Journal of Information Technology Management

Print ISSN: 2008-5893

Online ISSN: 2423-5059

A New S-Box Design by Applying Bat Algorithm-Based Technique

Maiya Din *

*Corresponding Author, Scientist, M.Sc., DRDO HQrs, New Delhi, India. E-mail: anuragimd@gmail.com

Saibal K. Pal

Scientist, Ph.D., DRDO, Delhi, India. E-mail: skptech@yahoo.com

S.K. Muttoo

Professor (Retd.), PhD., 505, Patel Society, Dwarka, New Delhi, India. E-mail: drskmuttoo@gmail.com

Sushila Madan

Professor, Ph.D., LSR College, University of Delhi, New Delhi, India. E-mail: sushila.madan@gmail.com

Abstract

Substitution boxes (S-boxes) are very important nonlinear components used for achieving strong confusion for enhancing cryptographic security in most of the block ciphers. Designing cryptographically strong S-boxes has been a major research domain for the designers of symmetric cryptosystems. In the proposed research work, the Bat Algorithm-based swarm technique is proposed to design strong S-boxes. The developed swarm technique obtains cryptographic strong S-boxes. The authors analyze the cryptographic strength of the obtained S-box by evaluating properties like Bijectivity, Nonlinearity, Bit-Independence Criterion, Linear Probability, and Differential Uniformity. The obtained performance parameters for the designed new S-box by the swarm technique are compared with some recently reported S-boxes in the literature. The designed S-box has good cryptographic strength. The designed S-box has good cryptographic strength like nonlinearity=110.75 and an average Strict Avalanche Criterion (SAC) value=0.506. For the constructed S-box, most of the Differential uniformity components are four and show uniform distribution approximately. The proposed new S-box is also free from the fixed points.

Keywords: Cryptography, Block Cipher, S-box; Nonlinearity, Bat Algorithm

Journal of Information Technology Management, 2023, Vol. 15, Issue 3, pp. 85-98
Published by the University of Tehran, Faculty of Management
doi: https://doi.org/ 10.22059/jitm.2023.93626
Article Type: Research Paper

Received: April 03, 2023
Received in revised form: June 13, 2023
Accepted: July 20, 2023
Published online: August 26, 2023

Introduction

Cryptography is a branch of cryptology concerned with the design of cryptosystems (Stallings, 2012).

With the advancement of communication technologies, the design of encryption techniques for secure communication of confidential information over an insecure channel, have attracted the major attention of the research community. Encryption algorithms are mainly categorized as stream and block ciphers. The principles of confusion and diffusion are ensured during the design phase of block ciphers (e.g., DES, IDEA, PRESENT, RC6, and AES). An S-box is a vital and most common crypto primitive in block ciphers. The S-box employed within a block cipher imparts confusion and makes the system resilient to defend against cryptanalytic attacks applied by the cryptanalyst for breaking the crypts. In most of the block ciphers, the S-box is an important nonlinear element due to fulfilling crypto characteristics like Nonlinearity, Bit-Independence Criterion, Linear Probability, Differential Uniformity, and Bijectivity. The strength of the employed S-box contributes significantly in the overall cryptographic security of the cryptosystem.

The cryptographic strength of block ciphers mainly depends on applied S-boxes. There are three types of S-box designing process, known as, random search-based techniques, algebraic techniques, and heuristic-based techniques. The finite Field inversion scheme is used in algebraic techniques for constructing an S-box satisfying required cryptographic properties. The 'Random Search' based techniques are easy to implement and produce a large number of S-boxes. However, the quality of generated S-boxes using random search are far from the S-boxes produced by algebraic techniques. The third type of designing process applies various heuristics to design S-boxes satisfying most of the cryptographic strength parameters. There is a performance gap between algebraic and heuristics-based techniques. Therefore, meta-heuristics-based iterative swarm techniques are applied for constructing S-boxes with enhanced cryptographic strength.

Review of Related Work

S-box is defined as a set of Boolean functions. An n×m S-box is a function S: {0, 1} n →{0, 1}m, here n and m are positive integers indicating the size of input and output string respectively. This maps an input string to an output string nonlinearly. In many ciphers, deployed S-boxes have the same size of i/p and o/p strings. S-boxes of size (4×4) and (8×8) are utilized in many block ciphers. The S-box (8×8) is used in highly secure ciphers like AES and the S-box (4×4) is generally used in lightweight cryptosystems like PRESENT Cipher (Stinson, 2013).

The correlation between the input-output strings is also determined to satisfy the security criteria of the S-box. Nonlinearity is a very important cryptographic property that imparts confusion for resisting linear cryptanalysis of block ciphers (Hussain et al., 2012; Özkaynak & Yavuz, 2013). After the development of linear and differential cryptanalytic attacks on S-boxes of DES cipher, a need was felt for the design of cryptographically strong S-boxes. Consequently, several S-box design schemes were evaluated which were based on optimization techniques. Tang et al. proposed a novel technique using a discrete Baker map for the construction of the S-box (Tang et al., 2005). An S-box construction method based on chaotic maps was presented by Jakimoski (Jakimoski & Kocarev, 2001). Wang et al. presented a design technique based on genetic algorithm and chaos to compute an improvised S-box in the proposed work (Wang et al., 2012). Ahmed and Bhatia proposed a nature-inspired swarm-based technique by applying Ant Colony Optimization (ACO) to generate strong S-boxes (Ahmad et al., 2015b). Farah et al. presented a technique for achieving S-box by applying a "Teaching–Learning-Based Optimization (TBLO)" swarm algorithm (Farah et al., 2017).

The firefly algorithm is an iterative, nature-inspired swarm technique developed by Yang. The algorithm has been applied by researchers to solve real-world optimization problems. Many researchers have developed several variants of the algorithm in their research work (Fister et al., 2013; Gandomi et al., 2013). Since the proposed FA, many modifications have been made to the schemes which provided many variants to solve real-world continuous and constrained optimization cases (Yang, 2015).

According to Yang (2009), Many variants of the algorithm also developed in the last decades due to performance comparison between FA and other well-known swarm techniques.

The Bat swarm algorithm has shown its capability in addressing various issues of optimization, however, its search behavior usually depends on initial positions and controlling parameters. The performance of the algorithm can be improved further by initializing it from a suitable initial point. In meta-heuristic-based optimization techniques, generally, random initialization is done in every execution of the program of the algorithm (Parpinelli & Lopes, 2011; Tsai et al., 2012; Yang et al., 2013).

Swarm intelligence-based techniques are also applied in the cryptanalysis of several ciphers (Laskari et al., 2007; Din et al., 2019b; and Din et al., 2019a). In the last two decades, some methods have been developed to design S-boxes by employing chaotic maps due to their nonlinear chaotic behavior (Webster & Tavares, 1986; Asim & Jeoti, 200; Özkaynak, 2019; Lambić, 2014; Ahmad et al., 2015a; and Ahmad et al., 2018).

According to the above reported motivating research on S-box design the authors propose a swarm technique by employing an anature-inspired Bat Algorithm (BA).

In this research paper, related work on the design of S-boxes by applying swarm-based techniques is reviewed. The next Section describes the Bat Algorithm and BA-based computational intelligence technique. The performance results of the achieved S-box are discussed in the last Section.

Bat Algorithm

A nature-inspired Bat Swarm Algorithm was proposed by Yang (2010a).

BA is a population-based meta-heuristic technique for solving optimization problems. It was inspired by the echolocation behavior of microbats, with varying pulse rates of emission and loudness. Each microbat of the swarm searches for an optimal solution to the considered problem.

This algorithm is naturally inspired form the social behavior of bats. The capability of echolocations of these bats composed a great competent manner to detect prey, avoid obstacles, and locate their rooster crevices in the dark based on sensing distances. To formalize the bat algorithm optimally, the following bat's echolocation characteristics should be idealized:

(i) Object distances are always perfectly sensed by the echolocation system on bats. This makes the ability to differentiate between different objects even in darkness.

(ii) Bats are flying randomly with velocity v, fixed frequency fmin at position xi and fluctuating wavelength (λ) and loudness form large loudness (A0) to minimum loudness (Amin) to search for its prey. Wavelength or frequency can be changed instantly by adjusting the pulse emission rate $r \in [0,1]$ based on the closeness of the bat objective.

(iii) Variation of the loudness parameter takes values between A0 and Amin.

The pseudo-code of the bat algorithm is mentioned in Figure 1. It starts with the initialization of all the echolocation system variables. The initial location of all bats swarms should be initialized as initial solutions.

$$f_i = f_{min} + (f_{max} - f_{min})\beta \quad (1)$$

$$v_i^t = v_i^{t-1} + (x_i^t - x^*)f_i \quad (2)$$

$$x_i^t = x_i^{t-1} + v_i^t \tag{3}$$

Here;

β: Random vector drawn from uniform distribution β∈[0, 1].

x*: Current global best location (solution) which is located after comparing all the solutions among all the bats.

fi: Frequency which is drawn uniformly from [fmin, fmax]

A random walk with direct exploitation is used for the local search that modifies the current best solution according to the equation:

$$x_{new} = x_{old} + \varepsilon A^t \tag{4}$$

Here;

ε : is a random number ∈[-1, 1].

At: is the average loudness of all the best at this time step.

ri: is the rate of pulse emission

For each bat, as soon as the prey is found, the bat loudness decrease, and the pulse emission rate increase. Both loudness and pulse emission expressed mathematically as follows:

$$A_i^{t+1} = \alpha A_i^t \tag{5}$$

$$r_i^{t+1} = r_i^0[1 - \exp(-\gamma t)] \tag{6}$$

$$A_i^t \rightarrow 0 \text{ and } r_i^t \rightarrow r_i^0 \text{ as} t \rightarrow \infty \tag{7}$$

Here; α : is constant 0 <α< 1 and γ : is constant γ> 0

Figure 1.

Bat Algorith

Begin
Step 1: Initialization
Set the generation counter t = 1
Initialize the population of NP bats P randomly and each bat corresponding to a potential solution
Loudness A_i; Define:
Pulse frequency Q_i;
Initial velocities V (i = 1, 2, …, NP);
Pulse rate r_i;
Step 2: Loop
While((the termination criteria are not satisfied) or (t < Max-Generation))
Do Generate new solutions by adjusting frequency, and updating velocities and locations/solutions
[(4) – (6)]
If *(rand >* r_i*)* **then**
Select a solution among the best solutions;
Generate a local solution around the selected best solution
End If
Generate a new solution by flying randomly
If *(rand <* A_i *&* $f(x_i) < f(x^*)$*)* **then**
Accept the new solutions
Increase r_i and reduce A_i
End If
Rank the bats and find the current best $x_t^u = t + 1$;
Step 3: End While
Step 4: Post-processing the results and visualization.
End.

Proposed Bat Algorithm-Based Technique

The BA is based on automated subdivision and has multimodality handling capacity. According to the subdivision criteria, the fireflies try to discover optimized solutions by applying parallel optimizations, particularly when the population size is sufficiently higher compared to the modes. The proposed technique based on BA for designing 8×8 Substitution-boxes is as follows:

Initialization of Swarm Bats

(1) Initialize n: No. of bats

(2) Initialize each bat with a random permutation vector (P), having values in [0,255] without repetition.

(3) Array P is reshaped into a 16×16 two-dimensional matrix representing the initial S-box, which shows the initial position of the

ith bat (f_i).

(4) Steps 2 to 3 are repeated for generating all n bats such that each bat represents one initial S-box.

Fitness function

The fitness value of each bat is calculated based on the nonlinearity of each S-box as follows:

4.2.1 Nonlinearity of S-box is calculated using the following equation:

$$N_f = 2^{n-1}(1 - 2^{-n} \max_{w \in GF(2^n)} |WS \prec f \succ (w)|) \tag{8}$$

Here, the Walsh spectrum f_x is determined as per the equation as follows:

$$WS\langle f \rangle(w) = \sum_{x \in GF(2^n)} (-1)^{f(x) \oplus x.w} \tag{9}$$

for the Eq. (4) $w\ \in$ GF(2n) and the dot product (x.w) is computed as follows:

$$x.w = x_1.w_1 \oplus \dots\dots \oplus x_n.w_n$$

The fitness value of ith bat (fi) is determined using the following equation.

$$fitness = 112 - NL(f_i) \tag{10}$$

Here, 112 is an optimal value of S-Box nonlinearity mentioned in the Block Cipher AES (Stinson, 2013), and the NL function determines the nonlinearthity of the bat corresponding to the S-box. In the current iteration, the bat with the lowest fitness value provides the best S-box.

$$r_{ij} = \sqrt{\sum_{k=1}^{d} \left(P_{i,k} - P_{j,k}\right)^2} \tag{11}$$

Where, Pi,k represents kth element of the S-box corresponding to ith bat and d is set to 255.

After every movement of bats, the obtained values are checked for keeping within the specified threshold values: 0 and 255. This is ensured by satisfying the Bijective property of the S-Box corresponding to the bat.

Minor Adjustment

An S-box (8×8) is represented as a vector of 256 elements. As per the Bijective property of an S-box, each element occurs only once in the vector. Sometimes, this property is not fully satisfied by the computed S-box in the Bat algorithm. So, minor adjustment is required in the obtained S-box such that each item from 0 to 255 occurs only once to fully satisfy the Bijective property.

Performance Analysis of the Proposed S-box

The proposed Bat algorithm-based technique is implemented using MATLAB software (Ver. R2017a). The developed software is run on the 3.0 GHz computing machine. Various trials are conducted for different values of the control parameters of the algorithm to compute cryptographic strong S-boxes.

The parameters of proposed technique are taken as: $\alpha = 0.25$, $\beta = 0.15\ to\ 0.35$, $\gamma = 1.0$, iterations = 750, swarm size (N) = 20. The obtained S-box is shown in the following Table:

Table 1.

The S-Box Computed by Proposed Technique

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

Many performance measures for an S-box are defined by expert designers to examine the cryptographic security strengths of an S-box. Biham and Shamir proposed the differential uniformity criteria through its differential cryptanalysis of DES block cipher based on differential probability, calculated using I/O XOR distribution (Menezes et al., 2018; Biham & Shamir, 1991). Matsui (1993) proposed linear cryptanalysis based on linear approximation probability. Dawson et al. applied information theoretic concepts for designing and evaluating cryptographic strong S-boxes (Dawson & Tavares, 1991). Therefore, cryptographic security of designed S-boxes is ensured by nonlinearity, bijectivity, strict avalanche criterion (SAC), bit-independence criterion (BIC), differential uniformity, and linear approximation probability (LP) (Wang et al., 2009).

Bijective Property

The bijectivity implies that each item of the S-box occurs only once.

$$wt(\sum_{i=1}^{n} a_i f_i) = 2^{n-1} \tag{12}$$

Where $a \in [0,1]$ and $wt(\,)$ is the Hamming weight? As per equation (13), bijectivity is fulfilled when all Hamming weights are equal to 128 for the designed S-box.

Nonlinearity

The nonlinearity of the S-box is calculated using equation (3). The non-linearity is connected to the immunity of block cipher and plain text confusion. This is computed using the Walsh spectrum as mentioned in equation (4). The computed nonlinearity corresponding to the Boolean functions of the obtained S-box is shown in the following table and depicted in Figure 2.

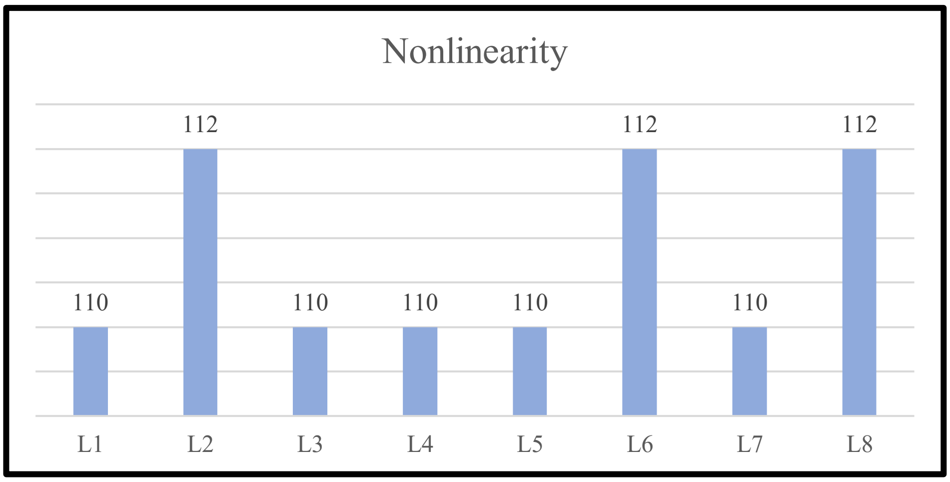
Table 2.

Nonlinearity of S-box

S-box	L_1	L_2	L_3	L_4	L_5	L_6	L_7	L_8	Min.
NL	110	112	110	110	110	112	110	112	110

Figure 2.

Nonlinearity of 8 Boolean Functions of S-box



Strict Avalanche Criteria

Webster and Tavares (Webster & Tavares, 1986) defined SAC by ensuring that there is a probability of ½ for a change in all the output bits when a single input bit is flipped. SAC is ensured through a dependence matrix and offsets values of SAC are computed using the following equation:

$$\boldsymbol{S(f) = \frac{1}{n^2}\sum_{1\le i\le n} sum_{1\le j\le n}\left|0.5 - p_{i,j}(f)\right|} \tag{13}$$

Bit Independence Criteria (BIC)

This stipulates the pair-wise independence of all avalanche variables for any avalanche vector generated by the complementing of a single plaintext. The criteria were defined by Webster and Tavares (Webster & Tavares, 1986).

I/O XOR distribution

For an ideal S-box, the obtained XOR distribution table should be balanced since the imbalance distribution table helps in applying differential cryptanalytic attack. A cryptographic strong S-box should be differential uniform. For a given S-box, differential uniformity is measured using equation (14).

$$DP(f) = \left[\frac{\#\{x\in X \mid f(x)\oplus f(x\oplus \triangle x)\} = \triangle y}{2^n}\right] \tag{14}$$

Here, X contains all the input values and the total number of elements is shown by 2n. Table 3 shows the I/O XOR distribution for the obtained S-box.

Table 3.

I/O XOR Distribution

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

Linear approximation probability

The LP probability is determined as the maximum imbalance value. Thus, mask a selects input bits that have parity equal to the output bits selected by mask b. The equation (15) is applied for computing LP.

$$LP = \max_{a,b\neq 0} \left| \frac{\#\{x|x.a=f(x).b\}}{2^n} \right| - 0.5 \tag{15}$$

For a given S-box, LP should be as small as possible. An S-box with lower LP has more resistance against linear cryptanalysis than other S-boxes with higher LP.

Table 4.

The Dependence Matrix for the Computed S-Box

0.469	0.492	0.500	0.484	0.461	0.469	0.539	0.469
0.445	0.508	0.516	0.516	0.508	0.484	0.461	0.563
0.578	0.531	0.398	0.516	0.453	0.414	0.531	0.477
0.453	0.516	0.602	0.383	0.555	0.523	0.438	0.477
0.563	0.461	0.531	0.539	0.641	0.500	0.453	0.563
0.492	0.477	0.531	0.461	0.633	0.500	0.484	0.508
0.398	0.555	0.484	0.461	0.523	0.625	0.469	0.539
0.492	0.563	0.570	0.523	0.508	0.531	0.453	0.453

A swarm technique applying the Bat Algorithm is proposed for designing strong S-boxes. The above-mentioned results of the S-box obtained by the swarm technique have shown that most of the criteria of a good S-box are fulfilled. The obtained S-box has high immunity against differential cryptanalytic attacks. The computed S-box is free from the fixed points. The maximum nonlinearity of the S-box is 112 and the minimum is 110. The average nonlinearity stands at 110.75. The average SAC value for the S-box is 0.506.

According to the Table of Input/Output XOR distribution, DP values are approximately uniformly distributed and most of the values are four.

Table 5.

Performance Comparison of the New S-box

S-box design	Nonlinearity			SAC	I/O XOR
Technique	Min.	Max.	Avg.	Avg.	Max. DP
(Hussain et al., 2012)	102	108	104.7	0.5056	12
(Özkaynak & Yavuz, 2013)	103	109	105.1	0.5061	10
(Ahmad et al., 2015b)	106	110	107.0	0.5015	10
(Farah et al., 2017)	104	108	106.5	0.4991	10
(Özkaynak , 2019)	106	108	106.7	0.4941	10
(Ahmed et al., 2019)	106	108	107.5	0.4943	10
(Alhadawi et al., 2021)	106	110	108.5	0.4995	10
(Lambić, 2017)	106	108	106.7	0.5034	10
New S-box by Proposed Technique	110	112	110.75	0.506	6

According to Table 5, the performance indicators for the constructed S-box are compared with some recently reported S-boxes. The maximum nonlinearity of the S-box is 112 and the minimum is 108. The computed average nonlinearity is 110.75. The average SAC value for the S-box is 0.506. The obtained maximum DP value in the I/O XOR distribution table is 6. The proposed new S-box is also free from the fixed points.

The authors addressed the issues of designing cryptographic strong S-boxes. An efficient Bat algorithm-based technique is proposed to generate S-boxes. The computed S-box has better performance as compared with some of the existing S-boxes. The achieved S-Box meets most of the cryptographic requirements. The authors plan to focus on designing strong S-boxes having different sizes of inputs and outputs by developing efficient swarm techniques based on other meta-heuristics and swarm intelligence.

Acknowledgments

My sincere thanks to the co-authors for their valuable technical guidance and motivation in carrying out the proposed research work and for supporting me in my research paper submission.

Conflict of interest

The authors declare no potential conflict of interest regarding the publication of this work. In addition, the ethical issues including plagiarism, informed consent, misconduct, data fabrication and, or falsification, double publication and, or submission, and redundancy have been completely witnessed by the authors.

Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article.

References

Ahmad, M., Ahmad, F., Nasim, Z., Bano, Z., & Zafar, S. (2015). Designing chaos based strong substitution box. In *2015 eighth International Conference on contemporary computing (IC3)* (pp. 97-100). IEEE.

Ahmad, M., Bhatia, D., & Hassan, Y. (2015). A novel ant colony optimization-based scheme for substitution box design. *Procedia Computer Science*, *57*, 572-580.

Ahmad, M., Seeru, F., Siddiqi, A. M., & Masood, S. (2018). Dynamic 9× 9 substitution-boxes using Chaos-based heuristic search. In *Soft Computing: Theories and Applications* (pp. 839-851). Springer, Singapore.

Ahmed, H. A., Zolkipli, M. F., & Ahmad, M. (2019). A novel efficient substitution-box design based on firefly algorithm and discrete chaotic map. *Neural Computing and Applications*, *31*(11), 7201-7210.

Alhadawi, H. S., Majid, M. A., Lambić, D., & Ahmad, M. (2021). A novel method of S-box design based on discrete chaotic maps and cuckoo search algorithm. *Multimedia Tools and Applications*, *80*(5), 7333-7350.

Asim, M., & Jeoti, V. (2008). Efficient and Simple Method for Designing Chaotic S-Boxes. *ETRI journal*, *30*(1), 170-172.

Biham, E., & Shamir, A. (1991). Differential cryptanalysis of DES-like cryptosystems. *Journal of CRYPTOLOGY*, *4*(1), 3-72.

Dawson, M. H., & Tavares, S. E. (1991). An expanded set of S-box design criteria based on information theory and its relation to differential-like attacks. In *Workshop on the Theory and Application of Cryptographic Techniques* (pp. 352-367). Springer, Berlin, Heidelberg.

Din, M., Pal, S. K., & Muttoo, S. K. (2019). Analysis of RC4 crypts using PSO based swarm technique. In *Harmony Search and Nature Inspired Optimization Algorithms* (pp. 1049-1056). Springer, Singapore.

Din, M., Pal, S. K., & Muttoo, S. K. (2019). Applying PSO based technique for analysis of geffe generator cryptosystem. In *Harmony Search and Nature Inspired Optimization Algorithms* (pp. 741-749). Springer, Singapore.

Farah, T., Rhouma, R., & Belghith, S. (2017). A novel method for designing S-box based on chaotic map and teaching–learning-based optimization. *Nonlinear dynamics*, *88*(2), 1059-1074.

Fister, I., Fister Jr, I., Yang, X. S., & Brest, J. (2013). A comprehensive review of firefly algorithms. *Swarm and Evolutionary Computation*, *13*, 34-46.

Gandomi, A. H., Yang, X. S., Talatahari, S., & Alavi, A. H. (2013). Firefly algorithm with chaos. *Communications in Nonlinear Science and Numerical Simulation*, *18*(1), 89-98.

Hussain, I., Shah, T., & Gondal, M. A. (2012). A novel approach for designing substitution-boxes based on nonlinear chaotic algorithm. *Nonlinear Dynamics*, *70*(3), 1791-1794.

Jakimoski, G., & Kocarev, L. (2001). Chaos and cryptography: block encryption ciphers based on chaotic maps. *IEEE transactions on circuits and systems i: fundamental theory and applications*, *48*(2), 163-169.

Lambić, D. (2014). A novel method of S-box design based on chaotic map and composition method. *Chaos, Solitons & Fractals*, *58*, 16-21.

Lambić, D. (2017). A novel method of S-box design based on discrete chaotic map. *Nonlinear dynamics*, *87*(4), 2407-2413.

Laskari, E. C., Meletiou, G. C., Stamatiou, Y. C., & Vrahatis, M. N. (2007). Cryptography and cryptanalysis through computational intelligence. In *Computational Intelligence in Information Assurance and Security* (pp. 1-49). Springer, Berlin, Heidelberg.

Matsui, M. (1993). Linear cryptanalysis method for DES cipher. In *Workshop on the Theory and Application of Cryptographic Techniques* (pp. 386-397). Springer, Berlin, Heidelberg.

Menezes, A. J., Van Oorschot, P. C., & Vanstone, S. A. (2018). *Handbook of applied cryptography*. CRC press.

Özkaynak, F. (2019). Construction of robust substitution boxes based on chaotic systems. *Neural Computing and Applications*, *31*(8), 3317-3326.

Özkaynak, F., & Yavuz, S. (2013). Designing chaotic S-boxes based on time-delay chaotic system. *Nonlinear Dynamics*, *74*(3), 551-557.

Parpinelli, R. S., & Lopes, H. S. (2011). New inspirations in swarm intelligence: a sur vey. *International Journal of Bio-Inspired Computation*, *3*(1), 1-16.

Stallings W. (2012). *Cryptography and Network Security*, Pearson Publications, London.

Stinson, D.R. *(2013). Cryptography: Theory and Practice*. 3rd Edition, Chapman & Hall/CRC Publication, 2013. 593p.

Tang, G., Liao, X., & Chen, Y. (2005). A novel method for designing S-boxes based on chaotic maps. *Chaos, Solitons & Fractals*, *23*(2), 413-419.

Tsai, P.W., Pan, J. S., Liao, B. Y., Tsai, M. J., and Istanda, V (2012). Bat algorithm inspired algorithm for solving numerical optimization problems. *Applied Mechanics and Materials*, 134– 137.

Wang, Y., Wong, K. W., Li, C., & Li, Y. (2012). A novel method to design S-box based on chaotic map and genetic algorithm. *Physics Letters A*, *376*(6-7), 827-833.

Wang, Y., Xie, Q., Wu, Y., & Du, B. (2009). A software for S-box performance analysis and test. In *2009 International Conference on Electronic Commerce and Business Intelligence* (pp. 125-128). IEEE.

Webster, A., and Tavares, S. (1986): On the design of S-boxes. In: Advances in cryptology. Lecture notes in computer science. *Proc of CRYPTO85,* pp 523-534.

Yang X-S (2009). Firefly algorithms for multimodal optimization. *International Symposium on Stochastic Algorithms*, 169-178p

Yang X-S (2015). Analysis of firefly algorithm and automatic parameter tuning. In emerging research on swarm intelligence and algorithm optimization, *IGI Global*, 36-49

Yang, X. S. (2010). A new metaheuristic bat-inspired algorithm. In *Nature inspired cooperative strategies for optimization (NICSO 2010)* (pp. 65-74). Springer, Berlin, Heidelberg.

Yang, X. S. (2010). *Nature-inspired metaheuristic algorithms*. Luniver press.

Yang, X. S., Cui, Z., Xiao, R., Gandomi, A. H., & Karamanoglu, M. (Eds.). (2013). *Swarm intelligence and bio-inspired computation: theory and applications*. Newnes.

Bibliographic information of this paper for citing:

Din, Maiya; Pal, Saibal K.; Muttoo, S.K. & Madan, Sushila (2023). A New S-box design by Applying Bat Algorithm based technique. *Journal of Information Technology Management*, 15 (3), 85-98. https://doi.org/ 10.22059/jitm.2023.93626