012.

[16] R. M. May, "Simple mathematical models with very complicated dynamics," *Nature*, vol. 261, no. 5560, pp. 459–467, 1976.

[17] M. Henon, "A two-dimensional mapping with a strange attractor," *Communications in Mathematical Physics*, vol. 50, no. 1, pp. 69–77, 1976.

[18] T. Addabbo, A. Fort, S. Rocchi, and V. Vignoli, "Digitized chaos for pseudo-random number generation in cryptography," in *Chaos-Based Cryptography*, L. Kocarev and S. Lian, Eds., pp. 67–97, Springer, Berlin, Heidelberg, 2011.

[19] M. Jessa, "Designing security for number sequences generated by means of the sawtooth chaotic map," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 53, no. 5, pp. 1140–1150, 2006.

[20] L. Cong and W. Xiaofu, "Design and realization of an FPGA-based generator for chaotic frequency hopping sequences," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 48, no. 5, pp. 521–532, 2001.

[21] K. Kelber, "N-dimensional uniform probability distribution in nonlinear autoregressive filter structures," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 47, no. 9, pp. 1413–1417, 2000.

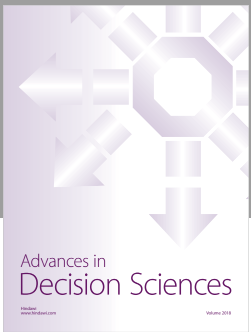
[22] A. D. Barnard, J. R. Silvester, and W. G. Chambers, "Guaranteeing the period of linear recurring sequences (mod 2^e)," *IEE Proceedings E (Computers and Digital Techniques)*, vol. 140, no. 5, pp. 243–246, 1993.

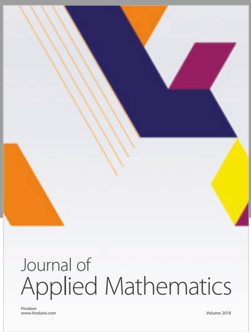
[23] D. Lambić and M. Nikolić, "Pseudo-random number generator based on discrete-space chaotic map," *Nonlinear Dynamics*, vol. 90, no. 1, pp. 223–232, 2017.

[24] L. Bassham, A. Rukhin, J. Soto et al., *A statistical test suite for random and pseudorandom number generators for cryptographic applications, NIST special publication 800–22, Rev.1-a*, National Institute of Standards and Technology, Gaithersburg, MD, USA, 2010.

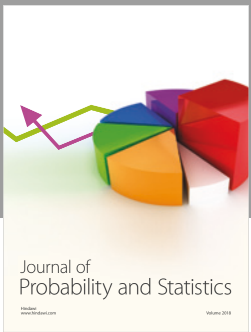
[25] G. Alvarez and S. Li, "Some basic cryptographic requirements for chaos-based cryptosystems," *International Journal of Bifurcation and Chaos*, vol. 16, no. 8, pp. 2129–2151, 2006.

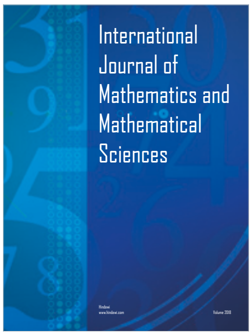
Advances in
Operations Research

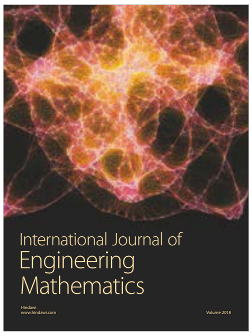

Advances in
Decision Sciences


Journal of
Applied Mathematics

The Scientific
World Journal

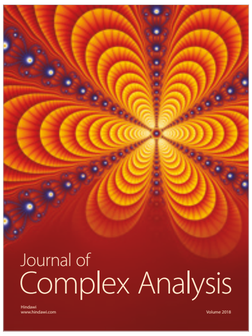

Journal of
Probability and Statistics

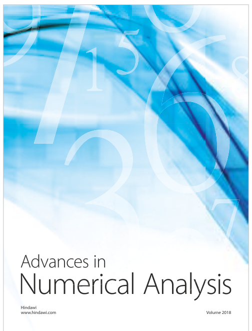

International
Journal of
Mathematics and
Mathematical
Sciences

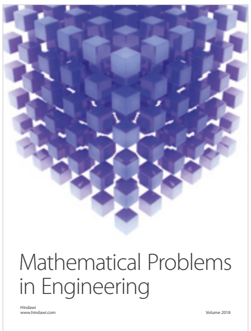

International Journal of
Engineering
Mathematics

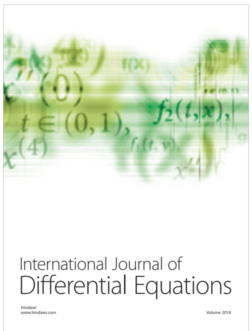
Journal of
Optimization

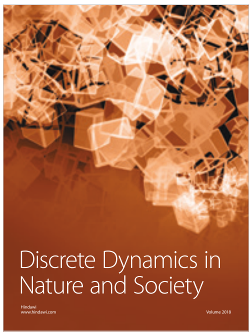
International Journal of
Analysis

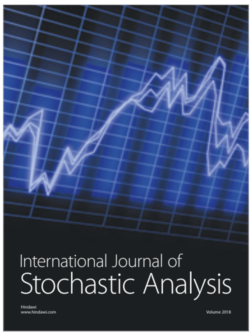

Journal of
Complex Analysis


Advances in
Numerical Analysis


Mathematical Problems
in Engineering

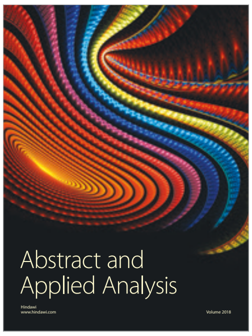

International Journal of
Differential Equations

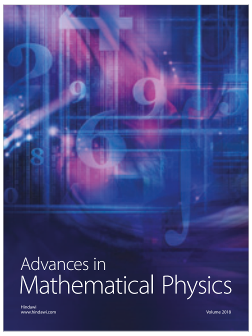

Discrete Dynamics in
Nature and Society


International Journal of
Stochastic Analysis

Journal of
Mathematics

Journal of
Function Spaces


Abstract and
Applied Analysis


Advances in
Mathematical Physics