entropy MDPI

Article

Novel Low-Power Construction of Chaotic S-Box in Multilayer Perceptron

Runtao Ren [1,2,3,*,†], Jinqi Su [2,*,†], Ban Yang [4], Raymond Y. K. Lau [3] and Qilei Liu [2]

[1] School of Modern Post, Xi'an University of Posts and Telecommunications, Xi'an 710061, China
[2] School of Management and Economics, Xi'an University of Posts and Telecommunications, Xi'an 710061, China
[3] Department of Information Systems, City University of Hong Kong, Kowloon Tong, Hong Kong, China
[4] School of Cyberspace Security, Xi'an University of Posts and Telecommunications, Xi'an 710121, China
* Correspondence: runtaoren@gmail.com (R.R.); sujinqi@xupt.edu.cn (J.S.)
† These authors contributed equally to this work.

Abstract: Multilayer perceptron is composed of massive distributed neural processors interconnected. The nonlinear dynamic components in these processors expand the input data into a linear combination of synapses. However, the nonlinear mapping ability of original multilayer perceptron is limited when processing high complexity information. The introduction of more powerful nonlinear components (e.g., S-box) to multilayer perceptron can not only reinforce its information processing ability, but also enhance the overall security. Therefore, we combine the methods of cryptography and information theory to design a low-power chaotic S-box (LPC S-box) with entropy coding in the hidden layer to make the multilayer perceptron process information more efficiently and safely. In the performance test, our S-box architecture has good properties, which can effectively resist main known attacks (e.g., Berlekamp Massey-attack and Ronjom–Helleseth attack). This interdisciplinary work can attract more attention from academia and industry to the security of multilayer perceptron.

Keywords: S-box; multilayer perceptron; information theory; cyber security

Citation: Ren, R.; Su, J.; Yang, B.; Lau, R.Y.K.; Liu, Q. Novel Low-Power Construction of Chaotic S-Box in Multilayer Perceptron. *Entropy* **2022**, *24*, 1552. https://doi.org/10.3390/e24111552

Academic Editor: Luis Javier Garcia Villalba

Received: 30 September 2022
Accepted: 26 October 2022
Published: 28 October 2022

1. Introduction

Multilayer perceptron (MLP) is a multilayer feedforward network model with one-way propagation [1–3]. Because of its high nonlinear mapping ability, MLP is one of the most basic network models in neural network research. From the perspective of information processing, MLP is an abstract simulation of biological neural networks to establish a simple biological neuron model. Therefore, the basic structure of MLP is based on the logic of biological neuron model. The most typical MLP includes three layers: input layer, hidden layer, and output layer (as shown in Figure 1). In the hidden layer, each node is equivalent to a perceptron, and each node represents a specific output function, which is called activation function [4]. The connection between each two nodes represents a weighted value for the signal passing through the connection, which is called the weight. The output of MLP will be different due to the difference of weight value and excitation function, and its powerful fitting ability can be used to solve more complex problems. MLP itself is usually the approximation of some algorithm or function in nature, and it may also be the expression of a logical strategy. With the gradual deepening of the research on MLP, it has great research potential in both theoretical research and application. At present, MLP has been applied in many commercial and industrial fields and has brought varying degrees of productivity improvement (e.g., pattern recognition, function approximation, and optimal prediction) [5–7].

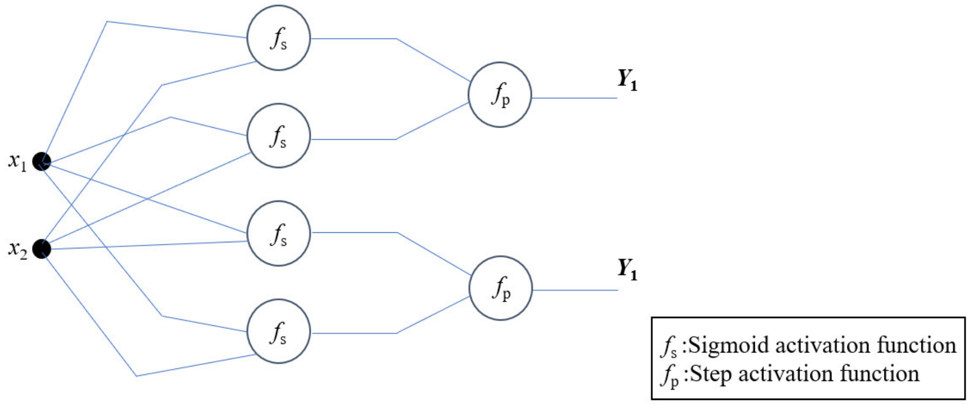
x_1
x_2
f_s
f_s
f_s
f_s
f_p
f_p
Y_1
Y_1
f_s :Sigmoid activation function
f_p :Step activation function

Figure 1. Structure of the LPC S-box in multilayer perceptron.

The interior of multilayer perceptron is a highly nonlinear information processing system composed of multilayer single perceptron interconnection [8]. When neurons are in two different states of activation or inhibition, it can be called a nonlinear relationship. The network composed of neurons with a threshold has better performance, which can improve fault tolerance and storage capacity. Each neuron of the multilayer perceptron receives the input of a large number of other neurons, and generates the output through the parallel network, affecting other neurons. This mutual restriction and interaction between the networks realizes the nonlinear mapping from the input state to the output state space [9,10]. However, the overall performance of multilayer perceptron is not the superposition of the performance of local neurons, and its nonlinear mapping ability is limited. Therefore, when processing information with higher complexity, we can introduce stronger nonlinear components (e.g., S-box) to make up for this defect. As a nonlinear component of multilayer perceptron, S-box is also an important part of symmetric cipher (e.g., block cipher) [11–13].

The higher dimension multilayer perceptron has very complex nonlinear dynamic behaviors, which contain various functions (e.g., activation function and step function). As Piotr et al. remarked [14], securely enhance multilayer perceptron must satisfy certain conditions, the higher the dimension of S-box, the more statistical analysis it can be applied to multilayer perceptron, which is complex for algorithm designers and malicious password analysts. Yet, the carefully designed S-box is just a strong nonlinear Boolean (vector) map with avalanche characteristics. This means, in fact, that the S-box can be considered as a black box that transforms any input vector into a balanced vector in a non-predictable way (i.e., nonlinear). Figure 1 shows the architecture of different component combinations to achieve balance.

At present the methods of S-boxes applied on multilayer perception for improving security complexity are based on two parallel methodologies [15,16]. The first one mainly uses mathematical theories and statistical investigations, while the other is additionally supported by the practitioner’s experience. To second a link between the both methodologies we propose design a structure of S-box based on multilayer perceptron to process the data from the overall point of view, this method can enhance its information processing ability and improve the overall security.

Accordingly, for the complex nonlinear architecture of multilayer perceptron, if the computational cost of data itself (e.g., compressed data) is reduced, its nonlinear mapping ability can also be improved. Modern lossless data compression program is realized through the combination of general compression technology and entropy coding. The role of lossless data compression algorithm is to increase the randomness of data (i.e., increase entropy). On the other hand, the entropy coding method compresses an original data unit into the minimum code as the compressed data. Although each of these technologies can achieve the compression effect, the combination will produce a better compression ratio.

Then, they maximize the entropy of the compressed data (such as arithmetic coding and Hoffman coding). Therefore, in the hidden layer, compression technology and entropy coding can be combined to increase the throughput of the multilayer perceptron, reduce redundant information, and reduce computational overhead.

Based on the above factors, we combine the methods of cryptography and information theory to introduce S-box into multilayer perceptron with Levenshtein entropy coding-based in the hidden layer to make the multilayer perceptron process information more efficiently and safely.

In previous studies, there is a scheme to apply S-box to multilayer perceptron. Arrañaga et al. proposed a scheme that S-box can apply it to multilayer perceptron [17]. But it does not test the cryptographic characteristics of the implemented S-box, nor is it applied to the replacement of image processing.

Zhu et al. [18] proposed an improved chaotic map for image encryption system. The framework preliminarily analyzes the conventional technologies of one-dimensional chaotic system, replacement box production structure and image encryption algorithm. The double chaotic S-box algorithm includes forward backward confusion diffusion operation to enhance the performance of image encryption system. The shortcomings of the proposed model do not discuss the side channel attack in the encryption framework, nor is it applied to multilayer perceptron.

A.S. et al. [19] involved an S-box with hybrid prediction and adaptive chaos, and calculated and analyzed various performance parameters. However, it is based on embedded system rather than multilayer perceptron, which makes image processing less obvious.

Yang et al. [20] designed a new chaotic S-box diffusion method based on 2d-mccm, which improved the security and efficiency, and proposed a new image encryption algorithm, but it was not constructed based on multilayer perceptron.

Zhang et al. [21] introduced the learning algorithm based on multilayer perceptron and S-box to improve the security integrity of the system. The proposal uses the new way of approach to decompose the huge input into two equal parts and each part is trained by two perceptrons combined with a special 172P perceptron. It also convenient to quickly train the weight-threshold values of the Boolean function in the network through DNA-like learning algorithm.

Kotlarz and Kotulski [22] discussed the use of S-boxes to implement cryptographic schemes in multilayer perceptron. To realize the elementary permutation, 2-bit and 3-bit block of bits were transformed into a block of bits as a combination of small blocks (i.e., multilayer perceptron). The permutation of 16-bit blocks is realized in this model. The advantage of this method is that it makes use of the fragmentary training sets for each block, at the server side, and the complete training set for the whole multilayer perception in order to realize the cryptographic algorithm. This fragmentary training system gives some security for the algorithm update process provided the internal structure of the complete topology remains secret.

For the above research, their research scheme not only failed to study the S-box through Boolean function, resulting in the inability to analyze the cryptographic properties, but also failed to combine the S-box and multilayer perceptron into image processing.

For the above research, their research scheme not only failed to study the S-box through Boolean function, resulting in the inability to analyze the cryptographic properties, but also failed to combine the S-box and multilayer perceptron into image processing. We give a low-power chaotic S-box based on multilayer perceptron to optimize, and apply it to image processing to improve its information processing ability in the multilayer perceptron. The application of LPC S-box in multilayer perceptron is shown in Figure 2. The contributions of this article are as follows:

(1) We combine the methods of cryptography and information theory to design a low-power chaotic S-box, and carry out noise reduction and entropy coding compression

in the hidden layer, so that the multilayer perceptron can process images more efficiently and safely.

(2) Our S-box has the function of replacement, which aims to confuse the binary string after encoding and compression, prevent the computer from being invaded illegally and maliciously decode the image, so as to encode the image that is difficult to be processed by the computer and has low processing efficiency efficiently and safely.

(3) We selected a group of S-boxes with good cryptographic performance for performance testing. The results show that our scheme can effectively resist algebraic attacks, DPA attacks, etc., which can not only make up for the limited nonlinear mapping ability of multilayer perceptron, but also improve the security of multilayer perceptron model.

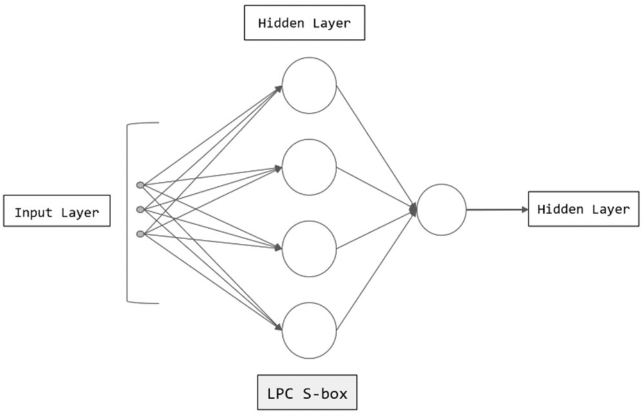

Figure 2. Structure of the LPC S-box in multilayer perceptron.

2. Preliminaries

2.1. Boolean Function

Let n and m be two positive integers, and the vector space $F_2^n \to F_2^m$ mapping is called (n, m) function (i.e., multi output Boolean function or vector Boolean function), where n is the multi-input and m is the multi-output. When m = 1, it can be called a single output Boolean function. When m = n, we call this multi output Boolean function S-box.

2.2. Algebraic Degree

Let F(x) be a (n, m) function, then the algebraic degree of F(x) is defined as:

$$deg\ F = min\{deg\ (v \cdot F)|0 \neq v \in F_2^m\}$$

where $v \cdot F$ is called the component function, and the algebraic degree of the multi output Boolean function is the minimum value of the non-zero linear combination of all its component functions.

2.3. Algebraic Immunity

The algebraic immunity of n-ary Boolean function F is expressed by AI (F) and is defined as:

$$AI(F) = min\ \{deg\ g\ |0 \neq Ann(F) \text{or}\ Ann(1 + F)\}$$

where $Ann(F) = \{g\ |g \epsilon B_n, Fg = 0\}$, G is called the annihilator of Boolean function F.

2.4. Differential Uniformity

Let F(x) be a (n, m) function, then the differential uniformity of F(x) is:

$$\delta_F = \max_{0 \neq \alpha \epsilon F_2^n} \max_{\beta \epsilon F_2^m} |\{x \epsilon F_2^n | F(x + \alpha) - F(x) = \beta\}|$$

The difference uniformity of $F(x)$ satisfies $2^{n-m} \leq \delta_F \leq 2^n$, if it δ_F is the maximum value 2^n, F(x) is an affine function, α and $\beta \in F_2^n$, n is the multi-input, m is the multi-output. If δ_F is the minimum value 2^{n-m}, it is called a fully nonlinear function.

3. Methodology

In this paper, a lightweight chaotic S-box is constructed based on entropy coded images and embedded in multiple hidden layers of multilayer perceptron. The application of chaotic S-box in multilayer perceptron is shown in the figure. The chaotic S-box based on entropy coded images is an efficient and safe construction scheme. Figure 3 shows the interaction of our scheme.

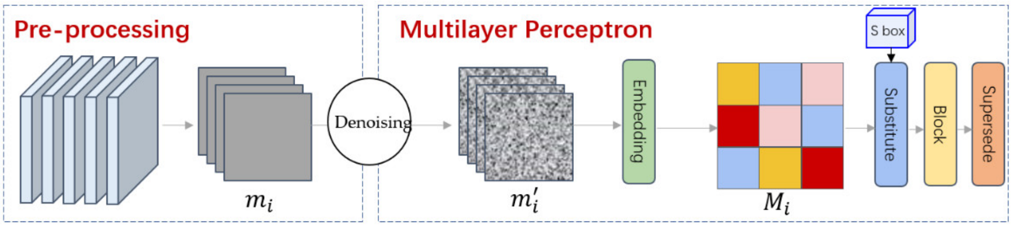

Figure 3. Interaction of our scheme.

Our scheme includes the following algorithms: pre-processing, denoising, embedding, substitute, block, and supersede. The specific definition of each algorithm is as follows:

(1) Pre-processing: first, the original image is processed at the input layer of the multilayer perceptron, and a series of binary sequences are obtained through specific algorithms.
(2) Denoising: in the first hidden layer, the pixels of the input image are filtered to obtain a noise reduction image with redundant information removed.
(3) Embedding: the algorithm performs Levenshtein entropy coding based on each image and completes weighting to obtain a compressed image in the second hidden layer.
(4) Substitute: this algorithm converts the image into a one-dimensional sequence by generating the initial parameters of chaos. The blurred image can be obtained by shifting the image according to the sequence.
(5) Block: grouping the long sequence after replacement.
(6) Supersede: the operation is to input the one-dimensional sequence into the S-box sequentially to obtain a new sequence.

4. Syntax

4.1. Pre-Processing

This algorithm runs by the input layer of deep feed-forward artificial neural network. The deep structure comprises many layers of non-linearly activating nodes. Each neuron-like node is connected from one layer to another. The input of one layer is connected to another layer with different adjustable weights to form a complete neural network. When the original image is input to the deep feed-forward artificial neural network, it will be extracted by equation $W(t) = \sum_{i=1}^{n} m_i \gamma_1 + d$, where $W(t)$ is the activity of the neurons in the input layer at a time t, m_i denotes the input $M \times N$ size image with adjustable weight, γ_1 is the weights among the input and hidden layer, and d represents the bias. The following Algorithm 1 describes the implementation process of Pre-Processing.

Algorithm 1 Pre-processing.

```
Input: $m_i$
Output: $W(t)$
extract features $W(t) = \sum_{i=1}^{n} m_i \gamma_1 + d$
Return the activity of the neurons $W(t)$
```

4.2. Denoising

This algorithm is executed by the first hidden layer. Its purpose is to improve the image contrast and eliminate the unwanted pixels in an image. In this algorithm, the adaptive sigma filter (smoothing filter) places the pixels of the input image in the adaptive kernel (i.e., window) with different sizes. Then, it organizes the neighboring pixels V_n (i.e., $V_{i,j-1}$, $V_{i-1,j}$, $V_{i,j+1}$, $V_{i+1,j}$) in the form of a matrix with 'i' row and 'j' column. In the kernel area, the intensity values of pixels are sorted in ascending order. Finally, the noise pixels are removed from the filter window by the central pixel $V_{i,j}$ to obtain the denoised image m_i'. The following Algorithm 2 describes the implementation process of Denoising.

Algorithm 2 Denoising.

```
Input: $m_i$
Output: $m_i'$
divides the original picture $m_i$ into segments;
//segments are horizontal and vertical;
arranges the pixel in a kernel size $R*R$;
picks the central pixels $V_{i,j}$;
picks the neighboring pixels $V_n$;
measures the deviation $\tau_{ij} = \sum |V_{i,j} - V_n|$;
removes the noisy pixels from the filter window;
Return the denoised image $m_i'$
```

4.3. Embedding

This algorithm (i.e., Levenshtein entropy encoding-based compression) is run by the second hidden layer. Embedding can make pictures lossless compression, reduce storage costs, and improve transmission efficiency. For this algorithm, each image is weighted by $\omega = \rho(m_i')$, where ρ indicates a weight assigned to m_i', the sum of weight value is equal to one (ρ = 1), and the resultant code is termed a complete code. The length of codeword and the weighted path length are determined. Then, the probability and entropy are calculated. Finally, the compressed images are obtained at the second hidden layer by the output of previous layer $B(t)$ and the output of deep learning $O(t)$. The following Algorithm 3 describes the implementation process of Embedding.

Algorithm 3 Embedding.

```
Input: $m_i'$
Output: $M_i$
assigns the weight $\omega = \rho(m_i')$;
sets the length of codeword $C(w) = (\psi_1, \dots, \psi_n)$;
sets the weighted path length $\omega_n = \rho l_c$;
//$l_c$ is the lengths of the code words
calculates the probability of pixels $P = 2^{-l_c}$;
calculates the entropy $H(E) = -\rho\, log_2\, \rho$;
calculates $B(t) = \sum_{i=1}^{n} m_i' a_1 + a_2\, b(t-1)$;
calculates $O(t) = B(t) a_3$;
//$a_1$ is the weight of hidden layers;
```

```
//$a_2$ is a weight between input and hidden layers;
//$a_3$ is an adjustable weight of two layers;
//$b(t-1)$ is the output from first hidden layer;
Return the compressed image $M_i$
```

4.4. Substitute

The algorithm first generates the secret key of 256bit binary number, and then the secret key will be divided into 32 8-bit binary numbers $k_i(i = 1,2 \cdots ,32)$. Next, the system generates the initial parameters (*w*$_0$, *e*, *q*, *x*$_0$, *v*$_1$, *v*$_2$) of chaos, converts the picture into a one-dimensional sequence *D* with the size of *M* × *N*, and then substitutes the chaotic parameters into the formula:$w_{n+1} = 1 - q|ew_n(1 - w_n) - (1/q)|$, where $w_n \in [0,1]$, $e \in [0,2]$, $q \in [0,4]$, *e* and *q* are system parameters. The sequence *T* with the length of *M* × *N* can be obtained by the above formula for eliminating the transient benefits. Then, the substitution operation is carried out. Sequence *T′* is obtained by arranging the values in sequence *T* from small to large. Displacement sequence *T′′* can be obtained according to the position information of the elements of sequence *T′* in sequence *T*. Finally, image $M_i{}'$ can be obtained by displacement of image M_i according to the sequence. The following Algorithm 4 describes the implementation process of Substitute.

Algorithm 4 Substitute.

```
Input: $M_i$
Output: $M_i{}'$
generates the $w_0 = \frac{(k_1 \oplus k_2 \oplus k_3 \oplus k_4 \oplus k_5 \oplus k_6) + \sum_{i=1}^{32} k_i}{2^8} mod(1)$;
generates the $q = \frac{(k_6 \oplus k_7 \oplus k_8 \oplus k_9 \oplus k_{10} \oplus k_{11}) + \sum_{i=1}^{32} k_i}{2^8} mod(1) + 1$;
generates the $e = \frac{(k_{11} \oplus k_{12} \oplus k_{13} \oplus k_{14} \oplus k_{15} \oplus k_{16}) + \sum_{i=1}^{32} k_i}{2^8} mod(1) + 2$;
generates the $x_0 = \frac{(k_{17} \oplus k_{18} \oplus k_{19} \oplus k_{20} \oplus k_{21} \oplus k_{22}) + \sum_{i=1}^{32} k_i}{2^8} mod(1)$;
generates the $v_1 = \frac{(k_{22} \oplus k_{23} \oplus k_{24} \oplus k_{25} \oplus k_{26} \oplus k_{27}) + \sum_{i=1}^{32} k_i}{2^8} mod(1) + 3$;
generates the $v_2 = \frac{(k_{27} \oplus k_{28} \oplus k_{29} \oplus k_{30} \oplus k_{31} \oplus k_{32}) + \sum_{i=1}^{32} k_i}{2^8} mod(1) + 3$;
//$l_c$ is the lengths of the code words
displaces the $M_i$ to one-dimensional sequence $D$;
eliminates transient effects to get sequence $T$;
executes substitution operation;
Return the image $M_i{}'$
```

4.5. Block

The block algorithm is to segment the image to facilitate the next supersede algorithm. The following Algorithm 5 describes the implementation process of Block.

Algorithm 5 Block.

```
Input: $M_i{}'$
Output: $A_i{}'$
divides the $M_i{}'$ to $m \times n$ numbers block with $M/m \times N/n$ size;
converts pixels of each block into one-dimensional sequence $A_i$;
Return the sequence $A_i$
```

4.6. Supersede

The specific process of this S-box operation is to input the one-dimensional sequence A_i into the S-box in turn to obtain the sequence B_i. The following Algorithm 6 describes the implementation process of Supersede.

Algorithm 6 Supersede.

```
Input: $A_i$
Output: $B_i$
run equation $S(B_i) = A_i, (i = 1,2,\cdots, m \times n)$;
Return the sequence $B_i$
```

5. Benchmark Test

In this section, we estimate the index performance of our S-box and other schemes. Experimental environment for performance analysis is as follows: the processor is Intel® Core™ i5-8300H CPU @2.30 GHz; the system type is a 64-bit operating system. Based on this system, this paper uses C programming language to calculate nonlinearity, differential uniformity, and transparency order operations.

In the multilayer perceptron, a series of operations of S-box can be used in the hidden layer. Because the S-box has a high degree of nonlinearity, it is related to the nonlinear mapping of the hidden layer. The nonlinearity, difference uniformity, and transparency order involved in this paper can be applied to the adaptive function of multilayer perceptron, which plays an important role in the optimization of S-box. The relation of attributes of S-box based on multilayer perceptron is shown in Table 1 below.

Table 1. Relation.

Multilayer perceptron	S-box
Adaptability	Reduce manual intervention
Fault tolerance	Nonlinearity
Weight coefficient	Differential uniformity
Threshold coefficient	Transparency order
Iterative training	Multi turn transformation
Ergodicity	Confusion principle
Stability	Reliability
Low-power	Differential power analysis

We classify all the optimal 4-bit S-boxes and generate 16 optimal S-boxes under different nonlinearity and uniformity conditions according to the affine equivalence principle (i.e., optimum in differential, linear, and algebraic attacks). In this paper, a new scheme of S-box with multilayer perceptron based on Levenshtein entropy coding algorithm is proposed, and the performance of various scheme 4-bit S-box is tested.

Ta Thi Kim Hue et al. [23] defined the 4-bit optimal S-box for the first time, that is, the bijective S-box whose nonlinearity and differential uniformity reach the critical value 4 at the same time. We have found the representative elements of eight types of optimal 4-bit S-boxes and used intuitive symbols G_i to represent different schemes. The optimal 4-bit S-box of our S-box is named $G_{0,1,2,3,4,5,6,7}$. The 4-bit S-box proposed by Canteaut et al. [24] named $G_{8,9}$The data of ours 8 optimal 4-bit S-box and Canteaut et al. S-boxes [24] are shown in Table 2.

Table 2. The 4-bit S-box representation.

Scheme	S-box	Representative Element
LPC S-box	S_0	0, 2, 10, 3, 4, 6, 9, 14, 11, 7, 5, 1, 12, 8, 13, 15
	S_1	6, 10, 14, 2, 15, 8, 13, 1, 12, 9, 7, 4, 5, 0, 3, 11
	S_2	2, 1, 6, 12, 4, 10, 15, 7, 3, 5, 13, 11, 9, 8, 14, 0
	S_3	4, 8, 3, 15, 11, 7, 12, 0, 9, 5, 14, 2, 6, 10, 1, 13
	S_4	10, 13, 4, 5, 7, 3, 9, 12, 14, 6, 0, 15, 8, 1, 2, 11
	S_5	3, 5, 8, 2, 13, 4, 12, 6, 7, 0, 9, 10, 15, 11, 1, 14
	S_6	9, 13, 8, 11, 3, 15, 5, 0, 14, 7, 1, 4, 12, 10, 2, 6
	S_7	9, 12, 15, 0, 1, 8, 2, 11, 3, 14, 13, 4, 5, 10, 6, 7
Canteaut et al. [24]	S_8	0, 6, 14, 1, 15, 4, 7, 13, 9, 8, 12, 5, 2, 10, 3, 11
	S_9	0, 9, 13, 2, 15, 1, 11, 7, 6, 4, 5, 3, 8, 12, 10, 14

5.1. Nonlinearity

In order to resist linear cryptographic attacks, Boolean functions used in cryptosystems should be as far away from the Hamming distance of all affine functions as possible [25,26]. The nonlinearity $NL(F)$ of Boolean function F is defined as the minimum Hamming distance between F and all affine functions. The nonlinearity of each scheme is obtained from the data input into the S-box, as in Figure 4.

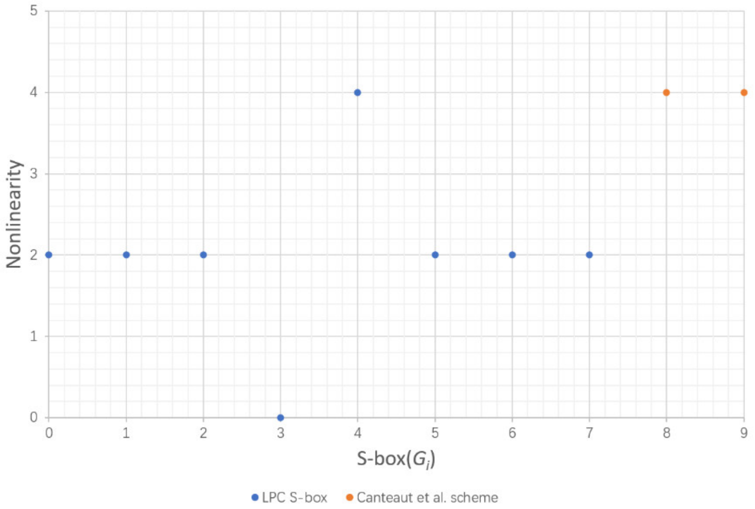

Figure 4. Nonlinearity of each scheme.

The upper bound of the nonlinearity of a n × n S-box is $2^{n-1} - 2^{\frac{n}{2}-1}$, and the upper bound of the nonlinearity of a 4-bit S-box is 6. It can be seen from Figure 4 that the nonlinearity of LPC S-box is up to 4, which is high for 4-bit, because it is difficult to construct an S-box close to the upper bound. The higher the nonlinearity, the closer the nonlinear ability between input and output is to the upper bound, the stronger the corresponding anti-linear attack ability. Hence, LPC S-box has excellent ability to resist linear cryptographic attacks

5.2. Differential Uniformity

The value of differential evenness of Boolean function is inversely related to the ability to resist differential cryptographic attacks. The differential uniformity of each scheme is obtained from the data input into the S-box, as in Figure 5.

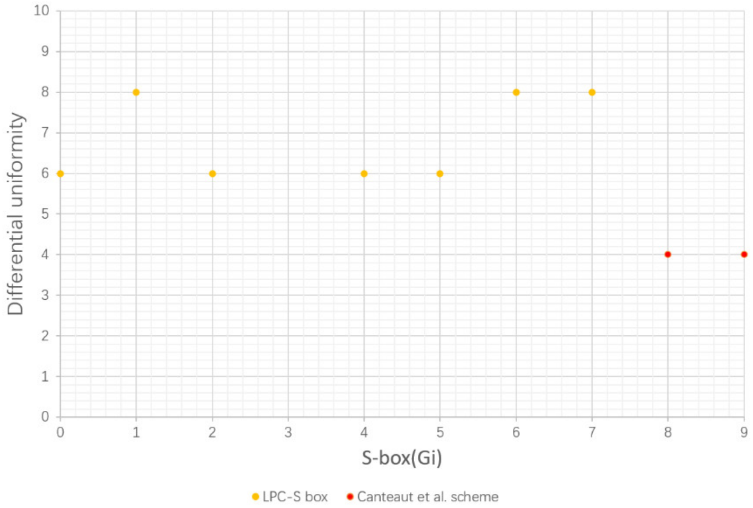

Figure 5. Differential uniformity.

The range of differential uniformity for n × n S-boxes is $0 \leq \delta_F \leq 2^n$, and the range of differential uniformity for 4-bit S-boxes is $0 \leq \delta_F \leq 16$. By comparison, the difference uniformity of the LPC S-box is up to 8, and the difference uniformity of the Canteaut et al. [24] scheme is all 4. In terms of security performance, the smaller the maximum value of the differential propagation probability, the stronger the S-box resists differential attacks.

Hence, LPC S-box has excellent ability to resist differential attacks.

5.3. Transparency Order

Transparency order is an indicator to measure the ability of Boolean functions to resist differential power analysis (DPA) attacks. The lower the transparency order of Boolean function, the stronger the ability to resist DPA attack. The DPA attack has nothing to do with the security of the algorithm, which is an attack based on the characteristics of the algorithm on the device [27]. The transparency order of each scheme is obtained from the data input into the S-box, as in Figure 6.

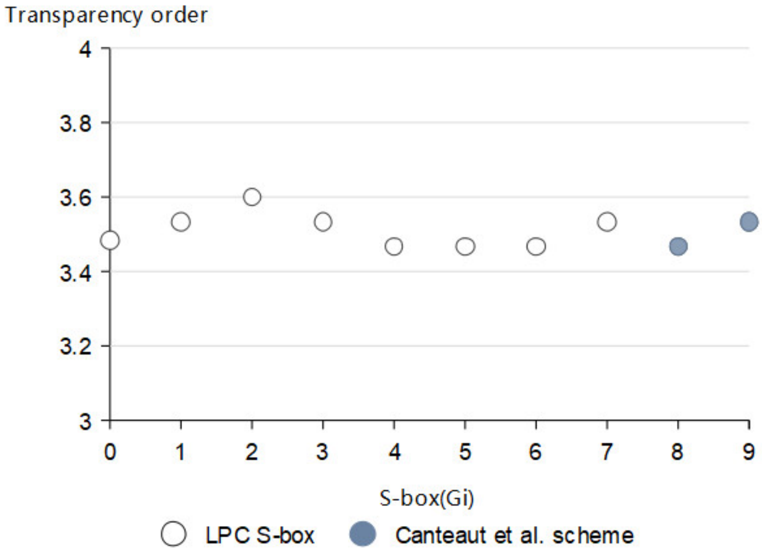

Figure 6. Transparency order of each scheme.

According to the above definition analysis, the transparency order, and nonlinearity of the 4-bit S-box are negatively correlated. When the S-box has high nonlinearity, its transparency order is low. It can be seen from Figure 6 that the average level of transparency order of LPC S-box is slightly smaller than that of the Canteaut et al. [24] scheme. Correspondingly, LPC S-box has excellent ability to resist DPA attacks.

5.4. Algebraic Degree

For resisting the Berlekamp Massey-attack and the Ronjom-Helleseth attack, the Boolean function in the S-box must have a high algebraic degree. The algebraic degree of function F, expressed by $deg(F)$, is the number of variables contained in the highest order term in its algebraic normal form. We obtain their algebraic degree by inputting the data of S-box, and obtain the data shown in Figure 7 according to the simulation calculation.

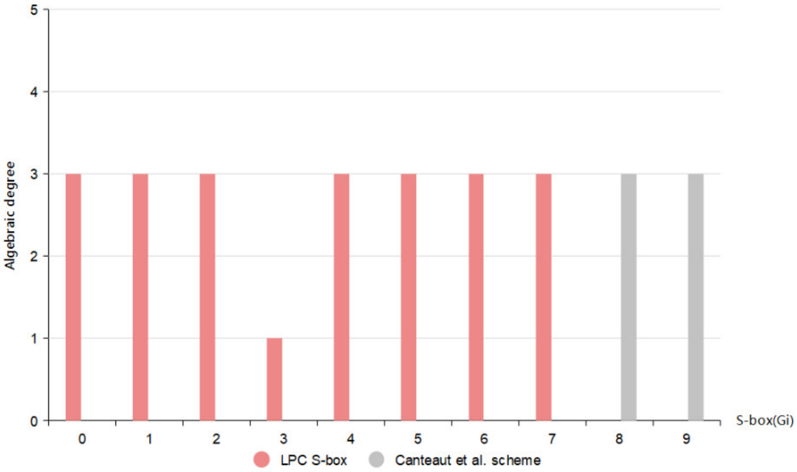

Figure 7. Algebraic degree of each scheme.

The upper bound for the number of algebras of the 4-bit S-box is 4, and it can be seen from the figure that the number of algebras of the LPC S-box is up to 3. This has a higher algebraic degree for a 4-bit S-box, because it is difficult to construct an S-box close to the upper bound, and the closer the algebraic number is to the upper bound, the stronger the anti-algebraic attack capability of the corresponding S-box. Therefore, both the LPC S-box and the Canteaut et al. [24] scheme have outstanding resistance to Berlekamp Massey-attack and the Ronjom-Helleseth attack.

5.5. Algebraic Immunity

Algebraic attacks are often used in stream cipher systems, threatening the security of the entire cryptosystem. In order to resist algebraic attacks, people have proposed a new security index of Boolean functions (e.g., algebraic immunity). The algebraic immunity of n-ary Boolean function F is expressed by $AI(F)$. We obtain their algebraic immunity by inputting the data of S-box, and obtain the data shown in Figure 8 according to the simulation calculation.

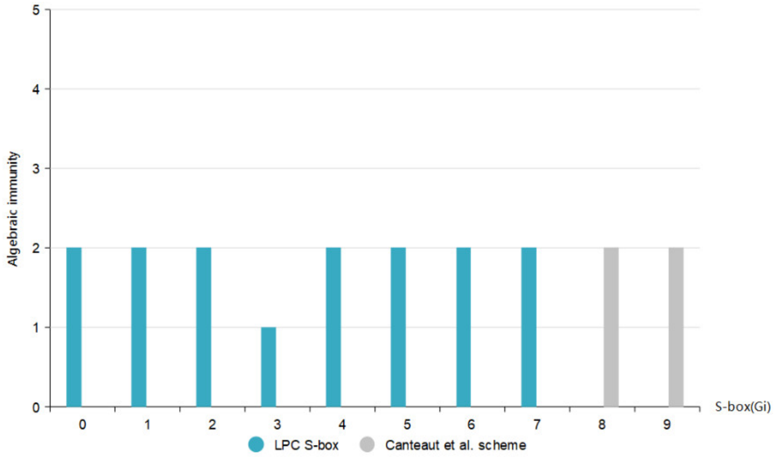

Figure 8. Algebraic immunity of each scheme.

The upper bound of the algebraic immunity of an n × n S-box is $AI(F) \le \left\lceil \frac{n}{2} \right\rceil$, and the upper bound of the algebraic immunity of a 4-bit S-box is 2. As can be seen from Figure 7, the algebraic immunity of LPC S-boxes all reach the upper bound of 2, which has the highest algebraic immunity for 4-bit S-boxes, so the constructed S-boxes are optimal. Hence, LPC S-box has excellent ability to resist algebraic attacks.

5.6. Overall Performance Analysis

In the benchmark test, the cryptographic performance index of the S-box of the existing scheme is compared, which is based on the verification of the security of the S-box in the multilayer perceptron. The nonlinearity, transparency order, algebraic degree, and algebraic immunity selected in the experiment are supplementary explanations for the security of the S-box. The data itself reflects the security of the 4-bit S-box. The calculation results of the transparency order verify that the correlation between the original data and the output data is reduced through the S-box of the hidden layer, and a better anti-DPA performance is obtained. Related research has proved that power consumption is the main function of Hamming weight of data privacy in displacement operation [28]. If Hamming weight of data privacy is disclosed, attackers can determine the number of bits of data privacy by solving a series of linear equations. In DPA, the attacker associates the power consumption with the data value manipulated in the replacement process, and uses the corresponding function solution to obtain data privacy information. Therefore, a high-performance transparency order will reduce power consumption. High nonlinearity is also difficult for attackers to solve linear equations [29], so the calculation of nonlinearity can also show the correlation between S-box and low power consumption. Therefore, we tested based on the cryptographic performance indicators of the S-box in the multilayer perceptron, and proved that LPC S-box reduced the overhead for the overall system operation in terms of computing, thus achieving the goal of low power consumption.

6. Summary

In this paper, a low-power chaotic S-box is designed based on the multilayer perceptron, and the existing image is preprocessed, denoised, entropy coded compression, S-box replacement, and other operations through the algorithm to make the image processing more efficient and safe. By comparing the performance of eight types of low-power 4-bit S-boxes with that of some commonly used lightweight 4-bit S-boxes, we know that the LPC S-box has excellent performance. In the performance test of Boolean functions in low-power S-boxes, LPC S-box has good nonlinearity, differential uniformity, and transparency, and can effectively resist linear attacks, differential attacks, and DPA attacks.

Author Contributions: Conceptualization, R.R.; methodology, R.R. and B.Y.; software, B.Y.; validation, B.Y.; formal analysis, R.R.; investigation, R.R. and B.Y.; resources, J.S., R.Y.K.L. and Q.L.; data curation, B.Y.; visualization, R.R.; writing—original draft preparation, R.R. and B.Y.; writing—review and editing, R.R.; supervision, J.S., R.Y.K.L. and Q.L.; project administration, J.S., R.Y.K.L. and Q.L.; funding acquisition, J.S., R.Y.K.L. and Q.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Key scientific research project of Shaanxi Provincial Department of Education (No. 18JZ052); Shaanxi Provincial Social Science Foundation (2022D161); Research on major theoretical and practical problems of philosophy and Social Sciences in Shaanxi Province general project later funded project "Research on cultivation mechanism and Countermeasures of data factor market in Shaanxi Province"; Research project of high quality development special research project of Shaanxi Provincial People's Congress Financial and Economic Committee (2022HZ1500); Shaanxi Province Philosophy and Social Sciences Major Theoretical and Practical Problems Research General Project Later Funding Project (SX-318); Communications Soft Science Project of the Ministry of Industry and Information Technology (2021-R-44); General project of Xi'an soft science research (21RKYJ0019); the Research Grants Council of the Hong Kong Special Administrative Region, China (Project: CityU 11507219); the CityU SRG (Project: 7005780); Humanities and social sciences fund of the Ministry of Education (Youth Project) under Grant 20YJC630086; Key scientific research program of Shaanxi Provincial Department of Education (20JT058).

Institutional Review Board Statement: Not applicable.

Data Availability Statement: The data used to support the findings of this study are included within the article.

Conflicts of Interest: The authors declare that they have no conflict of interest.

References

1. Ahmadlou, M.; Al-Fugara, A.; Al-Shabeeb, A.R.; Arora, A.; Al-Adamat, R.; Pham, Q.B.; Al-Ansari, N.; Linh, N.T.T.; Sajedi, H. Flood susceptibility mapping and assessment using a novel deep learning model combining multilayer perceptron and autoencoder neural networks. *J. Flood Risk Manag.* **2020**, *14*, e12683. https://doi.org/10.1111/jfr3.12683.
2. Arias del Campo, F.; Guevara Neri, M.; Vergara Villegas, O.; Cruz Sánchez, V.; Ochoa Domínguez, H.; García Jiménez, V. Auto-adaptive multilayer perceptron for univariate time series classification. *Expert Syst. Appl.* **2021**, *181*, 115147. https://doi.org/10.1016/j.eswa.2021.115147.
3. Wu, B.; Qin, J. A list-ranking framework based on linear and non-linear fusion for recommendation from implicit feedback. *Entropy* **2022**, *24*, 778. https://doi.org/10.3390/e24060778.
4. Al Bataineh, A.; Kaur, D.; Jalali, S. Multi-layer perceptron training optimization using nature inspired computing. *IEEE Access* **2022**, *10*, 36963–36977. https://doi.org/10.1109/access.2022.3164669.
5. Rather, S.; Bala, P. A hybrid constriction coefficient-based particle swarm optimization and gravitational search algorithm for training multi-layer perceptron. *Int. J. Intell. Comput. Cybern.* **2020**, *13*, 129–165. https://doi.org/10.1108/ijicc-09-2019-0105.
6. Semwal, V.; Raj, M.; Nandi, G. Biometric gait identification based on a multilayer perceptron. *Robot. Auton. Syst.* **2015**, *65*, 65–75. https://doi.org/10.1016/j.robot.2014.11.010.
7. Faraji, J.; Ketabi, A.; Hashemi-Dezaki, H.; Shafie-Khah, M.; Catalao, J. Optimal day-ahead self-scheduling and operation of prosumer microgrids using hybrid machine learning-based weather and load forecasting. *IEEE Access* **2020**, *8*, 157284–157305. https://doi.org/10.1109/access.2020.3019562.
8. Li, Q.; Lu, K.; Wu, K.; Zhang, H.; Sun, X.; Wu, X.; Xiao, D. A novel high-speed and high-accuracy mathematical modeling method of complex MEMS resonator structures based on the multilayer perceptron neural network. *Micromachines* **2021**, *12*, 1313. https://doi.org/10.3390/mi12111313.
9. Valpola, H.; Karhunen, J. An unsupervised ensemble learning method for nonlinear dynamic state-space models. *Neural Comput.* **2002**, *14*, 2647–2692. https://doi.org/10.1162/089976602760408017.
10. Su, Y.; Lu, X.; Zhao, Y.; Huang, L.; Du, X. Cooperative communications with relay selection based on deep reinforcement learning in wireless sensor networks. *IEEE Sens. J.* **2019**, *19*, 9561–9569. https://doi.org/10.1109/jsen.2019.2925719.
11. Nandan, V.; Rao, R. Low-power AES S-box design using dual-basis tower field extension method for cyber security applications. *Complex Intell. Syst*. **2021**, 1–9. https://doi.org/10.1007/s40747-021-00556-x.
12. Lu, Q.; Zhu, C.; Wang, G. A novel S-box design algorithm based on a new compound chaotic system. *Entropy* **2019**, *21*, 1004. https://doi.org/10.3390/e21101004.
13. Liu, H.; Kadir, A.; Xu, C. Cryptanalysis and constructing S-Box based on chaotic map and backtracking. *Appl. Math. Comput.* **2020**, *376*, 125153. https://doi.org/10.1016/j.amc.2020.125153.
14. Kotlarz, P.; Kotulski, Z. On application of neural networks for S-Boxes design. *Adv. Web Intell*. **2005**, *3528*, 243–248. https://doi.org/10.1007/11495772_38.

15. Sheik, S.A.; Muniyandi, A.P. Secure authentication schemes in cloud computing with glimpse of artificial neural networks: A review. *Cyber Secur. Appl.* **2023**, *1*, 100002. https://doi.org/10.1016/j.csa.2022.100002.
16. Noughabi, M.N.A.; Sadeghiyan, B. Design of S-boxes based on neural networks. In Proceedings of the 2010 International Conference on Electronics and Information Engineering, Kyoto, Japan, 1–3 August 2010. https://doi.org/10.1109/iceie.2010.5559741.
17. Arrañaga, J.D.R.; Chavarin, J.A.S.; Panduro, J.J.R.; Alvarez, E.C.B. New S-box calculation for Rijndael-AES based on an artificial neural network. *ReCIBE Revista Electrónica de Computación Informática Biomédica Y Electrónica* **2017**, *6*, 49–69.
18. Zhu, S.; Wang, G.; Zhu, C. A secure and fast image encryption scheme based on double chaotic S-boxes. *Entropy* **2019**, *21*, 790. https://doi.org/10.3390/e21080790.
19. Aruna, S.; Usha, G. HPAC-sbox- a novel implementation of predictive learning classifier and adaptive chaotic s-box for counterfeiting sidechannel attacks in an IOT networks. *Microprocess. Microsyst.* **2021**, *81*, 103737. https://doi.org/10.1016/j.micpro.2020.103737.
20. Yang, C.; Wei, X.; Wang, C. S-box design based on 2D multiple collapse chaotic map and their application in image encryption. *Entropy* **2021**, *23*, 1312. https://doi.org/10.3390/e23101312.
21. Zhang, X.; Chen, F.; Chen, B.; Cao, Z. A new scheme for implementing S-box based on neural network. In Proceedings of the 2015 International Conference on Computational Science and Computational Intelligence (CSCI), Las Vegas, NV, USA, 7–9 December 2015. https://doi.org/10.1109/csci.2015.9.
22. Kotlarz, P.; Kotulski, Z. Neural network as a programmable block cipher. advances in information processing and protection. In *Advances in Information Processing and Protection*; Springer: Boston, MA, USA, 2007; pp. 241–250. https://doi.org/10.1007/978-0-387-73137-7_21.
23. Hue, T.T.K.; Hoang, T.M.; Tran, D. Chaos-based S-box for lightweight block cipher. In Proceedings of the 2014 IEEE Fifth International Conference on Communications and Electronics (ICCE), Danang, Vietnam, 30 July–1 August 2014. https://doi.org/10.1109/cce.2014.6916765.
24. Canteaut, A.; Duval, S.; Leurent, G.; Naya-Plasencia, M.; Perrin, L.; Pornin, T.; Schrottenloher, A. Saturnin: A suite of lightweight symmetric algorithms for post-quantum security. *IACR Trans. Symmetric Cryptol.* **2020**, *2020*, 160–207. https://doi.org/10.46586/tosc.v2020.is1.160-207.
25. Liu, L.; Zhang, Y.; Wang, X. A novel method for constructing the S-box based on spatiotemporal chaotic dynamics. *Appl. Sci.* **2018**, *8*, 2650. https://doi.org/10.3390/app8122650.
26. Sălăgean, A.; Stănică, P. Improving bounds on probabilistic affine tests to estimate the nonlinearity of Boolean functions. *Cryptogr. Commun.* **2021**, *14*, 459–481. https://doi.org/10.1007/s12095-021-00529-4.
27. Kubota, T.; Yoshida, K.; Shiozaki, M.; Fujino, T. Deep learning side-channel attack against hardware implementations of AES. *Microprocess. Microsyst*. **2021**, *87*, 103383. https://doi.org/10.1016/j.micpro.2020.103383.
28. Liu, Z.; Zeng, Y.; Zou, X.; Han, Y.; Chen, Y. A high-security and low-power AES S-box full-custom design for wireless sensor network. In Proceedings of the 2007 International Conference on Wireless Communications, Networking and Mobile Computing, Shanghai, China, 21–25 September 2007. https://doi.org/10.1109/wicom.2007.622.
29. Yang, L.; Hong, S.; Xu, Y. The nonlinearity and Hamming weights of rotation symmetric Boolean functions of small degree. *AIMS Math.* **2020**, *5*, 4581–4595. https://doi.org/10.3934/math.2020294.