



S-Box Generation Algorithm by Constructing the Non-Singular Adjacency Matrix Using the Genetic Algorithm

Javokhir Abdurazzokov

Digital Technologies and Artificial Intelligence Development Research Institute, Tashkent, Uzbekistan

Email address:

javohirjon-1992@gmail.com

To cite this article:

Javokhir Abdurazzokov. (2024). S-Box Generation Algorithm by Constructing the Non-Singular Adjacency Matrix Using the Genetic Algorithm. *American Journal of Science, Engineering and Technology*, 9(1), 14-20. <https://doi.org/10.11648/j.ajset.20240901.12>

Received: December 20, 2023; **Accepted:** January 4, 2024; **Published:** January 18, 2024

Abstract: In today's applications of block ciphers, the substitution box (S-box) serves as a critical nonlinear component that is essential for generating complex ciphertext. S-boxes that exhibit lower differential uniformity and increased nonlinearity are more adept at resisting cryptanalytic efforts. The paper proposes that the construction of an 8x8 S-box can be accomplished by selecting non-singular adjacency matrices derived from graph parameters generated by a genetic algorithm. This selection is followed by an affine transformation. This method uses any graph with 8 vertices and its edge count, resulting in a non-singular adjacency matrix. The S-box is then generated by an affine mapping technique using the non-singular adjacency matrix, similar to the approach of the Rijndael algorithm. The effectiveness and reliability of the resulting S-box was rigorously tested against various cryptographic standards. The robustness evaluation included factors such as non-linearity, differential approximation probability, linear approximation probability and strict avalanche criteria. A thorough investigation confirmed that the newly created S-box met the required algebraic properties. Furthermore, a comparative analysis was performed to evaluate the performance of this novel S-box against the most recent counterparts in the literature. In terms of defense against potential malicious exploits, the results indicate a significant advantage. Overall, the results of this study underscore the significant promise and advantages of the proposed S-box-centric cryptographic strategy, positioning it as an attractive alternative to conventional encryption techniques.

Keywords: S-Box, Nonsingular Matrix, Adjacency Matrix, Affine Transformation, Nonlinearity, Strict Avalanche Criterion

1. Introduction

In the current era of digital progress, where data transmission, storage and protection are ubiquitous, it is crucial to have strong security systems in place. Encryption methods, based on the principles of cryptography, are fundamental to ensuring the security of data communication and storage [1, 13].

S-Box is essential to achieve advanced encryption performance in various elements of the cryptographic architecture. It is widely used in standards of symmetric encryption algorithms similar to those used in Substitution-Permutation (SP) or Feistel network-based systems [2]. In symmetric key cryptography, block ciphers and stream ciphers are the two primary forms of encryption. The integration of artificial intelligence with cryptanalysis has revolutionized the ever-evolving landscape of cryptography, where ensuring security and maintaining confidentiality

remain paramount [3]. Achieving this balance often requires a deep understanding of mathematical techniques, statistical analysis of cryptographic systems, and various computational strategies. In general, S-boxes are developed using a variety of computational methods that exploit the properties of Boolean functions [4]. The development of analysis methods for code breaking, including linear, integral, differential, and algebraic analysis, continues to drive the need for stronger substitution boxes (S-boxes). Recent advances in computational techniques have led to more robust methods for designing S-boxes [5, 6].

Recent work has demonstrated the importance of linear and differential uniformity, two attributes that evaluate a substitution box's (S-box's) resilience to differential and linear attacks [7, 8]. In 2014, a technique for the creation of an S-box with enhanced algebraic and differential properties was formulated [9].

In recent times, novel non-traditional mathematical

approaches and algorithms have emerged for the creation of stationary and changing S-boxes [10, 11, 13].

By 2021, a simple and effective dynamic and key-dependent method using linear trigonometric transformations to create substitution boxes (S-boxes) had been introduced [12]. In 2021, an innovative approach to generating S-boxes by compiling bitwise operations derived from an identity function emerged [13, 14]. One of the factors considered when designing an S-box is strict avalanche effectiveness. S-boxes with optimal or close to optimal strict avalanche effectiveness are rare [15, 16]. Notwithstanding the formulation of algorithms for computing S-box values, as explained earlier, the creation of an S-box that can withstand various conditions remains a major focus of cryptographic research.

The paper presents an approach to generate robust S-boxes by constructing nonsingular adjacency matrices using selected graph features in genetic algorithms. The affine transformation is performed using the adjacency matrix. By using the proposed approach, the reliability of S-boxes is further improved within the given strict avalanche efficiency parameters.

2. Materials and Methods

2.1. Implementation of the S-Box in the AES

S-Boxes are widely used in modern symmetric key cryptographic algorithms. The AES's S-box acts as a mechanism that maps each input bit to the output bits through predefined look-up tables (LUTs).

By analyzing the foundations of the Galois field, we can see that if p represents a non-zero element in the principal ideal field R , then the operation R/p yields a field if p is not factorizable. If p is a prime number and the exponentiated set q is equal to p raised to the power of n , we have the ability to represent the finite field containing q elements as $GF(q)$. If we consider a vector space spanning $GF(p)$, which can be formulated as the result $a_i \in GF(2^8)$ field of the repeated XOR procedure applied to $GF(p)$ for n iterations, we construct the representation of $GF(q) = GF(p^n)$ as shown in equation (1) [13].

$$GF(q) = \frac{GF(p)[x]}{m(x)} \quad (1)$$

where $m(x)$ is the generative polynomial of degree n within the bounds of the $GF(q)$, we come to the fundamental irreducible polynomial chosen by the AES. This polynomial serves as the basis for the construction of the $GF(2^8)$ field, $m(x) = x^8 + x^4 + x^3 + x + 1$. The AES S-box operates on the byte scale. It operates within the $GF(2^8)$ field. So the representation of each entity is like the following 7th-degree polynomial: (2)

$$a_8x^7 + a_7x^6 + a_6x^5 + a_5x^4 + a_4x^3 + a_3x^2 + a_2x + a_1 \quad (2)$$

where $a_i \in GF(2)$ and are applicable, the addition is determined by the XOR operation, while the product is determined by the polynomial multiplication modulo the generating polynomial.

The S-box acts as a mapping operation within the Galois field $GF: 2^8 \rightarrow 2^8$. The S-box is constructed using the multiplicative inversion within $GF(2^8)$, the irreducible polynomial $m(x) = x^8 + x^4 + x^3 + x + 1$, and then an affine transformation. In other words, each element is subjected to the process $x \mapsto Ax^{-1} + b$. In this context, $A \in GF_8(2)$ represents the entire linear group of 8 degrees over $GF(2)$, and $b \in GF(2^8)$ is called the displacement vector, with A and b described as follows:

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \text{ and } b = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} \quad (3)$$

The result of the above is an S-box with 256 components [13, 17].

2.2. Adjacency Matrix Representing the Graph

The adjacency matrix can be used as a visual representation of the graph. In graph theory and computing, an adjacency matrix acts as a square matrix used to illustrate a bounded graph. Whether or not pairs of vertices in the graph are connected is indicated by the entries in the matrix. Adjacency matrix: If two vertices are connected by a single path, they are called adjacency vertices [18]. Furthermore, if a vertex is connected to itself, it is considered to be adjacent to itself. Let G be a graph with n vertices and m edges. The adjacency matrix A of G is an $n \times m$ matrix $A = [a_{ij}]$ whose n rows correspond to n vertices and m columns correspond to m edges (4) [13].

$$a_{ij} = \begin{cases} 1, & \text{if } \{v_i, v_j\} \text{ is an edge of } G \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

2.3. Generation of Nonsingular Adjacency Matrix Using Genetic Algorithms

Genetic algorithms are a cornerstone of evolutionary computing, an approach that mimics the process of natural selection to solve complex optimization and search problems. Inspired by the principles of biological evolution, they exploit the power of genetic processes. In this section, the problem of generating a nonsingular adjacency matrix of genetic algorithms was considered. This algorithm is shown in Table 1.

Table 1. Generating nonsingular adjacency matrices using a genetic algorithm.

Algorithm 1: Generating a nonsingular adjacency matrix with genetic algorithms

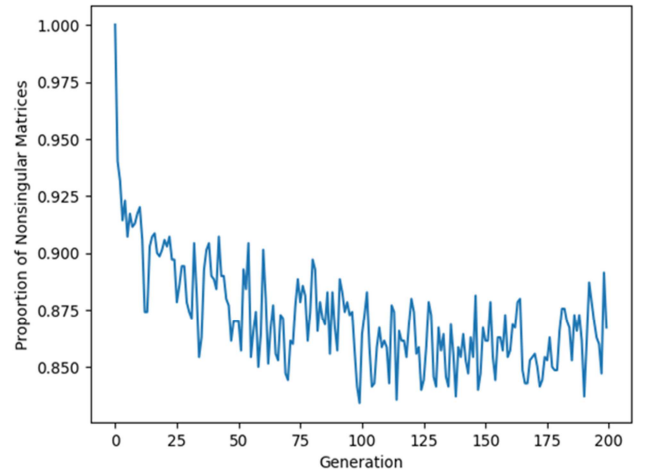
Input:
 N: Number of individuals in the population.
 V: Number of vertices in the adjacency matrix.
 G: Number of generations to run the algorithm.
 μ : Probability of mutation.
 Output: A nonsingular adjacency matrix, if one is found.

- 1 Initialize Population
- 2 For each individual i in the population $\{1, 2, \dots, N\}$: $f(A_i) = \begin{cases} 0 & \text{if } \text{rank}(A_i) = V \\ 1 & \text{otherwise} \end{cases}$
- 3 Create a $V \times V$ binary matrix A_i
- 4 For each j, k in $\{0, 1, \dots, V-1\}$, set:
- 5 $A_i(j, k) = \begin{cases} 1 & \text{if } (k = j-1 \text{ and } j > 0) \\ & \text{or } (k = j+1 \text{ and } j < V-1) \\ 0 & \text{otherwise} \end{cases}$
- 6 Repeat for Each generation $g \in \{1, 2, \dots, G\}$
- 7 Evaluate Fitness
- 8 For each individual A_i , compute fitness $f(A_i)$:
- 9 Selection
- 10 Perform tournament selection to choose $N/2$ parents.
- 11 Crossover
- 12 For each pair of selected parents, create offspring:
- 13 Choose a random crossover point c .
- 14 Create offspring matrix O using parents A_x and A_y : $O(j, :) = \begin{cases} A_x(j, :) & \text{if } j < c \\ A_y(j, :) & \text{if } j \geq c \end{cases}$
- 15 Mutation
- 16 For each offspring O , mutate each element $O(j, k)$ with probability μ .
- 17 Create New Generation
- 18 Combine parents and offspring to form a new population of size N .
- 19 Find Nonsingular Matrix
- 20 After G generations, search for a nonsingular adjacency matrix in the population.
- 21 If: Nonsingular matrix A_{final} is found:
- 22 Return A_{final} .
- 23 Else:
- 24 Return "No nonsingular adjacency matrix found."
- 25 End Algorithm.

This structured approach provides a clear and step-by-step overview of the Genetic Algorithm. It emphasizes the key operations of initialization, fitness evaluation, selection, crossover, mutation and generation evolution. The goal is to evolve a population towards a non-singular adjacency matrix.

When the input values of Algorithm 1 are $N = 700$, $V = 8$, $G = 200$, and $\mu = 0.01$, the proportion of nonsingular matrices over generations by generations is shown in Figure 1, and the average fitness score by generations and graph are shown in Figure 2.

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \end{bmatrix} \quad (5)$$

**Figure 1.** Proportion of Nonsingular Matrices Over Generations.

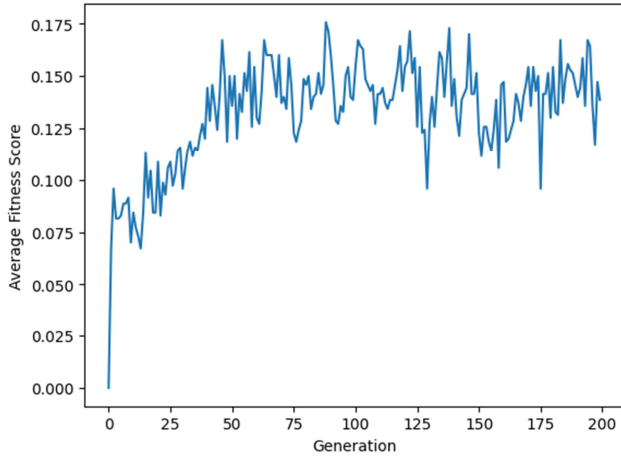


Figure 2. Average Fitness Score Over Generations.

3. Results

3.1. Affine Transformation by Non-Singular Adjacency Matrix

To implement the affine transformation used in the Rijndael's algorithm, the matrix $A(5)$ generated by the genetic algorithm is the irreducible polynomial $m(x) = x^8 + x^7 + x^6 + x^5 + x^4 + x + 1$, $b = (1, 0, 1, 0, 1, 0, 1, 1)$ -vector, and the invariant expression given in the vector algorithm was chosen. The non-singular adjacency matrix consists of the following set. For the graph $G=(V, U)$ $V=\{0, 1, 2, 3, 4, 5, 6, 7\}$, 8 vertices and $U=\langle(0, 1), (0, 2), (0, 6), (1, 0), (1, 4), (1, 6), (1, 7), (2, 0), (2, 1), (2, 2), (2, 4), (3, 3), (3, 4), (3, 5), (3, 6), (3, 7), (4, 0), (4, 2), (4, 3), (5, 4), (5, 6), (6, 0), (6, 1), (6, 3), (6, 4), (6, 5), (7, 2), (7, 6)\rangle$ contains a total of 36 elements with 28 edges. Reflecting the chosen parameters, affine transformation was executed utilizing the Rijndael algorithm, and the computed value of $S\{8 \times 8\}$ is displayed in Table 2 in hexadecimal format.

3.2. Performance Analysis of Proposed S-Box

To evaluate the efficiency of the algorithm, we executed it in Python and performed a thorough verification of the resulting substitution box (S-box). We compared the nonlinearity of the initially generated S-box and then the minimum, maximum, and average values of the Differential Approximation Probability (DAP) and the Strict Avalanche Criterion (SAC) with other research results.

3.3. Nonlinearity

Substitution box (S-box) Nonlinearity is a critical attribute in cryptographic algorithms, especially block ciphers, to ensure resistance to linear cryptanalysis. It refers to how much

the output of an S-box differs from that of any linear or affine function. The greater the nonlinearity, the more difficult it is for an attacker to approximate the S-box using linear equations, thereby increasing the security of the cipher. Nonlinearity is typically measured by the Hamming distance between the S-box output and the closest linear or affine function. Optimizing the nonlinearity of the S-box is critical to developing robust encryption schemes. To calculate the nonlinearity, the Walsh-Hadamard substitution technique is used, designated as (6).

$$S_{\langle f \rangle}(\omega) = \sum_{x \in GF(2^n)} (-1)^{f(x) \oplus x \cdot \omega} \quad (6)$$

The scalar product between $\omega \in GF(2^n)$, $x \cdot \omega$ and ω in a finite field. The nonlinearity of the $n \times n$ S-box is computed using Equation (7) [13].

$$N_f = 2^{n-1} \left(1 - 2^{-n} \max_{\omega \in GF(2^n)} |S_{\langle f \rangle}(\omega)| \right) \quad (7)$$

When calculating the nonlinearity of the eight f_i ($1 \leq f_i \leq 8$) Boolean functions in our designed 8×8 S-boxes, we obtained a remarkable average nonlinearity score of 112, with each individual score being 112. To evaluate the effectiveness of this S-box, we compared it with some recent chaos-based S-boxes in Table 3, which demonstrated its remarkable performance. Compared to other S-boxes, our proposed S-box showed superior statistical results in terms of minimum, maximum, and average nonlinearity values. High nonlinearity values for all eight Boolean functions in S-boxes are essential, as they help to reduce the input-output correlation and enhance the cryptographic robustness of the S-box [13].

3.4. Strict Avalanche Criterion

The Strict Avalanche Criterion (SAC) is an essential property of cryptographic substitution boxes (S-boxes) used to measure the responsiveness of output bits to changes in input bits. Introduced by Webster and Tavares in 1986, the SAC criterion is widely used to evaluate the robustness of cryptographic elements [19]. To determine whether the S-Box satisfies the SAC property, every possible pair of inputs differing by a single bit is generated, and the resulting output pairs are closely examined for analysis [20, 21]. The number of output bit variances among these pairs is then tallied and divided by the total number of output bits. An S-box is considered to satisfy the SAC property if this quotient is approximately 0.5. Table 4 shows the SAC matrix of the proposed S-box.

Table 2. Proposed S-Box in Values of Hexadecimal.

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	ab	0c	76	ed	9e	58	30	67	cf	66	c3	46	05	ee	31	be
1	65	f5	fc	ce	6d	82	e5	ae	61	37	b1	59	c8	a9	d6	28
2	7d	15	88	26	9f	e2	a8	62	a7	0f	a5	3d	ea	81	b4	bc

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
3	af	92	56	8e	5f	5e	0e	bb	36	9c	e7	b8	10	1f	bd	db
4	c5	00	03	0a	33	09	f9	72	02	d8	b9	cc	2a	34	2e	25
5	a3	4e	f7	b2	ef	6c	c0	5c	63	c2	24	86	40	b7	9a	7f
6	79	df	c7	04	87	6b	ad	12	90	51	5d	1a	3a	18	c9	19
7	9b	3c	83	4f	a6	8f	48	2d	1c	6f	95	1b	57	4b	d5	f8
8	f3	06	7e	5b	ff	41	e8	21	84	64	69	d9	80	7a	4c	fe
9	32	13	54	74	85	29	e4	f0	f1	22	d7	8d	23	e1	78	53
a	71	01	3f	08	c4	52	de	f4	7c	fa	6a	fd	ec	43	11	d1
b	e3	b3	a0	cd	b5	a4	77	8c	7b	55	c1	3e	1d	45	89	ca
c	17	2b	07	5a	bf	2c	ac	dd	ba	38	39	e6	35	98	50	da
d	8b	9d	d4	49	dc	d2	8a	cb	93	70	c6	f6	fb	97	0b	27
e	d0	b0	0d	3b	68	99	f2	75	6e	96	60	e9	a1	d3	a2	73
f	14	e0	eb	16	94	42	47	44	4a	b6	20	1e	91	aa	4d	2f

3.5. Differential Probability

Differential Probability (DP) measures the probability that a given input variance will result in a given output variance. It's determined by the ratio of the number of input pairings that produce the desired output variance to the total number of possible input pairings. Ideally, the substitution box should exhibit differential uniformity, where an input variance Δx_i matches an output variance Δy_i with a particular value, resulting in a particular reflection likelihood for each i . The differential convergence likelihood (DP), a metric for evaluating differential uniformity, measures this property of a given S-box (8).

$$DP(\Delta x \rightarrow \Delta y) = \left(\frac{\#\{x \in X \mid S(x) \oplus S(x \oplus \Delta x) = \Delta y\}}{2^m} \right) \quad (8)$$

where x is the set of all possible input bit values and 2^m is the number of its elements [13].

Table 3. Comparison of Nonlinearity Scores and Algebraic Degree of Some $\{8 \times 8\}$ S-boxes.

S-box	Nonlinearity			deg(f)
	Min	Max	Mean	
Proposed S-box	112	112	112	7
In [10]	110	112	110	7
In [22]	112	112	112	7
In [23]	112	112	112	7
In [24]	112	112	112	7
In [25]	112	112	112	7
In [26]	112	112	112	7
In [20]	112	112	112	7
AES [27]	112	112	112	7
SM4 [29]	112	112	112	7

Table 4. SAC Matrix Values of the Proposed S-box.

bit0	bit1	bit2	bit3	bit4	bit5	bit6	bit7
0,5156	0,5156	0,4844	0,5000	0,4531	0,5469	0,4844	0,5469
0,4688	0,5000	0,5000	0,4844	0,5156	0,4531	0,4844	0,4844
0,4531	0,5156	0,4844	0,4688	0,4531	0,5156	0,5156	0,4844
0,5469	0,5469	0,5156	0,4844	0,5000	0,4531	0,4688	0,5156
0,5000	0,5156	0,5000	0,4844	0,5000	0,5000	0,5313	0,5313
0,5156	0,5000	0,5156	0,5156	0,5313	0,5000	0,5156	0,5156
0,5000	0,4688	0,4844	0,4688	0,5000	0,5313	0,5156	0,4531
0,4844	0,5469	0,4688	0,5313	0,4844	0,5000	0,5313	0,5000

3.6. Linear Approximation Probability

The Linear Approximation Probability (LAP) of an S-box in cryptography measures its susceptibility to linear cryptanalysis. It quantifies how closely the input-output relationship of an S-box can be approximated by a linear function. As a key component in symmetric key ciphers such as AES, the security of the S-box depends largely on its LAP. A lower LAP is preferable, indicating greater resistance to linear attacks by obscuring input-output correlations. The LAP, determined by comparing S-box outputs with those of a linear approximation, is essential for evaluating and improving the security of cryptographic systems. Matsui's definition defines the linear convergence

probability of an S-box on the basis of the criteria mentioned above (9).

$$LAP = \max_{\Gamma_x, \Gamma_y \neq 0} \left| \frac{\#\{x \in X \mid x \bullet \Gamma_x = S(x) \bullet \Gamma_y\}}{2^m} - \frac{1}{2} \right| \quad (9)$$

where Γ_x and Γ_y are input and output masks, respectively; X is the set of all possible input bits, and 2^m is the number of its elements [13, 28]. The outcomes of the Strict Avalanche Criterion (SAC), Linear Approximation Probability (LAP), and Differential Probability (DP) evaluations with the suggested substitution box (S-box) and others are presented in Table 5.

Table 5. Comparison of SAC, LAP and DP of some 8×8 S-boxes.

S-box	SAC				LAP	DP
	Min	Max	Mean	Square deviation		
Proposed S-box	0.4531	0.5469	0.5000	0.0276	0.0625	4/256
In [10]	0.4219	0.5781	0.4953	0.0322	0.125	10/256
In [22]	0.4065	0.5859	0.4980	0.0426	0.0625	4/256
In [23]	0.4375	0.5625	0.5010	0.0323	0.0625	4/256
In [24]	0.4375	0.5469	0.4978	0.0340	0.0625	4/256
In [25]	0.3906	0.5781	0.4960	0.0353	0.0625	4/256
In [26]	0.4375	0.5469	0.4980	0.0349	0.0625	4/256
In [20]	0.4375	0.5625	0.5022	0.0354	0.0625	4/256
AES [27]	0.4531	0.5625	0.5048	0.0314	0.0625	4/256
SM4 [29]	0.4375	0.5625	0.4997	0.0346	0.0625	4/256

4. Conclusions

In summary, this research has developed an improved substitution box (S-box) design technique that achieves higher nonlinearity and improves strict avalanche criteria by selecting a non-singular neighboring matrix via a genetic algorithm. The proposed S-boxes were constructed using Rijndael's affine transformation. The robustness properties of the S-box, which include nonlinearity, differential probability, strict avalanche criterion, and linear approximation probability, were evaluated against other recently developed S-boxes. The minimum Strict Avalanche Criterion (SAC) value of the proposed S-box was 0.45, with a maximum of 0.54 and an average of 0.5. Furthermore, the comparative study, including graphs and tables, indicates that the proposed S-box exhibits superior defense against linear and differential attacks, especially in terms of the SAC parameter. The potential of these approaches to provide robust S-box values for symmetric block cipher algorithms will be demonstrated in future research.

ORCID

Javokhir Abdurazzokov: 0000-0002-8052-0078

Conflicts of Interest

The author declares no competing interests.

References

- [1] Abdurakhimov B, Boykuziyev I, Abdurazzokov J. Encryption systems and the history of their development. *InterConf*, 2022. <https://doi.org/10.51582/interconf.19-20.01.2022.085>.
- [2] Feistel H, Notz WA, Smith JL. Some cryptographic techniques for machine-to-machine data communications. *Proceedings of the IEEE* 1975; 63: 1545–54. <https://doi.org/10.1109/PROC.975.10005>.
- [3] Abdurakhimov B, Abdurazzokov J, Lingyun L. Analysis of the use of artificial neural networks in the cryptanalysis of the SM4 block encryption algorithm. *AIP Conf Proc* 2023; 2812: 020048. <https://doi.org/10.1063/5.0161859>.
- [4] Sattarov A. B, Abdurahimov B. F. An algorithm for constructing S-boxes for block symmetric encryption. *Universal Journal of Mathematics and Applications* 2018; 1: 29–32. <https://doi.org/10.32323/ujma.393155>.
- [5] Zhu D, Tong X, Zhang M, Wang Z. A New S-Box Generation Method and Advanced Design Based on Combined Chaotic System. *Symmetry (Basel)* 2020; 12: 2087. <https://doi.org/10.3390/sym12122087>.
- [6] Marochok S, Zajac P. Algorithm for Generating S-Boxes with Prescribed Differential Properties. *Algorithms* 2023; 16: 157. <https://doi.org/10.3390/a16030157>.
- [7] Abdurakhimov B, Boykuziev I, Allanov O, Xidirov B. Differential characteristics of reflections of Kuznyechik encryption algorithm. *2021 International Conference on Information Science and Communications Technologies (ICISCT)*, IEEE; 2021, p. 1–4.
- [8] Irfan M, Shah T, Siddiqui GF, Rehman A, Saba T, Bahaj SA. Design of Nonlinear Component of Block Cipher Using Gravesian Octonion Integers. *IEEE Access* 2023; 11: 2138–47. <https://doi.org/10.1109/ACCESS.2022.3217211>.
- [9] Zhang W, Pasalic E. Highly Nonlinear Balanced S-Boxes With Good Differential Properties. *IEEE Trans Inf Theory* 2014; 60: 7970–9.
- [10] 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: 827–33.
- [11] Wang Y, Zhang Z, Zhang LY, Feng J, Gao J, Lei P. A genetic algorithm for constructing bijective substitution boxes with high nonlinearity. *Inf Sci (N Y)* 2020; 523: 152–66. <https://doi.org/10.1016/j.ins.2020.03.025>.
- [12] Zahid AH, Tawalbeh L, Ahmad M, Alkhayyat A, Hassan MT, Manzoor A, et al. Efficient Dynamic S-Box Generation Using Linear Trigonometric Transformation for Security Applications. *IEEE Access* 2021; 9: 98460–75.
- [13] Kim G, Kim H, Heo Y, Jeon Y, Kim J. Generating Cryptographic S-Boxes Using the Reinforcement Learning. *IEEE Access* 2021; 9: 83092–104.
- [14] Ahmad M, Malik M. Design of chaotic neural network based method for cryptographic substitution box. *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*, IEEE; 2016, p. 864–8. <https://doi.org/10.1109/ICEEOT.2016.7754809>.
- [15] Li L, Liu J, Guo Y, Liu B. A new S-box construction method meeting strict avalanche criterion. *Journal of Information Security and Applications* 2022; 66: 103135. <https://doi.org/10.1016/j.jisa.2022.103135>.

- [16] Abdurazzokov J, Abdurakhimov B, Boykuziev I, Allanov O. Algorithm for Generating Robust S-Boxes Using Adjacency Matrix Parameters. *2023 10th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI), IEEE*; 2023, p. 372–7. <https://doi.org/10.1109/EECSI59885.2023.10295947>.
- [17] Gangadari BR, Ahamed SR. Analysis and algebraic construction of S-Box for AES algorithm using irreducible polynomials. *2015 Eighth International Conference on Contemporary Computing (IC3), IEEE*; 2015, p. 526–30. <https://doi.org/10.1109/IC3.2015.7346738>.
- [18] Islam M, Faruk O, Kar S, Islam M. Matrix Representation of Graph Theory with Different Operations. *IOSR Journal of Mathematics* 2022; 18: 8–27.
- [19] Tran MT, Bui DK, Duong AD. Gray S-Box for Advanced Encryption Standard. *2008 International Conference on Computational Intelligence and Security, IEEE*; 2008, p. 253–8. <https://doi.org/10.1109/CIS.2008.205>.
- [20] Mahmood Malik MS, Ali MA, Khan MA, Ehatisham-Ul-Haq M, Shah SNM, Rehman M, et al. Generation of Highly Nonlinear and Dynamic AES Substitution-Boxes (S-Boxes) Using Chaos-Based Rotational Matrices. *IEEE Access* 2020; 8: 35682–95. <https://doi.org/10.1109/ACCESS.2020.2973679>.
- [21] Aboytes-González JA, Murguía JS, Mejía-Carlos M, González-Aguilar H, Ramírez-Torres MT. Design of a strong S-box based on a matrix approach. *Nonlinear Dyn* 2018; 94: 2003–12.
- [22] Nitaj A, Susilo W, Tonien J. A New Improved AES S-box with Enhanced Properties. *IACR Cryptol EPrint Arch* 2020; 2020: 1597.
- [23] Nizam Chew LC, Ismail ES. S-box Construction Based on Linear Fractional Transformation and Permutation Function. *Symmetry (Basel)* 2020; 12: 826.
- [24] Chen G, Chen Y, Liao X. An extended method for obtaining S-boxes based on three-dimensional chaotic Baker maps. *Chaos Solitons Fractals* 2007; 31: 571–9. <https://doi.org/10.1016/j.chaos.2005.10.022>.
- [25] Daemen J. AES Proposal : Rijndael, 1998.
- [26] Webster AF, Tavares SE. On the Design of S-Boxes. *Advances in Cryptology — CRYPTO '85 Proceedings, Berlin, Heidelberg: Springer Berlin Heidelberg*; n.d., p. 523–34. https://doi.org/10.1007/3-540-39799-X_41.
- [27] Mahboob A, Asif M, Siddique I, Saleem A, Nadeem M, Grzelczyk D, et al. A Novel Construction of Substitution Box Based on Polynomial Mapped and Finite Field With Image Encryption Application. *IEEE Access* 2022; 10: 119244–58. <https://doi.org/10.1109/ACCESS.2022.3218643>.
- [28] Matsui M. Linear Cryptanalysis Method for DES Cipher, 1994, p. 386–97. https://doi.org/10.1007/3-540-48285-7_33.
- [29] Pu S, Guo Z, Liu J, Gu D, Yang Y, Tang X, et al. Boolean Matrix Masking for SM4 Block Cipher Algorithm. *2017 13th International Conference on Computational Intelligence and Security (CIS), IEEE*; 2017, p. 238–42. <https://doi.org/10.1109/CIS.2017.00059>.