A New S-Box Generation Method and Advanced Design Based on Combined Chaotic System

Ding Zhu 1, Xiaojun Tong 1,*, Miao Zhang 1,2 and Zhu Wang 2

1 School of Computer Science and Technology, Harbin Institute of Technology, Weihai 264209, China; t19s130319@stu.hit.edu.cn
2 School of Information, Harbin Institute of Technology, Weihai 264209, China; wangzhu@hit.edu.cn
* Correspondence: tong_xiaojun@hit.edu.cn (X.T.); zm@hitwh.edu.cn (M.Z.); Tel.: +86-130-61181-039 (X.T.)

Received: 5 November 2020; Accepted: 11 December 2020; Published: 15 December 2020

Abstract: The construction of substitute box (S-box) has always been an important research direction in cryptography. This paper proposes a new S-box generation method and advanced design based on combined chaotic system. Firstly, our paper proposes a new combined chaotic system and analyze its dynamic behavior. Next, we construct S-box by combining the generated pseudo-random sequence with the linear congruence random number generator, and the standard mapping is introduced to scramble the initial S-box. Then, the S-box optimization method based on advanced genetic algorithm is proposed in this paper. We design adaptive of S-box coding, selection operator, crossover operator and mutation operator to avoid the shortcomings of low calculation efficiency and non-convergence of optimization results in traditional genetic algorithm. Finally, through a lot of security analysis experiments and comparison with other papers, the optimized S-box has better cryptography characteristics and can resist linear attacks and differential attacks.

Keywords: substitute box; combined chaotic system; advanced genetic algorithm

1. Introduction

Cryptographic algorithms are widely used in many fields, which can transmit important personal data through remote channels in computer and communication technology. In general, symmetric and asymmetric are two basic system of cryptographic. Symmetric cipher is also known as private key cryptography, asymmetric cipher is known as public key cryptography. They have their own advantages: symmetric cipher is faster, while asymmetric cipher is more secure.

In addition, block cipher is one class in cryptographic, substitute box (S-box) is the only non-linear component in block cipher, which plays an important role in the security of encryption algorithm. Therefore, how to build a strong S-box has become the attention issue. Chaotic system refers to the existence of seemingly random irregular motion in a deterministic system. Its behavior is characterized by uncertainty, unrepeatability and unpredictability. According to its characteristics, it is very suitable for building S-boxes.

Aim to distinguish the performance of S-box, there are some criteria to estimate whether S-box is strong or not, such as linear approximation probability, the output bits independence criterion, equiprobable input/output XOR distribution, the strict avalanche criterion (SAC), nonlinear criterion [1]. The evaluation criteria are specific, so we can optimize the S-box by optimizing these values, there are many optimization algorithms, such as simulated annealing algorithm [2], ant colony algorithm [3], genetic algorithm [4], particle swarm optimization algorithm [5]. Among them, genetic algorithm has been widely used because of its convenience, it is suitable to be used to optimize S-box, because we can regard multiple S-boxes as input individuals of genetic algorithm.
The purpose of our study was not to focus on how to design cryptographic algorithm, we aimed to propose the method to construct and optimize the S-box. One advantage of our study is that the criteria can reflect the performance of S-box, another advantage is that the optimization S-box can be applied in any block cipher algorithms. The main contributions are as follows:

- Our paper proposes a new combined chaotic system and analyzes its dynamic behavior.
- We construct S-box by combining the generated pseudo-random sequence with the linear congruence random number generator, and the standard mapping is introduced to scramble the initial S-box.
- The S-box optimization method based on advanced genetic algorithm is proposed in this paper.
- Through a lot of security analysis experiment and comparison with other papers, the optimized S-box has better cryptography characteristics and can resist linear attacks and differential attacks.

Therefore, this paper proposes a new combined chaotic system and the S-boxes construction method. We design self-adaptive processing of multiple operators in general genetic algorithm to optimize performances of the S-boxes. Through a lot of security analysis experiments and comparison with other papers, the optimized S-box has better cryptography characteristics and can resist linear attacks and differential attacks.

The remaining part of this paper is organized as follows. In Section 2, we have listed some review related to construction and optimization of S-box. The new combined chaotic system is proposed in Section 3, and analyzes its dynamic behavior. Section 4 introduces the method of S-box generation based on combined chaotic system. In Section 5, we propose an advanced genetic algorithm to optimize the S-Box. In Section 6, we evaluate the proposed S-box by a lot of security analysis experiments and comparison with other papers. In Section 7, we have a comprehensive analysis of the advantage of the proposed method. Finally, we present the conclusion in Section 8.

2. Related Work

2.1. Review of S-Box

In recent years, many methods of S-box construction have been proposed. Tian [6] designed a chaotic S-box based on the intertwining logistic map and bacterial foraging optimization. Ahmed [7] proposed an S-box using Gaussian distribution and linear fractional transform based on the BoxMuller transform, polarization decision and central limit algorithm. Khan [8] proposed a systematic design methodology to generate a chaotic S-box using difference distribution table. Isa [9] proposed a heuristic method called the bee waggle dance for designing of S-box. Rafiq [10] proposed an innovative scheme of S-box based on the action of projective linear groups on the projective line and the permutation triangle groups. Ahmad [11] proposed an S-box based on artificial bee colony optimization and the chaotic map. Grosso [12] proposed an innovative design of S-boxes using cubic polynomial mapping. Shahzad [13] proposed a method for obtaining random bijective S-boxes based on improved one-dimensional discrete chaotic map. Khan [7] proposed an S-box construction algorithm based on Gaussian distribution, linear fractional transformation and center restriction algorithm; Asif [8] proposed an S-box construction algorithm based on differential distribution table; Ullah [14] proposed an S-box construction algorithm based on box linear fractional transformation of chaotic system; Belazi and El-Latif [15] proposed a simple S-box generation algorithm based on sine chaotic map. Khan [7] proposed an S-box using Gaussian distribution and linear fractional transform, which is constructed by employing a linear fractional transform based on the BoxAASMuller transform, polarization decision and central limit algorithm. Rafiq [10] developed an innovative scheme of S-box based on the action of projective linear groups on the projective line, and the permutation triangle groups. Zahid [12] proposed an innovative design of S-boxes using cubic polynomial mapping, the use of cubic polynomial maintains the simplicity of the S-box construction method.
2.2. Review of Genetic Algorithm

The linear probability (LP) and differential probability (DP) of the normal method are not good, and the ability to resist linear and differential attacks is not ideal. In addition, many scholars turned their attention to some mathematical intelligent algorithms. For example, Guesmi et al. [16] proposed an S-box optimization method based on genetic algorithm, and optimized the generated S-box with nonlinearity as the fitness value; Ilivanov et al. [17] proposed an S-box construction and optimization algorithm based on the genetic algorithm of reverse work, which can quickly generate multi-dimensional bijective S-boxes. Wang et al. [18] proposed an 8 × 8 S-box design scheme based on chaos mapping and genetic algorithm, which turned the construction of S-boxes into a traveling salesman problem. Through this method makes full use of the characteristics of chaotic mapping and evolution process, a better performance S-box is obtained. However, there are some problems in the application of this genetic algorithm, such as the low efficiency of calculation and the non-convergence of optimization results, which lead to the low performance of the optimized S-boxes.

3. Chaotic System Analysis

\[ y = k \cdot x_{n-1} (1 - x_{n-1}) \]  \hspace{1cm} (1)
\[ y = k \cdot \sin(\pi \cdot x_{n-1}) \]  \hspace{1cm} (2)

There are two one-dimensional chaotic maps. Equation (1) is called as logistic map, where \( x_{n-1} \) is a state variable, \( k \in (0,4] \) is a control parameter and \( n \) is the number of iterations. The bifurcation diagram and Lyapunov exponent of Equation (1) are shown in Figure 1a,b. Sine map is showed as Equation (2). \( k \in (0,4] \) is a control parameter, The bifurcation diagram and Lyapunov exponent of Equation (1) are shown in Figure 1c,d. We can find that the ergodicity is not good and there exist some periodic windows of logistic map and sine map, their Lyapunov exponent are low, none of them is more than 1, which shows that the above two chaotic systems are defective and chaotic dynamic behavior can be improved.

![Figure 1](image-url)

*Figure 1. (a) Bifurcation diagrams of logistic map. (b) Lyapunov exponents of logistic map. (c) Bifurcation diagrams of sine map. (d) Lyapunov exponents of sine map.*
Enhance the ergodicity, randomness and enlarge the key space to make it suitable for cryptography. Then, this paper constructed the combined chaotic system based on Equations (1) and (2):

\[
x_n = \begin{cases} 
  k \cos(1 - \arccos(x_{n-1})), & x > \frac{\pi}{4} \\
  \frac{k}{\pi} \sin(\pi x_{n-1}^i), & x \leq \frac{\pi}{4}
\end{cases}
\]

(3)

where \(0 < k \leq 4\), the following is the dynamic behavior analysis of the combined chaotic system, mainly analyzing the following performance characteristics: Lyapunov exponent, bifurcation diagram, sequence uniformity and initial value sensitivity.

- **Lyapunov exponent of the combined chaotic system**

Lyapunov exponent describes the chaotic system quantitatively. For one-dimensional chaos, as long as one positive Lyapunov exponent of the chaotic system means that the system is chaotic. Lyapunov exponent is denoted as following Equation (4):

\[
LE = \lim_{n \to \infty} \frac{1}{n} \ln \left| \frac{dF(x)}{d(x)} \right|_{x=x_0}
\]

(4)

where \(F(x)\) is the chaotic system. \(LE\) is the Lyapunov exponent, it represents the exponent of the exponential separation caused by each iteration in multiple iterations.

As illustrated in Figure 1, the Lyapunov exponents of the system are all positive in the parameter space, and the maximum Lyapunov exponent is 1.265 \((k = 0.13)\), which shows that the system is chaotic.

- **Bifurcation diagram of the combined system**

Bifurcation diagram is another important concept to study the chaotic characteristics of chaotic system, which means that the dynamic state of chaotic system may change with the change of parameters, resulting in bifurcation phenomenon.

Figure 2a is bifurcation diagrams of the combined system, which shows that the system is in a chaotic state all the time, without obvious bifurcation phenomenon. It proves that the dynamic state of the system does not change with the change of parameters, and the system is always in a stable chaotic state. Therefore, the system is a chaotic system with good chaotic characteristics.

![Figure 2. (a) Bifurcation diagrams of the combined system. (b) Lyapunov exponent of the combined chaotic system.](image-url)

- **Sequence uniformity**
The chaotic sequences generated by a strong chaotic system iterated should be uniform. Figure 3 represents the number of occurrences of X values in different ranges for different iterations. Figure 4 shows the distribution of the chaotic sequence after 5000 iterations times of the combined chaotic system. The X-axis represents the number of iterations, the Y-axis represents frequency of the sequence value of iterations and the different colors represent the different range of values. According to the maximum Lyapunov exponent of the combined chaotic system, we set \( X_0 = 0.05, t = 0.13 \) as initial conditions. As illustrated in Figures 3 and 4, the chaotic sequences generated are evenly distributed in the whole range, and can completely cover the whole value range. Therefore, the combined chaotic system has strong performance in sequence uniformity and can be used to generate chaotic pseudo-random sequences.

**Figure 3.** Statistics of the number of occurrences of X values in the same range.

**Figure 4.** Sequence uniformity.

- **Initial value sensitivity**

  Set \( X_0 = 0.12345, X_0 = 0.12345 + 10^{-10} \) and \( t = 0.13 \) as initial conditions, and conditions of other parameters are the same, then the two chaotic sequences are generated through 100 iterations. The system is behaving chaotically with this condition; when \( t = 0.13 \), the system has the largest Lyapunov exponent, as shown in Figure 5. The chaos sequences begin to diverge and show a stable state of divergence after the number of iterations close to 11. In the long-term behavior of the system, the system has initial value sensitivity. Therefore, the combined chaotic system proposed in this paper is
a chaotic system with strong performance and can be used to generate chaotic pseudo-random sequences.

Figure 5. Initial value sensitivity.

4. S-Box Generation Method Based on Combined Chaotic System

4.1. Construction of S-Box

The construction steps of S-box in this paper are shown as the Figure 6:

Step 1. Generate chaotic pseudo-random sequence according to Equation (3).

Step 2. Discretization of pseudo-random sequences, the phase space [0,1] of chaotic sequence is equally divided into 65,536 parts. The chaotic sequence $\{x_i\}_{i=1}^{N}$ is transformed into the corresponding phase space with the help of Equation (5), then we obtain the discrete pseudo-random sequence $\{seq\}_{i=1}^{N}$, where N is the iteration times.

$$Seq(i) = \begin{cases} 
\text{ceil}( (1 - 2 \times \arcsin(x_i)/\pi) \times 65536) & \text{if } 0 \leq x_i < 1 \\
65535 & \text{if } x_i = 1
\end{cases}$$ (5)

Step 3. The randomness of the pseudo-random sequences will be reduced after discretization, the linear congruence random number generator (LCGs) is used to avoid the short-period phenomena with the help of Equation (6).

$$LCG(i) = (a \times LCG(i-1) + b) \mod(m)$$ (6)
where \(a, b, c\) are constant, \(a\) is multiplier, \(b\) is increment, \(m\) is module. Set \(a = 16807, b = 0, m = 2^{32} - 1\). Therefore, the periodic value of LCGs is \(2^{31} - 1\), then we obtain the new discrete sequence \(\{y_i\}_{i=1}^{N}\) with the help of Equation (7).

\[
y(i) = \text{Seq}(i) \oplus LCG(i)
\]

(7)

Step 4. Convert the chaotic sequence \(\{y_i\}_{i=1}^{N}\) into 256 integers with the help of Equation (8).

\[
z(i) = y(i) \mod 256
\]

(8)

Select 256 non-repeated from sequence \(\{z_i\}_{i=1}^{N}\) and record their positions. Rearrange the sequence with ascending order then obtain a new sequence \(\{S_i\}_{i=1}^{256}\).

4.2. Scrambling of S-Box

The scrambling steps of S-box in this paper is shown as the Figure 7:

![Figure 7. S-box scrambling.](image)

Step 1. Scrambling the sequence \(\{S_i\}_{i=1}^{256}\), generated in the Section 4.1 with the help of standard mapping. The standard mapping follows Equation (9):

\[
\begin{align*}
x_{r+1} &= (x_r + y_r) \mod 2\pi \\
y_{r+1} &= (y_r + k \sin(x_r + y_r)) \mod 2\pi
\end{align*}
\]

(9)

Step 2. Obtain two sequences \(\{x_i\}\) and \(\{y_i\}\) by step 1. Then the elements in initial S-box are scrambled by row left and column rotation, and the sequence is discretized to \([0, 15]\) with the help of Equation (10):

\[
\begin{align*}
row_i &= \text{floor}(x_i + 0.5) \* 16 + 1 \\
col_j &= \text{floor}(y_j + 0.5) \* 16 + 1 \quad i = 1, 2, ..., 16
\end{align*}
\]

(10)

where \(row_i\) represents rotate left by \(row_i\) bits for the \(i\)-th row of the initial S-box, \(col_j\) represents rotate the \(col_j\) bits for the \(j\)-th column of the initial S-box.

4.3. Security Analysis of Chaotic Pseudo-Random Sequences

In order to verify the randomness of the discrete chaotic pseudo-random sequence obtained from the combined chaotic system, SP800-22 is tested in this paper. SP800-22 has 16 test items. The test items meet the requirements if \(p\)-value > 0.01. Only if all the 16 test items meet the requirements, the ciphertext generated by cryptographic algorithm can be considered as good random performance. The selected text should be large enough and divided into 100 groups. The results as shown in Table 1:

Table 1. SP800-22 result.
| Test Suite                      | Pass Rate | p-Value   | Result |
|--------------------------------|-----------|-----------|--------|
| Frequency                      | 0.97      | 0.5748    | SUCCESS |
| Block Frequency                | 1.00      | 0.6253    | SUCCESS |
| Cumulative                     | 0.98      | 0.8712    | SUCCESS |
| Runs                           | 0.99      | 0.1623    | SUCCESS |
| Lonest Run                     | 1.00      | 0.3458    | SUCCESS |
| Rank                           | 0.99      | 0.5182    | SUCCESS |
| FFT                            | 0.97      | 0.7923    | SUCCESS |
| Non-Overlapping Template       | 1.00      | 0.8322    | SUCCESS |
| Overlapping Template           | 1.00      | 0.6573    | SUCCESS |
| Universal                      | 1.00      | 0.8812    | SUCCESS |
| Approximate Entropy            | 0.99      | 0.1678    | SUCCESS |
| Random Excursion               | 0.97      | 0.5893    | SUCCESS |
| Random Excursions Variant      | 1.00      | 0.7694    | SUCCESS |
| Serial                         | 1.00      | 0.6199    | SUCCESS |
| Serial                         | 0.99      | 0.0765    | SUCCESS |
| Linear Complexity              | 1.00      | 0.8192    | SUCCESS |

The chaotic sequences generated by the combined chaotic system proposed have strong pseudo randomness. The proposed is suitable for constructing S-boxes. We can dynamically generate the S-boxes after the above steps, which is the preparation for the Section 4 about the S-box optimization based on the advanced genetic algorithm.

5. S-Box Optimization Based on Advanced Genetic Algorithm

5.1. Traditional Genetic Algorithm

Genetic algorithm (GA) is a kind of population optimization algorithm derived from the evolution of biological world, which was proposed by J. Holland in 1957. The general process is to calculate the fitness of each individual by calculating the objective function of each individual in the group and comparing the relationship with the fitness function, then select two individuals with larger fitness value to cross and mutate to generate new individuals. The algorithm is terminated when it reaches iteration times. Figure 8 shows the process of traditional genetic algorithm.

![Figure 8. The process of traditional genetic algorithm.](image)

5.2. The Design of S-Boxes Optimization Based on Advanced GA

5.2.1. Adaptive S-Boxes Coding

The S-boxes coding format should be determined before optimizing S-boxes by GA. There are generally two ways: decimal coding and binary coding. This paper proposes different encoding methods for S-boxes encoding in different stages according to characteristics of different encoding format. Select decimal coding in the iterative and crossover stages and select binary coding when
calculating fitness value, which improves the speed of calculating the fitness value and simplifies the de-duplication stage of iterative operation and cross operation.

5.2.2. Advanced Fitness Function

A strong S-box has good performance in nonlinearity, differential uniformity and strict avalanche effect according to the evaluation criteria. However, the design of fitness function is too simple in traditional GA to comprehensively satisfy the performance standard of S-box. Therefore, the fitness function in this paper shown as Equation (11):

\[ F(s) = aF(N_s) + bF(\delta_s) + cF(B_s) \]  \hspace{1cm} (11)

where \( F(s) \) is fitness function, \( F(N_s), F(\delta_s), F(B_s) \) represent nonlinear, difference uniformity, strict avalanche effect; \( a, b \) and \( c \) represent the relevant weights coefficients, respectively.

Through the extensive experiment, a piecewise weighting function is designed in this paper as following Equation (12):

\[
F(s) = \begin{cases} 
    N_s - \frac{1}{4} \delta_s - B_s & N_s < 10 \text{ and } \delta_s > 10 \\
    N_s - \frac{3}{4} \delta_s - B_s & N_s < 10 \text{ and } \delta_s \leq 10 \\
    \frac{5}{3} N_s - \frac{1}{4} \delta_s - B_s & N_s \geq 10 \text{ and } \delta_s > 10 \\
    \frac{5}{3} N_s - \frac{3}{4} \delta_s - B_s & N_s \geq 10 \text{ and } \delta_s \leq 10 
\end{cases} \]  \hspace{1cm} (12)

5.2.3. Advanced Selection Operator

The purpose of selection operator is to select good individuals for the population and eliminate the bad ones in GA. Firstly, the individuals are sorted according to the fitness value, then \( N \) individuals with high fitness value are selected as the next generation. However, there exists many problems in this process. In the early stage of evolution, there are few excellent individuals and their distribution is scattered. In the late stage of evolution, there are many excellent individuals and their distribution is centralized. It is easy to cause the non-convergence of evolution if selection rate is fixed. Therefore, this paper proposes an adaptive selection probability as shown in Equation (13):

\[
p_a = \begin{cases} 
    p_{\text{max}} & \text{if } \frac{q}{Q} \leq \frac{3}{5} \\
    p_{\text{min}} & \text{if } 1 \geq \frac{q}{Q} > \frac{3}{5} 
\end{cases} \]  \hspace{1cm} (13)

where \( q \) is \( q \)-th iteration, \( Q \) is the total number of iterations, \( p_a \) is the selection probability, \( p_{\text{max}} \) is the maximum selection probability, \( p_{\text{min}} \) is the minimum selection probability.

5.2.4. Advanced Crossover Operator

The purpose of crossover operator is to generate new individuals in the iterative process. The crossover rate is a fixed constant in traditional GA. However, a fixed crossover probability may cause that there are not enough individuals in the early stage, or the calculation speed cannot be guaranteed, the stable convergence of the algorithm cannot be guaranteed in the later stage. Therefore, this paper proposes an adaptive crossover probability as shown in Equation (14):
\[
p_h = \begin{cases} 
p'_m \times e^{-q/\bar{Q}} & \text{if } p'_m \times e^{-q/\bar{Q}} < p'_{\min} \\ p'_{\min} & \text{else} \end{cases}
\]

where \(q\) is \(q\)-th iteration, \(Q\) is the total number of iterations, \(p_b\) is the selection probability, \(p'_{\max}\) is the maximum crossover probability, \(p'_{\min}\) is the minimum crossover probability.

5.2.5. Advanced Mutation Operator

The purpose of mutation operator is to increase the diversity of population in traditional GA. The algorithm is not easy to converge and the evolutionary randomness is enhanced when the mutation rate is too large, the algorithm may cause he local optimal solution and the population diversity is too small when the mutation rate is too small. Therefore, this paper proposes the adaptive mutation rate. Firstly, the individuals with larger adaptability should have a lower mutation probability to retain the good characteristics, and the individuals with smaller adaptability should have a larger mutation probability. Secondly, the mutation rate should decrease gradually and the algorithm tends to be stable with the increase of the number of iterations. The adaptive mutation probability designed in this paper as shown in Equation (15):

\[
p_c = \begin{cases} 
p''_{\max} \times e^{-|F(x)/F_{\max}| - q/\bar{Q}} & \text{if } p''_{\max} \times e^{-|F(x)/F_{\max}| - q/\bar{Q}} > p''_{\min} \\ p''_{\min} & \text{else} \end{cases}
\]

where \(F_{\max}\) is the maximum fitness of the current population, \(F(x)\) is the fitness of the individual mutated, \(p''_{\max}\) is the maximum mutation probability, \(p''_{\min}\) is the minimum mutation probability.

5.3. The Design of S-Boxes Optimization Based on Advanced GA

The S-boxes optimization algorithm in this paper combines the S-box construction method proposed in the Section 3 with the advanced GA, including population initialization stage, individual evaluation, selection stage, crossover stage, mutation stage and termination condition determination. Algorithm 1 shows the pseudo code of chaotic s-box optimization. The steps are as follows:

**Step 1: Initialization**

Initialize each element \(x\) in the S-box generated in Section 3 with the help of Equation (16):

\[
X = \text{Floor}(x \times 2^8)
\]

Add each \(X\) to sequence \([S]\), and the output is the individual of the initial population. Repeat the above steps until all the populations are initialized.

**Step 2: Individual evaluation**

Calculate the fitness value of the S-box according to the fitness function, then arrange the individuals in ascending order according to the fitness values. The operation continues unless the number of iterations reaches the threshold or the maximum fitness value in the population is greater than the predetermined value.

**Step 3: Selection stage**

Calculate the selection probability \(p_a\) in the current iteration stage, then select \(N_1\) excellent individuals with the help of Equation (17):

\[
N_1 = p_a \times T
\]

where \(T\) is the number of initial populations.

**Step 4: Crossover stage**
Calculate the number of populations \( N_2 \) generated in the crossover according to Equation (18). Select the individual with the largest fitness value as the parent-1, and the \( i \)-th individual as the parent-2, add parent-1 and parent-2 to the crossover population. The above operation continues until the number of individuals generated by the crossover operator is not greater than \( N_2 \). Cross descendants of the output as elements in \([S']\)

\[
N_2 = p_b \times T \tag{18}
\]

\[
i = \text{Floor}(T \times x') \tag{19}
\]

where \( p_b \) is the crossover probability, \( x' \) is the element of the sequence \([S]\).

**Step 5: Mutation stage**

Calculate the number \( N_3 \) of individuals to be mutated according to Equation (20). Select the \( i \)-th individual in the cross population to perform the mutation operation. Then exchange the \((P_1 + i)\)-th and \((P_2 + i)\)-th individuals to generate an individual mutated. The above operation continues until the number of individuals generated by the mutation operator is not greater than \( N_3 \).

\[
N_3 = p_c \times T \tag{20}
\]

\[
P_1 = \text{Floor}(N_2 \times x'') \tag{21}
\]

\[
P_2 = \text{Floor}((N_2 - 1) \times x'') \tag{22}
\]

where \( p_c \) is the mutation probability, \( x'' \) is the element of the sequence \([S']\).

**Step 6: Termination condition determination**

If the number of iterations is greater than the threshold \( Q \), obtain the individual with the maximum fitness in the optimization process as the optimal solution, then the algorithm is terminated.

**Algorithm 1.** The pseudo code of chaotic s-box optimization.

```
Algorithm The method of S-boxes optimization

Input: The initial S-box population \([S], Q, T, p_{max}', p_{min}', p_{max}', p_{min}', p_{max}', p_{min}'\)
Output: The optimal S-box \( S_{p} \)
1: read the initial S-box \([S]\);
2: sort(F([S]));
3: while(m < Q & & n < T) do
4: if(m/Q <= 3/5)
5: pa = \( p_{max}' \);
6: N1 = pa \times T;
7: else pa = \( p_{min}' \);
8: N1 = pa \times T;
9: end if;
10: Selection(N1);
11: if(\( p_{max}'' \times e^{-\gamma} < p_{min}'' \))
12: pb = \( p_{min}'' \);
13: N2 = pb \times T;
14: else pb = \( p_{max}'' \);
15: N2 = pb \times T;
16: end if;
17: Obtain S1: F(S1) = max(F([S]));
18: Obtain S2: i = Floor(T \times x');
```
5.4. Proposed S-Boxes

Select initial conditions $x = 0.02485$, $t = 1$, $T = 10000$, $p_{\text{max}} = 0.15$, $Q = 200$, $p_{\text{min}} = 0.05$, $p'_{\text{max}} = 0.5$, $p''_{\text{min}} = 0.3$, $p''_{\text{max}} = 0.5$, $p''_{\text{min}} = 0.3$.

Generate the S-box according to the above steps as shown in Table 2.

| 0  | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | A  | B  | C  | D  | E  | F  |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 0  | 244| 83 | 117| 150| 190| 111| 102| 107| 128| 200| 92 | 223| 247| 174| 198|
| 1  | 241| 47 | 95 | 71 | 184| 242| 113| 48 | 30 | 135| 50 | 10 | 202| 110| 22 |
| 2  | 101| 44 | 53 | 13 | 140| 28 | 58 | 168| 196| 132| 199| 70 | 11 | 206| 252|
| 3  | 98 | 90 | 89 | 109| 66 | 61 | 169| 170| 214| 136| 2 | 232| 130| 154|
| 4  | 246| 158| 67 | 39 | 51 | 76 | 87 | 1 | 139| 37 | 121| 176| 24 | 185| 178|
| 5  | 175| 14 | 212| 225| 46 | 12 | 219| 142| 29 | 226| 0 | 81 | 179| 243| 127|
| 6  | 165| 205| 119| 180| 217| 97 | 118| 112| 64 | 159| 94 | 255| 77 | 249| 134|
| 7  | 211| 65 | 181| 43 | 161| 57 | 99 | 201| 108| 115| 155| 187| 123| 208| 173|
| 8  | 238| 222| 248| 216| 182| 237| 152| 25 | 250| 143| 146| 172| 18 | 194| 5 |
| 9  | 192| 239| 8 | 254| 221| 80 | 35 | 75 | 195| 21 | 229| 213| 62 | 224| 42 |
| A  | 149| 68 | 114| 86 | 15 | 27 | 245| 144| 227| 88 | 105| 141| 72 | 38 | 210|
| B  | 122| 56 | 73 | 236| 19 | 103| 7 | 129| 233| 209| 52 | 54 | 133| 163| 93 |
| C  | 36 | 107| 186| 55 | 125| 191| 106| 45 | 69 | 60 | 85 | 91 | 116| 240| 218|
| D  | 220| 74 | 145| 49 | 151| 164| 230| 26 | 31 | 79 | 197| 84 | 253| 23 | 6 |
| E  | 96 | 166| 183| 59 | 167| 251| 120| 148| 189| 160| 231| 215| 235| 33 | 228|
| F  | 9  | 193| 3 | 188| 204| 104| 32 | 4  | 40 | 156| 78 | 63 | 16 | 41 | 138|

6. Performance Analysis

In order to verify the cryptography characteristics of S-boxes, according to the performance standard evaluation system of S-boxes, this paper analyzes the performances of the proposed S-boxes included nonlinear criterion, the strict avalanche criterion, differential uniformity, the output bits independence criterion and linear approximation probability.

6.1. Nonlinear Criterion

Nonlinear criterion is an important characteristic of S-boxes performance evaluation system. The higher the nonlinearity, the stronger the ability of S-box to resist nonlinear attack. By definition, $f: F_2^n \rightarrow F_2$ can be considered as N-Variable Boolean function. The nonlinear criterion is denoted by

$$N_f = \min_{i \in N} d_H(f, I)$$ (23)
where $L_n$ is an affine function set, $d_H(f,l)$ is the Hamming distance between $f$ and $l$. In general, $f$ is denoted by Walsh spectrum as Equation (24):

$$S_f(\omega) = \sum_{\omega \in GF(2^n)} (-1)^{d_H(f,x) \cdot \omega}$$  \hspace{1cm} (24)$$
with $\omega \in GF(2^n)$, and $\omega \cdot x$ is the dot product between $x$ and $\omega$. The nonlinearity is shown in Equation (25):

$$N_f = 2^n - (1 - 2^{-n} \max |S_{\omega \in GF(2^n)}(\omega)|)$$  \hspace{1cm} (25)$$

Nonlinearity is a criterion of linear cryptanalysis. The higher the nonlinearity, the stronger the ability of S-boxes to resist nonlinear attack.

6.2. The Strict Avalanche Criterion (SAC)

$S(x) = [f_1(x), \ldots, f_m(x)] : F_2^n \rightarrow F_2^m$ satisfies the strict avalanche effect, which means that each output bit changes with each input bit, and the probability of change is strictly 0.5.

In general, the strict avalanche effect of S-boxes can be measured by SAC correlation matrix. The S-boxes satisfy the strict avalanche effect if each of the matrix correlation is close to 0.5.

Set $e_i = [\delta_{i,1}, \delta_{i,2}, \ldots, \delta_{i,m}]^T$, $\delta_{i,j}$ is denoted as Equation (26):

$$\delta_{i,j} = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases}$$  \hspace{1cm} (26)$$

$(\cdot)^T$ is transpose matrix, showed as Equation (27):

$$P_{i,j} = 2^{-n} \sum_{x \in B^n} f_i(x) \oplus f_j(x \oplus e_i)$$  \hspace{1cm} (27)$$

The estimation bias of SAC correlation is measured by Equation (28):

$$S(f) = \frac{1}{n^2} \sum_{1 \leq i < j \leq n} \left| \frac{1}{2} - P_{i,j}(f) \right|$$  \hspace{1cm} (28)$$

Table 3 shows the comparison of our S-boxes and other S-boxes about the values of SAC values. The S-boxes satisfy the strict avalanche criterion when every value of the matrix is close to 0.5.

|   | 110 | 109 | 107 | 108 | 112 | 108 | 109 |
|---|-----|-----|-----|-----|-----|-----|-----|
| 113 | -   | 108 | 110 | 108 | 109 | 112 | 106 |
| 109 | 105 | -   | 116 | 108 | 110 | 109 | 114 |
| 114 | 107 | 109 | -   | 106 | 117 | 109 | 109 |
| 112 | 109 | 108 | 107 | -   | 110 | 109 | 110 |
| 108 | 112 | 107 | 106 | 112 | -   | 114 | 110 |
| 114 | 109 | 112 | 110 | 109 | 109 | -   | 112 |
| 117 | 111 | 109 | 110 | 109 | 113 | 112 | -   |

6.3. Equiprobable Input/Output XOR Distribution

Differential probability is a characteristic of S-box, it is a measure of criterion of equiprobable input/output XOR distribution.

Set $S(x) = [f_1(x), \ldots, f_m(x)] : F_2^n \rightarrow F_2^m$ as a multi-output function. Differential uniformity $\eta$ is denoted as Equation (29):
\[
\eta = \frac{1}{2^n} \max_{\alpha \in F_2^n} \max_{\beta \in F_2^n} \left| \{ x \in F_2^n : S(x + \alpha) - S(x) = \beta \} \right|
\] (29)

Differential uniformity also can be expressed by the different approximation probability like as Equation (30):
\[
DP_f = \max_{\Delta x \in \Delta y} \left( \frac{\# \{ x \in X \mid f(x) \oplus f(x + \Delta x) = \Delta y \}}{2^n} \right)
\] (30)
where \(2^n\) is the cardinality of all the possible input values, \(\Delta x\) is input differences, \(\Delta y\) is output differences. The smaller values of \(DP_f\) give better resistance to differential cryptanalysis.

6.4. The Output Bits Independence Criterion

By inverting the plaintext bits to generate vector sets, if the vector sets are independent of each other, the S-boxes satisfy the output bits independence criterion. The independence of avalanche vector pairs can be measured by calculating the correlation coefficient. The correlation coefficient is denoted as Equation (31). Set a given sum of two variables, \(A\) and \(B\).
\[
\rho\{A, B\} = \frac{\text{cov}\{A, B\}}{\sigma_A \sigma_B}
\] (31)
where \(\rho\{A, B\}\) is the correlation between \(A\) and \(B\), \(\text{cov}\{A, B\} = E\{AB\} - E\{A\}E\{B\}\) is the covariance between \(A\) and \(B\), \(\sigma_A^2 = E\{A^2\} - (E\{A\})^2\) and \(\sigma_B^2 = E\{B^2\} - (E\{B\})^2\).

Set two output bits, \(f_i\) and \(f_k\). It shows that S-boxes satisfy the output bits independence criterion if the non-linearity of \(f_i \oplus f_k\) is high and the strict avalanche effect is close to the 0.5.

Tables 3 and 4 show that the minimum of BIC-nonlinearity is 105 and the BIC-SAC is so close to the ideal value 0.5.

Table 4. BIC-strict avalanche criterion (SAC) of the optimized S-box.

|               | 0.503887 | 0.498913 | 0.506238 | 0.502119 | 0.493289 | 0.494276 | 0.500157 |
|---------------|----------|----------|----------|----------|----------|----------|----------|
| 0.507813      | 0.496994 | 0.506836 | 0.507813 | 0.492811 | 0.506602 | 0.495117 |
| 0.480469      | 0.503672 | -        | 0.496094 | 0.500376 | 0.499871 | 0.501728 | 0.49972 |
| 0.500497      | 0.502930 | -        | 0.500198 | 0.500492 | 0.498081 | 0.498814 |
| 0.499882      | 0.49971 | 0.500836 | 0.507831 | -        | 0.501001 | 0.500321 | 0.500427 |
| 0.501813      | 0.500501 | 0.501121 | 0.50192 | 0.50036 | -        | 0.500492 | 0.50513 |
| 0.500499      | 0.500507 | 0.500177 | 0.499213 | 0.499172 | 0.499925 | -        | 0.498210 |
| 0.500130      | 0.500498 | 0.499918 | 0.500116 | 0.500091 | 0.500492 | 0.499981 | -        |

6.5. Linear Approximation Probability

Linear approximation probability (LP) is defined as
\[
LP_f = \max_{a, b \neq 0} \left( \frac{\# \{ x \in X \mid x \cdot a = f(x) \cdot b \} - 2^{n-1}}{2^n} \right)
\] (32)
where \(a\) is the input and \(b\) is the output. The smaller values of \(LP_f\) gives better resistance to linear cryptanalysis.

6.6. Performance Comparison

Table 5 shows the comparison of our S-boxes proposed and other S-boxes from references [16–24].
Table 5. Comparison of S-boxes proposed and other S-boxes.

| S-Boxes           | Average Nonlinearity | SAC   | DP     | BIC-Nonlinearity | BIC-SAC | LP   |
|-------------------|----------------------|-------|--------|------------------|---------|------|
| Constructed S-box | 107.25               | 0.5019| 0.03900| 105.75           | 0.5005  | 0.0629|
| Optimized S-box   | 110.75               | 0.5005| 0.01560| 110.02           | 0.5005  | 0.0156|
| Cui [19]          | 112.00               | 0.5007| 0.01560| 112              | 0.4997  | 0.0156|
| Çavuşoğlu [17]    | 106.25               | 0.5039| 0.03910| 103.35           | 0.5059  | 0.0791|
| Wang [20]         | 110.50               | 0.4937| 0.03910| 103.85           | 0.5033  | 0.0625|
| Belazi [21]       | 112.00               | 0.5115| 0.03130| 103.78           | 0.4982  | 0.0479|
| Lambic [22]       | 106.75               | 0.5010| 0.03910| 104.07           | 0.5005  | 0.0706|
| X. Wang [23]      | 106.75               | 0.4998| 0.03910| 104.14           | 0.4998  | 0.0706|
| Ozkaynak [24]     | 106.75               | 0.4971| 0.03910| 102.92           | 0.5008  | 0.0791|
| Ullah [25]        | 106.00               | 0.5034| 0.04690| 105.28           | 0.4980  | 0.0627|
| AES [26]          | 107.25               | 0.5049| 0.01560| 112              | 0.5046  | 0.0706|

Comparing the results in the Table 3, some insights we can summarize as follows:

- The average nonlinearity of the constructed S-box-based construction method in Section 3 is 107.5, and the optimized S-box based on the advanced GA in Section 4 is 110.75, which is better than all other S-box algorithms except for literature 16 and 19. Therefore, our S-box has very good nonlinear characteristics and can resist linear attacks.
- The SAC value of S-box in this paper is 0.0019 different from 0.5000, and the performance is better than that of references 17, 18, 19, 22, 23, 24. The performance of the optimized S-box is better than that of all other S-box algorithms except for reference 24.
- The value of the XOR distribution of equiprobable input/output of our constructed S-box is 0.03900 and the optimized S-box is 0.01560, which illustrates resistance to differential cryptanalysis.
- The BIC-nonlinearity of our constructed S-box is 105.75 and the BIC-SAC is 0.5005. The BIC-nonlinearity of optimized S-box is 110.02 and the BIC-SAC is 0.5005, which illustrates good output bits independence.
- The LP of our constructed S-box is 0.0629 and the optimized S-box is 0.0156, which illustrates resistance to linear cryptanalysis.
- In conclusion, both the constructed S-box and optimized S-box have a good performance in nonlinear criterion, the strict avalanche criterion, differential uniformity, the output bits independence criterion and linear approximation probability.

7. Comprehensive Analysis of Proposed Method

Our proposed method contains the construction of S-box and the optimization of S-box. The quality of this construction method depends on the chaotic system used. The chaotic system is controlled by parameters $k$ and $\chi$, the proposed method is to generate S-box population by determining the optimal range of these two parameters, the obtained S-box population can guarantee the quality of the S-box. In addition, the speed of the proposed advanced genetic algorithm can be converged, our proposed method presents the following advantages after comparing with method [19–26]:

- The S-box of proposed method is determined in advance, which can ensure the quality of optimization.
- The proposed optimization algorithm controls the degree of optimization through the fitness function; therefore, we can obtain the satisfactory S-box by controlling the fitness function.
- The comparison of performance illustrated that the optimized S-box have a better performance in above performance than method [19–27], which shows the optimized S-box has better cryptography characteristics and can resist linear attacks and differential attacks.
8. Conclusions

In this paper, a new combined chaotic system is proposed based on the chaotic theory, which has good chaotic characteristics proved by analyzing its dynamic characteristics. Next, this paper proposes a method of generating S-boxes based on the combined chaotic system. Finally, an advanced genetic algorithm is proposed to optimize the performances of S-boxes. The proposed advanced genetic algorithm is effective to avoid the shortcomings of low calculation efficiency and non-convergence of optimization results in traditional genetic algorithm. Through a lot of security analysis experiments and comparison with other papers, the optimized S-box has better cryptography characteristics and can resist linear attacks and differential attacks. The constructed S-box can be used in information hiding, image encryption, communication security technology and other fields.

Author Contributions: D.Z., X.T., Z.W. and M.Z. wrote and edited the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: This work was supported by the following projects and foundations: project ZR2019MF054 supported by Shandong Provincial Natural Science Foundation, the National Natural Science Foundation of China (No.61902091) and Innovation Research Foundation of Harbin Institute of Technology (HIT.NSRIF.2020099), the Foundation of Science and Technology on Information Assurance Laboratory (No.KJ-17-004), Equip Pre-research Projects of 2018 supported by Foundation of China Academy of Space Technology (No.WT-TXYY/WLZDFHJY003), 2017 Weihai University Co-construction Project.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Adams, C.; Tavares, S. The structured design of cryptographically good S-boxes. J. Cryptol. 1990, 3, 27–41.
2. Abdulsalam, A.A. Keystream Generator Based On Simulated Annealing. J. Appl. Comput. Sci. Math. 2011, 5, 48–53.
3. Sreelaja, N.K.; Pai, G.A.V. Stream cipher for binary image encryption using Ant Colony Optimization based key generation. Appl. Soft Comput. 2012, 12, 2879–2895.
4. Wang, Y.; Zhang, Z.; Zhang, L.Y.; Feng, J.; Gao, J.; Lei, P. A genetic algorithm for constructing bijective substitution boxes with high nonlinearity. Inf. Sci. 2020, 523, 152–166.
5. Pathak, V.K.; Singh, A.K.; Singh, R.; Chaudhary, H. A modified algorithm of Particle Swarm Optimization for form error evaluation. Tn-Tech. Mess. 2017, 84, 272–292.
6. Tian, Y.; Lu, Z. Chaotic S-box: Intertwining Logistic Map and Bacterial Foraging Optimization. Math. Problms Eng. 2017, doi:10.1155/2017/6969312.
7. Khan, M.F.; Ahmed, A.; Saleem, K. A Novel Cryptographic Substitution Box Design Using Gaussian Distribution. IEEE Access 2019, 7, 15999–16007.
8. Khan, M.A.; Ali, A.; Jeoti, V.; Manzoor, S. A Chaos-Based Substitution Box (S-Box) Design with Improved Differential Approximation Probability (DP). Iran. J. Sci. Technol. Trans. Electr. Eng. 2018, 42, 219–238.
9. Isa, H.; Jamil, N.; Zaba, M.R. Construction of Cryptographically Strong S-Boxes Inspired by Bee Waggle Dance. New Gener. Comput. 2016, 34, 221–228.
10. Rafiq, A.; Khan, M. Construction of new S-boxes based on triangle groups and its applications in copyright protection. Multimed. Tools Appl. 2019, 78, 15527–15544.
11. Ahmad, M.; Doja, M.N.; Sufyan Beg, M.M. ABC Optimization Based Construction of Strong Substitution-Boxes. Wirel. Pers. Commun. 2018, 101, 1715–1729.
12. Zahid, A.H.; Arshad, M.J. An Innovative Design of Substitution-Boxes Using Cubic Polynomial Mapping. Symmetry 2019, 11, 437.
13. Shahzad, I.; Mushlaq, Q.; Razaq, A. Construction of New S-Box Using Action of Quotient of the Modular Group for Multimedia Security. Secur. Commun. Netw. 2019, 2019, 1–10.
14. Belazi, A.; El-Latif, A.A.A. A simple yet efficient s-box method based on chaotic sine map. Optik 2017, 130, 1438–1444.
15. Çavuşoğlu, Ü.; Zengin, A.; Pehlivan, I.; Kaçar, S. A novel approach for strong s-box generation algorithm design based on chaotic scaled zhongtang system. *Nonlinear Dyn.* 2017, 87, 1081–1094.

16. Guesmi, R.; Farah, M.A.; Kachouri, A.; Samet, M. A novel design of Chaos based S-Boxes using genetic algorithm techniques. In Proceedings of the 2014 IEEE/ACS 11th International Conference on Computer Systems and Applications (AICCSA), Doha, Qatar, 10–13 November 2014; pp. 678–684.

17. Ilvanov, G.; Nikolov, N.; Nikova, S. Reversed genetic algorithms for generation of bijectives-boxes with good cryptographic properties. *Cryptogr. Commun.* 2016, 8, 247–276.

18. Wang, Y.; Wong, K.W.; Li, C.; Li, Y. A novel method to design S-box based on chaotic map and genetic algorithm. *Phys. Lett. A* 2012, 376, 827–833.

19. Cui, L.; Cao, Y. A new S-box structure named affine power-affine. *Int. J. Innov. Comput. Inf. Control* 2007, 3, 751–759.

20. Wang, Y.; Lei, P.; Wong, K.-W. A method for constructing bijective S-box with high nonlinearity based on chaos and optimization. *Int. J. Bifurc. Chaos* 2015, 25, 1550127.

21. Belazi, A.; Abd El-Latif, A.A.; Diaconu, A.V.; Rouma, R.; Belghith, S. Chaos-based partial image encryption scheme based on linear fractional and lifting wavelet transforms. *Opt. Lasers Eng.* 2017, 88, 37–50.

22. Lambic, D. A new discrete-space chaotic map based on the multiplication of integer numbers and its application in S-box design. *Nonlinear Dyn.* 2020, 1–13, doi:10.1007/s11071-020-05503-y.

23. Wang, X.; Çavuşoğlu, Ü.; Kacar, S.; Akgul, A.; Pham, V.T.; Jafari, S.; Alsaaedi, F.E.; Nguyen, X.Q. S-box based image encryption application using a chaotic system without equilibrium. *Appl. Sci.* 2019, 9, 781.

24. Ozkaynak, F. Construction of robust substitution boxes based on chaotic systems. *Neural Comput. Appl.* 2019, 31, 3317–3326.

25. Ullah, A.; Jamal, S.S.; Shah, T. A novel scheme for image encryption using substitution box and chaotic system. *Nonlinear Dyn.* 2018, 91, 359–370.

26. Daemen, J.; Rijmen, V. *Design of Rijndael: AES—Advanced Encryption Standard*; Springer Science and Business Media: Berlin, Germany, 2002.

**Publisher’s Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).