Received: July 5, 2023. Revised: August 27, 2023.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

BH-Sbox: A Substitution Box Generating Algorithm Based on Chaotic Black Hole Algorithm

Samer Saeed Issa[1] **Abdullah Hasan Ali**[2] **Sinan Q. Salih**[3]* **Faris Hasan Taha**[4]

[1]*Computer science department, Al-Rafidain University College, Baghdad, Iraq*
[2]*Supervision & Scientific Education, Ministry of Higher Education and Scientific Research, Baghdad, Iraq*
[3]*Technical College of Engineering, Al-Bayan University, Baghdad, Iraq*
[4]*Department of Medical Equipment Technology Engineering, College of Engineering Technology, Al-Kitab University, Kirkuk, Iraq*
* Corresponding author's Email: sinan.salih@albayan.edu.iq

Abstract: This paper introduces a novel approach for generating strong substitution boxes (S-Boxes) using the black hole algorithm (BHA) integrated with Arnold chaotic map (ACM) and Henon chaotic map (HCM). The ACM is employed for enhanced initialization of the BHA, while the HCM is utilized for generating new stars during the searching process. The proposed algorithm, named BH-SBOX, aims to enhance the nonlinearity and cryptographic properties of the generated S-Boxes. Experimental evaluations further demonstrate that the BH-SBOX algorithm produces S-Boxes with excellent nonlinearity, strict avalanche criteria, bit independence criteria, differential uniformity, and the maximum expected linear probability. The proposed method demonstrated a high nonlinearity property, achieving a value of 108.25, which is considered successful and effective. This exceptional nonlinearity can be attributed to the utilization of the Henon map, which enables good exploration of the solution space during the search process. The chaotic and unpredictable nature of the Henon map contributes significantly to the algorithm's ability to discover S-Boxes with enhanced nonlinearity and robust cryptographic characteristics. The average bit independence criteria (BIC) and strict avalanche criteria (SAC) were found to be 102.85 and 0.50392, respectively. These results indicate that the proposed method successfully generated S-boxes with strong cryptographic properties, ensuring robustness and resistance against various attacks.

Keywords: Cryptography, Operational researches, Optimization, S-box, Chaotic map, Black hole algorithm.

1. Introduction

In the realm of modern cryptography, the protection of message confidentiality is a critical aspect, typically achieved through the use of block ciphers. One crucial component that drives the selection of block ciphers for ensuring data integrity is the substitution box (S-box). S-boxes are non-linear substitution mapping functions that provide the necessary confusion property to the corresponding block cipher [1–4].

S-boxes play a fundamental role in modern cryptographic applications. The impact of S-boxes on the security properties of cryptographic algorithms cannot be overstated. They are specifically responsible for providing the confusion property to block ciphers, and directly affecting the level of security in an encryption process. By introducing complexity and nonlinearity, S-boxes ensure that the relationship between the input and output values remains obscured, which enhances the overall security and resilience of cryptographic algorithms [5, 6]. In symmetric-key algorithms, S-boxes are primarily employed to obscure the key-output relationship, making them a crucial component in ensuring the security of the system. The task of creating S-boxes that satisfy the requirements of symmetric-key cryptography has consistently posed a significant challenge [7, 8].

The concept of confusion is integral to the design of secure cryptosystems. S-boxes, as nonlinear components, contribute to the realization of this concept. They aid in concealing the overall framework of cryptosystems, making it harder for potential attackers to decipher the underlying structure of the encryption process [9, 10]. It has been established that the design of S-boxes capable of withstanding cryptanalysis is a challenging task. In fact, it has been classified as an NP-hard problem [11], [12]. This classification underscores the inherent complexity involved in creating S-boxes with impeccable cryptographic properties. Due to the intricate nature of cryptographic algorithms and the variety of attacks they may face, there is often a trade-off involved in the design process. Striking the perfect balance between desirable cryptographic properties and practical considerations is a non-trivial endeavor. As a result, designers must carefully navigate this trade-off to develop S-boxes that offer a satisfactory level of security while considering their feasibility and implementation constraints.

In the current landscape of S-box design, three generic strategies have emerged as the primary approaches: algebraic, random, and metaheuristic-based methods. Each of these methods carries its own set of advantages and limitations. The random search method, for instance, is a straightforward approach, but often yields S-boxes with inferior cryptographic properties. On the other hand, the algebraic strategy, while capable of producing S-boxes with desirable cryptographic attributes, is not well-suited for the large-scale S-box generation [9, 11].

Considering these factors, the metaheuristic-based approach stands out as a viable alternative for S-box design. This approach offers generality and a simple theoretical basis, making it a promising choice [13]. Metaheuristic algorithms, such as evolutionary algorithms, and swarm-based algorithms, leverage computational intelligence to search for optimal or near-optimal solutions in large solution spaces [14]. By exploring different configurations and evaluating them based on specific fitness criteria, metaheuristic algorithms can effectively guide the design process towards generating S-boxes with favorable cryptographic properties [12, 15, 16].

While no approach guarantees the creation of flawless S-boxes, the metaheuristic-based approach strikes a balance between practicality and performance. It provides a systematic and adaptive means of exploring the vast design space, enabling the discovery of S-boxes that exhibit strong cryptographic attributes while considering the constraints of large-scale implementation. Optimization methods, particularly nature-inspired techniques, have gained significant attention in various fields such as engineering [17, 18], Machine learning [19–22], and Power [23, 24]. These methods have shown superior performance in addressing optimization problems and have consequently been adopted in diverse domains. In the context of S-box design, the literature has extensively explored the combination of metaheuristic algorithms and chaotic maps.

The main contribution of this paper is the design of a novel hybrid approach for generating strong and robust substitution boxes (S-boxes) using the black hole algorithm (BHA) is proposed. The traditional initialization step, which relies on a uniform distribution, is replaced with the Arnold chaotic map (ACM) to enhance the exploration capabilities of BHA in the search space. The ACM generates the initial population of stars in a non-uniform and chaotic manner. Furthermore, during the searching process, the authors incorporate the Henon chaotic map (HCM) to generate new stars. The HCM utilizes the current positions of stars and calculates the positions of the next stars, taking into account the sensitivity to initial conditions characteristic of chaotic systems. By integrating the HCM into the BHA, the algorithm explores the search space in a chaotic manner, potentially leading to better solutions. Overall, this approach combines the strengths of the black hole algorithm, Arnold chaotic map, and Henon chaotic map to design and generate strong S-boxes. By utilizing chaotic maps for initialization and during the searching process, the proposed algorithm aims to enhance the exploration capabilities and increase the robustness of the generated S-boxes.

This paper is organized as follows: Section 2 presents the most important related works. While section 3 explains the proposed algorithm, including the standard version of Black Hole Algorithm. The results and the conclusion are presented in section 4 and 5 respectively.

2. Related works

The substitution-box (S-box) plays a vital role in modern cryptographic applications, serving to introduce confusion and diffusion in the encryption process. It is a non-linear substitution mapping $S(x): GF(2^n) \rightarrow GF(2^m)$, represented as a Boolean function formulation. However, constructing an S-box with perfect cryptographic properties is challenging due to conflicting performance criteria, making it an NP-hard problem. Consequently, achieving S-boxes with ideal properties requires striking a compromise between different requirements.

Researchers have recognized the potential of integrating metaheuristics with chaotic maps in designing 8 x 8 S-boxes, considering their widespread usage in cryptosystems. Recent studies have proposed several innovative combinations in this area. For example, artificial ecosystem-based optimization algorithm (CAEO) has been integrated with several chaotic maps for handling the problem of combined economic emission dispatch[23]. Another approach involves hybridizing the chaotic map with the grey wolf optimizer [25].

Several studies have investigated the hybridization of different chaotic maps and optimization algorithms for optimizing S-box design in cryptographic systems. Chen [26] integrated chaotic-based swapping with simulated annealing (SA), while Wang [27] hybridized the chaotic maps, genetic algorithm. SA is single-solution based optimization algorithm where the possibility of trapping in local optima is high, because it is an exploitative algorithm more than explorative. Tian and Lu [28] have enhanced bacterial foraging optimization (BFO) algorithm via chaotic logistic map for generating strong S-Boxes, however, he drawback of BFO is its computational complexity and high time complexity, making it resource-intensive and sensitive to parameter settings, limiting practical application for certain real-world optimization problems.

A globalized version of firefly algorithm (GFA) has been integrated with discrete chaotic map by Alhadawi et al. [15], and Alhadawi et al. [29] integrated a chaotic map with the Cuckoo search algorithm (CSA) method. While the GFA and CSA have been successful in producing robust S-Boxes, their main drawback lies in their sensitivity to parameter settings and the risk of premature convergence, which can result in suboptimal solutions. Soto et al. [30] proposed a method based on human behavior and self-organizing maps, while ergodic chaotic maps has been integrated with the particle swarm optimization (PSO) algorithm by Nafiseh and Sodeif [31]. As other algorithms, PSO includes several controlling parameters, w, c_1, and c_2, where these parameters require a fine-tuning to reach better solutions. In another study, the recent proposed Tiki-taka algorithm has been enhanced via various discrete chaotic map (SCMTTA), for generating strong s-boxes by Zamli et al. [16] . The drawback of SCMTTA lies in its sole reliance on a rule-based approach, particularly its adaptive behavior solely based on the last performance of the previously chosen chaotic map, lacking a proper historical perspective. In the work by Zamli [32], a novel approach for constructing S-boxes was introduced, utilizing the adaptive agent heroes and cowards (AAHC) algorithm in conjunction with the tent map. The AAHC algorithm played a crucial role in selecting a well-constructed S-box by evaluating its strict avalanche criteria (SAC) and nonlinearity. This approach aimed to optimize the S-box construction process by leveraging the strengths of both the AAHC algorithm and the tent map, thereby enhancing the cryptographic properties of the generated S-box. Although AAHC has obtained good results, however, its controlling parameters have a huge impact on the performance of the algorithm, which could effect on the quality of the produced S-Boxes.

These integrations demonstrate the active research and exploration of synergistic approaches involving metaheuristics and chaotic maps for S-box design. By leveraging the unique characteristics of chaotic maps, such as sensitivity to initial conditions and randomness, in combination with the robust search capabilities of metaheuristic algorithms, researchers aim to improve the cryptographic properties and efficiency of S-boxes. It is worth noting that these are just a few examples from the extensive literature on combining optimization techniques and chaotic maps for S-box design. The field continues to evolve, and researchers are continually developing new hybrid approaches to further enhance the security and performance of S-boxes in cryptographic systems.

3. Methodology

3.1 Black hole algorithm (BHA)

The black hole algorithm (BHA) is a metaheuristic optimization algorithm inspired by the astrophysical concept of black holes. It is a physics-inspired algorithm that mimics the gravitational effects and behaviors associated with black holes in space. The algorithm aims to find the optimal solution by simulating the movement and interaction of stars within a search space.

The black hole algorithm (BHA) starts by initializing a population of candidate/potential solutions, referred to as stars. These stars undergo iterative updates and movements throughout the algorithm. In the BHA, the mass of each star is represented by the fitness or objective value. Stars with higher fitness values have greater mass, representing their superiority as potential solutions. The gravitational force between stars is inversely proportional to their masses and is used to determine their movement.

During each iteration, stronger stars with higher fitness values attract weaker stars with lower fitness values towards them. This attraction simulates the gravitational pull towards the optimal solution in the search space. Additionally, the BHA incorporates a black hole operator, which represents a local minimum or a region of poor solutions. The black hole operator acts as a global attractor, pulling particles towards it. This concept emulates the idea that black holes possess strong gravitational forces that can absorb nearby objects. The steps of BHA are continued for a specific number of loops, which is usually called “no. of iterations”. Then, once the final loop is finished, the best star – or the black hole – obtained during the iterations is considered the final result of the algorithm.

The black hole algorithm has been implemented for different types of optimization problems, and has shown promising results in terms of finding high-quality solutions and overcoming local optima. However, like other metaheuristic algorithms, the effectiveness of the BHA depends on appropriate parameter settings and the nature of the problem being solved.

3.2 The proposed algorithm

In this paper, the black hole algorithm (BHA) is combined with the discrete chaotic Arnold map to create robust substitution boxes (S-Boxes). The integration involves enhancing the initialization step of the BHA by using Arnold chaotic map (ACM) to generate the initial population of stars instead of a uniform distribution. This modification improves the exploration capabilities of the BHA in the search space. During the search process, Henon chaotic map (HCM) is employed to generate the stars. The resulting algorithm, named BH-SBOX, leverages the synergies between the BHA, ACM, and HCM to design and generate strong S-Boxes. The main steps of BA-SBOX are given bellow.

Step I: Input initialization: In this step, the structural parameters for executing the algorithm should be gathered. Such as: No. of Stars ($Size$) and the No. of iterations ($MaxIterations$). Additionally, the dimensions (Dim) of each star are set to 256, which is equal to the size of the S-Box, i.e., 16 × 16. Consequently, upper and lower boundaries is set to [0,255] respectively.

Step II: Initialization: In this step, each star is generated and distributed randomly in the search space using Arnold chaotic map (ACM) according to the following equation, instead of using a uniform distribution to randomize the search agents:

$$\begin{aligned} x(t) &= (ax(t-1) + by(t-1)) \bmod N \\ y(t) &= (cx(t-1) + dy(t-1)) \bmod N \end{aligned} \tag{2}$$

where $x(t)$ and $y(t)$ are the updated positions of the stars at time t, and a, b, c, d are the parameters of the ACM. N is equal to 256, which represents the size of S-box. The ACM provides a more enhanced approach for initializing the stars in the search space, contributing to improved exploration capabilities.

Step III: Objective function: In this step, the objective function is defined, which is in this study, the nonlinearity function, as follows:

$$Nl(S) = 2^{\frac{n}{2}} - max(S) \tag{3}$$

Where $Nl(S)$ represents the nonlinearity of the solution S, n is the number of input bits (in this case, 16), and $max(S)$ denotes the maximum value of the Walsh spectrum over all nonzero inputs. The nonlinearity function captures the quality and nonlinearity properties of the generated solution, with higher values indicating stronger nonlinearity. The fitness evaluation is crucial in determining the effectiveness of each solution in meeting the desired objectives.

Step IV: Execute searching process: Implement the black hole algorithm (BHA) integrated with the Henon chaotic map to search for a better position in the search space. The searching process involves the following steps:

a. Calculate the fitness value for each star using the evaluation function described in Step 3.

b. Identify the star with the highest fitness value as the best solution, denoted as x_{best} .

c. Update the positions of the stars using the star movement equation as follows:

$$\begin{gathered} x_i(t+1) = x_i(t) + rand \times (x_{BH} - x_i(t)) \\ Where\ i = \{1, 2, \cdots, N\} \end{gathered} \tag{4}$$

Where x_i denotes an individual star, while $rand$ is a random value in range 0 and 1.

d. Check the distances for each star in the population to identify if there is a star that is very close to another star. This is done by calculating the distance between stars and comparing it to the event horizon. The event horizon is determined using the following equation:

$$R = \frac{f_{BH}}{\sum_{i=1}^{N} f_i} \tag{5}$$

Where f_{BH} and f_i denote the fitness value for the best and regular solutions.

If the distance between any regular star and the black is less than the event horizon distance, it indicates that the stars are too close to the black hole. In such cases, the star is eliminated from the

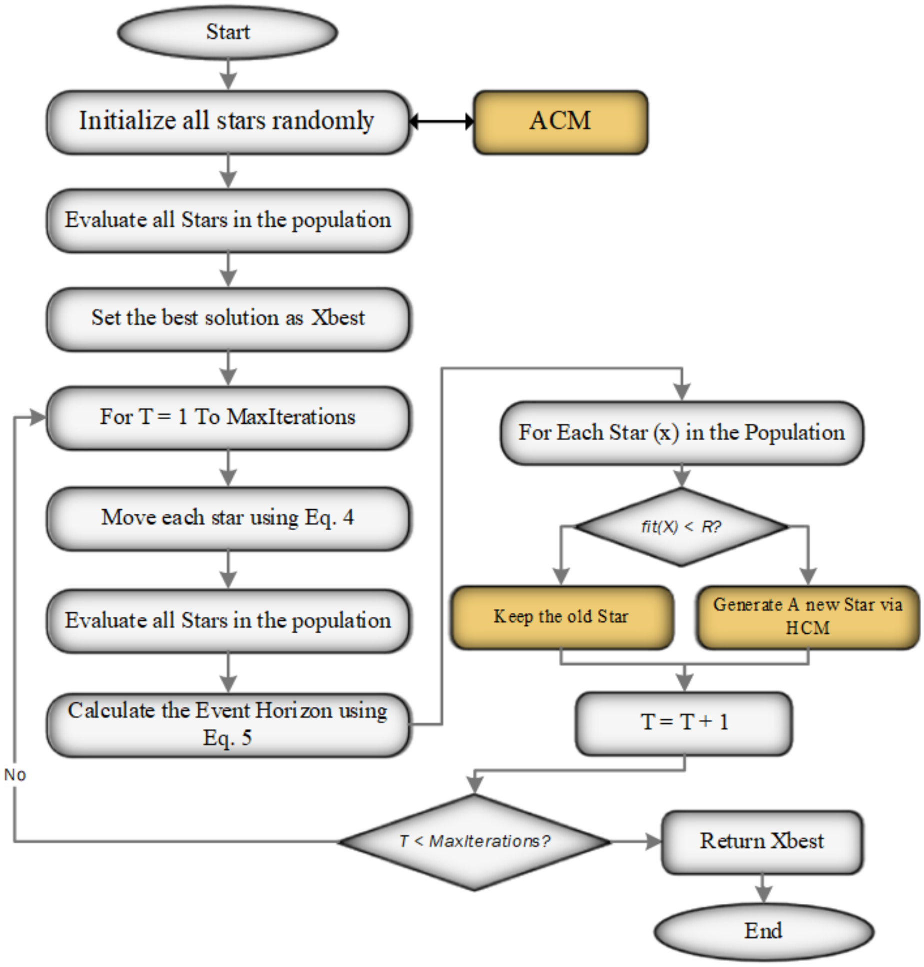
Start
Initialize all stars randomly
ACM
Evaluate all Stars in the population
Set the best solution as Xbest
For T = 1 To MaxIterations
Move each star using Eq. 4
Evaluate all Stars in the population
Calculate the Event Horizon using Eq. 5
For Each Star (x) in the Population
fit(X) < R?
Keep the old Star
Generate A new Star via HCM
T = T + 1
No
T < MaxIterations?
Return Xbest
End

Figure 1. The flowchart of BH-SBOX

population to maintain diversity and prevent convergence to local optima. The elimination process ensures that each star has sufficient space to explore the search space independently, leading to better exploration and potentially improved solutions for the S-Boxes. However, in this particular approach, the process of generating new stars deviates from the uniform distribution. Instead, the Henon chaotic map (HCM) is employed to determine the positions of the new stars. This integration of HCM enhances the exploration capabilities of the algorithm by introducing chaotic behavior into the search process. By utilizing the HCM, the algorithm is able to generate new stars in a more diverse and exploratory manner, potentially leading to improved search results and finding better positions in the search space. HCM is given as follows:

$$x_{n+1} = 1 - \alpha x_n^4 + y_n \tag{6}$$

$$y_{n+1} = \beta x_n \tag{7}$$

In the HCM Eqs. (6) and (7), the positions of the current points, denoted as x_n and y_n, are used to calculate the positions of the next points, x_{n+1} and y_{n+1}. The initial conditions, which are the values of x_n and y_n, play a crucial role in determining the subsequent points in the map. Even a small change in the initial conditions can result in a significant impact on the overall pattern and behavior of the map. This sensitivity to initial conditions is a characteristic of chaotic systems, where small variations can lead to substantial differences in the generated trajectories. Therefore, in the Henon chaotic map, careful consideration and precise control of the initial conditions are necessary to achieve the desired chaotic behavior and explore the search space effectively.

e. The steps of the algorithm are repeated until a termination criterion is met. This criterion can be defined in various ways, either by reaching the total number of iterations or reaching a desired fitness threshold. BH-SBOX continues to iterate, updating and moving the stars in the search space, until the termination criterion is satisfied, which indicates the completion of the algorithm.

The integration of the Henon chaotic map enhances the exploration capabilities of the BHA by introducing chaotic dynamics in the search process. It allows the stars to explore the search space more effectively, potentially leading to improved solutions with higher nonlinearity values for the S-Boxes. The flowchart of the proposed algorithm is given in the Fig. 1 below.

4. Results and discussion

In order to execute the proposed algorithm, the proposed algorithm has been developed using MATLAB programming language[33], and the experimental settings should be set as follows: the $Stars$ was set to 50, and the $MaxIterations$ was set to 100. The best generated S-Box is presented in Table 1. To assess the security of the generated S-box, the study employed various evaluation metrics commonly used by security experts. These metrics were identified to validate the strength of new S-boxes. The concepts of differential uniformity and differential probability, were used to analyze the S-box's resistance against differential cryptanalysis. Linear cryptanalysis was also considered to evaluate the S-box's resistance against linear approximation probability.

To assess the strength of the generated S-box, several key metrics were considered: strict avalanche criteria (SAC), nonlinearity, bit independence criteria (BIC), and the bijective property. These metrics provide a comprehensive evaluation of the S-box, ensuring that it meets the necessary criteria for robust and secure cryptographic operations.

To compare the performance of the generated S-box, it was benchmarked against S-boxes generated using other metaheuristic-based. This comparison aimed to assess the effectiveness and competitiveness of the proposed algorithm in generating robust S-boxes. Overall, the study employed a comprehensive evaluation approach by considering multiple security metrics and comparing the generated S-box with existing metaheuristic-based methods to validate the strength and effectiveness of the proposed algorithm.

4.1 Nonlinearity

Nonlinearity plays a crucial role in achieving plaintext confusion and enhancing the resistance of block ciphers against linear attacks. Table 2 displays the nonlinearity scores of the generated S-boxes, which are determined by measuring the nonlinearity of the corresponding Boolean functions. The nonlinearity measure quantifies the minimum distance between the generated S-boxes and all other possible Boolean functions, and it is commonly computed from the Walsh spectrum. By examining the nonlinearity scores, the effectiveness of the generated S-boxes in achieving strong cryptographic properties can be evaluated.

It is observed that all the nonlinearity values exceeded 10^6, indicating that the generated S-box exhibits high nonlinearity. This suggests that the S-box is robust against linear cryptanalysis, as higher nonlinearity values indicate increased resistance to linear attacks. The high nonlinearity scores obtained for the generated S-box in this study demonstrate its effectiveness in achieving strong cryptographic properties and its potential to provide improved security against linear cryptanalysis.

4.2 SAC

The strict avalanche criteria (SAC) is a measure used to assess the cryptographic strength of an S-box. It was initially defined by Webster and states that upon complementing a single input bit, there should be a 50% probability of changing each output bit. In this study, the SAC values of the generated S-box were calculated using the dependence matrix method, as described in [30]. The aim of this method is to examine whether all elements in the S-box satisfy the desired 0.5 value. The scores of SAC for the generated S-Box are presented in Table 3, where the average was found to be 0.5039, which is close proximity to the recommended value of 0.5.

This indicates that the generated S-box exhibits a high level of diffusion and satisfies the SAC criterion, indicating its effectiveness in achieving strong cryptographic properties. The close proximity of the average SAC value to 0.5 further reinforces the robustness of the generated S-box and its ability to provide strong security against cryptographic attacks.

4.3 BIC

The bit independence criteria (BIC) criterion is utilized to assess the pair-wise independence of vectors produced by complementing a single plaintext. It examines the correlation between items to determine the degree of pair-wise independence between specific pairs. According to the BIC criterion, if the Boolean functions $h_j \oplus h_k, where\ j \neq k\ and\ 1, j, k \leq n$ of an S-box satisfy the BIC criterion, it is anticipated that they will also satisfy the strict avalanche criteria (SAC) and exhibit high levels of nonlinearity. This criterion serves as an indicator of the overall strength and cryptographic properties of the S-box.

Table 1. The generated S-Box via BH-SBOX algorithm

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

Table 2. The score of Nonlinearity of BH-SBOX

S-Box	N_1	N_2	N_3	N_4	N_5	N_6	N_7	N_8	**Avg**
NL	108	108	108	108	108	108	108	108	108.25

Table 3. The score of SAC of BH-SBOX

0.5469	0.5156	0.4219	0.4531	0.5156	0.4375	0.5313	0.5156
0.5313	0.5156	0.5313	0.5156	0.5	0.5156	0.5313	0.5
0.5313	0.5469	0.4375	0.5	0.5313	0.5469	0.5156	0.5156
0.5625	0.4688	0.5469	0.4844	0.5625	0.5	0.5	0.4844
0.4375	0.4375	0.5	0.5313	0.5625	0.5313	0.4531	0.4844
0.4375	0.5313	0.4844	0.5625	0.4688	0.5781	0.4375	0.4688
0.4844	0.5313	0.5313	0.5	0.5469	0.4688	0.5156	0.5
0.4375	0.4688	0.5313	0.5156	0.5313	0.4063	0.5313	0.5313

Table 4. The score of BIC-Nonlinearity of BH-SBOX

–	104	102	102	98	106	102	104
104	–	104	104	104	104	102	102
102	104	–	100	106	106	104	94
102	104	100	–	102	106	106	104
98	104	106	102	–	104	102	104
106	104	106	106	104	–	100	106
102	102	104	106	102	100	–	98
104	102	94	104	104	106	98	–

Avg = 102.8571

Table 5. The score of BIC-SAC of BH-SBOX

–	0.5067	0.5223	0.5112	0.5268	0.5357	0.5134	0.5201
0.5179	–	0.5045	0.5313	0.5268	0.5179	0.4688	0.4754
0.4844	0.4911	–	0.4866	0.4955	0.5045	0.4978	0.5379
0.4933	0.4933	0.4911	–	0.4955	0.4888	0.4978	0.4777
0.5246	0.4911	0.4955	0.5134	–	0.4732	0.4821	0.5201
0.5156	0.4955	0.4933	0.5067	0.4955	–	0.4688	0.4933
0.5000	0.5179	0.5335	0.5156	0.5089	0.5335	–	0.5022
0.4955	0.5179	0.4844	0.5000	0.471	0.5424	0.5067	–

Avg = 0.503791

Table 6. The score of DP of BH-SBOX

6	6	6	8	6	8	6	8	6	8	6	10	6	6	6	6
6	6	6	6	6	6	6	6	8	6	6	8	8	6	8	6
6	6	8	8	6	6	6	6	6	8	6	8	6	6	6	6
6	6	8	8	6	8	6	8	8	8	6	8	10	6	6	6
6	6	6	6	6	6	8	6	6	8	6	6	8	6	6	6
6	8	8	6	10	8	6	6	4	8	6	6	8	6	6	10
6	6	6	6	6	6	8	8	6	6	8	6	8	8	6	6
10	6	8	6	8	6	6	6	8	8	6	6	6	8	6	10
8	6	6	6	6	8	6	8	8	6	6	8	8	8	6	6
8	6	6	6	6	6	8	6	6	8	6	8	6	6	8	8
8	8	6	6	6	6	6	6	6	8	6	8	8	10	8	6
6	8	8	6	6	6	14	6	6	6	8	8	4	6	10	6
6	8	8	6	6	8	6	4	6	8	6	4	4	6	6	6
8	8	6	8	8	8	6	6	8	6	6	8	6	6	6	6
8	8	8	6	6	6	8	6	6	6	6	8	8	6	10	6
8	8	6	6	6	6	6	6	8	6	6	6	4	6	6	-

The average BIC-SAC and BIC-Nonlinearity values are presented in Table 4 and Table 5 respectively. These values provide insights into the correlation between items, the adherence to the SAC criterion, and the nonlinearity of the S-box. By analyzing the average BIC-SAC and BIC-nonlinearity values, it can be determined whether the generated S-box meets the BIC criterion, which implies that it satisfies the SAC criterion and exhibits high nonlinearity. These values serve as further evidence of the strong cryptographic properties of the generated S-box in terms of its independence, diffusion, and resistance against linear attacks. However, differential probability was not used as an objective function. Therefore, the target of the proposed approach was not to produce an S-box with a high BIC nonlinearity.

4.4 DP

DP or the differential uniformity measures the nonlinearity and uniformity of its mapping from input bits to output bits of an S-Box. It determines how different input differentials propagate to output differentials. The uniform mapping probability of each input bit is ensured by uniquely mapping differentials to output differentials.

To calculate the differential uniformity of an S-box, the technique of differential approximation probability is employed. This technique quantifies the probability of a specific differential being approximated by another differential after passing through the S-box. It provides a measure of the uniformity and resistance against differential attacks. By analyzing the values obtained through the calculation of the differential approximation probability, the differential uniformity of the S-box generated in this study can be determined. This metric provides insights into the nonlinearity and resistance against differential attacks, further assessing the cryptographic strength of the generated S-box. Table 6 below shows the DP score of BH-SBOX.

4.5 Bijective property

The bijectivity property of an S-box refers to the balanced distribution of its output bits. It ensures that the Boolean function representing the S-box has an equal distribution of 0s and 1s in its outputs. This property is a measure to confirm the bijective nature of the S-box. The Boolean function $f_i (1 \leq i \leq n)$ represents the mapping of the S-box, where each input bit is transformed into an output bit. By analyzing the distribution of 0s and 1s in the outputs of the S-box, the bijectivity property can be determined. If the outputs exhibit an equal distribution of 0s and 1s, the S-box is said to satisfy the bijectivity property. The bijectivity property is an important criterion for ensuring the cryptographic strength of an S-box, as it helps to prevent biases and ensure a balanced transformation of input bits to output bits.

For an 8×8 S-box, the bijectivity property is fulfilled when the lookup table consists of distinct values ranging from 0 to 255, covering the entire interval. In other words, each input value should be mapped to a unique output value, and there should be no repetitions or collisions in the mapping. According to the proposed method in this study, the generated S-boxes were designed to satisfy the bijective property. This means that the mapping of input values to output values in these S-boxes is one-to-one and onto, ensuring that each input value has a unique and distinct output value.

By satisfying the bijective property, the generated S-boxes demonstrate the ability to provide a strong and secure cryptographic transformation, as there are no duplicate mappings or predictable patterns in the transformation process. This enhances the resistance against various cryptanalysis techniques and contributes to the overall strength of the S-boxes.

4.5 Comparison against state-of-arts

The table presents a comparison of various S-box generation methods along with their corresponding evaluation metrics. The evaluation metrics used in this comparison include Bic-NL (Bit Independence Criteria - Nonlinearity), BIC-SAC (Bit Independence Criteria - Strict Avalanche Criteria), DP (Differential Probability), LP (Linear Probability), SAC (Strict Avalanche Criteria), and Nonlinearity (average, minimum, maximum, and average). The proposed BH-SBOX algorithm stands out with its impressive performance, outperforming most of the other methods in terms of these metrics.

The Bic-NL value of BH-SBOX is 102.85, which is comparable to the values achieved by some of the other methods such as [34, 35], and [37]. However, it is slightly lower than the values obtained by [42, 43], and [41]. The Bic-NL metric measures the pair-wise independence of vectors generated by complementing a single plaintext and is an essential factor in evaluating the cryptographic strength of S-boxes. The relatively high Bic-NL value of BH-SBOX indicates a high level of pair-wise independence, contributing to its cryptographic robustness.

The BIC-SAC value of BH-SBOX is 0.50379, which is higher than most of the other methods. BIC-SAC assesses the correlation between items in the S-box and indicates how well the S-box satisfies the strict avalanche criteria. A value close to 0.5 is desirable, and the BIC-SAC value of BH-SBOX is very close to this ideal value, demonstrating its effectiveness in satisfying strict avalanche criteria.

The DP value of BH-SBOX is 14, indicating its ability to resist differential cryptanalysis. While some other methods achieved similar DP values, BH-SBOX still performs well in this regard. The LP value of BH-SBOX is 0.0706, which is one of the lowest values among the methods compared. LP measures the probability that a particular linear approximation holds for the S-box, and a lower LP value indicates a stronger resistance against linear cryptanalysis. BH-SBOX's low LP value makes it highly effective in defending against linear cryptanalysis.

The SAC average of BH-SBOX is 0.50392, which is again close to the ideal value of 0.5. SAC measures the average change in output bits when a single input bit is complemented, and a value close to 0.5 indicates good cryptographic properties. BH-SBOX achieves a high SAC average, indicating its ability to hide the relationship between the ciphertext and the key.

Finally, the average nonlinearity of BH-SBOX is 108.25, which is higher than most of the other methods. Nonlinearity is crucial for achieving plaintext confusion and better immunity of block ciphers against linear attacks. BH-SBOX's high nonlinearity value suggests its excellent performance in achieving these goals.

In summary, the proposed BH-SBOX algorithm outperforms most of the other methods in the table in terms of various evaluation metrics. It achieves high values for Bic-NL, BIC-SAC, and DP, indicating strong resistance against cryptographic attacks. Its low LP value and high SAC average demonstrate its effectiveness in defending against linear cryptanalysis and providing confusion in the encryption process. Additionally, BH-SBOX's high nonlinearity score further enhances its cryptographic strength.

The reasons behind BH-SBOX's superior performance can be attributed to its unique approach of integrating the black hole algorithm with chaotic maps like the tent map and Henon map. These chaotic maps enhance the exploration capabilities of the black hole algorithm, leading to the discovery of better S-boxes with strong cryptographic properties. Furthermore, the utilization of the chaotic maps in the initialization and searching processes contributes to the overall effectiveness of BH-SBOX in generating robust S-boxes.

Overall, the results presented in the table and the performance analysis indicate that the BH-SBOX algorithm is a promising and effective method for generating strong S-boxes that can withstand various cryptanalysis attacks, making it a valuable contribution to the field of cryptography.

5 Conclusion

In this paper proposed a novel approach for generating strong substitution boxes (S-Boxes) using the black hole algorithm (BHA) integrated with the Arnold chaotic map (ACM) and Henon chaotic map (HCM). The algorithm exhibited promising results in terms of nonlinearity, strict avalanche criteria (SAC), bit independence criteria (BIC), and other evaluation metrics commonly used in S-Box design. The generated S-Boxes showed high resistance against

Table 7. The comparison between BH-SBOX and other state-of-arts

Method	Bic-NL	BIC-SAC	DP	LP	SAC Avg	Nonlinearity		
						Min	Max	Avg.
BH-SBOX	102.85	0.50379	14	0.0706	0.50392	**108**	**110**	**108.25**
[34]	103.0	0.5066	12	0.1172	0.5044	102	110	107.00
[35]	103.21	0.5000	-	-	0.5351	102	108	105.25
[36]	103.26	0.4996	-	0.1171	0.5009	104	108	106.25
[37]	104.21	0.5030		0.1250	0.4993	104	110	106.00
[38]	103.50	-	-	-	0.4990	104	108	107.00
[39]	104.78	-	-	-	0.4973	106	110	107.25
[40]	104.21	0.5016	10	0.1484	0.5016	106	110	107.00
[41]	103.07	0.5029	10	0.1094	0.5029	106	110	107.50
[28]	104.35	0.4982	10	0.1250	0.4982	106	108	107.5
[42]	103.64	0.4996	10	0.1171	0.5002	104	110	108
[43]	104.57	0.4983	10	0.1172	0.4995	104	110	106.50
[26]	103.28	0.4969	10	0.1406	0.4980	102	106	104

linear cryptanalysis and demonstrated strong cryptographic properties. The utilization of chaotic maps in the initialization and searching processes enhanced the exploration capabilities of the algorithm, leading to the generation of robust S-boxes. Overall, the BH-SBOX paper contributes to the field of cryptographic algorithm design by offering an effective and efficient method for creating secure and reliable S-Boxes.

Future works for this study could include further exploration and optimization of the BH-SBOX algorithm by considering different variations of chaotic maps and integrating other metaheuristic algorithms. Additionally, the algorithm's performance can be evaluated on a wider range of cryptographic applications and benchmark problems. Furthermore, investigating the scalability of the algorithm and conducting comparative studies with existing state-of-the-art S-Box generation methods would contribute to its further development and validation.

Conflicts of interest

The authors declare no conflict of interest.

Author contributions

Conceptualization, SSI and AHA; methodology, SQS and SSI; software, SQS; validation, SSI, FHT, and SQS; formal analysis, AHA; investigation, AHA; data curation, SSI and SQS; writing—original draft preparation, SQS and FHT; writing—review and editing, SSI and AHA.

Acknowledgments

This work was supported by Al-Bayan University, College of Technical Engineering.

References

[1] C. E. Shannon, "Communication Theory of Secrecy Systems", *Bell Syst. Tech. J.*, Vol. 28, No. 4, pp. 656–715, 1949, doi: 10.1002/j.1538-7305.1949.tb00928.x.

[2] R. A. Amri, R. K. Murugesan, E. M. Alshari, and H. S. Alhadawi, "Toward a Full Exploitation of IoT in Smart Cities: A Review of IoT Anomaly Detection Techniques", *Lecture Notes in Networks and Systems*, pp. 193–214, 2022, doi: 10.1007/978-3-030-85990-9_17.

[3] T. Hai, M. Z. A. Bhuiyan, J. Wang, T. Wang, D. F. Hsu, Y. Li, S. Q. Salih, J. Wu, and P. Liu, "DependData: Data collection dependability through three-layer decision-making in BSNs for healthcare monitoring", *Inf. Fusion*, Vol. 62, pp. 32–46, Oct. 2020, doi: 10.1016/j.inffus.2020.03.004.

[4] S. A. M. A. Juboori, F. Hazzaa, S. Salih, Z. S. Jabbar, and H. M. Gheni, "Man-in-the-middle and denial of service attacks detection using machine learning algorithms", *Bull. Electr. Eng. Informatics*, Vol. 12, No. 1, pp. 418–426, Feb. 2023, doi: 10.11591/eei.v12i1.4555.

[5] M. S. M. Malik, M. A. Ali, M. A. Khan, M. E. U. Haq, S. N. M. Shah, M. Rehman, and W. Ahmad, "Generation of Highly Nonlinear and Dynamic AES Substitution-Boxes (S-Boxes) Using Chaos-Based Rotational Matrices", *IEEE Access*, Vol. 8, pp. 35682–35695, 2020, doi: 10.1109/ACCESS.2020.2973679.

[6] N. Kumar, V. M. Mishra, and A. Kumar, "Smart Grid Security by Embedding S-Box Advanced Encryption Standard", *Intell. Autom. Soft Comput.*, Vol. 34, No. 1, pp. 623–638, 2022, doi: 10.32604/iasc.2022.024804.

[7] K. S. Dhanalakshmi and R. A. Padmavathi, "A Survey on VLSI Implementation of AES Algorithm with Dynamic S-Box", *J. Appl. Secur. Res.*, 2022, doi: 10.1080/19361610.2020.1870403.

[8] G. Nissar, D. K. Garg, and B. U. I. Khan, "Implementation of security enhancement in AES by inducting dynamicity in AES S-box", *Int. J. Innov. Technol. Explor. Eng.*, 2019, doi: 10.35940/ijitee.J9311.0881019.

[9] H. S. Alhadawi, S. Q. Salih, and Y. D. Salman, "Chaotic Particle Swarm Optimization Based on Meeting Room Approach for Designing Bijective S-Boxes", In: *Proc. of International Conference on Emerging Technologies and Intelligent Systems*, pp. 331–341, 2022, doi: 10.1007/978-3-030-85990-9_28.

[10] K. Z. Zamli, H. S. Alhadawi, and F. Din, "Utilizing the roulette wheel based social network search algorithm for substitution box construction and optimization", *Neural Comput. Appl.*, Oct. 2022, doi: 10.1007/s00521-022-07899-7.

[11] A. I. Lawah, A. A. Ibrahim, S. Q. Salih, H. S. Alhadawi, and P. S. JosephNg, "Grey Wolf Optimizer and Discrete Chaotic Map for Substitution Boxes Design and Optimization", *IEEE Access*, Vol. 11, pp. 42416–42430, 2023, doi: 10.1109/ACCESS.2023.3266290.

[12] K. Z. Zamli, F. Din, and H. S. Alhadawi, "Exploring a Q-learning-based chaotic naked mole rat algorithm for S-box construction and optimization", *Neural Comput. Appl.*, 2023, doi: 10.1007/s00521-023-08243-3.

[13] W. A. Kissara and B. F. Hassan, "Determination of Fetal sex by Fetal anatomy parameters using a Fuzzy C-Mean Cluster", *Al-Kitab J. Pure Sci.*, 2023, doi: 10.32441/kjps.05.02.p2.

[14] S. A. Ahmed, H. Desa, and A. S. T. Hussain, "Performance Improvements Using Deep Learning Based Object-Identification", *Al-Kitab J. Pure Sci.*, Vol. 6, No. 1, pp. 1–13, Dec. 2022, doi: 10.32441/kjps.06.01.p1.

[15] H. S. Alhadawi, D. Lambić, M. F. Zolkipli, and M. Ahmad, "Globalized firefly algorithm and chaos for designing substitution box", *J. Inf. Secur. Appl.*, Vol. 55, p. 102671, Dec. 2020, doi: 10.1016/j.jisa.2020.102671.

[16] K. Z. Zamli, A. Kader, F. Din, and H. S. Alhadawi, "Selective chaotic maps Tiki-Taka algorithm for the S-box generation and optimization", *Neural Comput. Appl.*, Vol. 33, No. 23, pp. 16641–16658, Dec. 2021, doi: 10.1007/s00521-021-06260-8.

[17] S. Q. Salih, A. Sharafati, I. Ebtehaj, H. Sanikhani, R. Siddique, R. C. Deo, H. Bonakdari, S. Shahid, and Z. M. Yaseen, "Integrative stochastic model standardization with genetic algorithm for rainfall pattern forecasting in tropical and semi-arid environments", *Hydrol. Sci. J.*, 2020, doi: 10.1080/02626667.2020.1734813.

[18] S. Q. Salih, A. A. Alsewari, and Z. M. Yaseen, "Pressure Vessel Design Simulation: Implementing of Multi-Swarm Particle Swarm Optimization", *Proc. 2019 8th Int. Conf. Softw. Comput. Appl.*, pp. 120–124, 2019, doi: 10.1145/3316615.3316643.

[19] A. K. Shukla, P. Singh, and M. Vardhan, "A new hybrid wrapper TLBO and SA with SVM approach for gene expression data", *Inf. Sci. (Ny).*, Vol. 503, pp. 238–254, 2019, doi: 10.1016/j.ins.2019.06.063.

[20] S. Mirjalili, "How effective is the Grey Wolf optimizer in training multi-layer perceptrons", *Appl. Intell.*, 2015, doi: 10.1007/s10489-014-0645-7.

[21] N. S. Jaddi, S. Abdullah, and A. R. Hamdan, "Optimization of neural network model using modified bat-inspired algorithm", *Appl. Soft Comput. J.*, Vol. 37, pp. 71–86, 2015, doi: 10.1016/j.asoc.2015.08.002.

[22] S. Q. Salih, "A New Training Method Based on Black Hole Algorithm for Convolutional Neural Network", *J. Southwest Jiaotong Univ.*, Vol. 54, No. 3, pp. 1–10, 2019, doi: 10.1002/9783527678679.dg01121.

[23] M. H. Hassan, S. Kamel, S. Q. Salih, T. Khurshaid, and M. Ebeed, "Developing chaotic artificial ecosystem-based optimization algorithm for combined economic emission dispatch", *IEEE Access*, 2021, doi: 10.1109/ACCESS.2021.3066914.

[24] H. Guo, H. Tao, S. Q. Salih, and Z. M. Yaseen, "Optimized parameter estimation of a PEMFC model based on improved Grass Fibrous Root Optimization Algorithm", *Energy Reports*, Vol. 6, pp. 1510–1519, Nov. 2020, doi: 10.1016/j.egyr.2020.06.001.

[25] M. Kohli and S. Arora, "Chaotic grey wolf optimization algorithm for constrained optimization problems", *J. Comput. Des. Eng.*, 2018, doi: 10.1016/j.jcde.2017.02.005.

[26] G. Chen, "A novel heuristic method for obtaining S-boxes", *Chaos, Solitons & Fractals*, Vol. 36, No. 4, pp. 1028–1036, May 2008, doi: 10.1016/j.chaos.2006.08.003.

[27] Y. Wang, K. W. Wong, C. Li, and Y. Li, "A novel method to design S-box based on chaotic map and genetic algorithm", *Phys. Lett. Sect. A*

Gen. At. Solid State Phys., 2012, doi: 10.1016/j.physleta.2012.01.009.

[28] Y. Tian and Z. Lu, "Chaotic S-Box: Intertwining Logistic Map and Bacterial Foraging Optimization", *Math. Probl. Eng.*, 2017, doi: 10.1155/2017/6969312.

[29] H. S. Alhadawi, M. A. Majid, D. Lambić, and M. Ahmad, "A novel method of S-box design based on discrete chaotic maps and cuckoo search algorithm", *Multimed. Tools Appl.*, Vol. 80, No. 5, pp. 7333–7350, Feb. 2021, doi: 10.1007/s11042-020-10048-8.

[30] R. Soto, B. Crawford, F. G. Molina, and R. Olivares, "Human Behaviour Based Optimization Supported With Self-Organizing Maps for Solving the S-Box Design Problem", *IEEE Access*, Vol. 9, pp. 84605–84618, 2021, doi: 10.1109/ACCESS.2021.3087139.

[31] N. Hematpour and S. Ahadpour, "Execution examination of chaotic S-box dependent on improved PSO algorithm", *Neural Comput. Appl.*, 2021, doi: 10.1007/s00521-020-05304-9.

[32] K. Z. Zamli, "Optimizing S-box generation based on the Adaptive Agent Heroes and Cowards Algorithm", *Expert Syst. Appl.*, 2021, doi: 10.1016/j.eswa.2021.115305.

[33] A. A. Qasim and A. H. Sallomi, "Design and Analysis of Phased Array System by MATLAB Toolbox", *Al-Kitab J. Pure Sci.*, 2023, doi: 10.32441/kjps.04.01.p5.

[34] C. Adams and S. Tavares, "Good S-Boxes Are Easy To Find", in *Advances in Cryptology — CRYPTO' 89 Proceedings*, New York, NY: Springer New York, pp. 612–615, doi: 10.1007/0-387-34805-0_56.

[35] B. A. E. Atty, "Efficient S-box construction based on quantum-inspired quantum walks with PSO algorithm and its application to image cryptosystem", *Complex Intell. Syst.*, 2023, doi: 10.1007/s40747-023-00988-7.

[36] Z. Hua, J. Li, Y. Chen, and S. Yi, "Design and application of an S-box using complete Latin square", *Nonlinear Dyn.*, Vol. 104, No. 1, pp. 807–825, 2021, doi: 10.1007/s11071-021-06308-3.

[37] M. A. B. Farah, A. Farah, and T. Farah, "An image encryption scheme based on a new hybrid chaotic map and optimized substitution box", *Nonlinear Dyn.*, 2020, doi: 10.1007/s11071-019-05413-8.

[38] A. A. A. E. Latif, B. A. E. Atty, A. Belazi, and A. M. Iliyasu, "Efficient chaos-based substitution-box and its application to image encryption", *Electron.*, 2021, doi: 10.3390/electronics10121392.

[39] S. A. Jabber, A. M. A. A. Tameemi, and S. S. A. Jabbar, "Using Three-Dimensional Logistic Equations and Glowworm Swarm Optimization Algorithm to Generate S-Box", *Int. J. Intell. Eng. Syst.*, Vol. 15, No. 6, pp. 406–417, Dec. 2022, doi: 10.22266/ijies2022.1231.37.

[40] "Designing a Novel Efficient Substitution-Box by Using a Flower Pollination Algorithm and Chaos System", *Int. J. Intell. Eng. Syst.*, Vol. 15, No. 1, Feb. 2022, doi: 10.22266/ijies2022.0228.17.

[41] M. Ahmad, D. Bhatia, and Y. Hassan, "A Novel Ant Colony Optimization Based Scheme for Substitution Box Design", *Procedia Comput. Sci.*, Vol. 57, pp. 572–580, 2015, doi: 10.1016/j.procs.2015.07.394.

[42] H. A. Ahmed, M. F. Zolkipli, and M. Ahmad, "A novel efficient substitution-box design based on firefly algorithm and discrete chaotic map", *Neural Computing and Applications*, 2018. doi: 10.1007/s00521-018-3557-3.

[43] A. Farah and A. Belazi, "A novel chaotic Jaya algorithm for unconstrained numerical optimization", *Nonlinear Dyn.*, Vol. 93, No. 3, pp. 1451–1480, Aug. 2018, doi: 10.1007/s11071-018-4271-5.

[44] T. Farah, R. Rhouma, and S. Belghith, "A novel method for designing S-box based on chaotic map and Teaching–Learning-Based Optimization", *Nonlinear Dyn.*, Vol. 88, No. 2, pp. 1059–1074, Apr. 2017, doi: 10.1007/s11071-016-3295-y