Complex & Intelligent Systems (2023) 9:2959–2967
https://doi.org/10.1007/s40747-021-00556-x

ORIGINAL ARTICLE

Low-power AES S-box design using dual-basis tower field extension method for cyber security applications

V. Nandan[1] · **R. Gowri Shankar Rao**[2]

Received: 29 June 2021 / Accepted: 24 September 2021 / Published online: 27 October 2021
© The Author(s) 2021

Abstract

In cryptography, one among several investigated areas is the implementation of AES S-boxes. In this paper, a substitution-box is designed which follows combined data path using dual-basis tower field extension with Golod–Shafarevich theorem fed in immune genetic algorithm for optimization purpose for each and every block. The role of enhanced immune genetic algorithm is as follows: at first, chaotic system generates S-boxes population, these S-boxes which perform excellently are then optimized by a sequence of operators such as extraction of anti-agent and immune selection. Few criteria of S-boxes such as differential uniformity, nonlinear degree, and strict avalanche effect are analyzed. The obtained results are analyzed with CMOS 35 nm and 15 nm technologies to measure the performance of the proposed designs and was observed that the proposed one outperforms in power and area. The optimized S-box can be effectively applied for securing information. The proposed Golod–Shafarevich feeder Immune Genetic Algorithm S-box (GSIGA-Sbox) is compared with two baseline methods such as Reversed Genetic Algorithm S-box (RGA-Sbox) and Discrete Space Chaotic S-box (DSC-Sbox). As a result the proposed GSIGA-Sbox achieves encryption speed of 61 MHZ, decryption speed of 55 MHZ with 24% of power consumption for 35 nm CMOS technology and 57 MHZ encryption speed, 51 MHZ decryption speed with 28% of power consumption for 15 nm CMOS technology.

Keywords S-box · Optimization · Encryption · Decryption · Cryptography · Galois field

Introduction

There is huge demand for data in Communication and Network technologies. The data are handled by open frameworks. Encryption process changes data as structure closer and more sensible that depends on the investigation with no reasonable learning [1]. The objective behind this is to secure data by maintaining and keeping it away from illegal user access. Decryption is a process which transfers encoded information into an outline clearly [2]. Both these processes necessarily use some anonymous data named the key. Sometimes for encryption, the same key is used in both Encryption and separation; but with various systems, keys used for encryption and interpretation are distinctive [3]. In accordance with AES, data that have to be encrypted are split as block with equal size where each block is termed as a state. Based on the principle of Substitution–Permutation Network, a sequence of mathematical operations is performed by the algorithms on every state and produces cipher text [4]. Initially, in the algorithm, round key is added at state which then enters the main loop and performs the following four operations repeatedly: substitution of shift rows, bytes, mix columns and add round key [5]. At last, after these operations, the final iteration excludes mix columns. With several improvements, the efficiency of the original AES is improved with measures such as delay, area, and power consumption [6]. Among these looping operations, substitution of bytes is termed as substitution-box (S-Box) which transforms data in a nonlinear fashion by replacing every byte with a different byte. Substitution of bytes is performed mainly to confuse the data which has to undergo encryption using AES [7]. This byte substitution is achieved by

✉ V. Nandan
vnandanecee@gmail.com

R. Gowri Shankar Rao
gshankarrao@veltech.edu.in

[1] Department of Electronics and Communication Engineering, Veltech Rangarajan Dr. Sagunthala R&D Institute of Science & Technology, Chennai, Tamil Nadu, India

[2] Department of Physics, Veltech Rangarajan Dr. Sagunthala R&D Institute of Science & Technology, Chennai, Tamil Nadu, India

defining the multiplicative inverse of the state given in finite field next to affine transformations [8]. On the other hand, the substitution bytes are calculated and maintained in the look-up table of S-Box. Many approaches are presented in the literature for the structures of S-box designing, incorporating algebraically approaches, pseudorandom and heuristic methods. The algorithms for modern block are frequently utilized S-box design methods depending on robust algebraic relations. Nyberg presented most well-known method called S-box of the AES (Advanced Encryption Standard) block encryption algorithm. These substitution bytes for encryption differs from that of the decryption. Among the four looping operations, byte substitution is a complex operation [9]. Therefore, this paper proposes a method for less consumption of area, power, delay and components. Hence, researcher put more efforts to optimize byte substitution in terms of hardware complexity, time, and power consumption [10]. This contribution of this work is as follows.

- Presenting a novel approach to design a high parallel area-efficient S-Box optimization architecture for cryptosystem of AES.

The organization of this paper is as described as follows: the next section describes about existing work related to the proposed method, the third section gives detailed methodology about optimization-based S-box architecture, the fourth section shows the performance analysis with respect to various parameters. Finally, the paper is concluded in the last section with future work.

Related works

Several research works in the literature based on metaheuristic techniques involved in developing S-boxes has been investigated. The optimization-based 8×8 S-box mechanism is described below: Wang et al. in [11] discussed the characteristics of Genetic Algorithm (GA) used to develop 8×8 S-box. The tent map as well as chaotic logistic map were involved to initialize the starting populations and GA parameters. The adjustment phase was improved to generate more potential S-box. Guesmi et al. [12] used logistic map for designing initial S-box and differential chaotic Lorenz model for performing mutation operations and crossover at the time of GA optimization. Simulation and analysis of security demonstrated that this approach is applied in image encryption. Ahmad et al. [13] analyzed metaheuristics Ant Colony Optimization (ACO) for optimizing initial S-box where chaotic tent map is integrated with logistic map. This optimized S-box was developed with features of cryptography. The S-box generation approach is essentially appropriate for using in the strong block cryptosystem's design. In [14], Artificial Bee Colony (ABC) method based on hyperchaotic map was used to produce efficient 8×8 S-box. For initial population of S-boxes, 6D hyperchaos was utilized. From the results of simulation proved that the algorithm has cryptographically strong S-box for meeting the criteria of multiple cryptography. In [15], Bacteria Foraging Optimization (BFO) was also employed with logistic map for S-box optimization. From the results of experiment, investigated the S-box generation algorithm presented will produce an S-box with the characteristics of good cryptography. In [16], traveling salesman problem was used to generate strong S-box. From the results of statistics manifested, the potential substitution-box is cryptographically highly inspiring as in contrasted to few current investigations.

Farah et al. [17] described a Teaching–Learning-Based Optimization (TLBO) approach with chaotic map for designing S-box efficiently. This approach determined the optimized keys which satisfied the conditions given. Hussam et al. [18] presented an optimized initial S-box using Firefly Algorithm (FA) from a chaotic map with discrete-space. Zhang et al. [19] developed I-Ching Operators (ICO) for producing an optimal 8×8 S-box. In [20], Alzaidi et al. analyzed β-hill climbing individual-based optimization technique to construct 8×8 S-box utilizing a new discrete-chaotic map. In [21], a fusion technique which involved Particle Swarm Optimization (PSO) and Differential Evolution (DE) approaches for generating various $n \times n$ S-boxes. From the results of experiments, it is proved that the chaotic S-box presented by the FLDSOP algorithm efficiently resisted to several kinds of attacks in cryptanalysis. Solami et al. in [22] implemented a random Heuristic Search (HS) method for synthesis bijective S-boxes where hyperchaotic system was used. The anticipated method's performance comparison with current S-box proposals showed its dominance and efficiency for a strong bijective construction of S-box.

Proposed methodology

This section presents the preliminaries of well-organized AES S-box operations and highlights the conventional area-efficient architectures utilizing tower fields described in Normal Basis (NB). As shown in Fig. 1, the input parameters undergo the mapping process which is followed by the construction of dual field based on Polynomial bias and Normal bias. After the bias, the subfield is analyzed with file extension block which leads to construction of S-box with the assistance of enhanced genetic algorithm.

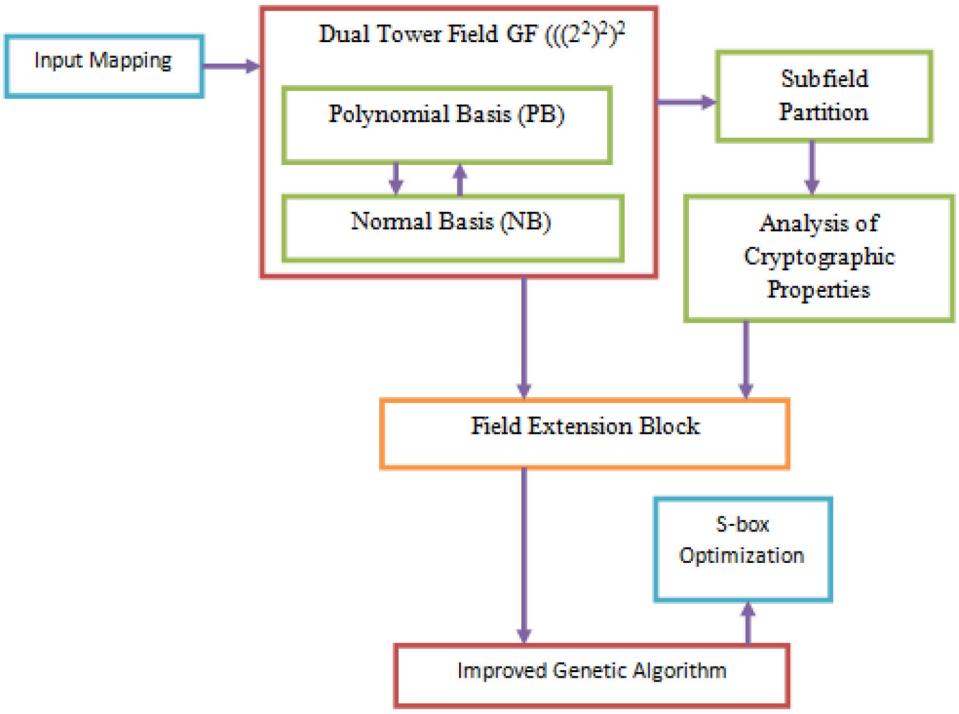

Fig. 1 Architecture for S-box with optimization method

Construction of dual-tower field GF((2²)²)²

Two various field construction sets are available which are the tower and composite field given as GF((2²)²)² and GF((2⁴)²), respectively, and are involved in computing S-boxes in AES. Moreover, subfields are described using Normal Basis (NB) or Polynomial Basis (PB). Here, tower field GF((2²)²)² is used over NB, as in Can right's approach, whereas various non-reducible complicated polynomials are used. In this proposed model, the field element $g = (g_7, g_6, \ldots g_0)$ ^{2}GF(2^8) is transformed to an isomorphic tower field $i = (A, B)$, and here (a_0, a_1, a_2, a_3) are represented by A and (b_0, b_1, b_2, b_3) are represented by B. The tower field is described by a non-reducible complicated polynomial over the subfield GF(((2²)² of the tower field:

$$p(y) = y2 + y + v = (y + \alpha)(y + 16),$$

where the subfield elements represented as α (its root) and v in tower field which are selected such that the polynomial used is non-reducible over GF(((2²)²)². Then, NB over GF(((2²)²)² is 16 g. Thus, a field element i is given by $i = A + B^{16}$, and here A and B are subfield elements of tower field. Likewise, the subfield is created with the help of over GF(((2²)² as irreducible polynomial of

$$q(z) = z2 + z + \mu = (z + \Omega)(z + \Omega 4).$$

The irreducible polynomial of degree 2, for constructing a binary field of the subfield GF(((2²)²,

$$R\left(t_0 = t^2 + t + 1 = (t + \beta)\left(t + \beta^2\right)\right),$$

is employed for generating NB$\{\beta, \beta^2\}$over GF(2): Thus, the elements of this fields are denoted as A_0, $A_1 \in$ GF(2^2) as $A_0 = (a_0, a_1) = a_{01} + a_{12}$ and $A_1 = (a_2, a_3) = a_{21} + a_{32}$, respectively.

Cryptographic properties of S-box

Definition 1 The function nonlinearity of set B_n is described as the least Hamming distance from the corresponding function to each linear function of B_n.

Coronary: Generally, the function nonlinearity $f \in B_n$ has an upper bound $2^{n-1} - 2^{n/2-1}$. If S-box having the highest nonlinearity is created, it does not give better estimates by linear functions; hence breaking up a cryptosystem is a difficult task. Total 1's in a binary vector v is the Hamming weight (H_w) of v. When the Hamming distance (H_d) of two binary vectors is equal, which means that the number of places where the respective entries vary.

Definition 2 A S-box of n × n is bijective when its output values are different ranging from 0 to $2^n - 1$. Based on the properties, the S-box is constructed as follows.

- A sequence S which is initially empty is defined.

- For the initial value x_0, to discard transient effect. Iterate 100 times.
- The current state value is given as x_0 where continuous iteration is performed. Then, X, which is an integer, is given as floor $(256 \times x_0)$ which gives the nearest integer for X.
- If X not present in S, append it else go to step 3.
- When the element count in S is less than 256, go to step 3 else S is the output
- Construct S-box of 8×8 from S which is utilized as the initial population.

Coronary: High nonlinearity was possessed by an S-box and differential probability and low linear are measured as a secured cryptographically. A novel approach is presented for the construction strong 8×8 S-boxes cryptographically through the application of an adjacency matrix on the Galois field GF(2^8). The adjacency matrix is acquired consistently to the closet diagram for the modular group's action PSL $(2, \mathbb{Z})$ on a projective line PL(F_7) for a finite field F_7.

Field extension using Golod–Shafarevich theorem

Estimation of Golod–Shafarevich for an infinite field general quadratic algebra having n generators is obtained and for quadratic relations $d \geq \frac{4(n1+n)}{g}$:

$$H(t) = \left|\left(1 - nt + \mathrm{d}t^2\right) - 1\right| = 1 + nt + \left(n^2 - d\right)t^2 + \left(n^3 - 2nd\right)t^3.$$

Assume field K and n, d, q$\in$N, $q > 3$, $d \leq n^2$, and $\{cj, k, m : 1 \leq j \leq d, 1 \leq k, m \leq n\}$ as variables which takes principles from field $\boldsymbol{K}$. Let I_c with $c = \{cj, k, m\}$ in Kh$\{\times 1,...,x_{ni}\}$ is obtained using $f_1,...,f_{\mathrm{d}}, f_{\mathrm{j}} = Pnk$, $m = 1\ cj, k, m \times k \times m$ and Rc $= Kh \times 1,..., x_{ni}/I_c$ algebra. The qth homogeneous element $(I_c)_q$ is clearly traversed using $\mu fj\nu$, and here j ranges from 1 to d and the two monomials are μ, ν in $K\ \{\times 1,..., x_{ni}\}$ with deg μ + deg $\nu = q - 2$. Thus, $(I_c)_q$ denotes the linear operator image L_c: K$\Omega \rightarrow Fq(n, K)$, and here Ω indicates a triplet (j, μ, ν) and Lc forwards a vector ej, μ, ν to $\mu fj\nu$. Rank rkL_c of L_c and dimension of $(I_c)_q$ is equal. Thus, dim (Rc)$q = n_q$ − dim $(I_c)q = n\ q - rc$, and here $rc = rk\ L_c$. When $K = Zp$ and p is prime, then $\delta(c) \neq 0$ for certain values of c whose coefficients are non-zeros as elements of Zp. Few coefficients of $\delta(c)$ are assumed to be a polynomial whose coefficients are in Z and are non-multiples of p and hence non-zero. $\delta(c) \neq 0$ for certain values of once Zp is replaced with Q. Likewise, argument illustrates that when hq (K, n, d) do not rely on fixed positive characteristic p's K and that $hq\ (K, n, d)\ 6\ hq(Zp, n, d)$ for any K.

Genetic algorithm-based S-box optimization

The S-boxes optimization algorithm in this paper combines the S-box construction method as mentioned above with the advanced GA, including population initialization stage, individual evaluation, selection stage, crossover stage, mutation stage and termination condition determination. Algorithm 1 shows the pseudocode of chaotic S-box optimization. The steps are as follows:

Step 1: Initialization

Initialize each element x in the S-box generated with the help of below equation:

$$X = \mathrm{Floor}\left(x \times 2^8\right).$$

Add each X to sequence $\{S\}$, and the output is the individual of the initial population. Repeat the above steps until all the populations are initialized.

Step 2: Individual evaluation

Compute the fitness for S-box according to the fitness function, then arrange the individuals in ascending order according to the fitness values. The operation continues unless the number of iterations reaches the threshold or the maximum fitness value in the population is greater than the predetermined value.

Step 3: Selection stage

Calculate the selection probability pa in the current iteration stage, then select N1 excellent individuals:

$$N_1 = \mathrm{pa} \times T.$$

where T is the number of initial populations.

Step 4: Crossover stage

Compute the number of populations N_2 generated in the crossover. The individual with the largest fitness is selected as parent-1, and the i-th individual as the parent-2, add parent-1 and parent-2 to the crossover population. The above operation continues until the total individuals obtained by the crossover operator is not greater than N_2. Cross-descendants of the output as elements in $\{S'\}$:

$$N_2 = \mathrm{pb} \times T$$
$$i = \mathrm{Floor}(T \times \times x_0),$$

where pb is the crossover probability, x_0 is the element of the sequence $\{S\}$.

Step 5: Mutation stage

Calculate the number N_3 of individuals to be mutated. Select the i-th individual in the cross-population to perform the mutation operation. Then, exchange the $(P_1 + i)$-th and $(P_2 + i)$-th individuals to generate an individual to be mutated. The above operation continues until the total individuals obtained by the mutation operator are not greater than N_3:

$$N_3 = pc \times T$$
$$P_1 = \text{Floor}(N_2 \times x')$$
$$P_2 = \text{Floor}\big((\text{N}_2 - 1) \times x''\big),$$

where pc is the mutation probability, and x 00 is the element of the sequence $\{S'\}$.

Step 6: Termination condition determination

If the number of iterations is greater than the threshold Q, obtain the individual with the maximum fitness in the optimization process as the optimal solution, then the algorithm is terminated. Genetic algorithm is applied to generate a maximum nonlinear S-box as follows:

Step 1. Compute nonlinearity, $nl0$ of the initial S-box (Sbox0) 2).

Step 2. Set $nl\ max = nl0, j = 0$ and $i = 1$.

Step 3. Crossover and mutation are performed on S-box as described as follows:

- With points $prow1i \in [2\ldots5]$ and $prow2i \in [8\ldots11]$, crossover row i and row $(16 - i + 1)$, here i ranges from 1 to 16.
- With points $pcol1i \in [2\ldots5]$ and $pcol2i \in [8\ldots11]$, crossover column i and column $(16 - i + 1)$, here i ranges from 1 to 16
- Permutation is applied with two points $pmut1i \in [2\ldots5]$ and $pmut2i \in [8\ldots11]$ in every line of S-box.
- This new S-box is termed as S-box 0 i, and here i ranges from 1 to n where n denotes total iterations.

Step 4. Compute nonlinearity nli of S-box 0 i.

Step 5. If $nli > nlmax$ then set: $nlmax = nli$, increase j by 1 and S-box$j$ = S-box 0 i. When $i \leq n$ then increment i and go to step 3, otherwise to step 6.

Step 6. Once step 3 is repeated n times, j S-boxes are obtained with maximum nonlinearity. Figure 2 shows the Genetic algorithm's flow chart-based S-box optimization.

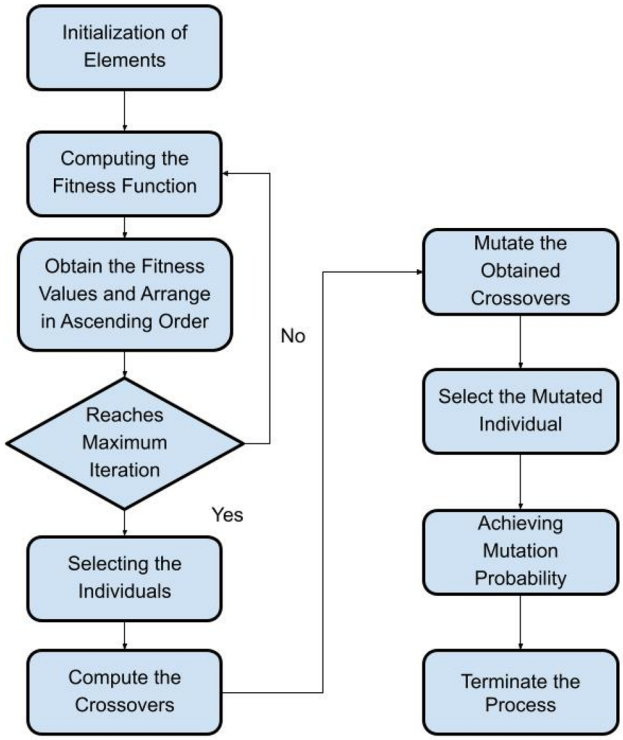

Fig. 2 Genetic algorithm's flow chart-based S-box optimization

Performance analysis

In the experiment, the benchmark is set with three parameters such as nonlinear degree criteria, differential uniformity criteria and strict avalanche effect criteria and hence the proposed Golod–Shafarevich feeder Immune Genetic Algorithm S-box (GSIGA-Sbox) is compared with Reversed Genetic Algorithm S-box (RGA-Sbox) and Discrete Space Chaotic S-box (DSC-Sbox). The designing of circuits is utilized by the xillinx tool and the layouts were drawn by the utilization of the CAD tool. The netlist of post-layout was then estimated with respect to aforementioned parameters that are obtained through the detailed simulations of transistor level through the usage of LTSpice ver4.13 CAD simulator.

- Nonlinear criteria are one significant measure to estimate the S-box performance. When nonlinearity is high, the capability of S-box is strong enough to withstand nonlinear attack. The criteria are given by

$$Nf = \min\,[dH(f, l)].$$

- The strict avalanche effect can be measured by SAC correlation matrix. The S-boxes satisfy the strict avalanche effect if each of the sac correlation matrix is close to 0.5.
- By inverting the plaintext bits to generate vector sets, if the vector sets are independent of each other, S-boxes satisfy the independence criterion of output bits. The independence of avalanche vector pairs can be measured by calculating the differential uniformity criterion.

Table 1 shows the comparison of existing RGA-Sbox and DSC-Sbox architecture with proposed GSIGA-Sbox architecture.

Figure 3 shows the comparison of various parameters between existing RGA-Sbox and DSC-Sbox with proposed GSIGA-Sbox where X axis shows various parameters such as Nonlinear Degree Criteria (NDC), Differential Uniformity Criteria (DUC) and Strict Avalanche Effect Criteria (SAEC). Y axis shows the values in percentage. When compared with the existing methods, the proposed GSIGA-Sbox architec-

Table 1 Comparison of various S-box architecture

Various S-box	Nonlinear degree criterion (%)	Differential uniformity criterion (%)	Strict avalanche effect criterion (%)
RGA-Sbox	74	47.3	23.6
DSC-Sbox	36.4	56	56%
GSIGA-Sbox	23.2	78	76.4

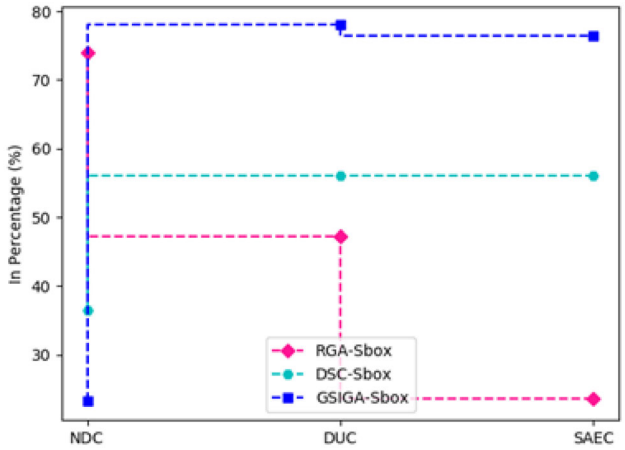

Fig. 3 Analysis for various S-box architecture

ture achieves 23.2% of NDC, 78% of DUC and 76.4% SAEC.

Analysis of Golod–Shafarevich exponent

There are three 1D chaotic maps. The first one is called as logistic map, where $xn + 1$ is a state variable, k^2 (0, 4] is a parameter of control and n denotes the total iterations. The second one is sine map and he third one is bifurcation map which are as shown as follows.

From Figs. 4, 5 and 6, it is concluded that the ergodicity is poor and there exists few periodic windows of logistic map and sine map, their exponent are low, none of them is more than 4, which shows that the above two chaotic systems are defective and chaotic dynamic behavior can be improved. The bifurcation diagram proves that dynamic state of the system is always in a stable chaotic state. Therefore, the system is a chaotic system with good chaotic characteristics.

Table 2 shows the comparison of proposed Golod–Shafarevich feeder Immune Genetic Algorithm S-box with 15 nm CMOS technology.

Figure 7 illustrates the analysis for proposed GSIGA-Sbox with 15 nm CMOS technology where X axis shows the time in milliseconds used for analysis and Y axis shows the average values obtained in percentage. It is found that the proposed

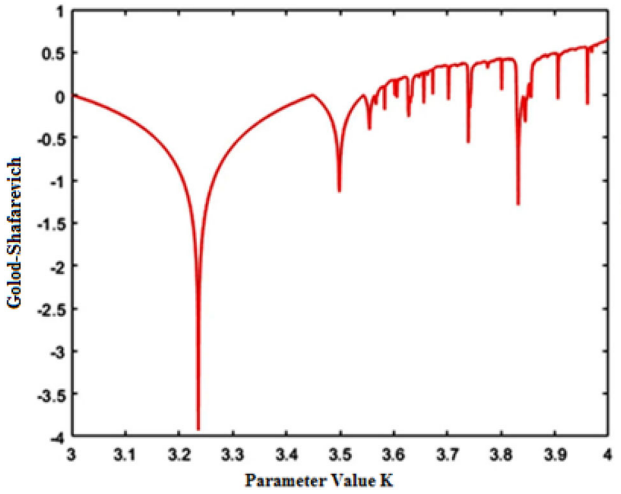

Fig. 4 Analysis of logistic map

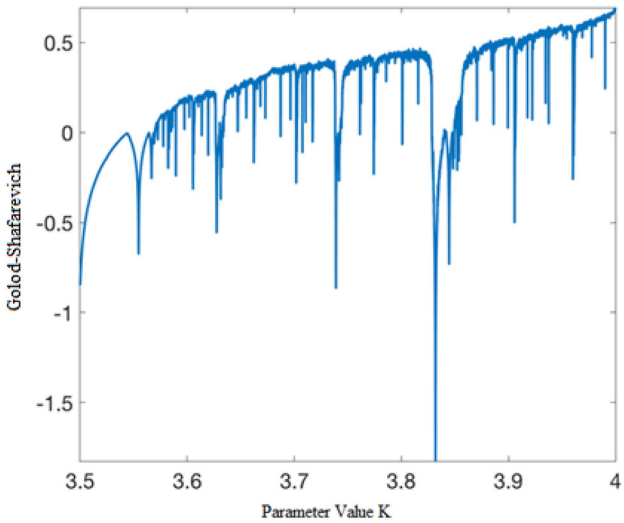

Fig. 5 Analysis of sine map

GSIGA-Sbox achieves better encrypt and decrypt speeds in less power consumption.

Table 3 indicates the comparison of Golod–Shafarevich feeder Immune Genetic Algorithm S-box proposed with 35 nm CMOS technology.

Figure 8 illustrates the analysis for proposed GSIGA-Sbox with 35 nm CMOS technology where X axis shows the time in milliseconds used for analysis and Y axis shows the average values obtained in percentage. It is found that the proposed GSIGA-Sbox achieves better encrypt and decrypt speeds in less power consumption.

Table 4 shows the comparison of existing RGA-Sbox and DSC-Sbox architecture with proposed GSIGA-Sbox architecture in terms of 15 nm and 35 nm CMOS technology.

Figure 9 shows the comparison of various parameters between existing RGA-Sbox and DSC-Sbox with proposed

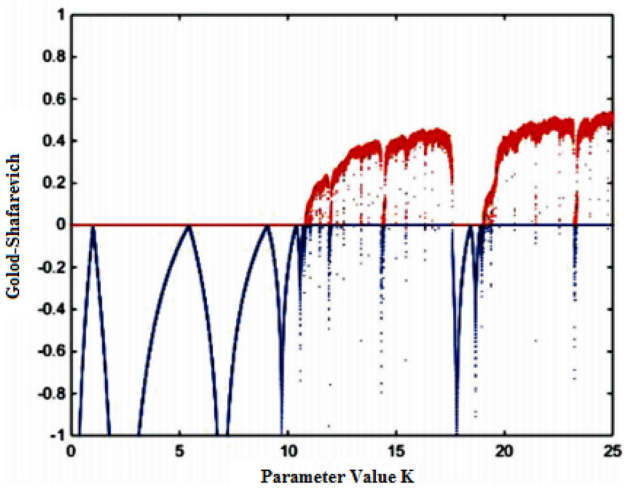

Fig. 6 Analysis of bifurcation map

Table 2 Comparison of proposed GSIGA-Sbox with 15 nm CMOS technology

Time (ms)	Encrypt in speed	Decrypt in speed	Power consumption (%)
5	56.4	50.6	27.4
10	56.9	50.8	27.7
15	57.1	51	28.2
20	57.5	51.4	28.7
25	57.9	51.8	28.9

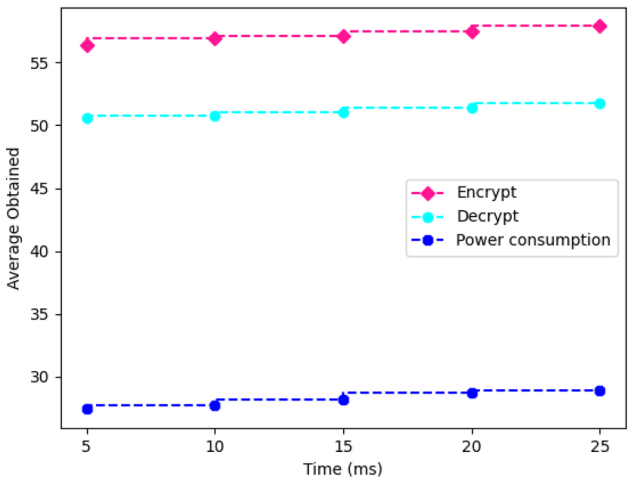

Fig. 7 Analysis for proposed GSIGA-Sbox with 15 nm CMOS technology

GSIGA-Sbox, whereas *X* axis shows various CMOS nm range and *Y* axis shows the average values. When compared with existing method, the proposed GSIGA-Sbox architecture achieves encryption speed of 61 MHZ, decryption speed of 55 MHZ with 24% of power consumption for 35 nm CMOS technology and 57 MHZ encryption speed, 51 MHZ

Table 3 Comparison of proposed GSIGA-Sbox with 35 nm CMOS technology

Time (ms)	Encrypt in speed (MHZ)	Decrypt in speed (MHZ)	Power consumption (%)
5	60.5	54.4	23.4
10	60.9	54.7	23.6
15	61	55.3	24.2
20	61.5	55.7	24.7
25	61.9	55.8	24.9

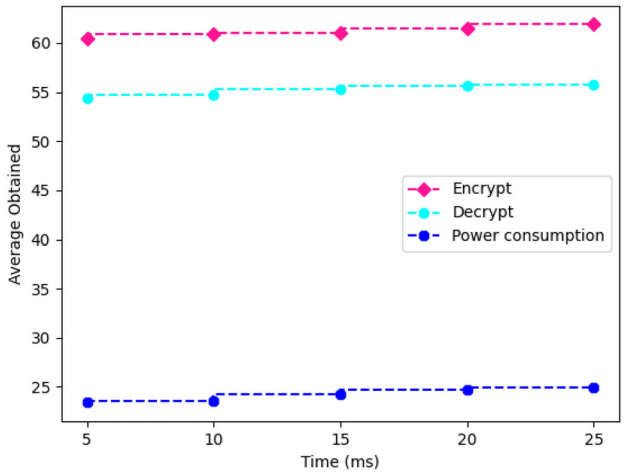

Fig. 8 Analysis for proposed GSIGA-Sbox with 35 nm CMOS technology

Table 4 Comparison of S-box with 15 nm and 35 nm CMOS technology

Techniques	Encrypt in speed (MHZ)	Decrypt in speed (MHZ)	Power consumption (%)
GSIGA-Sbox in 35 nm	61	55	24
GSIGA-Sbox in 15 nm	57	51	28
RGA-Sbox in 35 nm	45	42	30
RGA-Sbox in 15 nm	39	33	34
DSC-Sbox in 35 nm	35	31	47
	32	27	50

decryption speed with 28% of power consumption for 15 nm CMOS technology.

Conclusion

More attention is required to construct robust cryptographic substitution-boxes which is the major problem that has been

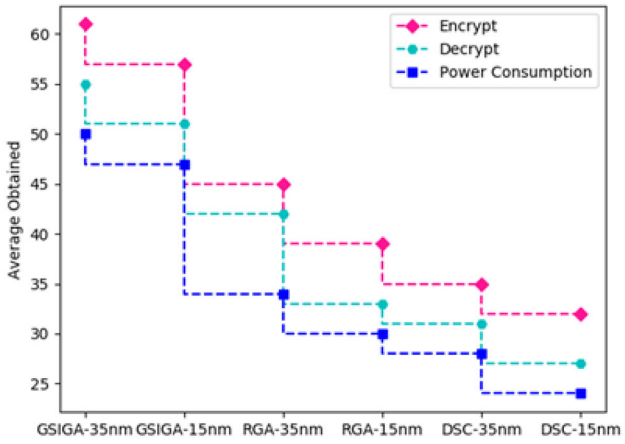

Fig. 9 Analysis for various S-box architecture

addressed. As the search space of S-boxes is broad, a random search approach is not the right choice as it provides no guarantee in quality of S-box. Conversely, optimization techniques have been examined for developing an automatic search mechanism for stronger S-boxes. For nonlinearly generating better S-boxes, a Golod–Shafarevich feeder Immune Genetic Algorithm S-box Algorithm (GSIGA) which is an optimization mechanism is presented in this research work. In this proposed S-box approach, S-box's initial population is randomly produced by utilizing new chaotic map. With the proposed optimization technique, consider the nonlinearity as fitness function to find the optimal S-box. The results obtained reveal that the S-boxes proposed offer higher nonlinearity and also satisfy other performance criteria. Further, the comparative analysis discloses the efficiency of optimization-based mechanism for S-boxes which is appropriate for applications involving cryptographic methods. The future work is to include power gating method to improve the overall speed.

Author contributions Both the authors equally contributed their skills and effort to produce this article.

Funding Not applicable.

Availability of data and materials Data and coding will be shared whenever it is required for the review.

Declarations

Conflict of interest Not applicable.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

1. Verma G, Shekhar S, Shashi Kant K, Verma V, Verma H, Pandey B (2016) SSTL IO standard based low power arithmetic design using CalanaKalanabhyamOn FPGA. Int J Control Autom 9(4):271–278
2. Verma G, Verma V, Sharma D, Kumar A, Verma H, Kalia K (2016) Design goal based implementation of energy efficient Greek unicode reader for natural language processing. Int J Smart Home 10(3):181–190
3. Hodjat A, Verbauwhede I (2006) Area-throughput trade-offs for fully pipelined 30 to 70 Gbits/s AES processors. IEEE Trans Comput 55(4):366–372
4. Wolkerstorfer J, Oswald E, Lamberger M (2002) An ASIC implementation of the AES SBoxes. In: Topics in Cryptology—CT-RSA 2002, Springer Verlag, pp 67–78
5. G. Bertoni, M. Macchetti, L. Negri, and P. Frangneto, "Power-efficient ASIC Synthesis of Cryptographic Sboxes," in Proc. the 14th ACM Creat Lakes symposium on VLSI (GLSVLSI 2004), pp. 277–281, 2004.
6. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V et al (2011) Scikit-learn: machine learning in python. J Mach Learn Res 12:2825–2830
7. Prabukumar M, Agilandeeswari L, Ganesan K (2019) An intelligent lung cancer diagnosis system using cuckoo search optimization and support vector machine classifier. J Ambient Intell Human Comput 10(1):267–329
8. Rodwald P, Mroczkowski P (2006) How to create" good "S-boxes?". In: 1st international Conference for young researchers in computer science. Control, electrical engineering and telecommunications ICYR, pp 18–20
9. Yin W, Mavaluru D, Ahmed M et al (2019) Application of new multiobjective optimization algorithm for EV scheduling in smart grid through theuncertainties. J Ambient Intell Human Computer 11:2071–2103
10. Morioka S, Katayama Y (2017) design methodology for a one-shot reed-solomon encoder and decoder. In: International Conference on computer design (ICCD '99), pp 60–67. IEEE, October 2017
11. Wang Y, Wong K-W, Li C, Li Y (2012) A novel method to design S-box based on chaotic map and genetic algorithm. Phys Lett A 376(6–7):827–833
12. Guesmi R, Farah MAB, Kachouri A, Samet M (2014) A novel design of Chaos based S-Boxes using genetic algorithm techniques. In: Proceedings of the 2014 11th IEEE/ACS International Conference on Computer Systems and Applications, AICCSA 2014, pp 678–684, Qatar, November 2014
13. Ahmad M, Bhatia D, Hassan Y (2015) A novel ant colony optimization based scheme for substitution box design. Proc Comput Sci 57:572–580
14. Tian Y, Lu Z (2016) S-box: Six-dimensional compound hyperchaotic map and artificial bee colony algorithm. J Syst Eng Electron 27(1):232–241
15. Tian Y, Lu Z (2017) Chaotic s-box: intertwining logistic map and bacterial foraging optimization. Math Probl Eng 6969311:11

16. Ahmad M, Mittal N, Garg P, Maftab Khan M (2016) Efficient cryptographic substitution box design using travelling salesman problem and chaos. Perspect Sci 8:465–468
17. Farah T, Rhouma R, Belghith S (2017) A novel method for designing S-box based on chaotic map and Teaching–Learning-Based Optimization. Nonlinear Dyn 88(2):1059–1074
18. Hussam M, Zolkipli F, Ahmad M (2018) A novel efficient substitution-box design based on firefly algorithm and discrete chaotic map. Neural Comput Appl 31:7201–7210
19. Zhang T, Chen CL, Chen L, Xu X, Hu B (2018) Design of highly nonlinear substitution boxes based on I-Ching operators. IEEE Trans Cybern 48:1–10
20. Alzaidi AA, Ahmad M, Doja MN, Solami EA, Beg MM (2018) A new 1D chaotic map and β-hill climbing for generating substitution-boxes. IEEE Access 6(1):55405–55418
21. Ye T, Zhimao L (2018) Chaotic S-box: six-dimensional fractional Lorenz–Duffing chaotic system and O-shaped path scrambling. Nonlinear Dyn 94(3):2115–2126
22. Al Solami E, Ahmad M, Volos C, Doja M, Beg M (2018) A new hyperchaotic system-based design for efficient bijective substitution-boxes. Entropy 20(7):525

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.