

An n -Dimensional Chaotic System Generation Method Using Parametric Pascal Matrix

Yinxing Zhang, Zhongyun Hua[✉], Member, IEEE, Han Bao[✉], Member, IEEE, Hejiao Huang[✉], and Yicong Zhou[✉], Senior Member, IEEE

Abstract—When high-dimensional chaotic systems are applied to many practical applications, they are required to have robust and complex hyperchaotic behaviors. In this article, we propose a novel n D chaotic system construction method using the Pascal-matrix theory. First, a parametric Pascal matrix is constructed. Then, an n D chaotic system can be generated by using the parametric Pascal matrix as the parameter matrix of the system. Theoretical analysis shows that the generated n D chaotic systems have robust and complex chaotic behaviors, and they become n D Arnold Cat maps by fixing the parameters as some special values. Performance evaluations demonstrate that the n D chaotic systems have more complex chaotic behaviors and better distribution of outputs compared with existing HD chaotic systems. A 4-D Arnold Cat map and a 4-D chaotic map with hyperchaotic behaviors are generated as two examples. The two chaotic maps are then simulated on a microcontroller-based hardware platform and the chaotic sequences are tested to show good randomness.

Index Terms—Arnold cat map, chaos complexity, chaotic system, hardware implementation, hyperchaotic sequence, nonlinear system.

I. INTRODUCTION

CHAOS is an intrinsic stochastic process in a deterministic dynamical system, and it is an evolution from an ordered state to a disordered state. The earliest discovery of chaotic behavior is in weather forecasting [1], and then a variety of chaotic

Manuscript received 11 July 2021; revised 8 December 2021 and 14 January 2022; accepted 27 January 2022. Date of publication 16 February 2022; date of current version 30 September 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 62071142, in part by the Guangdong Basic and Applied Basic Research Foundation under Grant 2021A1515011406, in part by the Shenzhen College Stability Support Plan under Grant GXWD20201230155427003-20200824210638001, and in part by the Research Committee at University of Macau under Grant MYRG2018-00136-FST. Paper no. TII-21-2918. (Corresponding author: Zhongyun Hua.)

Yinxing Zhang, Zhongyun Hua, and Hejiao Huang are with the School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, Shenzhen 518055, China (e-mail: yxzheng23@163.com; huazyum@gmail.com; hhuang@aliyun.com).

Han Bao is with the School of Microelectronics and Control Engineering, Changzhou University, Changzhou 213164, China (e-mail: charles-bao0319@gmail.com).

Yicong Zhou is with the Department of Computer and Information Science, University of Macau, Macau 999078, China (e-mail: yicongzhou@umac.mo).

Color versions of one or more figures in this article are available at <https://doi.org/10.1109/TII.2022.3151984>.

Digital Object Identifier 10.1109/TII.2022.3151984

phenomena are observed in natural and nonnatural systems, such as brain waves and traffic flow [2], [3]. Chaos system has initial state sensitivity, topological transitivity, unpredictability, ergodicity, and so on [1], [4]. These properties contribute to its wide applications in many fields, such as Internet of Things [5], secure communication [6], and intellectual property protection [7]. In addition, since some chaotic systems have very unique character and thus can be applied to some special applications. For example, the Chebyshev chaotic map has a semigroup property and it can be applied to build secure and efficient authentication schemes [8], [9].

In the past few decades, some classic low-dimensional (LD) chaotic systems have been proposed. For example, Hénon map is proposed as a simplified model of the Poincaré section of the Lorenz model and Logistic map is designed as a discrete-time demographic growth model analogous [10]. However, these maps have simple structures, few control parameters, and only one or two positive Lyapunov exponents (LEs). These properties make their trajectories predictive using some artificial intelligence techniques. These latest technologies are used to identify the control parameters [11], predict the states of the chaotic system [12], and estimate the initial states [13]. If the trajectories of a chaotic map can be successfully predicated, many of its applications (e.g., the chaos-based cryptography) will be ineffective [14]. Therefore, the complexity of chaotic sequences attracts the attention of scholars because of its importance in predicting chaotic sequences. For analyzing the complexity of chaotic sequences, a lot of algorithms have been introduced, including sample entropy [15], correlation dimension (CD) [16], and information entropy. Subsequently, to improve the complexity of LD chaotic systems, two kinds of methods are proposed. One is to perturb the variables or control parameters of chaos system [17], and the other is to design new chaotic systems with better complexity [18]. Although these methods can overcome the problems of LD chaotic systems lacking hyperchaotic behavior and discontinuous parameter intervals [19], [20], it is difficult for LD chaotic systems to exhibit stable performance [21], and the performance of the newly designed chaotic map lacks theoretical guarantee and depends on the experience [19].

Recently, many efforts have been devoted to designing high-dimensional (HD) chaotic systems with complex chaotic behaviors and abundant control parameters. Some representative works are as follows. Natiq *et al.* [22] used the sine function to generate n -dimensional (n D) discrete chaotic maps by treating existing chaotic maps as seed maps. Wu *et al.* [23] cascaded

many isomorphic chaotic maps to design an HD chaotic map. In [24], a 6-D discrete-time chaotic system with positive LEs is constructed by modular operation. Since the feedback controller strategy can provide a good solution for constructing chaotic systems, many n D discrete and continuous chaotic systems were developed using different settings of feedback controller [25]–[27]. Analysis results demonstrate that these chaotic systems can achieve many positive LEs and complex chaotic behaviors. However, the performance of the constructed chaotic systems using these strategies is highly dependent on the experience of debugging parameters. Besides, when a chaotic system is simulated in a platform with finite precision, the truncation of precision may cause the overlap of close trajectories, which makes the chaotic behavior become periodic behavior. Thus, developing new chaotic systems with complex chaotic behaviors is meaningful.

In this article, a new n D chaotic system (n D-CS) generation method is proposed using parameterized Pascal matrix. It has a simple structure and can generate n D discrete chaotic maps with robust chaotic behaviors. It first constructs n discrete linear equations and then constructs the coefficient parameter matrix using the Pascal-matrix theory. Specifically, the coefficient parameter is constructed by parameterizing the first row and first column of the Pascal matrix and different settings of parameter can result in different n D chaotic systems. Theoretical analysis proves that the generated n D chaotic systems have robust and complex chaotic behaviors. In particular, an n D Arnold Cat map can be generated by setting the parameters as some specific values. Performance evaluations and comparisons display the n D-CS and n D Arnold Cat map generated by the proposed method can achieve complex hyperchaotic behaviors and more uniformly-distributed outputs than other HD chaotic systems. A 4-D Arnold Cat map and a 4-D chaotic map with hyperchaotic behaviors are generated as two examples to verify the effectiveness of the proposed method. Moreover, to show the simple hardware simulation, the two chaotic maps are simulated on a microcontroller-based hardware platform. The experimental results demonstrate that the two chaotic maps can produce hyperchaotic sequences with high randomness.

The rest of this article is organized as follows. Section II presents the n D chaotic system generation method using the Pascal-matrix theory, and introduces the generation of n D Arnold Cat map using the proposed method. Section III evaluates the properties of the proposed chaotic maps and compares them with other HD chaotic maps. Section IV shows two examples of n D chaotic maps generated by our method. Finally, Section V concludes this article.

II. *n*-D CHAOTIC SYSTEM GENERATION METHOD

This section introduces the Pascal matrix [28], constructs a parametric Pascal matrix, and then generates n D chaotic systems using the parametric Pascal matrix.

A. Pascal Matrix

First, we give the definition of Pascal's triangle (also called the Yang-Hui triangle) as Definition 1.

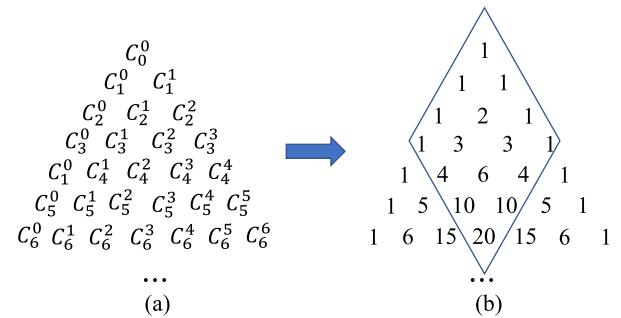


Fig. 1. Flowchart of the Pascal-matrix construction. (a) The coefficients of binomial expansions. (b) Pascal's triangle.

Definition 1: A triangle whose each element is the sum of its two adjacent elements in the preceding row is called the Pascal's triangle.

Using Pascal's triangle, a Pascal matrix can be generated and its generation process is shown in Fig. 1. As can be seen, Pascal's triangle is first constructed by expanding the binomial $(x + y)^n$. The values on both sides of the triangle are set to one and each of the other elements is the sum of its two adjacent elements in the preceding row. For example, $C_2^1 = C_1^0 + C_1^1$. Fig. 1(a) shows the calculation process of each element and Fig. 1(b) shows the obtained Pascal's triangle. Then, the Pascal matrix can be generated by specifying the matrix dimension in Pascal's triangle. For example, when the dimension $n = 4$, the 4-D Pascal matrix P can be generated by marking a 4-D diamond in Pascal's triangle and it is written as

$$P = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 2 & 3 & 4 \\ 1 & 3 & 6 & 10 \\ 1 & 4 & 10 & 20 \end{pmatrix}. \quad (1)$$

The Pascal matrix generated from Pascal's triangle has fixed elements. To increase the parameter space, we further derive a parametric Pascal matrix.

B. Methodology of Configuring Parametric Pascal Matrix P

The construction procedure of the parametric Pascal matrix P can be described as follows.

Step 1: Set the elements $p_{1,2}, p_{1,3}, \dots, p_{1,n}$ in the first row of the Pascal matrix P as some random positive integers and the element $p_{1,1}$ greater than or equal to $\max\{p_{1,j}\}$, where $j = 2, 3, \dots, n$.

Step 2: Set all the elements in the first column of the Pascal matrix P as some random positive integers.

Step 3: All the other elements are calculated by $p_{i,j} = p_{i-1,j} + p_{i,j-1}$ for $i, j = 2, 3, \dots, n$.

Algorithm 1 presents the pseudocode of abovementioned procedures for generating P using inputs p_r and p_c , where p_r and p_c are the elements of the first row and first column, respectively. The element $p_{1,1}$ is set to be greater than or equal to the maximum value of the first row p_r , ensuring that the matrix P is nonsingular. An n D chaotic system with complex

Algorithm 1: Algorithm for configuring the parametric Pascal matrix \mathbf{P} .

Input: $\mathbf{p}_r = \{p_{1,j}\}_{j=2}^n$ and $\mathbf{p}_c = \{p_{i,1}\}_{i=2}^n$ are pseudorandom positive integers.

Output: an n -D parameter matrices \mathbf{P} for n D-CS.

- 1: Initialize $\mathbf{P} \in \mathbb{R}^{n \times n}$;
- 2: $p_{1,1} \geq \max\{\mathbf{p}_r\}$;
- 3: $p_{1,(2:n)} = \mathbf{p}_r$;
- 4: $p_{(2:n),1} = \mathbf{p}_c$;
- 5: **for** $i = 2$ to n **do**
- 6: **for** $j = 2$ to n **do**
- 7: $\mathbf{P}(i,j) = p_{i-1,j} + p_{i,j-1}$;
- 8: **end for**
- 9: **end for**

chaotic behaviors can be generated using the parametric Pascal matrix \mathbf{P} as its parameter matrix.

C. Generation of n D Chaotic System

A general form of discrete map can be described as

$$\mathbf{v}(t+1) = \mathbf{P} \cdot \mathbf{v}(t) \mod N \quad (2)$$

where $\mathbf{v}(t) = \{v_1(t), v_2(t), \dots, v_n(t)\}^T \in \mathbb{R}^{n \times 1}$ denotes the state vector at observation time t , the function mod N is modular arithmetic to restrict the domain into the square, and \mathbf{P} is a parametric Pascal matrix generated by Algorithm 1. It is the parameter matrix of the map and it can be shown as

$$\mathbf{P} = \begin{pmatrix} p_{11} & p_{12} & \cdots & p_{1n} \\ p_{21} & p_{22} & \cdots & p_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ p_{n1} & p_{n2} & \cdots & p_{nn} \end{pmatrix}. \quad (3)$$

Obviously, the parameter matrix \mathbf{P} determines the properties of n D chaotic system. By configuring the elements of \mathbf{P} using some special values, the n D chaotic system can exhibit robust and expected chaotic behaviors.

D. Chaotic Behavior Analysis

The “chaos” is an observed phenomenon and there is no universal definition to describe its existence. To define chaos, researchers gave different definitions from different aspects. Since the LE measures the exponential divergence of a system’s two trajectories starting from extremely close initials, the system exponentially diverges and is sensitive to the initials with a positive LE [29]. The essence of chaotic behavior is that a system’s behavior stretches and folds in a bounded phase plane. Thus, according to the statements in [26] and [30], the properties of one positive LE and global boundedness are usually used as a guideline for chaos generation. The chaos in the sense of LE can be defined as Definition 2.

Definition 2: [26], [30] A discrete system is said to be chaotic in the sense of LE if it satisfies the two conditions: 1) it has at least one positive LE and 2) its phase space region is globally bounded.

The detailed equations of the n D chaotic system (2) can be described as follows:

$$\begin{cases} v_1 = p_{11}v_1(t) + p_{12}v_2(t) + \cdots + p_{1n}v_n(t) \\ v_2 = p_{21}v_1(t) + p_{22}v_2(t) + \cdots + p_{2n}v_n(t) \\ \vdots \\ v_n = p_{n1}v_1(t) + p_{n2}v_2(t) + \cdots + p_{nn}v_n(t) \end{cases}.$$

It is the Jacobian matrix $\mathbf{J}(\mathbf{v}(t))$ to the observed state $\mathbf{v}(t)$ can be expressed by

$$\mathbf{J}(\mathbf{v}(t)) = \begin{pmatrix} p_{11} & p_{12} & \cdots & p_{1n} \\ p_{21} & p_{22} & \cdots & p_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ p_{n1} & p_{n2} & \cdots & p_{nn} \end{pmatrix} \quad (4)$$

which is the parametric Pascal matrix \mathbf{P} in (3). Thus, the Jacobian matrix of the n D chaotic system (2) is independent of the observation state $\mathbf{v}(t)$.

The following Proposition 1 is introduced to illustrate that the proposed n D chaotic system is chaotic.

Proposition 1: The n D chaotic system in (2) has chaotic behavior if its n D parameter matrix \mathbf{P} is the parametric Pascal matrix configured by Algorithm 1.

Proof: The phase space of the n D chaotic system in (2) always stretches in a bounded region due to the modulo operation. Thus, the n D chaotic system is globally bounded and satisfies condition 2) of Definition 2.

According to the LE calculation [31], the LEs of the n D chaotic system can be expressed as

$$LE_i = \lim_{t \rightarrow \infty} \frac{1}{t} \sum_{k=0}^{t-1} \ln(\lambda_i^{\mathbf{v}(t)}) \quad (5)$$

where λ_i is the i th eigenvalue of the Jacobian matrix $\mathbf{J}(\mathbf{v}(t))$ of the system and $i \in \{1, 2, \dots, n\}$. The Jacobian matrix of the n D chaotic system (2) is independent of the observation state $\mathbf{v}(t)$, and it is the parametric Pascal matrix \mathbf{P} . Then, the n LEs of the n D chaotic system (2) is calculated as

$$LE_i = \lim_{t \rightarrow \infty} \frac{1}{t} \sum_{k=0}^{t-1} \ln(\lambda_i(\mathbf{P})) = \ln(\lambda_i(\mathbf{P})), \quad i = 1, 2, \dots, n. \quad (6)$$

For any square matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$, the sum of its eigenvalues is equal to the trace $tr(\mathbf{P})$ of the matrix. Then, the sum of the eigenvalues of the parametric Pascal matrix \mathbf{P} can be calculated as

$$\begin{aligned} \lambda_1 + \lambda_2 + \cdots + \lambda_n &= tr(\mathbf{P}) \\ &= p_{1,1} + p_{2,2} + \cdots + p_{n,n} \end{aligned} \quad (7)$$

where λ_i denotes the eigenvalues of \mathbf{P} and $i = 1, 2, \dots, n$.

According to the procedure of the parametric Pascal matrix in Algorithm 1, one can obtain that

$$\begin{aligned} p_{i,j} &= p_{i-1,j} + p_{i,j-1} \\ &= p_{i-1,j-1} + p_{i-2,j} + p_{i-1,j-1} + p_{i,j-2} \\ &= 2p_{i-1,j-1} + p_{i-2,j} + p_{i,j-2} \end{aligned}$$

where $i, j = 3, 4, \dots, n$. Because all the elements $p_{a,b}$ ($a, b = 1, 2, \dots, n$) are positive integers, $p_{i,i} > p_{i-1,i-1}$ when $i \geq 3$. Therefore, the trace of parametric Pascal matrix satisfies that $\text{tr}(\mathbf{P}) > n$. Combining the (7), one gets

$$\lambda_1 + \lambda_2 + \dots + \lambda_n > n.$$

This implies that at least one eigenvalue of the parametric Pascal matrix \mathbf{P} is greater than one. Consequently, according to the LE calculation in (6), one can get that at least one LE is positive. Then, the *nD* chaotic system satisfies condition 1) of Definition 2. This completes the proof. ■

E. Special Case of *nD* Chaotic System

The Arnold Cat map is a simple discrete system and has many unique properties, such as area preserving, determinant of one, integer entries of inverse matrix [32]. By setting the parameters of the *nD* chaotic system (2) as some special values, an *nD* Arnold Cat map can be generated.

When the elements of input \mathbf{p}_r and \mathbf{p}_c in Algorithm 1 are all one and set $p_{1,1} = 1$, the original Pascal matrix is generated. Using the original Pascal matrix as the parameter matrix in (2), an *nD* discrete Arnold Cat map can be generated. Since the original Pascal matrix is a symmetric matrix and its determinant is equal to one, it has the following properties.

Lemma 1: [28] Suppose $0 < \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$ are the *n* eigenvalues of the original Pascal matrix $\mathbf{P} = (p_{i,j})_{n \times n}$. They satisfy

$$\lambda_1 \lambda_2 \dots \lambda_n = 1 \quad (8)$$

and

$$\lambda_i = \frac{1}{\lambda_{n+1-i}} \quad (9)$$

where $i = 1, 2, \dots, n$.

Lemma 1 tells that the $\lfloor \frac{n}{2} \rfloor$ eigenvalues of an *nD* original Pascal matrix are larger than one. Since the eigenvalues of the original Pascal matrix can determine the LEs of the chaotic map, one can generate an *nD* Arnold Cat map with robust hyperchaotic behaviors using the original Pascal matrix as its parameter matrix.

Proposition 2: The *n* LEs of the *nD* Arnold Cat map satisfy that

$$LE_1 + LE_2 + \dots + LE_n = 0 \quad (10)$$

and

$$LE_i = -LE_{n+1-i} \quad (11)$$

where $i = 1, 2, \dots, n$.

Proof: Let $0 < \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$ be the *n* eigenvalues of the original Pascal matrix \mathbf{P} . According to (6), one can get that

$$\begin{aligned} LE_1 + LE_2 + \dots + LE_n &= \ln(\lambda_1) + \ln(\lambda_2) + \dots + \ln(\lambda_n) \\ &= \ln(\lambda_1 \lambda_2 \dots \lambda_n). \end{aligned}$$

Combining (8), one can obtain

$$LE_1 + LE_2 + \dots + LE_n = \ln(1) = 0.$$

Moreover, according to (9), the relationship of the *n* LEs of the Arnold Cat map can be derived as

$$\begin{aligned} LE_i &= \ln(\lambda_i) \\ &= \ln\left(\frac{1}{\lambda_{n+1-i}}\right) \\ &= -LE_{n+1-i} \end{aligned}$$

where $i = 1, 2, \dots, n$. This completes the proof. ■

Lemma 2: [28] The determinant of the inverse original Pascal matrix \mathbf{P}^{-1} is one, and \mathbf{P}^{-1} has the same eigenvalues as the original Pascal matrix \mathbf{P} .

Proposition 3: The chaotic map whose parameter matrix is the inverse original Pascal matrix \mathbf{P}^{-1} is also an *nD* Arnold Cat map.

Proof: Because the matrix \mathbf{P}^{-1} satisfies $\det(\mathbf{P}^{-1}) = 1$, the chaotic system in (2) is an *nD* Arnold Cat map when using it as the parameter matrix.

The newly generated *nD* Arnold Cat map also satisfies Proposition 2, because the parameter matrix \mathbf{P}^{-1} has the same eigenvalues as the original Pascal matrix \mathbf{P} . The *nD* Arnold Cat maps generated using original Pascal matrix \mathbf{P} and inverse Pascal matrix \mathbf{P}^{-1} are two completely different chaotic maps with different parameters. Using the same initial settings, they can generate different chaotic sequences with different chaotic behaviors. However, since the eigenvalues of the \mathbf{P} and \mathbf{P}^{-1} are the same, the *nD* Arnold Cat maps generated using \mathbf{P} and \mathbf{P}^{-1} have the same LEs.

III. PERFORMANCE ANALYSIS

This section investigates the performance of the proposed *nD* Arnold Cat map and *nD* chaotic system by using the measurements of the largest LE (lgtLE), CD [16], and information entropy. In addition, we also compare it with three existing HD-chaotic system generation methods including Chen's method [26], Shen's method [25], and Natiq's method [22].

To compare the performance of different chaotic map generation methods, our experiments generate many chaotic maps with different dimensions as examples for each generation method. However, for some generation methods in the pieces of literature, the authors provided only parameter settings for generating chaotic maps with a certain dimension. To obtain chaotic maps with various dimensions for each generation method and provide a fair comparison, our experiments set the parameters of all generation methods as the following rules.

- 1) If the parameter settings of generating chaotic maps with different dimensions are available in the literature, we directly use these parameter settings in our experiments.
- 2) If the parameter settings for generating chaotic maps with some dimensions are not provided in the literature, we set the parameters to the values to ensure that the generated chaotic maps achieve the best properties expected in the original literature.
- 3) Under the condition that the generated *nD* chaotic maps achieve the expected properties, we set the parameters to the same level with that in our *nD*-CS. Using the

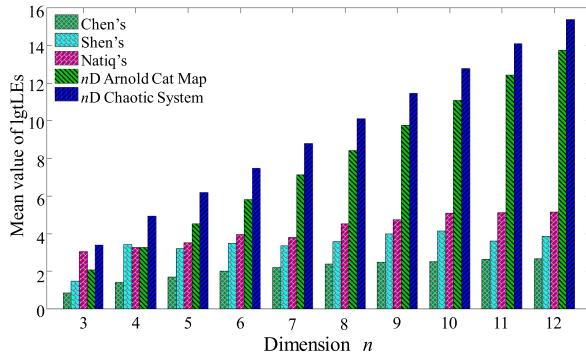


Fig. 2. LgtLEs for n D chaotic systems generated by the different generation methods with dimension $n \in \{3, 4, \dots, 12\}$.

abovementioned rules, each method can achieve their best performance expected in literature.

Consequently, the parameters of each HD-chaotic system generation method are set as follows. For the proposed n D chaotic system, when configuring the parameter matrix \mathbf{A} in Algorithm 1, all the elements in the sequences \mathbf{p}_r and \mathbf{p}_c are randomly obtained from the set $\{1, 2, \dots, 10\}$. When generating the n D Arnold Cat map, all the elements in the sequences \mathbf{p}_r and \mathbf{p}_c of Algorithm 1 are set to one. For existing generation methods, the control parameters in Shen's method are set to $a = -0.1$ and $\epsilon \in [1, 10]$, in Chen's method are set to $\epsilon \in [4, 10]$ and $\sigma = 0.01$, and in Natiq's method are set to $\beta = 5$, $\sigma = 1.5\pi$ and $\mu \in [1, 10]$. When an n D chaotic system is generated by each method, the control parameters of each method are randomly selected from their own set intervals.

A. Largest LE

According to the discussions in [27], the lgtLE is an important indicator for the complexity of a dynamical system. This section analyzes the proposed chaotic systems using lgtLE.

An experiment is designed to calculate the largest lgtLE value of the chaotic maps produced by different chaotic system generation methods. For each of the dimension $n \in \{3, 4, \dots, 12\}$, one n D Arnold Cat map and 100 n D chaotic maps are generated by different generation methods, and their lgtLEs are calculated. Fig. 2 plots the mean lgtLE value of the n D chaotic systems produced by each generation method. Note that the mean values are calculated from 100 n D chaotic maps. It shows that the n D Arnold Cat map and n D chaotic map have larger lgtLE values than the chaotic systems generated by other methods as the increase of dimension n . These imply that the proposed method can generate n D Arnold Cat map and n D chaotic map with complex chaotic behaviors. Besides, the n D chaotic map can achieve larger lgtLEs than the n D Arnold Cat map. Since their LEs are determined by the eigenvalues of parameter matrix \mathbf{P} , according to (6), their lgtLEs may be quite large for the specific setting of \mathbf{P} .

B. Correlation Dimension

The CD measures the space dimensionality that can be occupied by a data sequence [16]. A dynamical system with a

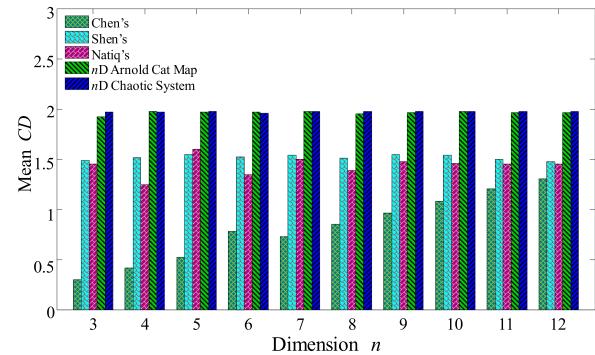


Fig. 3. Mean CDs for n D chaotic systems generated by different generation methods with dimension $n \in \{3, 4, \dots, 12\}$.

positive CD indicates that it has strange attractors. A larger CD means that the time series of the dynamical system can occupy a higher space dimensionality, which further indicates that the higher strangeness of the attractors.

Our experiments apply the nonlinear time-series analysis tool TISEAN 3.0.1¹ to obtain the CDs for different HD chaotic systems. The experiment is set as the same as the LE experiment for different dimension $n \in \{3, 4, \dots, 12\}$. For each generation method, randomly generate 100 n D chaotic maps with initial states $\mathbf{v}(0) \in [0, 1]$ and the CD of each n D chaotic map is the mean value of the n CDs of its n time series. Fig. 3 plots the mean CDs of these chaotic maps with different dimensions generated by different methods. One can see that with the increase of dimension n , the n D chaotic maps and n D Arnold Cat maps produced by our method have similar CDs. Besides, the chaotic maps generated by the proposed generation method have larger mean CDs than the chaotic systems generated by other methods. These verify that the chaotic maps produced by our method own higher complicated attractors.

C. Information Entropy

Information entropy is usually used to describe the randomness of a time series. The information entropy of a chaotic sequence can be calculated to show the randomness of the sequence. An n D chaotic map has n D phase space, which is composed of n D chaotic sequences. When uniformly dividing the chaotic sequence range of each dimension into M intervals, a total number of M^n subphase spaces can be obtained from the n D phase space. The information entropy of an n D chaotic sequence can explicitly be written as

$$H = - \sum_{k=1}^{M^n} P(k) \log_2 P(k) \quad (12)$$

where $P(k)$ is the probability of the states that are located into the k th subphase space. It is easy to see that information entropy is a nonnegative quantity and its theoretical maximum value is obtained when $P(1) = P(2) = \dots = P(M^n)$. That means when $P(k) = 1/M^n$ ($k = 1, 2, \dots, M^n$), the maximum information entropy value $H_{\max} = n \log_2 M$ can be obtained. A

¹[Online]. Available: https://www.pks.mpg.de/~tisean/archive_3.0.0.html

TABLE I
MAXIMUM INFORMATION ENTROPY VALUES FOR 100 3-D CHAOTIC MAPS GENERATED BY DIFFERENT CHAOTIC MAP GENERATION METHODS AGAINST DIFFERENT NUMBERS OF INTERVALS M

	Number of intervals M										
	3	5	7	9	11	13	15	17	19	21	23
Chen's [26]	1.9298	2.8900	4.0894	4.7157	5.3747	5.8642	6.3422	6.8186	7.2338	7.5774	7.9205
Shen's [25]	3.0150	4.8409	6.2126	7.5283	8.1390	8.6232	9.2662	9.7383	10.1235	10.6099	11.0335
Natiq's [22]	4.2557	6.1966	7.5028	8.4982	9.2842	9.9468	10.5263	10.9913	11.3942	11.7566	12.0726
n D-Arnold Cat map	4.6213	6.1554	7.1257	7.8616	8.4483	8.9350	9.3500	9.7157	10.0327	10.3242	10.5897
n D-Chaotic system	4.6347	6.8571	8.3330	9.4353	10.3182	11.0488	11.6741	12.2210	12.7068	13.1436	13.5402
H_{\max}	4.7549	6.9658	8.4221	9.5098	10.3783	11.1013	11.7207	12.2624	12.7438	13.1770	13.5707

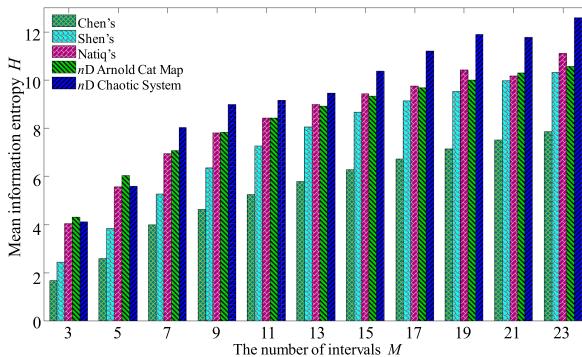


Fig. 4. Information entropy values for 100 3-D chaotic maps generated by each chaotic map generation method against different numbers of intervals $M \in \{3, 5, \dots, 23\}$.

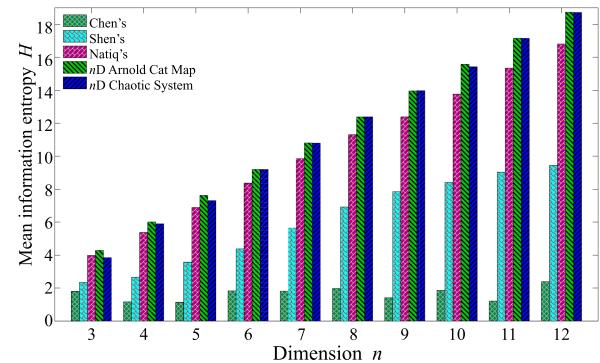


Fig. 5. Mean information entropy values for 100 n D chaotic maps produced by each chaotic map generation method by setting the number of intervals $M = 3$ against different dimensions $n \in \{3, 4, \dots, 12\}$.

larger information entropy indicates better uniform-distribution of the chaotic sequences, and further implies that the related chaotic map has higher complexity.

Two groups of experiments are performed to calculate the information entropy value of chaotic sequences. The first group of experiments calculate the information entropy values via the number of intervals $M = \{3, 5, \dots, 23\}$ by setting the dimension $n = 3$. We set the initial state $\mathbf{v}(0) \in [0, 1]$ and the length of chaotic sequence is M^4 , then the information entropy of each 3-D chaotic map can be calculated. Fig. 4 plots the mean information entropy values of the 100 3-D chaotic maps randomly generated by the different methods. As can be seen, the 3-D chaotic map generated by our method has the largest information entropy. Table I displays the maximum information entropy values of 100 3-D chaotic maps. It is obvious that the 3-D chaotic maps generated by our method own the largest information entropy values, and these maximum information entropy values approach to the theoretical maximum values.

The other group of experiments is designed to calculate the information entropy values via the dimension $n = \{3, 4, \dots, 12\}$ by setting the number of intervals $M = 3$. In these experiments, 100 n D chaotic maps are randomly generated by each generation method with different n . The initial states are set to $\mathbf{v}(0) \in [0, 1]$ and the length of chaotic sequence is set to 3^{n+1} , then the information entropy value of each chaotic map can be calculated. Shen's, Natiq's, and our methods can achieve larger information entropies with a larger n . This is because their outputs are distributed similarly for different dimensions n . However, for the

TABLE II
MAXIMUM INFORMATION ENTROPY VALUES OF THE n D CHAOTIC MAPS GENERATED BY DIFFERENT METHODS WITH DIMENSION $n = \{3, 4, \dots, 12\}$ BY SETTING $M = 3$

n	n D chaotic map Generation Methods					H_{\max}
	Chen's [26]	Shen's [25]	Natiq's [22]	n D-Arnold Cat map	n D-Chaotic system	
3	1.9527	2.9538	4.2349	4.5909	4.6639	4.7549
4	1.5219	3.6447	5.7438	6.1542	6.2043	6.3399
5	1.2903	4.8572	7.1637	7.7074	7.7222	7.9248
6	1.9695	5.5778	8.6708	9.2799	9.2875	9.5098
7	2.3336	6.5493	10.1369	10.8524	10.8610	11.0947
8	2.2242	7.5835	11.6325	12.4272	12.4291	12.6797
9	1.5376	8.7604	13.1037	14.0072	14.0072	14.2647
10	1.9509	9.9357	14.5675	15.5911	15.5895	15.8496
11	3.1147	11.3763	16.0269	17.1737	17.1735	17.4346
12	2.5046	12.5194	17.4851	18.7575	18.7575	19.0196

Chen's method, its outputs are distributed more concentratedly when the dimension n increases. This leads to that its information entropy does not increase with the increase of n . Fig. 5 shows the mean information entropy values of these 100 n D chaotic maps produced by different methods. It displays that the n D Arnold Cat map and n D chaotic maps generated by the proposed method can obtain higher mean information entropy values than the n D chaotic maps generated by other methods. Table II lists the maximum information values of the 100 n D chaotic maps. It can be seen that the maximum information entropy values of

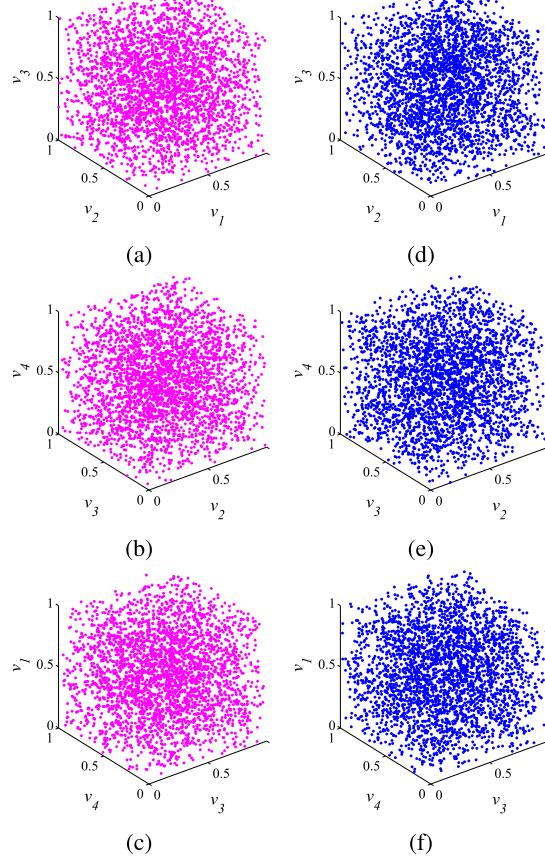


Fig. 6. Trajectories of different maps with 3000 states distributed in $v_1 - v_2 - v_3$ space, $v_2 - v_3 - v_4$ space, and $v_3 - v_4 - v_1$ space. (a)–(c) 4-D Arnold Cat map's trajectories. (d)–(f) 4-D chaotic map's trajectories.

our method are larger than that of other methods, and are very close to the theoretical maximum values. These show that our method can generate HD Arnold Cat maps and chaotic maps with uniform-distribution chaotic sequences.

IV. ILLUSTRATIVE EXAMPLES

To show the effect of the proposed n D chaotic system generation method, this section provides two examples: a 4-D Arnold Cat map and a 4-D chaotic map. When generating the 4-D Arnold Cat map, all the elements in the first column p_c and row p_r of Algorithm 1 are set to one. When generating the 4-D chaotic map, the p_c and p_r are randomly selected from the integer set $\{1, 2, \dots, 10\}$. Moreover, to display the easy hardware implementation, we simulate the 4-D Arnold Cat map and 4-D chaotic map on a microcontroller-based platform. Finally, we test the randomness of the chaotic sequences of the two chaotic maps.

A. 4-D Arnold Cat Map

When generating a 4-D Arnold Cat map, the first row and first column to construct the parameter matrix are $p_r = \{1, 1, 1\}$, $p_c = \{1, 1, 1\}$, and $p_{1,1} = 1$. Then, the parameter matrix of 4-D Arnold Cat map can be generated by running Algorithm 1 and

can be obtained as

$$\mathbf{P} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 2 & 3 & 4 \\ 1 & 3 & 6 & 10 \\ 1 & 4 & 10 & 20 \end{pmatrix}. \quad (13)$$

When setting the modular coefficient $N = 1$, a 4-D Arnold Cat map can be obtained as

$$\begin{pmatrix} v_1(t+1) \\ v_2(t+1) \\ v_3(t+1) \\ v_4(t+1) \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 2 & 3 & 4 \\ 1 & 3 & 6 & 10 \\ 1 & 4 & 10 & 20 \end{pmatrix} \times \begin{pmatrix} v_1(t) \\ v_2(t) \\ v_3(t) \\ v_4(t) \end{pmatrix} \bmod 1. \quad (14)$$

According to Lemma 2, one can get the inverse of matrix of the \mathbf{P} as

$$\mathbf{P}^{-1} = \begin{pmatrix} 4 & -6 & 4 & -1 \\ -6 & 14 & -11 & 3 \\ 4 & -11 & 10 & -3 \\ -1 & 3 & -3 & 1 \end{pmatrix}. \quad (15)$$

Using \mathbf{P}^{-1} as the parameter matrix in (14), we can also construct a 4-D Arnold Cat map. From Lemma 2, we know that the matrices \mathbf{P} and \mathbf{P}^{-1} have the same eigenvalues. Then, the LEs of the two 4-D Arnold Cat maps constructed by \mathbf{P} and \mathbf{P}^{-1} are the same. The LEs of the 4-D Arnold Cat map can be calculated from its parameter matrix. This is because the Jacobian matrix is independent of the observation state $\mathbf{v}(t)$. Then, its LEs are determined only by the Jacobian matrix, namely the parameter matrix \mathbf{P} . The four eigenvalues of Jacobian matrix \mathbf{P} are $\lambda_1 = 0.0380$, $\lambda_2 = 0.4538$, $\lambda_3 = 2.2034$, and $\lambda_4 = 26.3047$. Using the LE calculation method in (6), the four LEs of the two 4-D Arnold Cat map are $LE_1 = \ln(\lambda_1) = -3.2697$, $LE_2 = \ln(\lambda_2) = -0.7900$, $LE_3 = \ln(\lambda_3) = 0.7900$ and $LE_4 = \ln(\lambda_4) = 3.2697$. This indicates that the generated 4-D Arnold Cat map has two positive LEs and thus has hyperchaotic behavior.

To display the randomness of the 4-D Arnold Cat map, we plot its trajectory with the initial value $\mathbf{v}(0) = 0.1_{4 \times 1}$ in the different 3-D phase spaces. As can be seen from Fig. 6(a)–(c), the trajectories of the 4-D Arnold Cat map are randomly distributed in the 3-D phase spaces and this indicates the high randomness of the 4-D Arnold Cat map's outputs.

B. 4-D Chaotic Map

To generate a 4-D chaotic map, a 4-D parametric Pascal matrix should be generated. Suppose the first row and first column to construct the parameter Pascal matrix are $p_r = \{5, 7, 8\}$, $p_c = \{3, 7, 7\}$, and $p_{1,1} = 8$. According to Algorithm 1, the 4-D parametric matrix is generated as

$$\mathbf{P} = \begin{pmatrix} 8 & 5 & 7 & 8 \\ 3 & 8 & 15 & 23 \\ 7 & 15 & 30 & 53 \\ 7 & 22 & 52 & 105 \end{pmatrix}. \quad (16)$$

When setting the modular coefficient $N = 1$ in (2), a 4-D chaotic map is generated as

$$\begin{pmatrix} v_1(t+1) \\ v_2(t+1) \\ v_3(t+1) \\ v_4(t+1) \end{pmatrix} = \begin{pmatrix} 8 & 5 & 7 & 8 \\ 3 & 8 & 15 & 23 \\ 7 & 15 & 30 & 53 \\ 7 & 22 & 52 & 105 \end{pmatrix} \times \begin{pmatrix} v_1(t) \\ v_2(t) \\ v_3(t) \\ v_4(t) \end{pmatrix} \bmod 1. \quad (17)$$

Using the LE calculation method in (6), the four LEs of the 4-D chaotic map are $LE_1 = -1.5041$, $LE_2 = 1.1150$, $LE_3 = 2.2766$, and $LE_4 = 4.9304$. Obviously, three of the four LEs are positive, indicating that the chaotic map has hyperchaotic behavior.

To show the trajectory distribution of the 4-D chaotic map, we generate the trajectory of the 4-D chaotic map using the initial state $\mathbf{v}(0) = 0.1_{4 \times 1}$. Fig. 6 shows the trajectory of the 4-D chaotic map in different 3-D phase spaces. One can see that the trajectories of the 4-D chaotic map are fully distributed on the whole phase space, which indicates that the outputs generated by the 4-D chaotic map have good randomness.

C. Hardware Implementation

When chaotic systems are used in applications, a necessary step is to implement them on a hardware platform. To show the simple hardware implementation of the 4-D Arnold Cat map and the 4-D chaotic map, this section simulates them on a microcontroller-based platform.

The microcontroller has many good features including high integration, large storage, and strong environmental adaptability, it thus can be widely used in various industrial fields. In this experimental simulation, a digital hardware platform based on STM32F407VET6 (ARM 32-bit Cortex-M4 CPU) is constructed to run the proposed chaotic maps. The hardware circuit mainly includes the microcontroller STM32, a D/A converter TLV5618 with 12-bit, an oscilloscope TDS3054 C, and some other peripheral circuits. The mathematical model of 4-D Arnold Cat map and 4-D chaotic map, and their initial states are coded by using C language, and then download this program to the microcontroller. The generated chaotic digital signals can be converted into analog voltage signals by the D/A converter and the voltage signals will be displayed on the oscilloscope.

Here, we simulate the proposed 4-D Arnold Cat map and 4-D chaotic map in MATLAB software and microcontroller-based platform. The hardware devices used in the microcontroller-based experiment are illustrated in Fig. 7, and the initial states for 4-D Arnold Cat map and 4-D chaotic map are set to $\mathbf{v}(0) = 0.1_{4 \times 1}$. Fig. 8 plots the outputs of 4-D Arnold Cat map and 4-D chaotic map captured from the MATLAB and oscilloscope, which denote the software and hardware simulations, respectively. The first row of Fig. 8 shows MATLAB simulation results, and the second row shows microcontroller results. As can be seen, the outputs of the proposed 4-D Arnold Cat map and 4-D chaotic map are consistent in software and hardware platform. One of the bottlenecks in the application of chaos is that the trajectories of a chaotic system with the same initial values on different platforms are different. From the simulation results, this problem can be addressed. Moreover, one can see

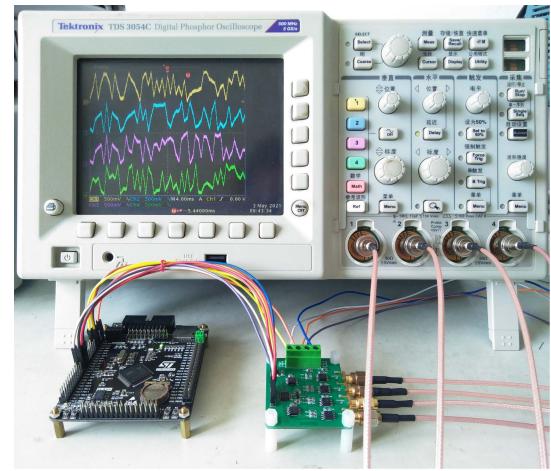


Fig. 7. Hardware devices of microcontroller-based experiment.

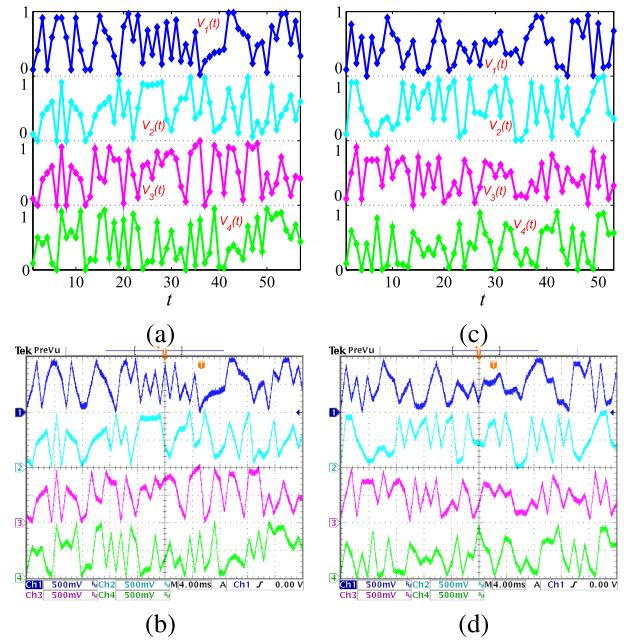


Fig. 8. Simulation results captured from software and hardware platform for (a)–(b) 4-D Arnold Cat map and (c)–(d) 4-D chaotic map. The chaotic sequences from top to bottom in the left figures (a) and (b) are the outputs of v_1, v_2, v_3, v_4 of 4-D Arnold Cat map, while the right figures (c) and (d) are the outputs of v_1, v_2, v_3, v_4 of 4-D chaotic map.

that each sequence v_i in Fig. 8 randomly oscillates within a fixed range. This means the feasibility and correctness of the hardware implementation of the 4-D Arnold Cat map and 4-D chaotic map.

D. Pseudorandom Number Generator

Since chaotic systems have unpredictability and initial value sensitivity, they can be used to design pseudorandom number generators (PRNGs). The randomness of pseudorandom numbers (PRNs) is highly dependent on the dynamic performance of the chaotic system when a chaotic system is used as a PRNG [33]. The 4-D Arnold Cat map and 4-D chaotic map shown in Section IV exhibit complicated dynamic behaviors,

they, thus, can show good performance when used as the PRNG. Here, we take these two chaotic maps as PRNGs and they are called 4DPCM-PRNG and 4DCM-PRNG, respectively.

A 4-D chaotic map can generate four chaotic signals. The generated chaotic signals are floating numbers in the digital platform. For each chaotic signal, it is first converted into binary numbers. Without loss of generality, the first chaotic signal is selected to generate PRNs. Assume that the first chaotic signal is $V = \{v_1, v_2, \dots, v_i, \dots\}$, and then the PRNs can be generated by

$$R = [v_i \times I] \quad (18)$$

where $I = 2^j$ ($j \in \mathbb{N}^+$) is a positive number, function $\lfloor x \rfloor$ denotes the largest integer not greater than x . When setting $j = 8$, it is easy to get that each v_i can produce eight binary numbers.

The National Institute of Standards and Technology (NIST) SP800-22 is used to measure the randomness of generated PRNs. It is a convinced test suite and contains 15 subtests, which aims to check the nonrandomness area of PRNs from various aspects. A significance level α is used to define the confidence interval in the test suite, and the number of binary sequences should be no less than the significance level α^{-1} . According to the recommendation in [34], the significance level α , binary sequence length, and the number of binary sequences are set to 0.01, 10^6 , and 128, respectively.

In our experiments, the initial states for 4-D Arnold Cat map and 4-D chaotic map are set to $\mathbf{v}(0) = \mathbf{0.1}_{4 \times 1}$, and 128 binary sequences with length 10^6 are generated using the PRNG and tested using the NIST SP800-22. Each subtest in the NIST SP800-22 outputs a pass rate and p -value. The p -value T measures the distribution of the 128 p -values. According to the discussions in [34], the pass rate and p -value T greater than 0.9609 and 0.0001 are considered to pass related subtest. The test results are listed in Table III. As can be seen, the pass rates and p -value T 's produced by different subtests are all larger than 0.9609 and 0.0001, respectively. This means that the presented 4-D Arnold Cat map and 4-D chaotic map have complex chaotic behavior and can generate PRNs with high randomness.

The advantages of our proposed method for PRNG are as follows. 1) According to the performance evaluation in Section III, our n D chaotic maps possess complex dynamics and can produce more uniform-distribution sequences. These are beneficial for generating PRNs [33]. 2) Each of our n D chaotic maps simultaneously produces n chaotic signals. Each signal can be used for generating PRNs. The disadvantages are summarized as follows. 1) The designed PRNG only needs one chaotic signal, while our n D chaotic maps produce many chaotic signals. This may lead to extra computational cost. 2) Each output state may contain many bits (may be longer than 32). However, the PRNs use only eight bits and discard the rest bits. This may reduce efficiency.

In addition, we discuss the advantages and disadvantages of our PRNG compared with several representative PRNGs. Many PRNGs were designed using linear theoretical models, such as linear congruential generator [35] and linear feedback shift register [36]. These linear-model-based PRNGs are simple

TABLE III
p-VALUES OF PRNs PRODUCED BY PRNG USING THE 4-D ARNOLD CAT MAP AND 4-D CHAOTIC MAP IN NIST SP800-22

No.	Sub-tests	4DPCM-PRNG		4DCM-PRNG	
		Pass rate p -value T			
1	Frequency	0.9766	0.2873	0.9922	0.0781
2	Block Frequency	1.0000	0.0602	1.0000	0.6545
3	Cum. Sums. (F)	0.9766	0.5009	0.9922	0.0282
3	Cum. Sums. (B)	0.9766	0.6025	1.0000	0.8195
4	Runs	1.0000	0.9114	1.0000	0.2041
5	Longest Run	0.9844	0.4686	1.0000	0.2430
6	Rank	0.9922	0.3115	0.9922	0.7231
7	FFT	1.0000	0.1952	1.0000	0.1626
8	Non-Ovla. Temp.*	0.9889	0.4143	0.9897	0.4066
9	Ovla. Temp.	0.9922	0.7231	1.0000	0.1951
10	Universal	0.9766	0.9573	0.9922	0.4071
11	Appr. Entropy	1.0000	0.4846	1.0000	0.9938
12	Ran. Exc.*	0.9928	0.4132	0.9935	0.3465
13	Ran. Exc. Var.*	0.9968	0.3193	0.9885	0.4033
14	Serial (1st)	0.9688	0.1952	1.0000	0.0372
14	Serial (2nd)	0.9844	0.1165	1.0000	0.7565
15	Linear Complexity	0.9922	0.1866	0.9844	0.9320
	Success No.	15/15	15/15	15/15	15/15

¹Note: The symbol * denotes the mean of multiple test results.

and have low computational cost. However, their generated PRNs often show relatively short periods and fail to pass the NIST test due to their linear structures [35], [37]. Recently, artificial-intelligence-based PRNGs have been constructed. For example, generative-adversarial-networks-based PRNGs [38] and recurrent-neural-networks-based PRNGs [39] show high randomness but they have significantly high computation cost and are time consuming during the training process. As a typical nonlinear system, chaotic systems have been widely used to construct PRNGs. Many chaos-based PRNGs have been developed and they have high randomness and can pass the NIST test [40], [41]. However, they may show weakness in cryptography [42]. This is because most of these chaotic systems are LD, may have periodic windows, or/and easily occur dynamic degradation in digital platforms.

Our PRNG is designed using the proposed HD chaotic systems. As discussed in Section III, our proposed HD chaotic systems have robust chaos and show better performance than existing HD systems. Since the performance of a chaos-based PRNG highly relies on the used chaotic system [33], the PRNs generated by our PRNG can achieve high randomness. The maximum pass rate in the NIST test can reach 100%, which can be seen from Table III. However, our PRNG may lead to a relatively slow generation speed if the dimension n is set to a large value.

V. CONCLUSION

In this article, we presented a new n -D chaotic system construction method based on the Pascal matrix. It was a simple construction method and can generate n D chaotic systems with robust chaotic behaviors. Theoretical analysis showed that the

generated chaotic system always has at least one positive LE. Setting the elements in the parameter matrix as some special values, an *n*D Arnold Cat map can be generated. Performance analysis demonstrated that the HD chaotic maps generated by our method had more complex dynamics properties and better uniform-distribution outputs than other HD chaotic maps. To show the effect of the proposed method in generating *n*D chaotic systems, a 4-D Arnold Cat map and a 4-D chaotic map with hyperchaotic behaviors were generated as two examples. The two chaotic maps were simulated on both software and hardware platforms. Comparing with the simulation results, it was found that the chaotic trajectories on different platforms were consistent. Finally, the two presented chaotic maps were tested using NIST SP800-22 and the results showed the high randomness. With complex hyperchaotic behaviors, the newly generated HD hyperchaotic maps can be applied to many chaos-based applications, such as Internet of Things [5], secure communication [6], and intellectual property protection [7]. Our future works will investigate these applications of the proposed chaotic system.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions that greatly contribute to improving the quality of this article.

REFERENCES

- [1] R. L. Devaney, *An Introduction to Chaotic Dynamical Systems*, 2nd ed. Boulder, CO, USA: Westview Press, 2003.
- [2] K. Li, H. Bao, H. Li, J. Ma, Z. Hua, and B. Bao, "Memristive Rulkov neuron model with magnetic induction effects," *IEEE Trans. Ind. Informat.*, vol. 18, no. 3, pp. 1726–1736, Mar. 2021.
- [3] D. Chen, "Research on traffic flow prediction in the big data environment based on the improved RBF neural network," *IEEE Trans. Ind. Informat.*, vol. 13, no. 4, pp. 2000–2008, Aug. 2017.
- [4] C. Li, K. Tan, B. Feng, and J. Lü, "The graph structure of the generalized discrete Arnold's cat map," *IEEE Trans. Comput.*, vol. 71, no. 2, pp. 364–377, Feb. 2022.
- [5] K. Muhammad, R. Hamza, J. Ahmad, J. Lloret, H. Wang, and S. W. Baik, "Secure surveillance framework for IoT systems using probabilistic image encryption," *IEEE Trans. Ind. Informat.*, vol. 14, no. 8, pp. 3679–3689, Aug. 2018.
- [6] L. Zhang, Z. Chen, W. Rao, and Z. Wu, "Efficient and secure non-coherent OFDM-based overlapped chaotic chip position shift keying system: Design and performance analysis," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 67, no. 1, pp. 309–321, Jan. 2020.
- [7] N. Lin, X. Chen, H. Lu, and X. Li, "Chaotic weights: A novel approach to protect intellectual property of deep neural networks," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 40, no. 7, pp. 1327–1339, Jul. 2021.
- [8] D. Abbasinezhad-Mood, A. Ostad-Sharif, S. M. Mazinani, and M. Nikooghadam, "Provably secure escrow-less Chebyshev chaotic map-based key agreement protocol for vehicle to grid connections with privacy protection," *IEEE Trans. Ind. Informat.*, vol. 16, no. 12, pp. 7287–7294, Dec. 2020.
- [9] H.-Y. Lin, "Dynamic ID authentication scheme using chaotic map," *Wireless Netw.*, vol. 24, no. 3, pp. 769–776, 2018.
- [10] S. Vaidyanathan and C. Volos, "Advances and applications in chaotic systems," in *Studies in Computational Intelligence*, vol. 636. Berlin, Germany: Springer, 2016.
- [11] Y. Xu, L. Mili, M. Korkali, and X. Chen, "An adaptive Bayesian parameter estimation of a synchronous generator under gross errors," *IEEE Trans. Ind. Informat.*, vol. 16, no. 8, pp. 5088–5098, Aug. 2020.
- [12] L. Lin, M. Shen, H. C. So, and C. Chang, "Convergence analysis for initial condition estimation in coupled map lattice systems," *IEEE Trans. Signal Process.*, vol. 60, no. 8, pp. 4426–4432, Aug. 2012.
- [13] M. Liu, S. Zhang, Z. Fan, S. Zheng, and W. Sheng, "Exponential H_∞ synchronization and state estimation for chaotic systems via a unified model," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 7, pp. 1114–1126, Jul. 2013.
- [14] S. Ergün, "On the security of chaos based true random number generators," *IEICE Trans. Fundam. Electron., Commun. Comput. Sci.*, vol. 99, no. 1, pp. 363–369, 2016.
- [15] J. S. Richman and J. R. Moorman, "Physiological time-series analysis using approximate entropy and sample entropy," *Amer J. Physiol.-Heart Circulatory Physiol.*, vol. 278, no. 6, pp. 2039–2049, 2000.
- [16] L. Lacasa and J. Gómez-Gardenes, "Correlation dimension of complex networks," *Phys. Rev. Lett.*, vol. 110, no. 16, 2013, Art. no. 168703.
- [17] L. Liu, J. Lin, S. Miao, and B. Liu, "A double perturbation method for reducing dynamical degradation of the digital baker map," *Int. J. Bifurcation Chaos*, vol. 27, no. 7, 2017, Art. no. 1750103.
- [18] Z. Hua, Y. Chen, H. Bao, and Y. Zhou, "Two-dimensional parametric polynomial chaotic system," *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published, doi: 10.1109/TSMC.2021.3096967.
- [19] S. Panahi, J. C. Sprott, and S. Jafari, "Two simplest quadratic chaotic maps without equilibrium," *Int. J. Bifurcation Chaos*, vol. 28, no. 12, 2018, Art. no. 1850144.
- [20] Z. Hua, Z. Zhu, Y. Chen, and Y. Li, "Color image encryption using orthogonal Latin squares and a new 2D chaotic system," *Nonlinear Dyn.*, vol. 104, no. 4, pp. 4505–4522, 2021.
- [21] S. Li, G. Chen, and X. Mou, "On the dynamical degradation of digital piecewise linear chaotic maps," *Int. J. Bifurcation Chaos*, vol. 15, no. 10, pp. 3119–3151, 2005.
- [22] H. Natiq, S. Banerjee, S. He, M. R. M. Said, and A. Kilicman, "Designing an *M*-dimensional nonlinear model for producing hyperchaos," *Chaos Solitons Fractals*, vol. 114, pp. 506–515, 2018.
- [23] Q. Wu, F. Zhang, Q. Hong, X. Wang, and Z. Zeng, "Research on cascading high-dimensional isomorphic chaotic maps," *Cogn. Neurodyn.*, vol. 15, no. 1, pp. 157–167, 2021.
- [24] C. Wang, C. Fan, K. Feng, X. Huang, and Q. Ding, "Analysis of the time series generated by a new high-dimensional discrete chaotic system," *Complexity*, vol. 2018, pp. 1–11, 2018.
- [25] C. Shen, S. Yu, J. Lü, and G. Chen, "Designing hyperchaotic systems with any desired number of positive Lyapunov exponents via a simple model," *IEEE Trans. Circuits Syst. I*, vol. 61, no. 8, pp. 2380–2389, Aug. 2014.
- [26] S. Chen, S. Yu, J. Lü, G. Chen, and J. He, "Design and FPGA-based realization of a chaotic secure video communication system," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 9, pp. 2359–2371, Sep. 2018.
- [27] C. Shen, S. Yu, J. Lu, and G. Chen, "Constructing hyperchaotic systems at will," *Int. J. Circuit Theory Appl.*, vol. 43, no. 12, pp. 2039–2056, 2016.
- [28] D. Edelman and G. Strang, "Pascal matrices," *Amer. Math. Monthly*, vol. 111, no. 3, pp. 189–197, 2004.
- [29] A. E. Motter, "Relativistic chaos is coordinate invariant," *Phys. Rev. Lett.*, vol. 91, no. 23, 2003, Art. no. 231101.
- [30] G. Chen and X. Wang, *Chaotification of Dynamical Systems: Theory, Method and Applications*. Shanghai, China: Shanghai Jiao Tong Univ. Press, 2006.
- [31] A. Wolf, J. B. Swift, H. L. Swinney, and J. A. Vastano, "Determining Lyapunov exponents from a time series," *Physica D: Nonlinear Phenomena*, vol. 16, no. 3, pp. 285–317, 1985.
- [32] C. Li, K. Tan, B. Feng, and J. Lu, "The graph structure of the generalized discrete Arnold's cat map," *IEEE Trans. Comput.*, vol. 71, no. 2, pp. 364–377, Feb. 2022.
- [33] S.-L. Chen, T. Hwang, and W.-W. Lin, "Randomness enhancement using digitalized modified logistic map," *IEEE Trans. Circuits Syst. II, Express Briefs*, vol. 57, no. 12, pp. 996–1000, Dec. 2010.
- [34] A. Rukhin *et al.*, "A statistical test suite for random and pseudorandom number generators for cryptographic applications," *Nat. Inst. Standards Technol.*, Gaithersburg, MD, USA, Special Publication 800-22 Revision 1a, Apr. 2010.
- [35] R. S. Katti, R. G. Kavasseri, and V. Sai, "Pseudorandom bit generation using coupled congruential generators," *IEEE Trans. Circuits Syst. II, Express Briefs*, vol. 57, no. 3, pp. 203–207, Mar. 2010.
- [36] E. Pasalic, "On guess and determine cryptanalysis of LFSR-based stream ciphers," *IEEE Trans. Inf. Theory*, vol. 55, no. 7, pp. 3398–3406, Jul. 2009.
- [37] T. Addabbo, M. Alioto, A. Fort, A. Pasini, S. Rocchi, and V. Vignoli, "A class of maximum-period nonlinear congruential generators derived from the Rényi chaotic map," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 54, no. 4, pp. 816–828, Apr. 2007.

- [38] M. De Bernardi, M. Khouzani, and P. Malacaria, "Pseudo-random number generation using generative adversarial networks," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discov. Databases*. Berlin, Germany: Springer, 2018, pp. 191–200.
- [39] Y.-S. Jeong, K. Oh, C.-K. Cho, and H.-J. Choi, "Pseudo random number generation using LSTMs and irrational numbers," in *Proc. IEEE Int. Conf. Big Data Smart Comput.*, 2018, pp. 541–544.
- [40] M. Garcia-Bosque, A. Pérez-Resa, C. Sánchez-Azqueta, C. Aldea, and S. Celma, "Chaos-based bitwise dynamical pseudorandom number generator on FPGA," *IEEE Trans. Instrum. Meas.*, vol. 68, no. 1, pp. 291–293, Jan. 2019.
- [41] C. E. C. Souza, D. P. B. Chaves, and C. Pimentel, "One-dimensional pseudo-chaotic sequences based on the discrete Arnold's cat map over \mathbb{Z}_{3^m} ," *IEEE Trans. Circuits Syst. II, Express Briefs*, vol. 68, no. 1, pp. 491–495, Jan. 2021.
- [42] A. Kumar Panda and K. Chandra Ray, "A coupled variable input LCG method and its VLSI architecture for pseudorandom bit generation," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 4, pp. 1011–1019, Apr. 2020.



Yinxing Zhang received the M.S. degree in fundamental mathematics from the Guilin University of Electronic Technology, Guilin, China, in 2019. He is currently working toward the Ph.D. degree in Computer Science and Technology from the Harbin Institute of Technology, Shenzhen, China.

His research interests include chaotic system and nonlinear system control.



Zhongyun Hua (Member, IEEE) received the B.S. degree from Chongqing University, Chongqing, China, in 2011, and the M.S. and Ph.D. degrees from the University of Macau, Macau, China, in 2013 and 2016, respectively, all in software engineering. He is currently an Associate Professor with the School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China. He has authored or coauthored more than 50 papers in the area of his research interests, receiving more than 2800 citations. His research interests include chaotic system, chaos-based applications, and multimedia security.

Prof. Hua is currently an Associate Editor for the *International Journal of Bifurcation and Chaos*.



Han Bao (Member, IEEE) received the B.S. degree in landscape design from the Jiangxi University of Finance and Economics, Nanchang, China, in 2015, the M.S. degree in art and design from Changzhou University, Changzhou, China, in 2018, and the Ph.D. degree in nonlinear system analysis and measurement technology from the Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2021.

In 2019, he visited the Department of Computer Science, The University of Auckland, New Zealand. He is currently a Lecturer with the School of Microelectronics and Control Engineering, Changzhou University, Changzhou, China. His research interests include memristive neuromorphic circuit, nonlinear circuits and systems, and artificial intelligence.



Hejiao Huang received the B.S. and M.S. degrees in mathematics from Shaanxi Normal University, Xi'an, China, in 1996 and 1999, respectively, and the Ph. D. degree in computer science from the City University of Hong Kong, Hong Kong, in 2004.

She was an Invited Professor with INRIA, Bordeaux, France. She is currently a Professor with the Harbin Institute of Technology, Shenzhen, China. Her research interests include cloud computing, network security, trustworthy computing, and formal methods for system design and wireless networks.



Yicong Zhou (Senior Member, IEEE) received the B.S. degree from Hunan University, Changsha, China, in 1992, and the M.S. and Ph.D. degrees from Tufts University, Medford, MA, USA, in 2008 and 2010, respectively, all in electrical engineering.

He is currently a Professor with the Department of Computer and Information Science, University of Macau, Macau, China. His research interests include image processing, computer vision, machine learning, and multi-media security.

Dr. Zhou was a recipient of the Third Price of Macau Natural Science Award as a sole winner in 2020 and a corecipient in 2014. Since 2015, he has been a leading Co-Chair of Technical Committee on Cognitive Computing in the IEEE Systems, Man, and Cybernetics Society. He is currently an Associate Editor for the *IEEE TRANSACTIONS ON CYBERNETICS*, *IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS*, *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY*, and *IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING*. He is a Fellow of SPIE (the Society of Photo-Optical Instrumentation Engineers) and was recognized as one of "Highly Cited Researchers" in 2020 and 2021.