Multimedia Tools and Applications (2024) 83:64899–64918
https://doi.org/10.1007/s11042-023-17720-9

A novel efficient S-box design algorithm based on a new chaotic map and permutation

Mingjie Zhao[1,2] · Zheng Yuan[1,2,3] · Lixiang Li[1] · Xiu-Bo Chen[1]

Received: 13 June 2022 / Revised: 19 October 2023 / Accepted: 21 November 2023 /
Published online: 18 January 2024

Abstract
The substitution box (S-box) is one of the extremely important components in the design of block cipher. An excellent S-box is necessary for the block cipher algorithm, and its cipher strength directly affects the security of the cipher algorithm. The differential uniformity of the S-box generated by the chaotic system is 10 or 12, which cannot effectively resist differential cryptanalysis. Aiming at the high differential uniformity of the S-box constructed by the chaotic system, a novel efficient S-box construction scheme based on a new chaotic map and permutation is proposed in this paper. In this scheme, the chaotic matrix is generated by a new chaotic map, and then is replaced by permutation sequences to generate S-boxes. Comparative analysis shows that the generated S-boxes have high nonlinearity, low differential uniformity, and satisfy SAC and BIC criteria, which can improve the ability of the algorithm to resist differential cipher attacks and linear cryptographic analysis.

Keywords S-box · Chaotic system · Permutation · Differential uniformity

1 Introduction

In many block cipher algorithms, the S-box is the primary nonlinear part, and its quality directly determines the performance and security level of the encryption scheme. Therefore, the construction of the S-box with excellent performance has become an important research topic, which has attracted the attention of numerous scholars [1]. There are various methods for constructing an S-box, such as random generation construction [2], heuristic method [3], mathematical construction method [4], cellular automata [5] and other methods.

The chaotic system has superior characteristics of uncertainty [6, 7], ergodicity [8, 9], sensitivity [10, 11] and pseudo-randomness, which make it suitable for constructing nonlinear S-box in block ciphers. In a chaotic system, two initial values with an extremely small difference will also produce completely different chaotic sequences, and the state of the chaotic system at a certain moment is unpredictable. Therefore, the S-box constructed by the

✉ Zheng Yuan
zyuan@tsinghua.edu.cn

Extended author information available on the last page of the article

chaotic system has favorable confusion and nonlinear characteristics. A lot of schemes have used the chaotic system to construct S-box [12–22].

To improve the nonlinearity and confusion performance of the S-box, traditional one-dimensional (1D) chaotic maps have been used in lots of schemes to construct the S-box [23–31]. However, the range of chaotic parameters of 1D chaotic maps is small. Attackers can use signal estimation technology to predict the chaotic values of 1D chaotic systems, which are vulnerable to security threats [32]. Therefore, many schemes use more complex and higher-dimensional chaotic maps or combined chaotic maps to construct S-box [33–45]. Yan et al. [33] put forward a novel S-box dynamic design based on CLS produced by LNECS, and simulation analysis shows that S-box can resist well-known attacks and cryptanalysis. Ahmad et al. [34] present a new S-box evolution scheme, which used the dynamics of the fractional-order time-delayed two-state Hopfield neural network system. It has been shown that time delay plays a vital role in increasing the nonlinearity of the S-box. Zhu et al. [37] used the combination of Logistic and Tent maps to obtain chaotic sequences and get static S-boxes, and then the static S-boxes were used as seeds to dynamically generate new S-boxes under the control of the mixed chaotic system and fitness function. Wang et al. [42] proposed a chaotic S-box construction method based on a memorable simulated annealing algorithm (MSAA). Using the nonlinearity and randomness of dynamic iteration of digital cascade chaotic maps to generate chaotic S-box sets. Turk et al. [44] proposed a method to create dynamic, reliable and fast S-boxes based on the Tent map. The simulation shows that the generated S-boxes meet the design criteria of S-boxes. Zheng et al. [45] proposed a dynamic S-box encryption algorithm. By using the characteristics of the chaos map, the chaotic idea is integrated into the construction of the S-box, and a new dynamic S-box generation method is obtained.

The inherent characteristics of the chaotic system provide a good foundation for constructing S-box. The constructed S-box improves its nonlinearity to a certain extent and can obtain a higher security level. However, the S-boxes generated by chaotic systems still have high differential uniformity, which cannot achieve all expected performance indicators [23]. Therefore, Khan et al. [46] proposed constructing an S-box based on a traditional logistic map to reduce differential uniformity, but the parameter range of the chaotic system they used is small.

For the problem of high differential uniformity of S-boxes constructed by chaotic systems and the small parameter range of traditional one-dimensional chaotic maps, this paper proposes a new S-box construction scheme based on a new logistic map and permutation to improve the difference uniformity. We use a new logistic map to generate a chaotic matrix, which is replaced by the permutation sequences to generate S-boxes. Through performance comparison analysis, it is proved that this new logistic map has good chaotic performance, large enough parameter space, strong randomness, and high sensitivity. And the S-boxes constructed in this paper have strong nonlinearity and low differential uniformity, which can improve the ability of the algorithm to resist differential attacks and linear analysis. The innovations of this work are concluded as follows:

(1) A new logistic map is adopted, which has superior chaotic performance, large parameter space, and strong sensitivity. It can effectively resist brute force attacks and is more suitable for cryptography applications.

(2) A simple and effective S-box construction method using a new logistic map and permutation sequence is proposed to improve the difference uniformity, which can enhance the construction efficiency of the S-box.

(3) Compared with other S-boxes, the proposed S-boxes have higher nonlinearity and lower differential uniformity, indicating that the proposed S-boxes have obvious advantages in resisting the attacks of differential cryptanalysis and linear cryptanalysis.

The rest of this paper is organized as follows. Section 2 presents the basics used in this paper. The new logistic map is introduced and analyzed in Section 3. In Section 4, we introduce the S-box generation method. Section 5 presents some basic security analyses of proposing S-boxes. The conclusion is presented in Section 6.

2 Basic knowledge

2.1 S-box

The S-box is a nonlinear transformation. An S-box with an n-bit input mapped to m-bit output is called an $n \times m$ S-box, also commonly known as multi-output Boolean functions or vector functions. The $n \times m$ S-box is defined as

$$S(x) = (S(x_1), \ldots, S(x_m)) : GF(2)^n \rightarrow GF(2)^m \tag{1}$$

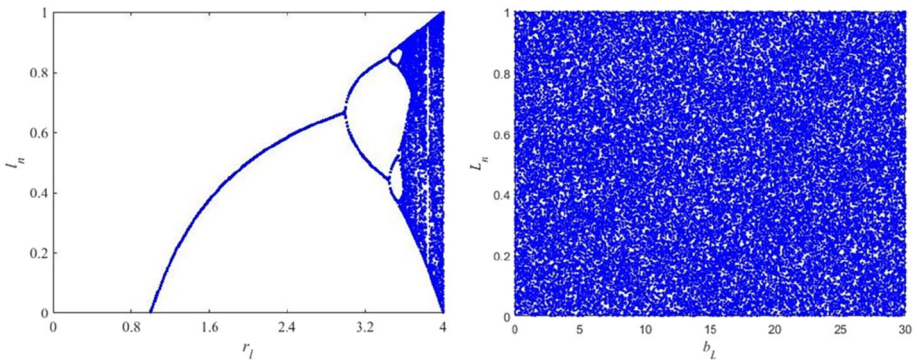
There are six common sizes of S-boxes, namely 4×4, 8×8, 8×32, 11×8, 13×8, 8×32. The most popular one is the 8×8 S-box.

2.2 Logistic map

Traditional 1D chaotic maps are simple in structure, easy to implement, and have excellent chaotic performance. The dynamical equation of the logistic map [47] is as follows

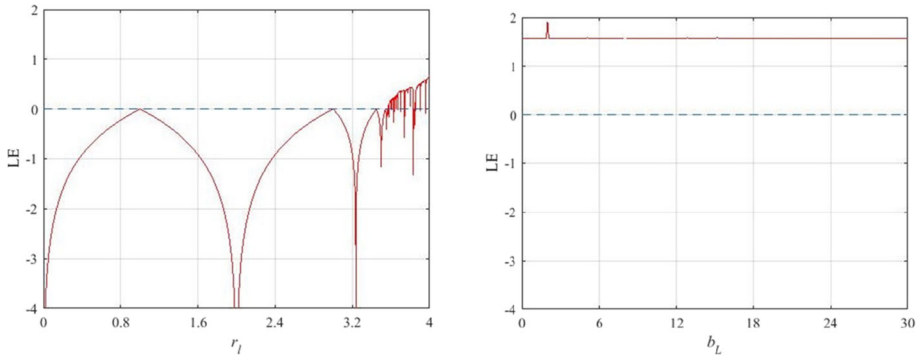
$$l_{n+1} = r_l \times l_n \times (1 - l_n) \tag{2}$$

where l_n is the logistic sequence, and $l_n \in [0, 1]$. r_l is the chaotic parameter of the logistic map. When $r_l \in [3.5699465, 4]$, this system is chaotic. The bifurcation diagram, Lyapunov exponential diagram, and iterative function diagram of the logistic map are shown in Figs.1(a), 2(a), and 3(a), respectively.



(a) traditional logistic map **(b)** new logistic map

Fig. 1 Bifurcation diagram



(a) traditional logistic map **(b)** new logistic map

Fig. 2 LEs analysis

2.3 Permutation

Let X be a non-empty set, and σ_X be the set of all bijections from X to itself. On the composition of σ_X map, $\circ$ constitutes a group, usually called σ_X as a symmetric group on X, the element in σ_X as the permutation on X, and the subgroup of σ_X as a permutation group.

Let $\{x_0, x_1, \ldots, x_k\} \subseteq \{0, 1, \ldots, n-1\}$, if the permutation f on $\{0, 1, \ldots, n-1\}$ is defined as

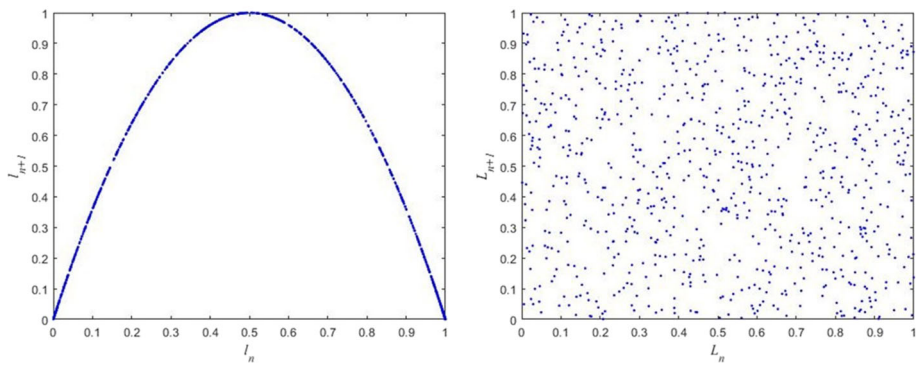
$$f(x) = \begin{cases} x_{i+1} \quad mod \quad k, & x \in \{x_0, x_1, \ldots, x_k\} \\ x_i, & x \notin \{x_0, x_1, \ldots, x_k\} \end{cases} \tag{3}$$

3 New logistic map

3.1 Definition

The new logistic map used in this paper is defined by [48]

$$L_{n+1} = \mathrm{mod}(b_L \times L_n \times (1 - L_n) - {L_n}^2/3) \times 2^{k_L}, 1) \tag{4}$$



(a) traditional logistic map **(b)** new logistic map

Fig. 3 Iterative function diagram

where L_n is a new logistic sequence, and $L_n \in [0, 1]$. b_L is the chaotic parameter. k_L is the number of iterations. And mod is the modulo function, which can control the value of the chaotic sequence within the range [0, 1]. 2^{k_L} is an adjustment function and iteratively removes transient effects by adjusting 2^{k_L}. When $k_L \in [7, 21]$, and $b_L \in [0, 30]$, this system is completely chaotic.

3.2 Performance analysis

Matlab 2018a is used to test the performance of the new logistic map. The chaotic parameters of the new logistic map are set to $b_L \in [0, 30]$, and $k_L = 16$. And the initial values of the chaotic sequence of the traditional and new logistic maps are all set to 0.3.

3.2.1 Bifurcation diagram

The bifurcation diagram gives the idea of the stability boundary and quantifies the sensitive dependence on the control parameters. It is obtained by giving the initial value of bifurcation parameters, calculating the iterative sequence of the system under the corresponding parameters, and drawing the chaotic sequence.

We intuitively analyze the performance of the traditional and new logistic maps through bifurcation diagrams, as shown in Fig. 1. From Fig. 1, we know the new logistic map greatly increases the chaotic parameter range of the traditional logistic map, and it is in a completely chaotic state within the parameter range, so the chaotic performance is superior.

3.2.2 Lyapunov exponent (LE)

LE is used to measure the sensitivity of the chaotic system to its initial conditions. It points out that at least one measurement value of the chaotic system is positive. When LE is negative or 0, the region is a periodic region, which will interrupt the chaotic region, and the system is in a stable state. When LE is positive, the larger LE is, the more sensitive the chaotic system is to the initial value.

Figure 2 shows the LEs analysis of the traditional and new logistic maps, the LEs of the new logistic map are all positive, which proves its chaotic superior performance.

3.2.3 Iterative function diagram

For an iterative dynamic system $x_{n+1} = f(x_n)$, its iterative function diagram can be expressed as: when x_n is the input, the output is x_{n+1}.

Figure 3 shows the iterative function diagrams of the traditional and new logistic maps. The iterative function diagram of the traditional logistic map is a regular curve, as shown in Fig. 3(a). The attacker can guess that it is a traditional logistic map from the curve in the iterative function diagram. However, the iterative function diagram of the new logistic map is randomly generated points without regularity. If the attacker gets the iterative function diagram in Fig. 3(b), the attacker cannot guess what kind of chaotic map it is. Therefore, the new logistic map has strong randomness and is difficult to predict.

3.2.4 Sensitivity analysis

The chaotic system is characterized by being sensitive to initial values. We judge the performance of the new logistic map by sensitivity analysis. We first set the initial value of the new logistic map to 0.5, and then reset the initial value to 0.500000000000001.

Figure 4 shows the sensitivity analysis of the new logistic map. The blue dotted line is the initial value of 0.5, and the double red line is the initial value of 0.500000000000001, although the initial value only differs by 1^{-16}, the chaotic curves of these two maps are quite different, which proves that the new logistic map has a strong sensitivity to the initial value.

4 Generate S-box

S-box plays a confusing role in the block cipher, and the sensitivity of chaotic to initial conditions is closely related to the confusion and diffusion characteristics of traditional cryptosystems [37]. Using a superior chaotic system can generate an S-box with excellent performance, which can improve the S-box capability of the algorithm.

In this paper, the new logistic map and permutation are used to generate 8-bit S-boxes. The new logistic map has excellent performance, which can improve the performance of the S-box in the algorithm. The steps of generating S-boxes are described as follows:

Step 1 Randomly select three numbers as the value of the chaotic parameter b_L, k_L, and the initial value L_0 of the new logistic map.

Step 2 Iterate the new logistic map for N times, delete the first 1500 values of the sequence and transform the selected numeric sequence L. Transform floating-point values of the numeric sequence L into integer sequences of 0∼255 by using (5) to obtain the chaotic sequence $(Z_1, Z_2, \ldots, Z_{N-1500})$. The specific generation of the chaotic sequence is shown in Algorithm 1.

$$Z_i = (floor(L_{i+1500} \times 10^{14})) \bmod 256 \tag{5}$$

where $i = 1, 2, \ldots, N - 1500$.

Step 3 Arranging chaotic sequences to construct a 16×16 chaotic matrix. If there are duplicate values, only the first numerical value will be kept, and then the same value that appears later will be deleted. Finally, the 16×16 chaotic matrix is obtained, denoted by

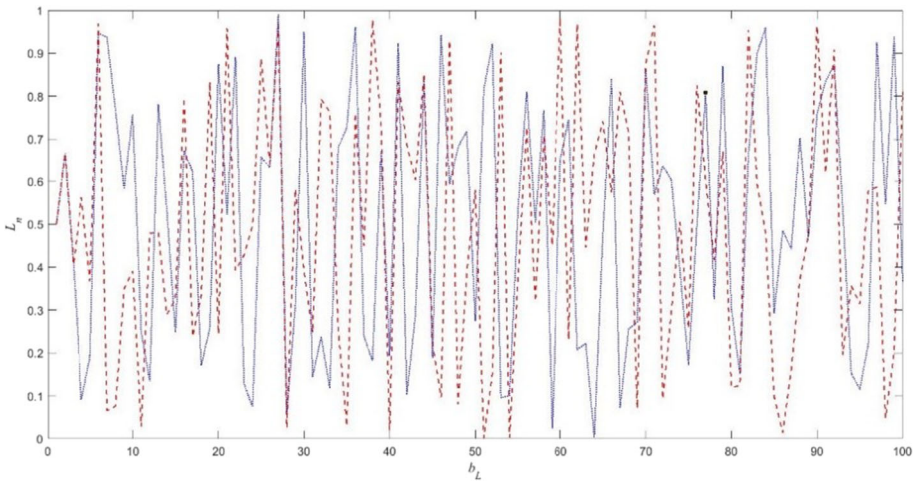


Fig. 4 Sensitivity analysis. The initial value of 0.5 is the blue dotted line and the initial value of 0.500000000000001 is the double red line

$(Y_0, Y_1, \ldots, Y_{255})$. Duplicate values are shown in (6). The concrete construction method is shown in Algorithm 2.

$$Y = unique\left(Z, ' stable'\right) \tag{6}$$

Step 4 According to the given permutation sequences, the chaotic matrix is permuted to generate the S-boxes. The construction method is shown in Algorithm 3. The flow chart of generating the S-box is shown in Fig. 5.

In this scheme, the initial parameters of the chaotic sequence are $b_L = 8$, $k_L = 16$, and $L_0 = 0.321$ to generated a 16×16 chaotic matrix, as shown in Table 1.

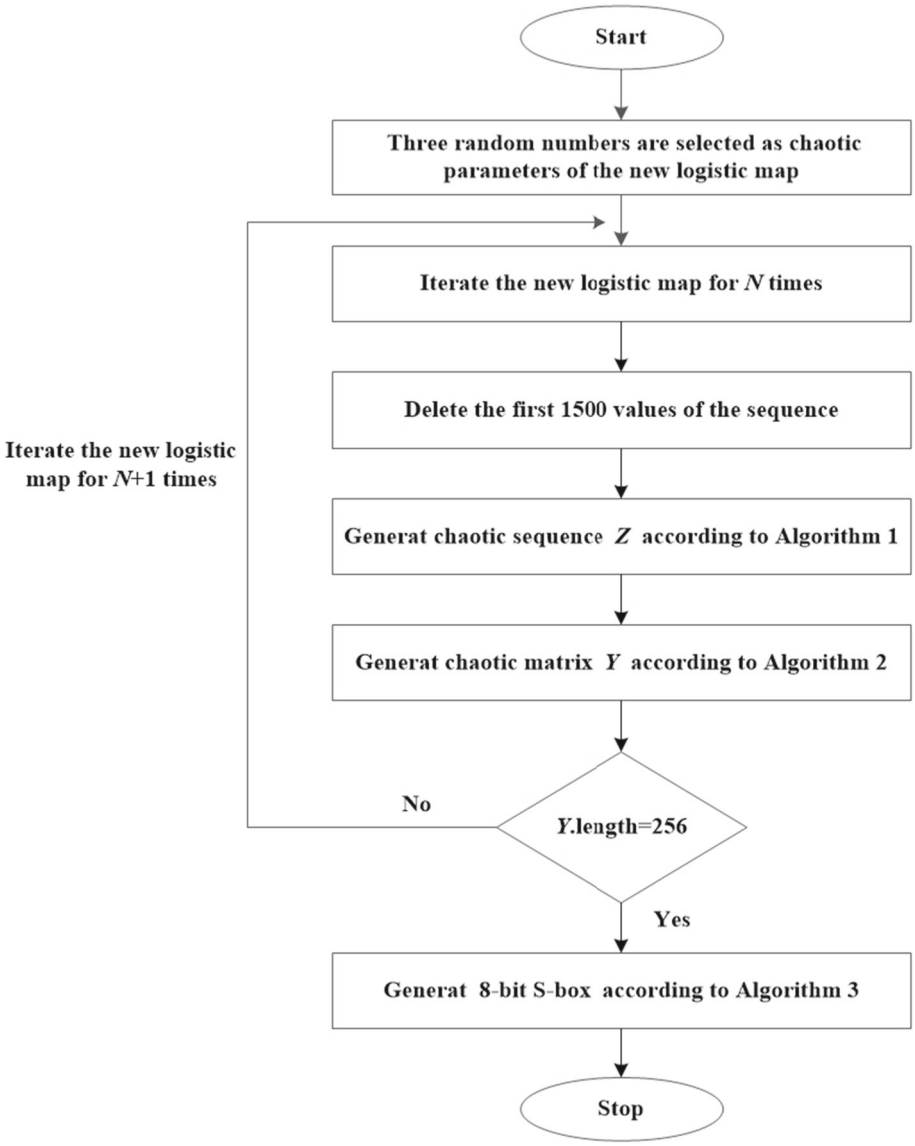


Fig. 5 Generate S-box flow chart

Table 1 Chaotic matrix

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	140	146	180	83	143	91	182	249	14	253	103	172	116	145	205	97
1	80	31	109	209	68	77	95	163	7	218	222	215	238	217	43	28
2	20	234	5	221	212	6	64	30	104	151	207	73	148	171	159	51
3	11	231	60	154	181	89	192	63	195	44	197	247	85	240	187	107
4	138	39	239	216	210	47	50	4	127	42	24	106	167	199	186	235
5	112	223	226	188	200	65	74	61	176	13	111	142	250	133	12	168
6	184	27	1	33	208	206	135	152	22	189	113	10	124	144	153	236
7	17	118	120	94	219	53	57	166	3	141	130	70	229	190	204	149
8	72	232	8	227	67	46	137	165	32	93	241	139	76	66	196	169
9	54	178	213	183	2	164	230	191	25	16	132	245	108	48	100	193
10	224	52	99	49	71	102	246	228	214	179	123	174	78	150	134	211
11	84	110	194	158	225	75	243	198	122	79	9	119	254	56	19	96
12	18	36	157	101	129	177	128	23	201	21	98	58	161	244	41	92
13	162	15	237	87	115	55	114	233	252	125	88	26	251	248	175	59
14	156	35	160	155	38	242	255	185	202	37	220	0	82	69	90	105
15	34	173	117	131	203	147	136	40	121	126	62	45	81	29	170	86

Algorithm 1 Generate chaotic sequence

Input: b_L, k_L and L_0
Output: $(Z_1, Z_2, \ldots, Z_{N-1500})$
1: Obtain the initial conditions of the new chaotic map
2: For $i = 0$ to N
2.1 Use the new logistic map to generate sequence
2.2 Perform N
3: To eliminate the transient effect, delete the first 1500 values of the sequence and obtain that the numeric sequence L // L is an array of floating-point values
4: For $j = 0$ to $L.length$
4.1 Convert each floating-point number of L into decimal system code and store it in Z. This process is completed by Eq. (5)
4.2 Perform j

Algorithm 2 Chaotic matrix

Input: $(Z_1, Z_2, \ldots, Z_{N-1500})$
Output: chaotic matrix
1: Traversal $(Z_1, Z_2, \ldots, Z_{N-1500})$
2: Delete all repeated values in the Z array, and obtain the first occurrence value of each value as a sequence Y, as shown in Eq. (6)
3: Select 256 non-repeating values of Y to construct a 16×16 matrix. If $Y.length < 256$, iterate the chaotic system $N + 1$ times
4: Update the sequence Y until $Y.length = 256$, and obtain a 16×16 chaotic matrix

Algorithm 3 Generate S-box

Input: chaotic matrix, permutation sequence
Output: the proposed 8-bit S-box
1: Iterate over all the values of the 16×16 chaotic matrix
2: According to the permutation sequence, assign the value of the next sequence to this sequence
3: Repeating step (2) until the S-box is completed

Two permutation sequences are given in this paper, permutation sequence 1 and permutation sequence 2, which are used to generate S-box$_1$ and S-box$_2$, as shown in Tables 2 and 3, respectively.

Permutation sequence 1:

(0, 130, 230, 9, 121, 151, 100, 112, 104, 29, 178, 131, 153, 132, 4, 18, 127, 251, 34, 115, 88, 120, 150, 66, 105, 124, 110, 215, 232, 55, 14, 33, 48, 212, 217, 92, 117, 134, 23, 160, 42, 246, 77, 129, 182, 243, 79, 12, 255, 198, 108, 39, 191, 168, 216, 53, 190, 169, 136, 194, 27, 193, 45, 123, 141, 83, 174, 118, 204, 98, 60, 76, 35, 26, 207, 10, 143, 25, 43, 109, 227, 106, 44, 28, 175, 125, 85, 52, 111, 102, 221, 241, 145, 46, 32, 116, 89, 128, 68, 133, 38, 196, 170, 5, 213, 200, 63, 249, 74, 183, 186, 226, 67, 181, 57, 22, 70, 114, 176, 154, 93, 15, 205, 157, 237, 161, 239, 218) (1, 171, 197, 187, 238, 138, 155, 47, 96, 81, 253, 24, 37, 185, 97, 247, 71, 78, 17, 167, 3, 195, 188, 223, 208, 54, 94, 231, 147, 16, 248, 173, 75, 2, 184, 177, 69, 233, 144, 95, 211, 224, 201, 229, 245, 225, 163, 86, 250, 51, 158, 72, 119, 61, 142, 99, 101, 103, 180, 219, 164, 210, 149, 254, 82, 159, 148, 220, 36, 234, 126, 209, 214, 122, 40, 73, 31, 11, 172, 244, 56, 41, 240, 242, 65, 139, 189, 236, 135, 252, 192, 113, 62, 87, 80, 179, 206, 30, 202, 58, 199, 162, 140, 6, 84, 165, 235, 59, 20, 107, 91, 166, 19, 13, 156, 7, 49, 152, 21, 203, 90, 8, 146, 50, 64, 222, 228, 137).

Permutation sequence 2:

(0, 242, 166, 250, 29, 65, 18, 113, 75, 136, 15, 50, 153, 119, 173, 126, 6, 198, 67, 93, 239, 46, 158, 151, 184, 196, 128, 200, 189, 108, 77, 3, 238, 167, 28, 105, 237, 204, 74, 154, 25, 30, 24, 244, 133, 64, 72, 41, 246, 218, 19, 202, 247, 205, 97, 145, 82, 107, 69, 228, 90, 14, 170, 220, 81, 99, 95, 130, 168, 219, 211, 53, 80, 230, 54, 109, 148, 38, 26, 37, 100, 201, 144, 175, 178, 7, 103, 137, 71, 87, 254, 63, 101, 47, 223, 122, 216, 225, 150, 140, 207, 31, 89, 10, 62, 58, 116, 180, 149, 57, 157, 76, 11, 169, 172, 83, 217, 22, 60, 48, 141, 88, 8, 162, 132, 115, 203, 193) (1, 2, 114, 13, 251, 5, 86, 243, 117, 177, 174, 229, 118, 73, 104, 213, 210, 106, 188, 227, 135, 78, 127, 147, 23, 35, 59, 236, 20, 42, 187, 231, 91, 111, 125, 155, 199, 235, 232, 40, 159, 123, 70, 226, 84, 94, 51, 68, 45, 163, 96, 249, 146, 27, 66, 110, 171, 142,

Table 2 The S-box$_1$ generated by proposed algorithm

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	8	174	122	101	109	55	200	231	213	141	169	78	86	108	234	244
1	121	228	149	145	10	58	50	224	6	73	92	36	211	194	98	172
2	219	11	94	222	220	79	129	96	42	34	136	144	238	70	20	184
3	115	25	138	100	236	19	12	205	151	95	23	68	167	196	61	126
4	175	139	189	75	46	37	120	186	166	28	198	180	221	232	31	116
5	158	29	193	134	102	181	62	112	3	72	14	246	53	97	185	87
6	223	40	85	206	17	152	248	225	217	229	148	142	30	155	233	135
7	22	187	84	176	13	137	161	240	230	191	104	66	153	65	15	45
8	210	243	255	16	143	64	163	81	157	146	245	56	182	188	33	218
9	168	159	60	80	251	170	239	208	77	67	133	51	249	69	127	2
10	207	105	76	74	237	0	209	83	252	32	91	177	203	106	57	190
11	132	47	227	41	26	44	131	9	110	27	160	90	59	82	179	214
12	118	171	215	254	123	119	124	99	107	242	197	111	1	48	43	103
13	192	114	164	156	125	201	130	202	89	250	140	71	212	173	38	162
14	21	49	216	113	93	147	253	183	63	54	204	247	165	52	241	88
15	117	178	39	235	195	35	199	4	150	24	154	5	18	7	226	128

Table 3 The S-box$_2$ generated by proposed algorithm

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	117	180	120	90	169	74	128	152	99	30	187	179	148	45	123	60
1	232	243	118	98	207	31	85	221	203	43	6	239	189	39	7	13
2	212	116	115	247	34	208	222	157	193	136	119	141	220	49	100	59
3	66	250	16	210	80	112	144	173	19	48	219	82	11	73	197	206
4	127	109	153	133	171	38	160	61	151	22	132	32	172	83	149	156
5	255	33	10	125	12	139	131	170	14	103	205	236	9	105	154	8
6	126	178	25	168	21	51	75	93	55	69	254	47	199	2	174	190
7	253	106	145	58	225	110	42	150	65	147	252	50	162	245	182	183
8	201	241	214	158	94	138	102	186	97	4	121	77	92	176	175	5
9	211	226	215	163	64	44	76	122	37	166	218	23	15	167	191	70
10	143	177	67	184	195	86	62	238	26	78	251	196	188	204	242	194
11	3	134	249	101	164	96	224	71	129	227	233	185	155	124	235	52
12	114	140	41	18	72	108	216	0	56	54	40	36	24	27	135	28
13	231	63	113	89	137	237	29	1	35	95	209	87	223	84	229	130
14	181	230	200	165	111	57	192	142	104	248	17	202	68	161	228	159
15	234	240	246	53	46	81	88	244	20	213	217	91	79	146	107	198

222, 124, 208, 49, 92, 186, 215, 98, 152, 233, 221, 176, 120, 85, 139, 21, 17, 182, 160, 4, 143, 34, 212, 134, 165, 255, 183, 164, 56, 190, 79, 224, 52, 16, 129, 138, 248, 32, 36, 240, 33, 12, 44, 234, 112, 9, 39, 194, 206, 102, 181, 191, 161, 197, 156, 209, 55, 241, 61, 43, 121, 245, 252, 185, 131, 179, 195, 192, 214, 253).

Take generating a 4-bit S-box as an example:

1) The initial parameters of the chaotic sequence are $b_L = 8$, $k_L = 16$, and $L_0 = 0.321$ to generated a chaotic matrix.

2) Iterate the new logistic map 150 times, delete the first 50 values of the sequence and transform the selected numeric sequence L. Transform floating-point values of the numeric sequence L into integer sequences of 0∼15 to obtain the chaotic sequence Z.

3) Arranging chaotic sequences to construct a 4×4 chaotic matrix Y. If there are duplicate values, only the first numerical value will be kept, and then the same value that appears later will be deleted. The chaotic matrix is Y = [11 6 9 5 13 10 4 7 12 15 14 0 3 8 1 2].

4) Permutation sequence is (4, 12, 9, 10, 14, 11, 5, 3, 13, 8, 6, 0, 15, 2, 7, 1).

It can be seen that the value 11 of the 0-th position of the initial S-box is replaced by the value 3 of the 12-th position, the value 6 of the 1-th position is replaced by the value 15 of the 9-th position, and so on, the value 2 of the 15-th position is replaced by the value 13 of the 4-th position, forming a cycle. Thus, the initial S-box Y = [11 6 9 5 13 10 4 7 12 15 14 0 3 8 1 2] is replaced by the permutation sequence (4, 12, 9, 10, 14, 11, 5, 3, 13, 8, 6, 0, 15, 2, 7, 1), and the new S-box generated after permutation is S-box= [3 15 14 1 0 10 5 8 12 4 11 2 9 7 6 13].

5 S-box performance analysis

To verify the performance of S-boxes constructed in this paper, this section will analyze the performance of S-boxes from bijectivity, nonlinearity, linear approximation probability, differential uniformity, strict avalanche criterion and bit independence criteria.

5.1 Bijectivity

Jakimoski et.al [49] proposed a test method for bijectivity. When $m = n$, the S-box satisfies the bijectivity if and only if the sum of the linear operations of each component Boolean function f_i is 2^{n-1}, that is

$$\omega t(\sum_{i=1}^{n} a_i f_i) = 2^{n-1} \tag{7}$$

where $\omega t()$ is the Hamming weight, $a_i \in [0, 1]$, and $(a_1, a_2, \ldots a_n) \neq (0, 0, \ldots, 0)$. In other words, if (7) holds, each f_i of the S-box is 0/1 balanced, and the S-box satisfies the bijective property.

According to the sufficient and necessary conditions for judging that the S-box satisfies the bijective characteristics, the average linear operation sum of the component Boolean functions of two 8-bit S-boxes constructed in this paper are 128, and each of the S-box is 0/1 balanced, which proves that S-boxes satisfy the bijectivity characteristics.

5.2 Nonlinearity

The nonlinearity is used to measure the ability of a cryptographic function to resist linear cryptanalysis. Nonlinearity is defined as

$f(x) : F_2^n \to F_2$ is an n-element Boolean function, and the nonlinearity of $f(x)$ is [50]

$$N_f = \min_{l \in L_n} d_H(f, l) \tag{8}$$

where L_n is the set of all-element linear and affine functions. $d_H(f, l)$ represents the Hamming distance between f and l.

If $S(x) = (f_1(x), \ldots, f_m(x)) : F_2^n \to F_2^m$ is a multi-output function, $f(x)$ is generally converted into Walsh spectrum when calculating the nonlinearity, then

$$S_{\langle f \rangle}(\omega) = \sum_{\omega \in GF(2^n)} (-1)^{f(x) \oplus x \cdot \omega} \tag{9}$$

where $\omega \in GF(2^n)$, "." represents the dot product operation, and $x \cdot \omega = x_1 \cdot \omega_1 \oplus x_2 \cdot \omega_2 \oplus \ldots \oplus x_n \cdot \omega_n$.

The nonlinearity expressed by the Walsh spectrum is

$$N_f = 2^{n-1}(1 - 2^{-n} \max_{\omega \in GF(2^n)} |S_{\langle f \rangle}(\omega)|) \tag{10}$$

where N_f is nonlinearity, and the greater its value, the stronger its ability to resist linear cryptanalysis.

To effectively resist linear attacks, the nonlinearity of a safe and effective S-box must be sufficiently large. Table 4 shows the nonlinearity of eight output Boolean functions of S-box$_1$ and S-box$_2$. The minimum nonlinearity of S-box$_1$ and S-box$_2$ is 108, the maximum nonlinearity is 112, and the nonlinearity of S-box$_1$ and S-box$_2$ is 104 and 106, respectively. Therefore, our S-boxes have good resistance to linear cryptanalysis.

5.3 Linear approximation probability

A secure cryptographic system should have strong confusion and diffusion effects. S-box with superior performance helps the cryptographic system to achieve strong confusion and

Table 4 Nonlinearity and LP

S-boxes	Nonlinearity of output Boolean functions								Nonlinearity	LP
	N_1	N_2	N_3	N_4	N_5	N_6	N_7	N_8		
S-box$_1$	112	112	108	110	108	112	112	110	104	0.093750
S-box$_2$	110	110	112	110	110	110	112	108	106	0.085938

diffusion effects through the nonlinear map between input and output data. Linear approximation probability (LP) is used to measure the resistance of the S-box to linear cryptanalysis, and its calculation formula is

$$LP = \max_{\alpha_x, \beta_x \neq 0} \left(\frac{\#\{x \in N \mid x \cdot \alpha_x = S(x) \cdot \beta_x\}}{2^n} - \frac{1}{2} \right) \tag{11}$$

where $N = \{0, 1, \ldots, 255\}$, α_x and β_x are the corresponding input and output masks ($\alpha_x \in N$, $\beta_x \in N$), and $\#\{x \in N \mid X\}$represents the number of x satisfying condition X.

With the lower LP of the S-box, it is proved that the greater the nonlinearity of the S-box, the stronger the ability of the function to resist linear attacks, and vice versa. Table 4 shows the LP between the proposed S-box$_1$ and S-box$_2$. It can be seen that the LP of S-box$_1$ and S-box$_2$ are 0.093750 and 0.085938, and the maximum value of LP is only 0.093750, so our proposed S-boxes have good resistance to linear cryptanalysis.

5.4 Differential uniformity

Since the introduction of differential cryptanalysis, many scholars have focused on designing a "strong" S-box to enhance the security of block ciphers based on the S-box. The ability of the S-box to resist differential cryptanalysis essentially depends on the difference distribution table and differential uniformity (DU). The difference distribution table is defined as

$2^n \times 2^m$-order matrix $\Lambda(S)$ is the difference distribution table of the S-box defined by [51]

$$\Lambda(S) = \begin{pmatrix} \lambda_{00} & \lambda_{01} & \cdots & \lambda_{0(2^m-1)} \\ \lambda_{10} & \lambda_{11} & \cdots & \lambda_{1(2^m-1)} \\ \vdots & \vdots & \vdots & \vdots \\ \lambda_{(2^n\text{-}1)0} & \lambda_{(2^n\text{-}1)1} & \cdots & \lambda_{(2^n\text{-}1)(2^m-1)} \end{pmatrix} \tag{12}$$

where $\lambda_{ij} = |\{x \in GF(2)^n | S(x) + S(x + \alpha_i) = \beta_j\}|$. $i = 0, 1, \ldots, 2^n - 1$. $j = 0, 1, \ldots, 2^m - 1$. α_i and β_j are the binary representation of i, j, respectively. Obviously

$$\lambda_{0j} = \begin{cases} 2^n & (j = 0) \\ 0 & (j \neq 0) \end{cases} \tag{13}$$

It must be pointed out that the distribution value of $\Lambda(S)$ is closely related to the operation on $GF(2)^n$, and only the case of bit-wise XOR is discussed here.

The key to differential analysis is to make use of special elements in the difference distribution table of the S-box. If the values of some elements are greater than those of other elements, the positions of these elements are particularly useful for differential attacks. Therefore, the concept of differential uniformity is introduced.

Suppose the difference distribution table of $n \times m$ S-box is $\Lambda(S) = \lambda_{ij}$, then the DU of S-box is called δ_S [51], and the formula is as follows

$$\begin{aligned}\delta_S &= \max\{\lambda_{ij} | i = 0, 1, \ldots, 2^n - 1, j = 0, 1, \ldots, 2^m - 1\} \\ &= \max_{\substack{0 \neq \alpha \in GF(2)^n \\ \beta \in GF(2)^m}} \delta_S(\alpha, \beta)\end{aligned} \tag{14}$$

where $\delta_S(\alpha, \beta) = |\{x \in GF(2)^n : S(x \oplus \alpha) + S(x) = \beta\}|$.

To resist differential cryptanalysis, the differential uniformity of the S-box should be as small as possible. The differential distribution tables of S-box$_1$ and S-box$_2$ are shown in Tables 5 and 6, respectively. The maximum value of the differential uniformity of our S-boxes is 6, which can effectively resist differential cryptanalysis.

5.5 Strict avalanche criterion

If a change in each input bit causes a change in each output bit with a probability of 0.5, then the map shown below satisfies the strict avalanche criterion (SAC)

$$S(x) = (f_1(x), f_2(x), \ldots, f_m(x)) : F_2^n \to F_2^m \tag{15}$$

Webster et.al [52] proposed to effectively judge whether the given $n \times m$ function f meets the SAC by constructing the correlation matrix. The value a_{ij} of each element of the correlation matrix is the correlation strength between the i-th bit of the ciphertext and the j-th bit of the plaintext. If each element of the correlation matrix has a value very close to 0.5, it can be said that the given function f satisfies the SAC.

The correlation matrices of the S-box$_1$ and S-box$_2$ are shown in Tables 7 and 8, respectively. It can be seen from the calculation that all elements of the correlation matrix are close to the theoretical value of 0.5.

5.6 Bit independence criteria

Webster et.al [52] proposed the bit independence criteria (BIC) of the S-box, as an important indicator for evaluating the performance of the S-box, which has good independence between bits and can effectively resist external attacks on the cryptographic system. The S-box can be regarded as a function $f : F_2^n \to F_2^n$, and its BIC calculation formula is as follows

$$BIC(f) = \max_{\substack{1 \le j,k \le n \\ j \neq k}} BIC(b_j, b_k) \tag{16}$$

where $BIC(b_j, b_k) = \max_{1 \le i \le n} |corr(b_j^{e_i}, b_k^{e_i'})|$.

Adams et.al [53] presented a method for measuring the independence of output bits. For a given Boolean function $f_j \oplus f_k$ of two output bits in an S-box, if it has a high nonlinearity and satisfies SAC as much as possible, it can be ensured that each output bit pair has a correlation coefficient close to 0 when any single input bit is inverted. Therefore, it can be proved whether S-box satisfies BIC by measuring whether any $f_j \oplus f_k$ of the S-box satisfies the nonlinear characteristic and SAC.

By calculating the nonlinearity and correlation matrix of $f_j \oplus f_k$ between any two output bits of the S-box, when each element of the $f_j \oplus f_k$ correlation matrix has a value very close

Table 5 Differential matrix for differential uniformity (DU) of S-box$_1$

4	4	4	4	4	4	6	4	4	4	4	4	4	6	4	4
4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
4	4	4	4	4	4	4	4	4	4	4	4	6	4	4	4
4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
4	4	6	4	4	4	4	4	4	4	4	6	4	4	4	4
4	4	4	4	4	6	4	4	4	4	4	6	4	4	4	6
4	4	4	6	4	4	4	4	4	4	4	4	4	4	4	4
4	4	4	4	4	4	6	4	4	4	4	4	6	4	4	4
4	4	4	4	4	6	4	6	4	4	4	4	4	4	4	4
4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
4	6	4	4	4	4	4	4	4	4	4	4	4	4	4	4
4	6	6	4	4	4	4	4	4	4	4	4	4	4	4	4
4	4	4	4	4	4	4	4	4	4	6	4	4	4	4	4
4	4	6	4	4	4	4	4	6	4	4	4	4	4	6	-

to 0.5, indicating that the $f_j \oplus f_k$ of the S-box satisfies the SAC. The nonlinearity of $f_j \oplus f_k$ is more than 100, which meets the nonlinear characteristics.

The BIC-SAC matrices of S-box$_1$ and S-box$_2$ are shown in Tables 9 and 10, and the BIC-Nonlinearity matrices are shown in Tables 11 and 12, respectively. All values of BIC-SAC are closed to 0.5, which satisfies the SAC. And all values of BIC-Nonlinearity are above 100, satisfying the nonlinear characteristics. Therefore, S-box$_1$ and S-box$_2$ have good independence between output bits.

Table 6 Differential matrix for differential uniformity (DU) of S-box$_2$

4	4	4	4	4	6	4	4	4	4	4	4	4	4	4	4
4	4	4	4	4	4	4	4	4	4	4	6	6	4	4	4
4	4	6	4	4	4	4	4	6	4	4	4	4	4	6	4
6	4	4	6	4	4	4	4	4	4	4	4	4	4	4	4
4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
4	4	4	4	4	4	4	6	4	6	4	4	4	6	4	4
4	4	4	4	6	4	4	4	4	4	4	6	4	4	4	4
6	4	4	4	4	4	6	4	4	4	4	4	4	4	4	4
4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
4	4	4	4	4	4	4	4	4	4	4	6	4	4	4	6
4	4	4	4	4	4	4	4	4	4	4	4	4	4	6	6
4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
4	4	4	4	4	4	4	4	4	4	6	4	4	4	4	4
4	4	4	6	4	4	4	6	4	4	4	6	4	4	4	-

Table 7 SAC matrix of S-box$_1$

0.4844	0.4844	0.5156	0.4375	0.5156	0.4531	0.5313	0.5000
0.5313	0.5156	0.5156	0.4688	0.5156	0.5000	0.5469	0.4531
0.4531	0.5469	0.4688	0.4688	0.4688	0.5156	0.5156	0.5000
0.5781	0.4844	0.5313	0.4531	0.5156	0.4688	0.5156	0.5469
0.4219	0.5313	0.5000	0.5156	0.5469	0.5625	0.4531	0.5469
0.4844	0.5156	0.4844	0.4219	0.5000	0.4844	0.5156	0.5781
0.4844	0.5469	0.5000	0. 4844	0.4844	0.4531	0.4844	0.4688
0.5156	0.5469	0.5469	0. 4844	0.5313	0.5781	0.4531	0.4688

Table 8 SAC matrix of S-box$_2$

0.4688	0.4375	0.5313	0.4375	0.5781	0.5625	0.5469	0.5469
0.4375	0.4844	0.4688	0.5469	0.4688	0.4688	0.4844	0.4844
0.4688	0.5156	0.4688	0.5156	0.5313	0.4844	0.5469	0.5469
0.5000	0.5156	0.4688	0.5313	0.5625	0.5313	0.4531	0.4844
0.4844	0.4688	0.4844	0.4375	0.5313	0.5313	0.5313	0.4844
0.4531	0.5469	0.5625	0.5156	0.4844	0.4844	0.4844	0.4688
0.4844	0.4844	0.4844	0.5469	0.5625	0.5313	0.5313	0.5000
0.5156	0.4844	0.5313	0.4844	0.4688	0.5000	0.5000	0.5313

Table 9 BIC-SAC matrix of S-box$_1$

0.5078	0.4980	0.5117	0.4922	0.4844	0.4844	0.5020	-
0.4961	0.5195	0.5000	0.5000	0.4824	0.4902	-	0.4902
0.5039	0.5176	0.5117	0.4863	0.5117	-	0.4980	0.4844
0.5234	0.5254	0.5176	0.5000	-	0.5098	0.5020	0.5176
0.5176	0.5020	0.5098	-	0.5000	0.5176	0.5254	0.5234
0.4844	0.4980	-	0.5117	0.4863	0.5117	0.5176	0.5039
0.4902	-	0.4902	0.4824	0.5000	0.5000	0.5195	0.4961
-	0.5020	0.4844	0.4844	0.4922	0.5117	0.4980	0.5078

Table 10 BIC-SAC matrix of S-box$_2$

0.4980	0.5078	0.5078	0.4922	0.5020	0.4941	0.4922	-
0.5078	0.4941	0.5020	0.5059	0.5098	0.5059	-	0.4941
0.5078	0.4941	0.5000	0.4941	0.4883	-	0.4961	0.4805
0.4902	0.4980	0.5195	0.4902	-	0.5078	0.4980	0.5020
0.5020	0.4980	0.5078	-	0.4902	0.5195	0.4980	0.4902
0.4805	0.4961	-	0.4883	0.4941	0.5000	0.4941	0.5078
0.4941	-	0.5059	0.5098	0.5059	0.5020	0.4941	0.5078
-	0.4922	0.4941	0.5020	0.4922	0.5078	0.5078	0.4980

Table 11 BIC-Nonlinearity matrix of S-box$_1$

110	110	110	108	112	110	110	-
110	110	112	110	110	110	-	110
110	110	110	110	110	-	108	108
110	110	110	108	-	106	110	110
110	110	106	-	108	110	110	110
108	108	-	110	110	110	110	110
110	-	110	110	110	112	110	110
-	110	110	112	108	110	110	110

5.7 Comparative analysis

To highlight the superior performance of the S-boxes proposed in this paper, the comparative analysis of the performance of S-box$_1$, S-box$_2$ and other S-boxes are compared and analyzed in terms of nonlinearity, linear approximation probability, differential uniformity, SAC and BIC, as shown in Table 13.

In terms of nonlinearity, we find that the minimum value of S-box$_1$ and S-box$_2$ is 104, which is the largest in all comparative literature. It can be seen that the S-boxes constructed in this paper have good nonlinearity and are resistant to linear analysis.

Based on the comparative analysis of the LP, it can be observed that the LP of the proposed S-boxes is the smallest among all S-boxes, indicating that they have the strongest ability to resist linear cryptanalysis.

From the comparative analysis of the DU, it can be seen that the largest value of differential uniformity of our S-boxes is 6, which is the smallest of all compared S-boxes, proving that the S-boxes proposed in this paper can resist differential cryptanalysis.

According to the comparative analysis of SAC, the average value of SAC in S-box$_1$ is 0.5015 and the average value of SAC in S-box$_2$ is 0.5027. The average values of S-box$_1$ and S-box$_2$ are close to the ideal value of 0.5, indicating that S-box$_1$ and S-box$_2$ conform to SAC characteristics.

As can be seen from the comparative analysis of BIC, the BIC-SAC values of the S-box$_1$ and S-box$_2$ are all close to the ideal value of 0.5. The BIC-Nonlinearity values of S-box$_1$ and S-box$_2$ are 109.71 and 110.14. There are the greatest values in all comparative S-boxes, which proves that the BIC performance of the proposed S-boxes is excellent.

In summary, the proposed S-boxes have strong nonlinearity, small LP and DU, satisfy SAC characteristics and excellent BIC, which can resist linear cryptanalysis and differential cryptanalysis.

Table 12 BIC-Nonlinearity matrix of S-box$_2$

110	110	108	110	110	112	112	-
110	110	110	108	110	110	-	110
112	110	110	110	110	-	110	112
110	110	110	108	-	112	108	112
112	108	112	-	108	110	110	110
112	110	-	110	110	110	110	112
110	-	110	110	108	110	110	110
-	112	112	110	110	108	110	110

Table 13 Comparative analysis of S-boxes performance

S-boxes	Nonlinearity	LP	DU	SAC	BIC	
					SAC	Non.
Ref.[23]	94	0.132813	10	0.4953	0.5021	104.07
Ref.[24]	94	0.132813	10	0.4995	0.5011	103.85
Ref.[25]	92	0.140625	12	0.5051	0.4993	103.78
Ref.[26]	94	0.132813	10	0.4976	0.5022	103.57
Ref.[27]	92	0.140625	14	0.5351	0.5000	103.21
Ref.[28]	94	0.132813	10	0.5059	0.5026	104.00
Ref.[29]	94	0.132813	10	0.4943	0.4982	104.35
Ref.[30]	94	0.132813	10	0.4990	0.5040	103.14
Ref.[31]	92	0.140625	12	0.4988	0.4969	102.86
Ref.[33]	96	0.125000	10	0.5065	0.5031	103.57
Ref.[34]	92	0.140300	10	0.5007	0.5034	102.57
Ref.[35]	96	0.125000	10	0.4991	0.5052	105.21
Ref.[36]	92	0.140625	10	0.5093	0.5025	103.07
Ref.[37]	95	0.128906	10	0.4995	0.4976	106.35
Ref.[38]	86	0.164063	12	0.5063	0.4976	103.86
Ref.[39]	94	0.132813	10	0.5048	0.5009	103.71
Ref.[40]	94	0.132813	10	0.5030	0.5070	103.90
Ref.[41]	94	0.132813	10	0.5009	0.4996	103.64
Ref.[42] S_1	92	0.140625	10	0.5007	0.5012	104.21
Ref.[43] S_1	94	0.132813	10	0.5037	0.4994	102.64
Ref.[43] S_2	94	0.132813	12	0.4915	0.4965	103.57
Ref.[43] S_3	94	0.132813	10	0.5156	0.4997	103.36
Ref.[43] S_4	92	0.140625	14	0.5034	0.4990	103.93
Ref.[44]	96	0.125000	12	0.5071	0.5009	104.29
Ref.[45]	93	0.136719	10	0.4988	0.5052	104.00
$S\text{-box}_1$	104	0.093750	6	0.5015	0.5029	109.71
$S\text{-box}_2$	106	0.085938	6	0.5027	0.4993	110.14

6 Conclusion

The design of the S-box in block cipher has always been an important research field in cryptography. Constructing S-box by the chaotic system can have strong randomness, but the differential uniformity of the generated S-box is 10 or 12. To solve this problem, this paper proposes a new efficient S-box construction scheme based on a new logistic map and permutation sequence. The chaotic matrix is generated by the new logistic map, and then is optimized by the permutation sequence to get the final S-box. Through performance analysis, proving that the new logistic map has strong chaotic performance, a large parameter space range, and high sensitivity to the initial value. The proposed S-boxes have high nonlinearity and their differential uniformity is only 6. They can enhance the ability of the cryptographic algorithm to resist differential cryptanalysis and linear cryptanalysis.

Acknowledgements This work was supported by the Open Fund of Advanced Cryptography and System Security Key Laboratory of Sichuan Province (Grant No. SKLACSS-202103), the National Natural Science Foundation of China (Grant No. 62032002 and 62176273), the BUPT Excellent Ph.D. Students Foundation (Grant No.CX2022141), and Building Point of First-class Undergraduate Specialty in Beijing Electronic Science and Technology Institute-Cryptographic Research and Technology.

Data Availability All comparative data indicators can be found in the reference

Declarations

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

References

1. Zhu D, Tong X, Zhang M, Wang Z (2020) A New S-Box Generation Method and Advanced Design Based on Combined Chaotic System. Symmetry 12(12):2087
2. Chen G (2008) A novel heuristic method for obtaining S-boxes. Chaos Solitons Fractals 36(4):1028–1036
3. Zahid AH, Lliyasu AM, Ahmad M et al (2021) A Novel Construction of Dynamic S-Box With High Nonlinearity Using Heuristic Evolution. IEEE Access 9:67797–67812
4. Zahid AH, Arshad MJ, Ahmad M (2019) A Novel Construction of Efficient Substitution-Boxes Using Cubic Fractional Transformation. Entropy 21(3):245
5. Szaban M, Seredynski F (2008) Designing cryptographically strong S-boxes with the use of cellular automata. Annales Umcs Informatica 8(2):27–41
6. Zhou C, Hu W, Wang L et al (2018) Turbo Trellis-Coded Differential Chaotic Modulation. IEEE Trans Circ Syst II Exp Briefs 65(2):191–195
7. Hua Z, Zhou Y (2017) Design of image cipher using block-based scrambling and image filtering. Inf Sci 396:97–113
8. Zhang LY, Zhang Y, Liu Y et al (2017) Security Analysis of Some Diffusion Mechanisms Used in Chaotic Ciphers. Int J Bifurcation Chaos 27(10):1750155
9. Wong KW, Lin Q, Chen J (2010) Simultaneous Arithmetic Coding and Encryption Using Chaotic Maps. IEEE Trans Circ Syst II Exp Briefs 57(2):146–150
10. Wang D, Zhang B, Qiu D et al (2018) On the Super-Lorenz Chaotic Model for the Virtual Synchronous Generator. IEEE Trans Circ Syst II Exp Briefs 65(4):511–515
11. Mao Y, Chen G (2005) Chaos-Based Image Encryption. Handbook of Geometric Computing 231–265
12. Tang G, Liao X, Chen Y (2005) A novel method for designing S-boxes based on chaotic maps. Chaos Solitons Fractals 23(2):413–419
13. Lambi D (2017) A novel method of S-box design based on discrete chaotic map. Nonlinear Dyn 87:2407–2413
14. Liu G, Yang W, Liu W et al (2015) Designing S-boxes based on 3-D four-wing autonomous chaotic system. Nonlinear Dyn 82:1867–1877
15. Özkaynak F, Yavuz S (2013) Designing chaotic S-boxes based on time-delay chaotic system. Nonlinear Dyn 74:551–557
16. Özkaynak F, Özer AB (2010) A method for designing strong S-Boxes based on chaotic systems. Phys Lett A 374(36):3733–3738
17. Khan M, Shah T, Mahmood H et al (2012) A novel technique for the construction of strong S-boxes based on chaotic Lorenz systems. Nonlinear Dyn 70:2303–2311
18. Khan M, Shah T (2015) An efficient construction of substitution box with fractional chaotic system. Signal Image Vid Process 9:1335–1338
19. Belazi A, Khan M, El-Latif AA et al (2017) Efficient cryptosystem approaches: S-boxes and permutation substitution-based encryption. Nonlinear Dyn 87:337–361
20. Özkaynak F, Çelik V, Özer A (2017) A new S-box construction method based on the fractional-order chaotic Chen system. Signal Image Vid Process 11:659–664
21. Cavusoglu U, Zengin A, Pehlivan I et al (2017) A novel approach for strong S-box generation algorithm design based on chaotic scaled Zhongtang system. Nonlinear Dyn 87:1081–1094
22. Ahmad M, Haleem H, Khan PM (2014) A new chaotic substitution box design for block ciphers. In: 2014 International conference on signal processing and integrated networks (SPIN). IEEE, pp 255–258

23. Wang Y, Zhang Z, Zhang LY et al (2020) A genetic algorithm for constructing bijective substitution boxes with high nonlinearity. Inf Sci 523:152–166
24. Alhadawi HS, Majid MA, Lambic D et al (2021) A novel method of S-box design based on discrete chaotic maps and cuckoo search algorithm. Multimed Tools App 80:7333–7350
25. Lu Q, Zhu C, Wang G (2019) A Novel S-Box Design Algorithm Based on a New Compound Chaotic System. Entropy 21(10):1004
26. Jiang Z, Ding Q (2021) Construction of an S-Box Based on Chaotic and Bent Functions. Symmetry 13(4):671
27. Hua Z, Li J, Chen Y, Yi S (2021) Design and application of an S-box using complete Latin square. Nonlinear Dyn 104:807–825
28. Zhu Z, Song Y, Zhang W et al (2020) A novel compressive sensing-based framework for image compression-encryption with S-box. Multimed Tools App 79:25497–25533
29. Ahmed HA, Zolkipli MF, Ahmad M (2019) A novel efficient substitution-box design based on firefly algorithm and discrete chaotic map. Neural Comput App 31:7201–7210
30. Artuger F, Özkaynak F (2021) An effective method to improve nonlinearity value of substitution boxes based on random selection. Inf Sci 576:577–588
31. Farah MAB, Guesmi R, Kachouri A et al (2020) A new design of cryptosystem based on S-box and chaotic permutation. Multimed Tools App 79:19129–19150
32. Persohn KJ, Povinelli RJ (2012) Analyzing logistic map pseudorandom number generators for periodicity induced by finite precision floating-point representation. Chaos Solitons Fractals 45(3):238–245
33. Yan W, Ding Q (2021) A Novel S-Box Dynamic Design Based on Nonlinear-Transform of 1D Chaotic Maps. Electron 10(11):1313
34. Ahmad M, Solami EA (2020) Evolving Dynamic S-Boxes Using Fractional-Order Hopfield Neural Network Based Scheme. Entropy 22(7):717
35. Alzaidi AA, Ahmad M, Doja MN et al (2018) A New 1D Chaotic Map and β-Hill Climbing for Generating Substitution-Boxes. IEEE Access 6:55405–55418
36. Tian Y, Lu Z (2017) Chaotic S-Box: Intertwining Logistic Map and Bacterial Foraging Optimization. Math Probl Eng 2017(3):1–11
37. Zhu H, Tong X, Wang Z et al (2020) A novel method of dynamic S-box design based on combined chaotic map and fitness function. Multimed Tools App 79:12329–12347
38. Cavusoglu U, Kacsr S, Pehlivan I et al (2017) Secure image encryption algorithm design using a novel chaos based S-Box. Chaos Solitons Fractals 95:92–101
39. Özkaynak F, Özer AB (2010) A method for designing strong S-Boxes based on chaotic Lorenz system. Phys Lett A 374(36):3733–3738
40. Lu Q, Zhu C, Deng X (2020) An Efficient Image Encryption Scheme Based on the LSS Chaotic Map and Single S-Box. IEEE Access 8:25664–25678
41. Farah MAB, Farah A, Farah T (2020) An image encryption scheme based on a new hybrid chaotic map and optimized substitution box. Nonlinear Dyn 99:3041–3064
42. Wang J, Zhu Y, Zhou C et al (2020) Construction Method and Performance Analysis of Chaotic S-Box Based on a Memorable Simulated Annealing Algorithm. Symmetry 12(12):2115
43. Özkaynak F (2020) On the effect of chaotic system in performance characteristics of chaos based s-box designs. Phys A Stat Mech App 550:124072
44. Türk Ö (2022) FPGA simulation of chaotic tent map-based S-Box design. Int J Circ Theor Appl 1–15
45. Zheng J, Zeng Q (2022) An image encryption algorithm using a dynamic S-box and chaotic maps. Appl Intell
46. Khan MA, Ali A, Jeoti V et al (2018) A Chaos-Based Substitution Box (S-Box) Design with Improved Differential Approximation Probability (DP). Iranian J Sci Technol Trans Electr Eng 42:219–238
47. May RM (1976) Simple mathematical models with very complicated dynamics. Nature 261:459–467
48. Tang Y, Zhao M, Li L (2020) Secure and Efficient Image Compression-Encryption Scheme using New Chaotic Structure and Compressive Sensing. Secur Commun Netw 2020(2):1–15
49. Jakimoski G, Kocarev L (2001) Chaos and cryptography: block encryption ciphers based on chaotic maps. IEEE Trans Circ Syst I 49(2):163–169
50. Adams C, Tavares S (1990) The structured design of cryptographically good S-boxes. J Cryptol 3:27–41
51. Chen H, Feng D (2004) An effective evolutionary strategy for bijective S-boxes. Evol Comput 2:2120–2123
52. Webster AF, Tavares SE (1985) On the Design of S-Boxes. Conference on the Theory and Application of Cryptology. Springer, New York, pp 523–534
53. Adams C, Tavares S (1995) Good S-Boxes Are Easy To Find. Conference on the theory and application of cryptology. Springer, New York, pp 612–615

54. Liu H, Liu J, Ma C (2022) Constructing dynamic strong S-Box using 3D chaotic map and application to image encryption. Multimed Tools App

Authors and Affiliations

Mingjie Zhao[1,2] · Zheng Yuan[1,2,3] · Lixiang Li[1] · Xiu-Bo Chen[1]

Mingjie Zhao
zhaomingjie0704@163.com

Lixiang Li
lixiang@bupt.edu.cn

Xiu-Bo Chen
flyover@163.com

1 Information Security Center, State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China

2 School of Cryptography Science and Technology, Beijing Electronic Science and Technology Institute, Beijing 100070, China

3 School of Cyber Science and Technology, University of Science and Technology of China, Hefei 230026, China