Designing Substitution Box Based on the 1D Logistic Map Chaotic System

Meryam Saad Fadhil¹,⁴, Alaa Kadhim Farhan¹ and Mohammad Natiq Fadhil¹

¹Computer Sciences Department, University of Technology, Baghdad, Iraq

⁴E-mail: 110897@student.uotechnology.edu.iq

Abstract. Many researchers have tried to use the chaotic systems in generating different cryptographic tools, in order to provide high level of security to the data passing through network. In modern symmetric cryptographic systems, one of the most essential functions is a nonlinear mapping Substitution process which is responsible for creating the confusion between plaintext and ciphertext. It replaces each input bits by other bits using the lookup table S-Box. In this paper a new S-Box based on a 1D logistic map chaotic system was designed. The chaotic sequence generated from the 1D logistic map was processed to generate the hexa code values, and after that this hexa code also processed to construct the new good S-Box. This proposal was evaluated using the S-Box tests criteria including avalanche, balanced, completeness, strict avalanche, and invertibility. The analysis results show that the proposed S-Box passes all these statistical tests, has large avalanche effect and therefore the S-Box can avoid many attacks. The construction of both Substitution-Box and its inverse take only few milliseconds, so it can be used in many lightweight block ciphers.

1. Introduction

The aim of any cryptosystem is to provide a secure way in transferring the sensitive information passing through networks or the Internet [1-3]. Confusion and Diffusion are essential features in any cryptosystem to provide randomness to the output [4]. Diffusion means hiding the relationship between the plaintext and ciphertext therefore cannot discern the original text. While the Confusion means complicating the relationship between the ciphertext and the key [5]. In block cipher, such as AES ,DES,CAST and other block cipher algorithms, one of the essential operations that responsible for creating the confusion is the nonlinear mapping S-Box[6]. It performs a substitution process using lookup table [7].S-box is an important tool used in modern symmetric cryptosystems (especially, in block ciphers) to provide higher level properties of security and bring nonlinearity to block ciphers [8-10]. Many researchers focused on constructing strong S-Box to provide high level of security in block ciphers. The researchers have been depended on many methods to generate good S-Box such as pseudo random number, polynomial mapping, DNA, chaos theory and many other methods. In 2016, Maram K and Gnanasekar [4] proposed new S-Box depends on pseudo-random-number-generators and shared-secret-key. It yields better results with the S-box parameters and can be embedded to popular cryptography algorithms. In 2019, a highly nonlinear S-p-boxes is presented by Naseer et al [11], their proposal provides a large number of highly non-linear substitution permutation boxes having the property of confusion as well as diffusion. In 2019, Zahid and Arshad [12] design a substitution-boxes using cubic polynomial mapping. In 2019, the same researcher et al. [13] proposed another S-Box based on Cubic Fractional Transformation. In 2020, Farhan et al. [7] proposed a new
approach to generate multiple S-Boxes based on RNA computing. In recent years the chaotic systems is also used for the construction of the good S-Boxes. In 2012, Khan et al [1] proposed a novel technique for the construction of strong S-boxes based on chaotic Lorenz systems. The performance of their proposal is evaluated by using the nonlinearity analysis, strict avalanche criterion, bit independence criterion, linear approximation probability, and differential approximation probability. In 2018, Alzaidi et al [5] proposed a new 1D discrete- chaotic map which has good chaotic behaviour and large chaotic range when compared with the original chaotic logistic map. They used this improved chaotic map and β hill climbing search technique in constructing efficient substitution-boxes (S-boxes). The analysis results show that the generated S-box has good cryptographic strength and found better as compared to S-boxes available. The researchers in [14-16] also depend on the chaotic systems to construct good cryptographic S-Boxes.

In this paper, we use new algorithms to generate an S-box based on the sequence generated from the 1D chaotic logistic map equation. In the first algorithm, the chaotic sequence is used to generate the hexa code while the second algorithm is responsible for processing this hexa code to generate the S-Box. The proposed S-Box designed in a way that it consumes few time in construction and at the same time have good statistical tests when it compared with other proposals. The analysis results show that this proposal is strong against attack. The rest of this paper is as follows: Section 2 presents the chaotic system and its role in cryptography; Section 3 presents the proposed method to construct S-Box based on 1D logistic map. In Section 4 presents the analysis of this proposal. The conclusion of this paper is presented in Section 5.

2. Chaotic systems

Chaos theory is a field of mathematics, based on a nonlinear and deterministic behaviour [17]. It has higher sensitivity to its initial conditions: initial values and control parameters; therefore, a slight change in its initial values or control parameters leads to huge change in the chaotic outputs [18]. The chaotic systems have high unpredictable and random look nature of their output signals [19]. These properties is connected to those of good cipher in cryptography, such as confusion and diffusion; which help many researchers to use chaotic systems in enhancing security level of many cryptographic systems [20,21]. Chaos theory can be used in image encryption algorithms, secure pseudo-random number generators, hash functions, block ciphers, stream ciphers, steganography and watermarking [22], and any other system requires unpredictable output. Chaotic maps are classified into continuous time and discrete time chaotic maps as shown in Table 1 [23]. The 1-dimensional logistic map is the most popular discrete chaotic system, which has a simple mathematical structure, and yet have complex chaotic behaviour. It is defined in equation (1) [10].

$$x_{i+1} = \mu \times x_i \times (1 - x_i)$$  

where \(x_0\) is the initial state, \(i\) is the number of iterations, and \(\mu\) is the system control parameter. The value of \(x_{i+1}\) is a number between zero and one for all \(i\) as shown in Figure 1 while the value of the control parameter \(\mu\) belongs to the interval \((0, 4)\) and gives better values when \(\mu\) near to equal 4 [10]. There are different behaviours of \(x\) depending on the \(\mu\) value as shown in Figure 2[20]. The 1D logistic map system is used in this paper to construct new S-Box.

| Chaotic Map Name       | Domain | Dimension |
|------------------------|--------|-----------|
| Logistic map           | Discrete | 1        |
| Piecewise linear chaotic | Discrete | 1        |
| Tent map               | Discrete | 1        |
| Gaussian map           | Discrete | 1        |
Discrete Standard map 2
Discrete Cat map 2
Discrete Baker map 2
Discrete Lorenz system 3
Continuous Rössler attractor 3
Continuous Chen system 3
Continuous Jerk equation 3

Figure 1. Variation of chaotic logistic map with iteration values [24].

Figure 2. Bifurcation diagram of the logistic map [24].
3. Research method

This paper presents a new method to construct S-Box which responsible for creating confusion in block ciphers. Due to the high compatibility between chaos theory and cryptography, we depend on a 1D chaotic logistic map system to create this new S-Box as shown in Figure 3. In this proposal, the S-Box generated as follow: in the first, we generate the hexa code from 1D logistic map as illustrated in algorithm 1, and after that, we use the hexa code to generate the new S-Box, algorithm 2 represent this process. The process of constructing inverse for the new S-Box is represented in algorithm 3.

![Figure 3. The proposed S-Box construction.](image)

The S-Box is generated as follow: in the first stage, the hexa code will be generated from the 1D logistic map system as illustrated in algorithm 1.

Algorithm 1: Generating Hexa code

**Input:** initial conditions for the 1D logistic map ($\mu$ and $x_0$)

**Output:** Hexa codes

**Begin:**

1. read initial conditions
2. For $i=0$ to 10000
   
   2.1: Generate sequence of numbers using 1D logistic map and store the results in $x$ // $x$ is an array of floating values
   
   2.2: Next $i$
3. For $j=0$ to $x$.length
   
   3.1: Convert every floating number of $x$ to hexa code and store it in $H$ this process is done using the programming instruction ($H[i] = Double.toHexString(x[i])$) // $H$ is an array of strings.
   
   3.2: Next $i$

**End**

After generating the hexa code, this hexa code will be used to generate the new S-Box table which can be used in encryption process. Algorithm 2 represents S-Box construction.

Algorithm 2: S_Box construction

**Input:** initial conditions for the 1D Chaotic logistic map ($x_0$ and $\mu$).

**Output:** new S_Box $(16 \times 16)$

**Begin**

1. Generate Hexa code using initial conditions according to algorithm 1 and store the result in $H$. // $H$ is an array of strings
2. For $i = 0$ to $H$.length
2.1: \( S(i) = H(i) \). substring \((5,7)\) // take two digits (from index \((5\) to \(7))\) of \(H(i)\) and put in \(S(i)\).

2.2: Next \(i\)

3. Delete any repeated values in \(S\) array.

4. Take only 256 values from \(S\) without repetition to construct new \(S\_Box\) \((16 \times 16)\).

End

Figure 4 represents an example about the construction of \(S\_Box\) using the chaotic sequence generated from the 1D logistic map equation and the initial conditions \(\mu = 3.71\) and \(x_0 = 0.201\).

| Chaotic sequence | Hexa code | Convert to Hexa String | Take 2 digits (from index 5 to 7) | Delete repeated values if exist, after that Insert numbers to \(S\_Box\) |
|------------------|-----------|-------------------------|----------------------------------|---------------------------------------------|
| 0.201            | 0x1.9ba5e353f7ceep-3 | ba                      | 10                               | 97                                          |
| 0.59582229       | 0x1.310f9e838e3f7p-1 | 10                      | 0x1.c970539e6a68dp-1              |                                             |
| 0.8934351092222684 | 0x1.e970539e6a68dp-1 | 10                      | 0x1.10970539e6a68dp-1             |                                             |
| 0.35322470302398096 | 0x1.69b3bc8e825dp-2 | 10                      | 0x1.b1f56afde09dp-1              |                                             |
| 0.8475755152531012 | 0x1.69b3bc8e825dp-2 | 10                      | 0x1.b1f56afde09dp-1              |                                             |
| .                | .         | .                      | .                                | .                                           |
| .                | .         | .                      | .                                | .                                           |
| .                | .         | .                      | .                                | .                                           |

**Figure 4.** Example of \(S\_Box\) construction.

In decryption process the inverse \(S\_Box\) will be used in order to extract the original data, the process of constructing the inverse for the new \(S\_Box\) is represented in algorithm 3 below.

**Algorithm 3: Inverse \(S\_Box\) for the constructed \(S\_Box\)**

**Input:** \(S\_Box\) \((16 \times 16)\).

**Output:** Inverse \(S\_Box\) \((16 \times 16)\).

**Begin**

1. Loop to all numbers in the \(S\_Box\) \((16 \times 16)\).
2. Extract each number from \(S\_Box\) and separate it into two digits (to represent the address of the value in inverse \(S\_Box\)).
3. The address of the two digits is also extracted and combined together to represent the new value that will be stored in the inverse \(S\_Box\).
4. The previous steps \((2 \cdot 3)\) is repeated until the inverse \(S\_Box\) is completed successfully.

**End**

Each value in the inverse \(S\_Box\) is generated as follow: as an example, the first value of the \(S\_Box\) is \((ba)\). This value is separated into two digits \((b)\) and \((a)\) to represent the address of the value in inverse \(S\_Box\). The address of \((ba)\) is extracted, which is row 0 and column 0 and combined together to represent the value that will be stored in the inverse \(S\_Box\). Therefore, the value in row \((b)\) and column \((a)\) of inverse \(S\_Box\) is \((00)\).
4. Results and discussion
The construction of both s-box and its inverse take only 59.75 milliseconds, which is measured by the netbeans IDE for java programming. This proposal also passes the S-Box tests criterion such as, avalanche, balanced, completeness, strict avalanche, and invertability, and each of these criterions is illustrated below and compared with other S-Boxes in related studies.

4.1. Balanced criteria (BC)
One of the essential S-Box tests criterions is to test the distribution of number of 0’s and 1’s in the output sequences, this distribution should be balanced [4,25]. Through performing this test, the new S-Box is balanced because it has equal or near to equal number of 0’s and 1’s, as shown in Table 2 which illustrate the BC test for two words using new S-Box compared with other related work.

| Method          | BC (2 words) | Computer | ABMNOPQR |
|-----------------|--------------|----------|----------|
|                 | No. 0’s | No. 1’s | No. 0’s | No. 1’s |
| [1]             | 39      | 25      | 30      | 34      |
| [4]             | 33      | 31      | 31      | 33      |
| [5]             | 34      | 30      | 27      | 37      |
| [10]            | 35      | 29      | 28      | 36      |
| Proposed S-Box  | 32      | 32      | 33      | 31      |

The Figures 5 and 6 represent the BC test for two words on the proposed S-Box compared with other S-Boxes in the related studies. In Figure 5 the number of ones and zeros are computed for the word "Computer" after replaced with new data from S-Box, while Figure 6 represents computing the number of ones and zeros for the word "ABMNOPQR" after replaced with new data from S-Box.
4.2. The completeness Criteria (CC)

This test means that every output bit depends on all input bits. Therefore; if there is one pair of plaintext vectors \((z_i, z_{i+1})\), \(z_i\) and \(z_{i+1}\) different in only one bit, so the output from \(z_i\) and \(z_{i+1}\) are different at least in bit k [26]. Each bit of the proposed S-Box depends on the initial values \((x_0\) and \(\mu)\) of the 1D logistic map; therefore, if there are two different values of \(x_0\) (they differ in only one digit after the decimal point), the S-Boxes generated by the first and the second \(x_0\) are different of each other. This property is one of the important features of using chaos and represents the strength of logistic map. It means that a slight change in initial conditions leads to a massive change in chaotic outputs. For example, the S-Box generated by \(x_0 = 0.201\) and \(\mu = 3.71\), and the S-Box, generated by \(x_0 = 0.0201\) and \(\mu = 3.71\) are different from each other. Therefore, the new S-Box satisfies the completeness criteria. The results of these two examples are shown in Tables 3, 4, 5, and 6, respectively.

**Table 3. S-Box values if \(x_0 = 0.201\) and \(\mu = 3.71\).**

|   | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 0 | BA | 10 | 97 | 9B | IF | AC |
| 1 | 13 | 20 | B4 | 95 | 1D | 7A |
| 2 | 04 | 9F | 67 | 3E | A9 | 12 |
| 3 | EE | 3A | 2E | E7 | E9 | 76 |
| 4 | AB | 4E | 61 | CF | BC | 6A |
| 5 | 9C | 58 | C3 | 09 | 3D | 92 |
| 6 | 6D | A8 | FF | 8B | 73 | 50 |
| 7 | 21 | C8 | A7 | 37 | D5 | 79 |
| 8 | 59 | FA | 39 | CB | A0 | B8 |
| 9 | 89 | EA | 8A | 05 | F7 | 35 |
| A | 07 | D6 | 47 | 1C | C5 | CC |
| B | 98 | 4B | 11 | 32 | DE | D0 |
| C | 45 | 7B | 5E | 52 | FC | C2 |
| D | 9E | 7D | E0 | B9 | 77 | 60 |

---

**Figure 6. BC test on the word "ABMNOPQR"** as an example and comparison with S-Boxes in related works.
Table 4. Inverse S-Box values if $x_0 = 0.201$ and $\mu = 3.71$.

| 0  | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | A  | B  | C  | D  | E  | F  |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 0  | 17 | A9 | 1E | 2E | 20 | 93 | 9F | A0 | FA | 53 | 27 | 0E | 5D | 39 | 3D |
| 1  | 01 | B2 | 25 | 10 | 4C | E9 | AB | EA | DA | DD | 62 | 89 | A3 | 14 | 1F |
| 2  | 11 | 70 | 16 | 8F | FB | BC | 3E | DF | 7B | CC | 99 | BE | 87 | 48 | 32 |
| 3  | 97 | A8 | B3 | 68 | 28 | 95 | 29 | 73 | E5 | 82 | 31 | EB | AE | 54 | 23 |
| 4  | 67 | 8A | 6C | F3 | E0 | C0 | E7 | A2 | FC | 07 | 3F | B1 | 0F | E2 | 41 |
| 5  | 66 | 6F | C4 | 26 | BD | 78 | B7 | 9E | 51 | 80 | 0C | 4F | CE | E4 | C3 |
| 6  | 6D | 42 | EC | C7 | B8 | 9C | D9 | 22 | 1D | 7E | 45 | 37 | CA | 60 | E3 |
| 7  | D8 | BA | 7D | 65 | F8 | 4E | 35 | D4 | C1 | 75 | 15 | 77 | DB | D1 | F4 |
| 8  | 6D | E8 | DC | 4D | 7A | 1B | 08 | E1 | 8E | 90 | 92 | 64 | F2 | 3A | CF |
| 9  | F1 | CD | 55 | 4B | F7 | 13 | A6 | 02 | B0 | 47 | 5A | 03 | 50 | 5E | D0 |
| A  | 84 | 06 | 6B | AF | AD | 56 | F5 | 72 | 61 | 24 | 0D | 40 | 05 | 98 | 2F |
| B  | 5F | 76 | FF | B9 | 12 | 96 | 7F | CF | 28 | 85 | 03 | D0 | 7C | 44 | 46 |
| C  | F9 | A7 | C6 | 52 | 9C | A4 | 59 | 3B | 71 | AA | 36 | 83 | A5 | 58 | 5C |
| D  | B5 | 0A | 18 | 88 | ED | 74 | A1 | 6A | 9B | 57 | AC | 2D | 1C | D7 | B4 |
| E  | D2 | 69 | 97 | F0 | 5B | EF | CB | 33 | 8D | 34 | 91 | 6E | FD | D6 | 30 |
| F  | 9A | 1A | C8 | 6E | 19 | 2C | 3C | 94 | 49 | F6 | 81 | 8B | C5 | 0B | DE |

Table 5. S-Box values if $x_0 = 0.0201$ and $\mu = 3.71$.

| 0  | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | A  | B  | C  | D  | E  | F  |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 0  | 95 | B4 | 15 | 56 | 06 | 3C | 99 | 29 | 40 | A2 | 80 | 4C | 6C | 8B | D4 |
| 1  | FC | 49 | 18 | 16 | 6E | 63 | 10 | CA | 5C | 86 | 3D | 9E | 19 | 67 | 6B |
| 2  | A1 | 36 | E0 | B0 | EC | DE | F9 | AE | EE | 3A | 2E | E7 | E9 | 76 | 6F |
| 3  | 8D | C6 | F6 | 0F | 25 | A6 | 22 | 37 | 58 | 03 | E6 | ED | 42 | A4 | 41 |
| 4  | 0C | B2 | 87 | F7 | 50 | AD | F3 | 1D | CE | 9C | AF | 72 | A8 | 1C | 5D |
| 5  | 52 | 83 | 59 | 0B | 08 | 32 | CD | 00 | F4 | 05 | 30 | A9 | 57 | 04 | 3F |
| 6  | 3B | 44 | 89 | 6D | 71 | 3E | 84 | 69 | CC | FA | 23 | 26 | 7F | F0 | 5A |
| 7  | 6A | 4E | 4B | D0 | C2 | A3 | DF | 33 | AC | B1 | 9F | B3 | E3 | DC | 2D |
| 8  | 54 | 61 | BC | 68 | 7C | 31 | 5E | F2 | 65 | 70 | A7 | E2 | 1F | DA | EF |
| 9  | 77 | 6E | 8C | 07 | 35 | D7 | 0A | 24 | DB | A5 | FF | 4A | 73 | AA | DB |
| A  | B9 | 9A | 7A | D1 | 27 | 9D | 5F | A0 | 0E | 4F | 28 | C1 | F1 | 17 | 82 |
| B  | DD | BB | 55 | 60 | B6 | 8E | 91 | 2F | 78 | 7D | E1 | EB | B8 | AB | 09 |
| C  | 81 | 13 | 39 | 1B | 62 | D2 | 01 | 97 | 46 | B5 | 8A | 7B | 2B | 2C | 43 |
| D  | 4D | F5 | 21 | 94 | 75 | 5B | 64 | 38 | 53 | BF | D5 | F8 | 47 | C4 | 85 |
| E  | C3 | 12 | BA | 11 | C5 | 74 | B7 | 20 | D3 | 8F | 79 | D6 | C9 | DF | C7 |
| F  | 2A | 1E | C8 | 66 | 7E | 34 | FD | 51 | 9B | 90 | CB | 14 | 88 | 45 | 1A |

Table 6. Inverse S-Box values if $x_0 = 0.0201$ and $\mu = 3.71$.

| 0  | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | A  | B  | C  | D  | E  | F  |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 0  | 57 | C6 | 0F | 39 | 59 | 59 | 04 | 93 | 54 | BE | 96 | 53 | 40 | 2E | A8 |
| 1  | 16 | E3 | E1 | C1 | FB | 02 | 13 | AD | 12 | 1C | FE | C3 | 4D | 47 | F1 |
| 2  | E7 | D2 | 36 | 6A | 97 | 34 | 6B | A4 | AA | 07 | F0 | CC | CD | 7E | B7 |
| 3  | 5A | 85 | 55 | 77 | F5 | 94 | 21 | 37 | D7 | C2 | 29 | 60 | 05 | 1A | 65 |
| 4  | 08 | 3E | 3C | CE | 61 | FD | C8 | DC | 6F | 11 | 9B | 72 | 0B | D0 | 71 |
| 5  | 44 | F7 | 50 | D8 | 80 | B2 | 03 | 5C | 38 | 52 | 6E | D5 | 18 | 4E | 86 |
| 6  | B3 | 81 | C4 | 15 | D6 | 88 | F3 | 1D | 83 | 67 | 70 | 1E | 0C | 63 | 91 |
| 7  | 89 | 64 | 4B | 9C | E5 | D4 | 2D | 90 | B8 | EA | A2 | CB | 84 | B9 | F4 |
| 8  | 0A | C0 | AE | 51 | 66 | DE | 19 | 42 | FC | 62 | CA | 0D | 52 | B0 | E9 |
| 9  | F9 | B6 | CF | DF | D3 | 00 | 5F | C7 | 1F | 06 | A1 | F8 | 49 | A5 | 1B |
| A  | A7 | 20 | 09 | 75 | 3D | 99 | 35 | 8A | 4C | 5B | 9D | BD | 78 | 45 | 27 |
| B  | 23 | 79 | 41 | 7B | 01 | C9 | B4 | E6 | BC | A0 | E2 | B1 | 82 | 9F | 14 |
4.3. Avalanche Criteria (AC)
The avalanche criteria is the essential criterion in block ciphers, which refers to how a simple change in the input bits leads to a large (avalanche) change in the output sequence, such as changing one bit from zero to one or vice versa leads to avalanche change in output. This criterion is a desirable feature for block cipher methods because of its result related to the computing of diffusion. The avalanche value should be within the range \([0, 1]\). The optimal value for avalanche effect is 0.5, which denotes that the avalanche criterion is satisfied [26]. Equation (2) used to calculate AC. Table 7 represents AC test.

\[
AC = \frac{\text{Number of Flipped Bits in Cipher Text}}{\text{Number of All Bits in Cipher Text}}
\]

As an example, when we change one bit from the letter "L" it become "M" and S-Box (L) is different than S-Box (M) in five bits from original eight bits therefore the AC is equal to 0.625 which means that it is within the optimal range. This test also compared with the related studies given in [1,4,5] and [10] as illustrated in Figure 7.

| Method          | Actual Data | ASCII | Hex | Binary | Replace in S-box | Binary Output | AC   |
|-----------------|-------------|-------|-----|--------|-----------------|---------------|------|
| [1]             | L           | 76    | 4C  | 0100   | 84              | 1000          | 4/8 =0.5 |
| Change 1 bit    | M           | 77    | 4D  | 0100   | 27              | 0010          |       |
| [4]             | L           | 76    | 4C  | 0100   | DB              | 1101          | 4/8=0.5 |
| Change 1 bit    | M           | 77    | 4D  | 0100   | 5C              | 0101          |       |
| [5]             | L           | 76    | 4C  | 0100   | 6D              | 1101          | 4/8=0.5 |
| Change 1 bit    | M           | 77    | 4D  | 0100   | 5B              | 0101          |       |
| [10]            | L           | 76    | 4C  | 0100   | BE              | 1011          | 3/8   |
| Change 1 bit    | M           | 77    | 4D  | 0100   | 7F              | 1110          | =0.375 |
| Proposed S-Box  | L           | 76    | 4C  | 0100   | 14              | 0111          |       |
| Change 1 bit    | M           | 77    | 4D  | 0100   | 83              | 1000          | 5/8   |
|                 |             |       |     |        |                 | 0011          | =0.625 |
4.4. **Strict Avalanche Criteria (SAC)**

The S-Box satisfies the strict avalanche criterion if the change of one bit in the input changes half of the output bits [27][28]. In other words, SAC is achieved if both the completeness and avalanche criteria were satisfied [7][29]. The proposed technique satisfies these criterions, so the SAC is also achieved.

4.5. **Invertability**

This test simply ensures that each input value to the S-Box is mapped into a unique output value, forcing the S-Box to be a one-to-one function. This test is necessary to enable the correct recovery (back substitution) which substitutes the values via the Inverse S-Box [29].

The S-box satisfy invertibility feature, if S-Box (L1) = S-Box (L2) such that L1= L2 for all inputs L1, and L2 [26]. Let L1= "a" and L2 = "a", i.e., L1=L2 so the output in hexadecimal of S-Box (L1) is "A8" and S-Box (L2) is "A8". Now, using the S-Box inverse on "A8", the output is "61" in hexadecimal after that convert to decimal ,the output is "97" which corresponds to char (97) = a. therefore ; the S-Box is invertable due to its ability to extract the original data.

5. **Conclusion**

In this paper, we design a new S-Box can be used in AES block cipher. Due to the high compatibility between the chaotic systems and cryptography, we depend on the 1D logistic map system in construction of this proposal. A slight change in initial conditions of logistic map equation leads to construct new S-Box different than before. In this proposal, the construction of the S-Box and its inverse take only few milliseconds. The new S-Box was evaluated using S-Box tests criteria such as: avalanche, balance, strict avalanche, completeness, and invertability. These statistical tests also compared with some related studies. From these results, yield that the new S-Box has good properties for encryption, in order to provide secure communication, and from the fast in construction both S-Box and its inverse, this S-Box is suitable for using in lightweight cryptography which is used for constraint devices. In the future work, we can use another chaotic system in addition to the logistic map in constructing the S-Box to increase the security level. Also, we can generate dynamic S-Box instead of static S-Box to use different S-Box in each round to increase the confusion.

![Figure 7. AC test for the proposed S-Box compared with related works.](image-url)
References

[1] M. Khan, T. Shah, H. Mahmood, M.A. Gondal, and I. Hussain 2012 A novel technique for the construction of strong S-boxes based on chaotic Lorenz systems. *Springer Science+Business Media Dordrecht*, pp.2303–2311.

[2] A. K. Farhan, G. H. A. Majeeed, and R. S. Ali 2017 Enhancement CAST Block Algorithm to Encrypt Big Data *Annual Conference on New Trends in Information & Communications Technology Applications IEEE*, pp. 80-85.

[3] A. K. Farhan and M. A. A. Ali, 2017 Database Protection System Depend on Modified Hash Function. *In Conference of Cihan University-Erbil on Communication Engineering and Computer Science*.

[4] B. Maram K and J M Gnanasekar 2016 Evaluation of Key Dependent S-Box Based Data Security Algorithm using Hamming Distance and Balanced Output. *TEM Journal* **5**, 1.

[5] A. A. Alzaidi, M. Ahmad, M. N. Doja, E. Al Solami, and M. M. S. Bega 2018 A New 1D Chaotic Map and β-Hill Climbing for Generating Substitution-Boxes *IEEE* **6**.

[6] M. Ahmad, M. N. Doja, and M. M. S. Beg 2018 ABC Optimization Based Construction of Strong Substitution-Boxes. *Springer Science+Business Media* pp 1715–1729.

[7] A. K. Farhan, R. S. Ali, H. R. Yassein, N. M. G. Al-Saidi, and G. H. A. Majeeed 2020 a new approach to generate Multi S-Boxes based on RNA computing *International Journal of Innovation Computing, Information and Control* **16** 1.

[8] X. Wang, Ü. Çavuşoğlu, S. Kacar, A. Akgul, V. Th. Pham, S. Jafari, F. E. Alsaadi and X. Q. Nguyen 2019 S-Box Based Image Encryption Application Using a Chaotic System without Equilibrium. *mdpi applied sciences* **9**.

[9] P. P. Mar and K. M. Latt 2008 New Analysis Methods on Strict Avalanche Criterion of S-Boxes World Academy of Science. *Engineering and Technology International Journal of Mathematical, Computational, Physical, Electrical and Computer Engineering* **2** 12.

[10] Q. Lu, C. Zhu and G. Wang 2019 A Novel S-Box Design Algorithm Based on a New Compound Chaotic System *mdpi, entropy* **21**.

[11] Y. Naseer, T. Shah, Dawood Shah and S. Hussain 2019 A Novel Algorithm of Constructing Highly Nonlinear S-p-boxes. *mdpi cryptography* **3**.

[12] A. H. Zahid and M. J. Arshad 2019 An Innovation Design of Substitution-Boxes using Cubic polynomial mapping. *mdpi symmetry* **11** 3.

[13] A. H. Zahid, M. J. Arshad, and M. Ahmad 2019 A Novel Construction of Efficient Substitution-Boxes Using Cubic Fractional Transformation. *mdpi entropy* **21** 3.

[14] X. Wang, Ü. Çavuşoğlu, S. Kacar, A. Akgul, V. Th. Pham, S. Jafari, F. E. Alsaadi and X. Q. Nguyen 2019 S-Box Based Image Encryption Application Using a Chaotic System without Equilibrium. *mdpi applied sciences* **9**.

[15] Q. Lu, C. Zhu and G. Wang 2019 A Novel S-Box Design Algorithm Based on a New Compound Chaotic System. *mdpi, entropy* **21**.

[16] M. Ahmad, E. Al-Solami, A. M. Alghamdi and M. A. Yousaf 2020 Bijective S-Boxes Method Using Improved Chaotic Map-Based Heuristic Search and Algebraic Group Structures. in *IEEE Access* **8** pp. 110397-110411, doi: 10.1109/ACCESS.2020.3001868.

[17] A. K. Farhan and H. Emad 2017 Mouse Movement with 3D Chaotic Logistic Maps to Generate Random Numbers. *Diyala Journal for Pure Sciences* **13** 3.

[18] H. Natiq, N. M. G. Al-Saidi, M. R. M. Said and A. Kilicman 2018 A new hyperchaotic map and its application for image encryption. *Springer, The European Physical Journal Plus*, 133(1) 6.

[19] G. Jakimoski and L. Kocarev February 2001 Chaos and Cryptography: Block Encryption Ciphers Based on Chaotic Maps. *IEEE Transactions on Circuits and Systems—I: Fundamental Theory and Applications* **48** 2.

[20] A. Abdulgader, M. Ismail, N. Zainal and T. Idbeaa 2015 Enhancement of AES Algorithm Based on Chaotic Maps and Shift Operation for Image Encryption. *Journal of Theoretical and Applied Information Technology* **71** 1.
[21] Y. Zhang, J. Hao and X. Wang 2020 An Efficient Image Encryption Scheme Based on S-Boxes and Fractional-Order Differential Logistic Map. IEEE Access 8 pp. 54175-54188 doi: 10.1109/ACCESS.2020.2979827.

[22] Ons Jallouli 2017 Chaos-based security under real-time and energy constraints for the Internet of Things, Thesis, Signal and Image processing, Universite de Nantes, English, tel-01633910.

[23] A. K. Farhan and R. S. Ali 2019 Enhancement AES based on 3D Chaos Theory and DNA Operations Addition. Karbala International Journal of Modern Science 5 2.

[24] N. Hazarika and M. Saikia 2014 A Novel Partial Image Encryption using Chaotic Logistic Map. International Conference on Signal Processing and Integrated Networks (SPN) IEEE.

[25] M. Ahmad, I. A. Khaja, A. Baz, H. Alhakami and W. Alhakami 2020 Particle Swarm Optimization Based Highly Nonlinear Substitution-Boxes Generation for Security Applications. IEEE Access 8 pp. 116132-116147 doi: 10.1109/ACCESS.2020.3004449.

[26] A. H. Saeed AL-Wattar 2019 A Review of Block Cipher’s S-Boxes Tests Criteria Iraqi Journal of Statistical Science First Student Conference, 29, pp[1-14].

[27] E. Tanyıldızı and F. Özkaynak 2017 A New Chaotic S-box Generation Method using Parameter Optimization of One Dimensional Chaotic Maps IEEE access.

[28] A. H. Zahid, E. Al-Solami, and M. Ahmad 2020 A Novel Modular Approach Based Substitution-Box Design for Image Encryption IEEE Access 8 pp. 150326-150340.

[29] N. B. Abdulwahed 2013 Chaos-Based Advanced Encryption Standard. Thesis, King Abdullah University of Science and Technology.