IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 7, 2020, accepted June 2, 2020, date of publication June 23, 2020, date of current version July 2, 2020.

*Digital Object Identifier 10.1109/ACCESS.2020.3004449*

# Particle Swarm Optimization Based Highly Nonlinear Substitution-Boxes Generation for Security Applications

**MUSHEER AHMAD[1], ISHFAQ AHMAD KHAJA[1], ABDULLAH BAZ[2], (Senior Member, IEEE), HOSAM ALHAKAMI[3], AND WAJDI ALHAKAMI[4]**

[1]Department of Computer Engineering, Jamia Millia Islamia, New Delhi 110025, India
[2]Department of Computer Engineering, College of Computer and Information Systems, Umm Al-Qura University, Mecca 21955, Saudi Arabia
[3]Department of Computer Science, College of Computer and Information Systems, Umm Al-Qura University, Mecca 21955, Saudi Arabia
[4]Department of Information Technology, College of Computers and Information Technology, Taif University, Taif 21974, Saudi Arabia

Corresponding author: Musheer Ahmad (mahmad9@jmi.ac.in)

This work was supported by the Deanship of Scientific Research at Umm Al-Qura University under Grant 19-COM-1-01-0015.

**ABSTRACT** Symmetric encryption has been considered as one of the essential means of ensuring security of end to end communication. The robustness and strength of modern day block encryption systems are based on the cryptographic features of substitution-boxes which are used to inject confusion ability during substitution-phases. In this paper, as an alternative to random, chaos or algebraic based construction methods, we propose to present an efficient method for the generation of cryptographic highly nonlinear substitution-boxes. The nature-inspired particle swarm optimization is reconnoitered to develop proposed method wherein the initial population is generated through a simple but with rich dynamics chaotic Renyi map. The anticipated method is analyzed for different scenarios such as change in population size, number of iterations, and linear increase in inertial weight. The performance assessment of generated S-boxes under standard criterions corroborate that the proposed method has excellent cryptographic features and found grander than many recent optimization based S-box methods. Moreover, an image encryption application of proposed S-boxes is also suggested to determine their suitability and applicability for image based security applications.

**INDEX TERMS** Particle swarm optimization, substitution-box, chaotic Renyi map, image encryption.

## I. INTRODUCTION

The ongoing advancement in communication technologies like Internet of Things, cloud computing, social media platforms and digital advanced gadgets have made it real fast to transmit digital data in past few years. The easy exchange of data in various forms like audio, video, images, etc., raises a concern of security and protection of this data. The basic security requirements like integrity, non-repudiation, authentication and confidentiality are mandatory for secure data transfer. The field of secure communication ensures the above mentioned criteria and also ensures to prevent the illegal spying, interruption or access [1]. The researchers preferred the schemes like digital watermarking for copyright protection. But for data protection and hiding, the most reliable techniques are cryptography and steganography. The sole aim of both these is to transform the data into some unintelligible and incomprehensible form for the intruder [2]. Cryptography is further classified as symmetric-key and asymmetric-key cryptography depending on the nature of key. The former has a single private key for encryption and decryption, both. Whereas, the later one has two different keys: the public key and the private key. The symmetric key cryptography has two broad known forms i.e., stream ciphers and block ciphers. The stream ciphers usually deal with encryption of plaintext data in bit-by-bit or byte-by-byte basis. Whereas, in block ciphers, the data is in the form of chunks of bits and encryption process is operated to obtain the chunks of encrypted data. Some of the modern day block ciphers are AES, IDEA, KASUMI, SAFER, SHARK, etc [3].

Mathematically, an $m \times n$ substiutution-box is defined as $S$: $\{0, 1\}^m \rightarrow \{0, 1\}^n$. Thus there are '$m$' component functions

The associate editor coordinating the review of this manuscript and approving it for publication was Jun Wang.

each being a map from '$m$' bits to 1 bit. An $m \times n$ S-box can be implemented as a lookup table with $2^m$ of $n$-bits each [4]. The relationship functions between the input and output values of the S-box must be chosen in the aim to satisfy the security criteria. One of the significant criterions is the nonlinearity [5]. The static S-box tables are used in ciphers such as DES, AES, etc, but in some ciphers the S-boxes are dynamically constructed from the key such as the Blowfish and Twofish encryption algorithms. S-boxes are the only nonlinear component in any symmetric encryption system. They follow the confusion principle presented by Claude Shannon in 1949. The confusion architecture is very effective in achieving secrecy if used correctly. Therefore, the S boxes need to be robust and efficient to tackle any sort of differential attack or attacks made on the bases of linear content of S-box. That is why it is vital to keep the nonlinearity in mind when designing an S-box [6].

For over two decades, much research has been dedicated to the use of chaos to generate nonlinear S-boxes. But, mostly the nonlinearity achieved by them has not been so impressive. To construct S-boxes with good nonlinearity, Khan *et al.* [7] applied a fractional linear transformation along with multiple chaotic systems to obtain an S-box. It is an easy and simple way but the nonlinearity content was not satisfactory enough. Later, in 2015, Ahmad *et al.* [8] proposed a new technique, in which the input elements of the S-box were generated using piecewise linear chaotic map, then raster and zigzag pattern scanning is applied to the initial S-box to obtain the final S-box. Özkaynak *et al.* [9] also presented a new S-box using the fractional order chaotic Chen system. Wang *et al.* [10] used a new three dimensional continuous chaotic map with infinite equilibrium points to design an S-box, but its nonlinearity was also not good enough. Liu *et al.* [11] proposed employing spatiotemporal chaos to generate random S-boxes. He used the non-adjacent coupled map lattices and Arnold's cat map to extract the spatiotemporal chaotic behavior of the system. Lambić [12] used existing chaos based S boxes [13], [14] to derive a new S-box by defining a new composition approach. Similarly, Tian and Lu [15] proposed a novel approach to constructing S-boxes. He proposed to use a comparatively new version of the logistic map, named as intertwined logistic map. He combined the intertwined logistic map [16] with the Bacterial foraging algorithm [17] to derive a new S-box. Zaibi *et al.* [18] proposed an approach to use 1-D chaotic maps like the logistic map and the piecewise linear chaotic map to generate a new S-box. The nonlinearity is not mentioned in the paper. Ahmad *et al.* [19] proposed to chaotically modify the trajectory of the piecewise linear chaotic map and logistic map to eliminate the gently decreasing peaks to obtain sharp peaks of the modified chaotic map. Then later, the modified map is scanned in a zigzag fashion to obtain better results to generate a random S-box. Belazi and El-Latif [20] proposed to employ the chaotic sine map to derive a new S-box with nonlinearity greater than 105. In more recent works, many schemes have been proposed to construct S-boxes, but they all capable to offer the low nonlinearity [21], [22].

In this paper, we investigated the particle swarm optimization with an aim to generate highly nonlinear substitution-boxes. The proposed method involves the initial population of S-boxes generation with the help of chaotic Renyi map, PSO based S-box optimization, adjustment (if needed) to maintain bijectivity of generated S-box. The main contributions of the paper are briefed as follows:

1. A particle swarm optimization approach based S-box generation method is proposed. The nonlinearity score is taken as the fitness parameter during optimization.
2. Chaotic Renyi map which has rich dynamical features is adopted for generation of initial S-boxes, and other random parameters of PSO for good exploration and exploitation of state space.
3. The well accepted standard performance parameters are chosen to evaluate the features of proposed S-boxes and method.
4. Comparative study is carried out which demonstrates the exceptionally well performance of proposed S-box work over many recent optimization based methods.
5. An image encryption algorithm utilizing the obtained S-boxes is also proposed which shows that they offer good encryption quality and features.

The outline of remaining portion of the paper is as follows. Some details of particle swarm optimization technique are provided in Section 2. The performance criterions for substitution-boxes are discussed in Section 3. Section 4 is develop to provide and discuss the proposed PSO based S-box generation method. The performance assessment and comparison of generated S-boxes under different scenarios are carried out in Section 5. The application of generated S-boxes for image encryption application is discussed, assessed, and compared in Section 6. Finally, the conclusions of the research findings are mentioned in Section 7.

## II. PARTICLE SWARM OPTIMIZATION

Particle swarm optimization is a population-based heuristic for global optimization put forward by Kennedy and Eberhart in 1995 [23]. PSO is based on notion of swarm intelligence (bird and fish flock movement behaviour). Birds are either scattered or go together while searching for food, before they locate the place where they can find the food. While searching the food, there is always one bird that smells food better than others, that is bird is perceptible of the place where the food can be seen, means having the food source information better than others. Since these birds continuously transmit information such as good information at any time while searching for food the flock will eventually move to the place where food can be found. In PSO, the solution, the birds moving starting with one spot onto the next, is equivalent to the improvement of the solution swarm, good information is equivalent to the most optimist solution, and the food asset is equivalent to

the most optimist solution during the entire course. The most optimist solution can be turned out in PSO calculation by the collaboration of every bird. This algorithm can be utilized to work out the intricate optimist problems. Owing to its many merits, PSO algorithm has been applied to numerous areas in optimization individually and in combination with other existing algorithms. It has been used widely in the fields such as neutral network training, function optimization, automatic adaptation control model classification, vague system control, machine study, the signal procession, and etc, [24].

In particle swarm optimization algorithm, population consists of "$N$" particles, and the position of each particle stands for the potential solution in $d$-dimensional space. Each particle's position in the swarm is affected by both the most optimist position during its movement (individual experience, called as personal best or *pBest* of particle) and the position of the most optimal particle in its neighborhood (near experience, called as global best or *gBest*). Particle in swarm fly into the search space by their exploration and exploitation capabilities and use personal best and global best positions to reach the best solution in PSO. Moreover, each particle is characterized by a velocity with which it explores the fitness function. The velocity and position of each particle are revised after each successive iteration of the algorithm. According to PSO dynamics, the speed and position of each particle are updated using the following formulations [25].

$$v_{id}^{k+1} = v_{id}^{k} + c_1 r_1 (pbest_{id}^{k} - x_{id}^{k}) + c_2 r_2 (gbest_{id}^{k} - x_{id}^{k}) \quad (1)$$

$$x_{id}^{k+1} = x_{id}^{k} + v_{id}^{k+1} \quad (2)$$

where, $v_{id}^{k}$ and $x_{id}^{k}$ stand for velocity and position of the particle "$i$" at its "$k$" times and the $d$-dimension quantity of its position; $pbest_{id}^{k}$ represents the d-dimension quantity of the individual "$i$" at its most optimist position at its "$k$" times. $gbest_{id}^{k}$ is the $d$-dimension quantity of the swarm at its most optimist position. The PSO parameters c1 and c2 represent the speeding figure to regulate the length when flying to the particle of the whole swarm and to the most optimist individual particle. The proper figures for c1 and c2 can control the speed of the particle's flying and the solution will not be the partial optimism; r1 and r2 represent random fiction in [0, 1] range. The motivation behind the selection of PSO for solving the problem of designing strong substitution-boxes includes: (1) PSO can be applied into both scientific and engineering use as it is based on intelligence, (2) the PSO calculations are very simple, and (3) it involves reduced number of parameters to tune and constraints acceptance compared to other derivative-free methods of optimizations. The procedure of PSO algorithm is provided as Algorithm-1.

**Algorithm 1** Particle Swarm Optimization

Initialize the number of particles $N$, particle positions $x_i$, velocity $v_i$, $c_1$, $c_2$, $r_1$, $r_2$, personal best $pbest_i$, global best $gbest$, maximum iterations $max\_itr$, where $i = 1, 2, \ldots, N$.
for $k \leftarrow 1$ to $max\_itr$ do:
  for $i \leftarrow 1$ to $N$ do:
    Compute the fitness of particle $x_i$ as $fitness(i)$
    if ($fitness(i) \geq fitness(pbest_i)$) then:
      $pbest_i \leftarrow x_i$
    endif
    if ($fitness(pbest_i)) \geq fitness(gbest)$) then:
      $gbest \leftarrow pbest_i$
    endif
    Update velocity of particle using Eqn. (1)
    Update position of particle using Eqn. (2)
  endfor
endfor

## III. SUBSTITUTION-BOX SECURITY PARAMETERS

The standard cryptographic parameters accounted for the security assessment of substitution-boxes by the researchers are discussed in this section. In general, the suite of performance criterion includes parameters such as bijectivity, nonlinearity, strict avalanche criterion, bits independence criterion, differential uniformity, linear approximation probability, these parameters are described as follows.

### A. BIJECTIVITY

The balancedness of an 8-bit Boolean function fi is confirmed when the output bitstream said to have equal of number of zeros and ones. Mathematically [26],

$$hwt\left(\sum_{i=1}^{8} x_i f_i\right) = 2^7 = 128$$

where, $hwt()$ stands for the hamming weight, $x_i \in \{0, 1\}$ and $(x_1, x_2, \ldots, x_8) \neq (0, 0, \ldots, 0)$. An S-box is said to hold the bijectivity property when the all eight candidate Boolean functions $f_i$ are balanced.

### B. NONLINEARITY

The nonlinearity measure of a Boolean function $f$ is computed by knowing the least distance of $f$ to the set of all affine functions [27]. Thus, the constituent functions of S-box should have standing nonlinearities scores. The nonlinearity $NL_f$ of any Boolean function $f$ is computed as:

$$NL_f = \frac{1}{2}(2^n - WH_{max}(f))$$

where, $WH_{\max}(f)$ is the Walsh-Hadamard transform of Boolean function $f$ [28]. A Boolean function is deemed frail if it tends to have poor nonlinearity. The maximization of nonlinearity of balanced Boolean functions is considered one of the prominent measures responsible for providing power against the any type of linear attacks [28], [29].

### C. STRICT AVALANCHE CRITERIA

The strict avalanche criterion was described by Tavares and Webster, which gets its base on the completeness effect's notion and the avalanche [26]. This criterion measures that

by making a single change in input bits, how much output bits get altered. The SAC assumed as satisfied when all the output bits are changed with a likelihood of 0.5, when only one input bit is flipped.

### D. BITS INDEPENDENCE CRITERIA

The input bits which remain unchanged are explored under bits independence criterion. The revamping of independent performance of pairwise variables of avalanche vectors and unaltered input bits are the assets of this measure. It is an effective criterion in symmetric cryptosystem, because by augmenting independence between bits, the recognition and prediction of patterns of the system is not possible [30].

### E. DIFFERENTIAL UNIFORMITY

The differential uniformity measures the resistivity of an S-Box against the differential cryptanalysis. The attack procedure of cryptanalysis was given by Biham and Shamir; it is related with developing imbalance on the input/output dissemination to assault block ciphers and S-boxes [31]. Confrontation to this cryptanalysis can be consummate if the EX-OR of each output has identical uniformity with the EX-OR value of each input [32]. If an S-box is uniform in input/output distribution, then it is said to be resistant. It is preferred that the largest value of differential uniformity (DU) in EX-OR table should be as small as possible. The differential uniformity for a Boolean function $f(x)$ is measured as:

$$DU(S) = \max_{\delta a \neq 0, \delta b} (\#\{a \in A | S(a) \oplus S(a \oplus \delta a) = \delta b\})$$

where, set X holds all probable input values and the figure of its elements is $2^n$. The largest value of EX-OR table for an S-box should be as small enough to resist the cryptanalysis.

### F. LINEAR APPROXIMATION PROBABILITY

The method of linear approximation probability (LAP) is helpful in calculating the imbalance of an incident. The largest value of imbalance of an event is measured with the help of the analysis introduced by Matsui in [33]. There must be no difference between output and input bits uniformity. Each of the input bits with its results in output bits is examined individually. If all the input elements are $2^n$, the class of all possible inputs is d and the masks applied on the equality of output and input bits are respectively $m_a$ and $m_b$, then maximum linear approximation is the maximum number of same results and calculated as:

$$LAP(S) = \max_{m_a, m_b \neq 0} \left| \frac{\#\{a \in S | a.m_a = b.m_b\}}{2^n} - 0.5 \right|$$

A lower value of this probability indicates that S-box is more capable to fight against linear cryptanalysis attack.

## IV. PROPOSED PSO BASED S-BOXES GENERATION

This section provides the description of optimized substitution-boxes construction using proposed PSO based method. In order to utilize the capableness of PSO algorithm, the 1-D chaotic Renyi map is adopted to provide the control parameters c1, c2, random parameters r1, r2 every time they needed during the PSO optimization phase, and to generate initial population of S-boxes. The chaos based initialization and updating of PSO parameters is motivated by the investigation which reports that the embedding of chaos during velocity updating results in good exploration and exploitation of search space and can improves the quality of results [34]–[36].

### A. CHAOTIC RENYI MAP

The chaotic Renyi is one-dimensional dynamical mapping which holds some rich dynamical characteristics compared to many contemporary 1-D chaotic maps in terms of high lyapunov exponent, excellent bifurcation, uniform coverage of whole state space, and uniform distribution. It is one of the simplest discrete chaotic models whose dynamics is governed by the following equation [37], [38].

$$x_{i+1} = (c.x_i)\ mod\ (1) \tag{3}$$

where, c is its control and bifurcation parameter, $x_i$ is state variable which lies in [0, 1]. The analysis shows that the Renyi map has chaotic phenomenon only when $c > 1$ because the positive lyapunov exponent (an indicator of chaoticity) is seen for any $c > 1$. The lyapunov exponent spectrum for different value of c ranging from 0 to 10 is shown in Figure 1(a), the bifurcation behavior of Renyi map is depicted in Figure 1(b). The phase attractor of chaotic Renyi map in 3-D projection is given in Figure 1(c-d). These plots make evident the good dynamical features of the chaotic map in Eqn. (3). Where as, the Figure 1(d) indicates the uniform distribution or the histogram of chaotic floating-point values generated from the Renyi map. The chaotic Renyi map is adopted to generate the parameters of PSO and initial population of S-boxes to begin the optimization process.

### B. ALGORITHM

This section deals with the proposed algorithm to generate S-boxes using particle swarm optimization.

#### 1) INITIALIZE THE POPULATION VECTOR

For S-box optimization problem, each individual $8 \times 8$ S-box is considered as the particle. Initial population of $8 \times 8$ S-boxes is generated with the help of chaotic Renyi map. The generated population is key dependent and dynamic. As per the heuristics mentioned in [39], the size of initial population generated is kept low. The *gen_sbox*() method aims to generate a random $8 \times 8$ S-box using chaotic map whose procedure is provided in Algorithm-2. The method is repeated for number of population times to initialize the population of S-boxes.

#### 2) CALCULATE FITNESS VALUE

The mean nonlinearity of each S-box (particle) in the population vector is calculated and considered as its fitness value.

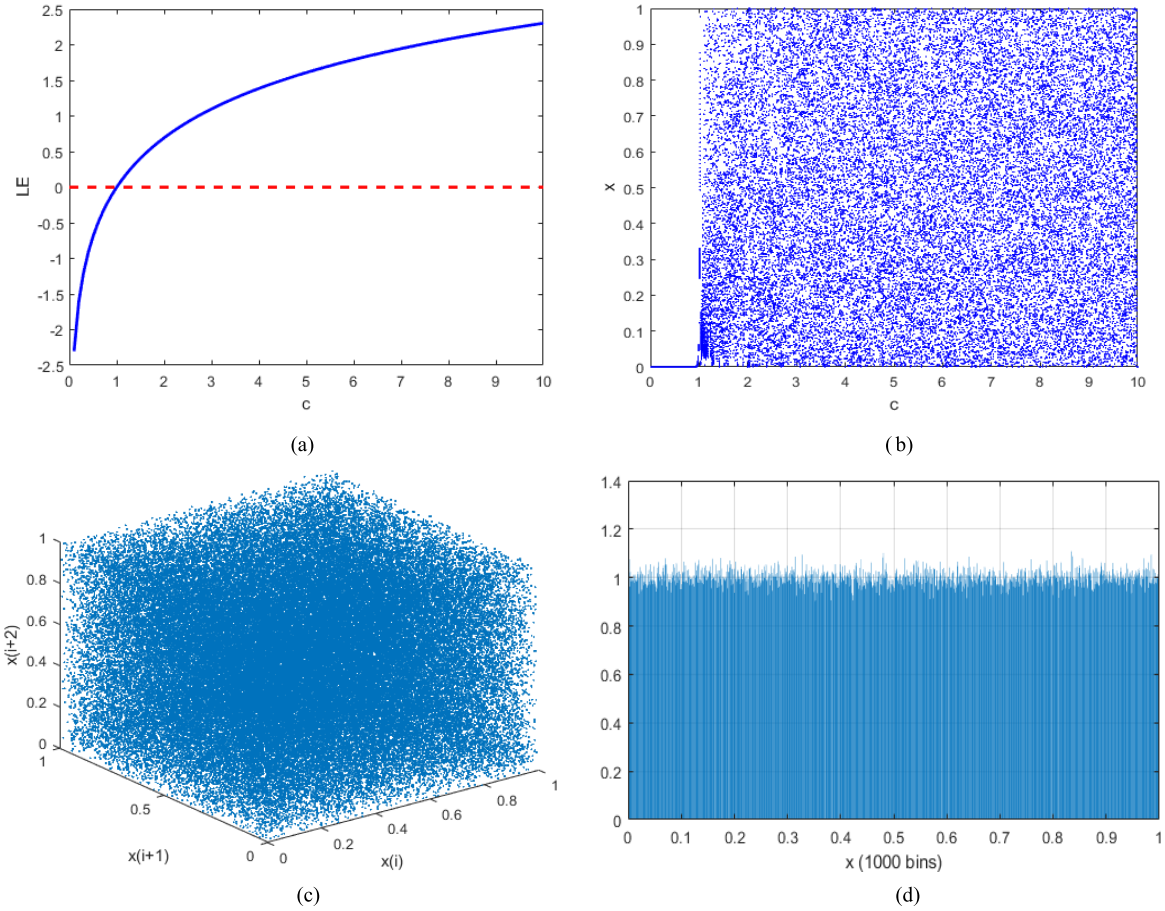



**FIGURE 1.** Dynamical features of chaotic Renyi map (a) lyapunov exponent diagram, (b) bifurcation diagram, (c) phase attractor plot in 3D, and (d) normalized histogram for 100000 chaotic points from map.

**Algorithm 2** Generate Initial S-Box

```
for k ← 1 to 256 do:
    xr ← renyi_map(xr, c, 1)
    A[k] ← xr
endfor
B ← sort(A)
for k ← 1 to 256 do:
u ← B[k]
for j ← 1 to 256 do:
    if (u == A[j]) then:
       sbox[k] ← j − 1
       break
    endif
endfor
endfor
return sbox, xr
```

### 3) INITIALIZE THE PSO VECTORS

The velocity vector is initialized to 0 and is updated on every successive iteration. Each individual position vector is initialized with the values of corresponding S-box in the population. The velocity vector is updated as per Eqn. (1) and position vector is updated as per Eqn. (2). The personal best vectors are initialized with the corresponding of S-boxes. The personal best vector gets revised with the S-box for which the newly generated population has better fitness value than previous one. The global best vector is initialized with S-box having highest nonlinearity in the population.

### 4) INITIALIZE THE PSO PARAMETERS

The PSO parameters like c1, c2, r1 and r2 are randomly initialized with the chaotic value produced using chaotic Renyi map. These parameters are updated randomly in each iteration during the optimization phase. However, the inertial weight $w$ of PSO is initialized (by default) to 0.6.

### 5) UPDATING AND ADJUSTMENT

The velocity and position vectors are updated using Eqns. (1) and (2). This process generates some redundant and negative values. However, the values of solution (S-box) are constraint to the range [0, 255] for the design of S-box. Therefore, in order to get rid of the possible redundant and negative values we apply preprocessing and adjustment process. In preprocessing, the negative values are made to lie in the desired range with some mathematics. In adjustment

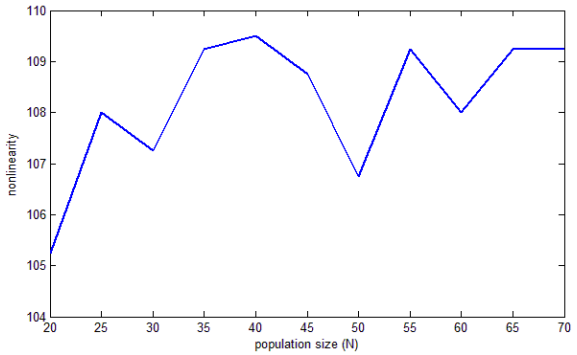



**FIGURE 2.** Variations in nonlinearity with increase in population size.

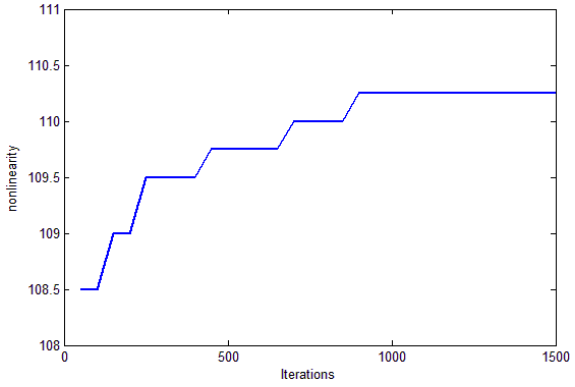



**FIGURE 3.** Variations in nonlinearity with increase in iterations.

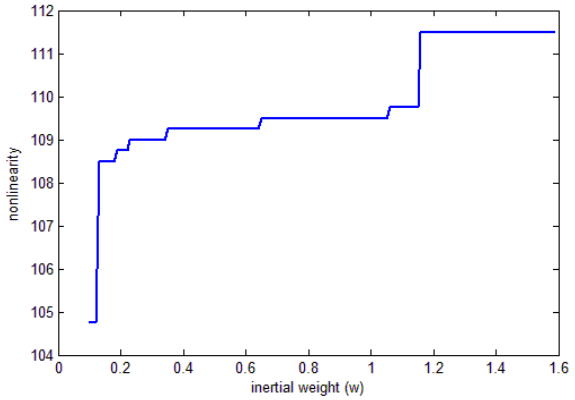



**FIGURE 4.** Variations in nonlinearity with linear increase in inertial weight.

process, we keep track of repeated values and replace them with the values that are missing so that the bijectivity of solution or population vector gets maintained. The fitness values are again calculated for newly generated population and the best yielding values among the new population (of size 2N, where N is initial population size) are preserved and are sent in next population. The personal best and global best are updated as mentioned before.

## V. PERFORMANCE ANALYSES AND S-BOX GENERATION

To simulate and analyze the performance of proposed S-box generation method using particle swarm optimization, we take the proposed method's input arguments as follows (without any loss of generality). The proposed method is analyzed under different scenarios and conditions by varying the population size, number of iterations, and inertial weight.

- Chaotic map's initial value, $xr = 0.1234$
- Chaotic map's parameter, $c = 137$
- Number of populations (default), $N = 40$
- Inertial weight (default), $w = 0.6$
- Maximum allowable iterations (default), $max\_itr = 250$

The results of optimization under different analyses are presented below.

### A. ANALYSIS FOR DIFFERENT POPULATION SIZE

Keeping the other input parameters as default; we vary the population size ($N$) in order to know the trend of best possible outcome in terms of S-boxes with optimized nonlinearity score. The effect of varying population size on nonlinearity of optimized S-boxes is shown in Figure 2. It has been noted that no steady state increment is seen if $N$ is increases from 20 to 70. But, the analysis suggests that the population size $N = 40$ is capable to generate an optimized S-box with nonlinearity of 109.5. Best possible S-box can be generated for $N = 40$. The best achievable S-box (named as S1) from this analysis is provided in Table 1. Based on this outcome, we keep the population size 40 fix to perform further analyses.

### B. ANALYSIS FOR DIFFERENT ITERATIONS

The trend of changing the number of maximum allowable iterations to carry out PSO optimization process is also explored and analyzed for $N = 40$ and $w = 0.6$. The effect of varying $max\_itr$ from 50 to 1500 is evaluated and nonlinearity results are plotted as Figure 3. Here, it is quite clear that the quality of S-boxes gets bettered in terms of their fitness score as the iteration gets increased. Optimized S-boxes with nonlinearity as high as 110.25 is achieved during this study, one such S-box is shown in Table 2 as S-box S2. However, no further improvisation on the fitness score is observed for $max\_itr$ above 1000. The value of $max\_itr$ as 1000 is chosen as favorable value of parameter for generating S-boxes with high nonlinearly.

### C. ANALYSIS FOR DIFFERENT INERTIAL WEIGHT

In PSO, the inertial weight factor $w$ is responsible for balancing between the global and local search competencies [40]. Instead of keeping the inertial weight w fix all the time, it is varied linearly in increasing order as per the rule given below.

$$w_{cur\_itr} = w_1 + (cur\_itr - 1)\left(\frac{w_2 - w_1}{max\_itr}\right) \tag{4}$$

where, $w_1$ and $w_2$ stands for initial and final value of inertial weight, $cur\_itr$ is current value of iteration, and $max\_itr$ is the maximum allowable iterations. In our problem, the increase in $w$ provides improvisation in getting the better results

**Algorithm 3** PSO Based Optimized S-Box Generation

Take input arguments as:
$N \leftarrow$ number of populations
*max_itr* $\leftarrow$ maximum number of iterations
xr $\leftarrow$ initial value of Renyi chaotic map
c $\leftarrow$ parameter of Renyi map
Generate initial population of S-boxes as:
*xr* $\leftarrow$ renyi_map(xr, c, 100) // *Iterating map for 100 times to remove transient effect of map*
*population* $\leftarrow$ zeros($2 \times N$, 256) // 256 is for $8 \times 8$ S-box
for $i \leftarrow 1$ to $N$
  [$sbox_i$, xr] $\leftarrow$ gen_sbox(xr, c)
  *population*[$i$] $\leftarrow sbox_i$
end for
*population*[1] $\leftarrow$ *aes_sbox*
Calculate the fitness score (nonlinearity) of each particle as:
for $i \leftarrow$1 to $N$ do:
*NL*[$i$] $\leftarrow$ nonlinearity(*population*[$i$])
end for
*NL_sorted* $\leftarrow$ sort(*NL*) // *in descending order*
*gBest* $\leftarrow$ *population*[1]
$pBest_i \leftarrow$ *population*[$i$]
*Vel* $\leftarrow$ zeros($N$, 256) // 256 is for $8 \times 8$ S-box
Take inertial weight $w$ // *default is* 0.6
Begin optimization phase as:
*While* (*max_itr* $> 0$) do:
  xr $\leftarrow$ renyi_map(xr, c); c1 $\leftarrow$ 2*xr
  xr $\leftarrow$ renyi_map(xr, c); c2 $\leftarrow$ 2*xr
  xr $\leftarrow$ renyi_map(xr, c); r1 $\leftarrow$ xr
  xr $\leftarrow$ renyi_map(xr, c); r2 $\leftarrow$ xr
  *NL* $\leftarrow$ *NL_sorted*
  for $i \leftarrow 1$ to $N$ do:
    for $j = 1$ to 256 do:
      *Vel*[i][j] $\leftarrow$ ceil($w$*Vel[i][j] + c1*r1*(*pBest*[i][j] - *population*[i][j]) + c2*r2*(*gBest*[j]-*population*[i][j]))
      if (*Vel*[i][j] $< 0$)
        *Vel*[i][j] $\leftarrow$ (*Vel*[i][j] + 256)*mod*(256)
      end if
      $X[i, j] \leftarrow$ int(*population*[i][j] + *Vel*[i][j])*mod*(256)
      *temp_sbox*[$j$] $\leftarrow X[i, j]$
    end for
    Perform adjustment to preserve the bijectivity in *temp_sbox*
    *population*[$N + i$] $\leftarrow$ *temp_sbox*
  end for
  for $i \leftarrow 1$ to $N$ do:
    *NL_sorted*[$N + i$] $\leftarrow$ nonlinearity(*population*[$N + i$])
  end for
  *NL_sorted* $\leftarrow$ sort(*NL_sorted*)
  Arrange *population* vector according to sorted fitness & keep first $N$ S-boxes and drop the rest
  for $i \leftarrow 1$ to $N$ do:
    if (*NL*[i] $<$ *NL_sorted*[i])
      *pBest*[i] $\leftarrow$ *population*[i]
    end if
  end for
  Update the gBest
  *max_itr* $\leftarrow$ *max_itr* - 1
end while
Ignore the first S-box and output the S-box with best fitness value from remaining S-boxes in *population* vector.

compared to decreasing the weight. For the setting $w_1 = 0.1$, $w_2 = 1.6$, *max_itr* $= 150$, the results of varying inertial weight $w$ as per Eqn. (4) is shown in Figure 4. Interestingly, the favorable variations in results have been seen and S-boxes

**TABLE 1.** Proposed optimized S-box S1.

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

**TABLE 2.** Proposed optimized S-box S2.

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

with nonlinearity as high as 111.5 are generated. One such optimized S-box out of this analysis is presented in Table 3 as S-box S3.

### *D. RESULTS AND COMPARISON OF S-BOXES*

The property of bijectivity an S-box ensures that it is a one-to-one mapping. Means, all possible output vectors should appear only once. Our all three S-boxes in Table 1 to 3 maintains the bijectivity as all 256 possible output values are distinct and appear only one time in each S-box. The bijectivity property is mandatory for S-boxes which are used in block ciphers based substitution-permutation networks (SPN) like AES, PRESENT, SHARK, SAFER, etc.

Designing S-boxes with high nonlinearity score have been one of the main challenges in the area of cryptography for last two decades. Through the proposed PSO based method, we are able to achieve S-boxes with high nonlinearities which are quite close to the best known $8 \times 8$ S-box nonlinearity of 112 [41]. The nonlinearity scores of all eight individual Boolean functions in each proposed S-boxes S1, S2 and S3 from our method and analyses are listed in Table 4. We compared the nonlinearity performance of our three S-boxes with the optimization-based S-boxes, wherein the optimization techniques such as bacteria foraging optimization (BFO) [15], JAYA optimization [42], cuckoo search (CS) [43], genetic algorithm (GA) [44], sine-cosine optimization (SCO) [45],

**TABLE 3.** Proposed optimized S-box S3.

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

**TABLE 4.** Nonlinearities of proposed optimized 8 × 8 S-boxes.

| S-box | $nl_1$ | $nl_2$ | $nl_3$ | $nl_4$ | $nl_5$ | $nl_6$ | $nl_7$ | $nl_8$ |
|---|---|---|---|---|---|---|---|---|
| S1 | 112 | 108 | 110 | 112 | 110 | 108 | 106 | 110 |
| S2 | 112 | 110 | 110 | 108 | 108 | 112 | 112 | 110 |
| S3 | 112 | 112 | 112 | 112 | 112 | 112 | 112 | 108 |

beta-hill climbing (BHC) search [46], firefly algorithm (FA) [47], artificial bee colony (ABC) optimization [48], teaching-learning based optimization (TLBO) [49], I-Ching's operators (ICO) optimization [50], ant colony optimization (ACO) [51], etc., in Figure 5 and Table 5. It is worth notable that the nonlinearities of our S-boxes are fairly high as compared to almost all optimization-based S-boxes recently investigated in [15], [42]–[44], [47]–[49], [51], [52]. Thus, the proposed PSO-based method is sufficiently efficient in generating highly nonlinear S-boxes.

We calculated the dependency matrices for each of proposed S-boxes in accordance with the procedure suggested in [26]. It has been found that all the entries of the dependency matrices are quite close to 0.5 with an average scores of 0.5068, 0.5046, and 0.5022 for S-boxes S1, S2, and S3, respectively. The proposed S-boxes satisfy the strict avalanche criteria well and SAC performance is comparable to S-boxes given in comparison Table 5.

The bits independence criterion is also checked following the Websters and Tavares guidelines [30]. Accordingly, the average BIC-NL scores for S-boxes S1, S2, and S3 are found as 106.86, 106.57, and 110.28, respectively. It is evident that the BIC performance of proposed S-boxes is considerably better compared to all the recently investigated optimization-based S-boxes listed in Table 5. Thus, the proposed method shows a dominating BIC performance.

The differential cryptanalysis given by Biham and Shamir can be mitigated if the S-box shows low differential uniformity. The differential uniformities of our S-boxes are computed as 8, 8, and 6 for S1, S2, and S3, respectively. As can be seen in the Table 5 that the proposed S-boxes have better potential to mitigate differential cryptanalysis than other S-boxes of the Table. Hence, the proposed S-boxes have excellent DU performance and exhibit great resistance against differential cryptanalysis.

The lowest value of linear approximation probability and high nonlinearity are sought to make the S-box robust against linear cryptanalysis. The LAP score of our three S-boxes are found as low as 0.1328, 0.1328, and 0.1094. The first two scores are better than [15], [43], [44], [51], [52], comparable to [42], [45], [50], and higher than [46]–[49]. Whereas, the LAP of third S-box S3 i.e. 0.1094 is comes out to be the best among all the S-boxes of the Table 5. Thus, the new S-boxes are able to offer better resistance to linear cryptanalysis than most of the recent optimization based S-boxes.

## VI. S-BOX BASED IMAGE SECURITY APPLICATION

This section reports the security application of proposed substitution-boxes in the area of image encryption. The S-boxes generated from proposed PSO based method are evaluated for their suitability and usage for securing the multimedia images. A novel simple image encryption scheme is suggested which is based on the generated S-boxes and chaotic Renyi map. The procedure of suggested new image encryption scheme based on S-box substitution is depicted through the flowchart shown in Figure 6.

The encryption process distorts the plain-image; the extent and quality of distortion determine the reliability of the

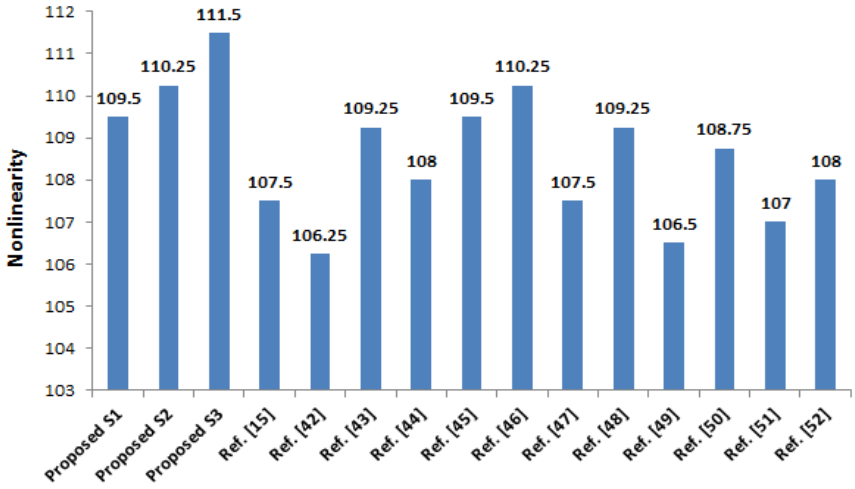



**FIGURE 5.** Comparison of mean nonlinearity scores of optimization based S-boxes.

**TABLE 5.** Comparison of cryptographic features of optimization-based 8 × 8 S-boxes.

| S-box method | Optimization | $NL_{min}$ | $NL_{avg}$ | SAC | BIC-NL | DU | LAP |
|---|---|---|---|---|---|---|---|
| Proposed S1 | PSO | 106 | 109.5 | 0.5068 | 106.86 | 8 | 0.1328 |
| Proposed S2 | PSO | 108 | 110.25 | 0.5046 | 106.57 | 8 | 0.1328 |
| Proposed S3 | PSO | 108 | 111.5 | 0.5022 | 110.28 | 6 | 0.1094 |
| Ref. [15] | BFO | 106 | 107.5 | 0.5093 | 103.07 | 10 | 0.1406 |
| Ref. [42] | JAYA | 104 | 106.25 | 0.5009 | 103.64 | 10 | 0.1328 |
| Ref. [43] | CS | 108 | 109.25 | 0.5075 | 102.93 | 12 | 0.1406 |
| Ref. [44] | GA | 108 | 108 | 0.5068 | 103.36 | 10 | 0.1406 |
| Ref. [45] | SCO | 108 | 109.5 | 0.4985 | 104.07 | 10 | 0.1328 |
| Ref. [46] | BHC | 110 | 110.25 | 0.5 | 105.21 | 10 | 0.125 |
| Ref. [47] | FA | 106 | 107.5 | 0.4943 | 104.35 | 10 | 0.125 |
| Ref. [48] | ABC | 108 | 109.25 | 0.4985 | 104.29 | 8 | 0.125 |
| Ref. [49] | TLBO | 104 | 106.5 | 0.4995 | 104.57 | 10 | 0.1172 |
| Ref. [50] | ICO | 108 | 108.75 | 0.4946 | 102.78 | 10 | 0.1328 |
| Ref. [51] | ACO | 106 | 107 | 0.5015 | 104.21 | 10 | 0.1484 |
| Ref. [52] | ABC | 106 | 108 | 0.5073 | 104 | 10 | 0.1523 |

encryption scheme. Therefore, it is imperative to examine the statistical and textural features of encrypted image through majority logic criterion (MLC). The MLC is a comprehensive exploration set of statistical performance metrics which includes the entropy, contrast, correlation, energy, and homogeneity as suggested in [53], [54]. The statistical stability of the generated S-boxes for offering quality of encryption effects is examined through this suite of MLC suite of analyses. Thus, the key motive of MLC analysis is the examination of the characteristics of S-boxes during the process of encryption. We applied our S-boxes to encrypt digital plain-images using our own encryption scheme and validate its suitability for multimedia security and image encryption applications. The S-boxes S1, S2, and S3 are applied separately to some standard images like *Cameraman*, *Baboon*, *Peppers*, each of size 256 × 256. These plain-images, encrypted images using S-boxes $S_1$, $S_2$, and $S_3$ along with their histograms are shown in Figure 7. Firstly, it can be seen that all the encrypted images have high visual distortion and indistinguishability. The distribution of pixels in the image is represented through the histograms. The non-uniform distribution of pixels in encrypted content may provide the clue of plain-image

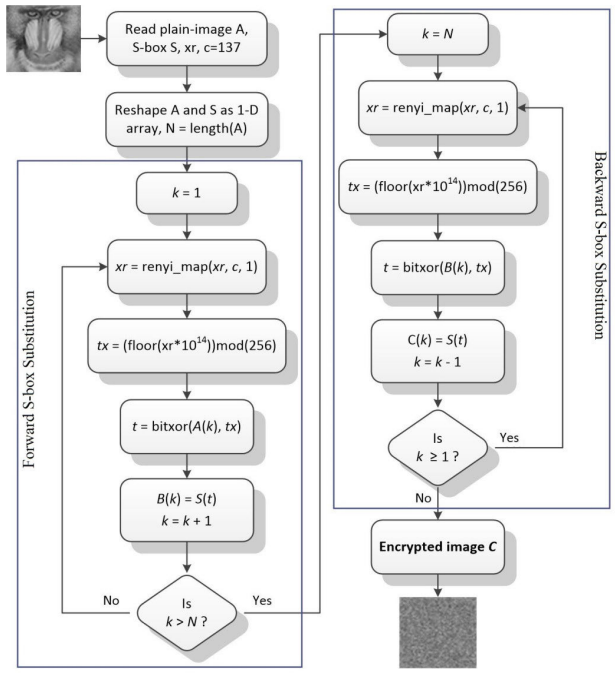



**FIGURE 6.** Flowchart of proposed image encryption scheme based on S-box substitution.

information and makes the encryption scheme susceptible to statistical attacks. But, as evident from the Figure 7 that the encrypted images from our scheme have significantly flat and uniform histograms. Thus, the results are showing excellent encryption effect in encrypted images. The statistical MLC performance and differential analysis are discussed in what follows.

### A. ENTROPY

The revolutionary concept of information entropy was coined by Claude E. Shannon in 1948. Entropy is the measure of randomness of information which is inherent in the variable's possible outcomes. Information entropy is usually measured in bits, corresponding to base 2, it is mathematically formulated as:

$$H(x) = \sum_i p(x_i) \log_2 \left( \frac{1}{p(x_i)} \right)$$

where, $p(x_i)$ is the probability of symbol of message source $x$. A higher value of entropy shows that the distribution of pixels gray values is more uniform. If the entropy of the encrypted image is way smaller than 8 (for an 8-bit encoded image), that indicates there are higher chances of predicting the image, thereby threatening its security.

### B. CORRELATION

The pixels values of an image vary between certain limits depending upon the number of bits used to encode one pixel of the image. The gray values of neighboring pixels decide the similarity or dissimilarity between the pixels. Resemblance of pixels to their neighboring pixels is measured through correlation coefficient. In the plain-images, there exists a strong correlation among neighboring pixels of a meaningful image. The correlation among pixels can be diminished through the encryption schemes to secure the sensitive content of the image. Therefore, negligible correlated values (near to zero) are considered robust in an insecure channel. The correlation coefficient for two sequences is given as:

$$\gamma = \sum \frac{(i - \mu i)(j - \mu j)}{\sigma_i \sigma_j}$$

where, $i$ represent the position of row and $j$ indicates its column value of image under examination. The parameters $\mu$ *and* $\sigma$ are the variance and standard deviation, respectively.

### C. ENERGY

The energy is applied to measure the localized change of the image. It's the rate of change in the color/brightness/magnitude of the pixels over local areas. This is especially true for edges of the things inside the image. For image energy analysis, we calculate the sum of squared members of gray level co-occurrences. As compared to the plain image energy, the energy of encrypted image is smaller as the values are randomly or uniformly distributed. Energy analysis involves the computation of its associated quantity which is calculated as:

$$E = \sum p(i, j)^2$$

where, $p(i, j)$ is the number of GLCM matrices.

### D. CONTRAST

In digital images, the contrast is the measure of luminance, or difference in it, which differentiates the objects from one another. Contrast is determined by difference in the color and brightness. Contrast analysis helps in visualizing the objects and the underlying information. The contrast and brightness of images are adjusted during the image processing for better visualization. Due to the nonlinear mapping of S-boxes the contrast varies directly with the randomness of images. Mathematically, contrast is computed as:

$$c = \sum |i - j|^2 \, p(i, j)$$

where, $p(i, j)$ represents the position of pixels in gray level co-occurrence matrix (GLCM).

### E. HOMOGEINITY

Homogeneity deals with the relationship of distribution of elements in gray level co-occurrence matrix (GLCM) with its diagonal. Its value depends on the diagonal of GLCM. Smaller the value of homogeneity indicates good encryption effect in the encrypted content. It follows the as per the equation given as:

$$h = \sum \frac{p(i, j)}{1 + |i - j|}$$

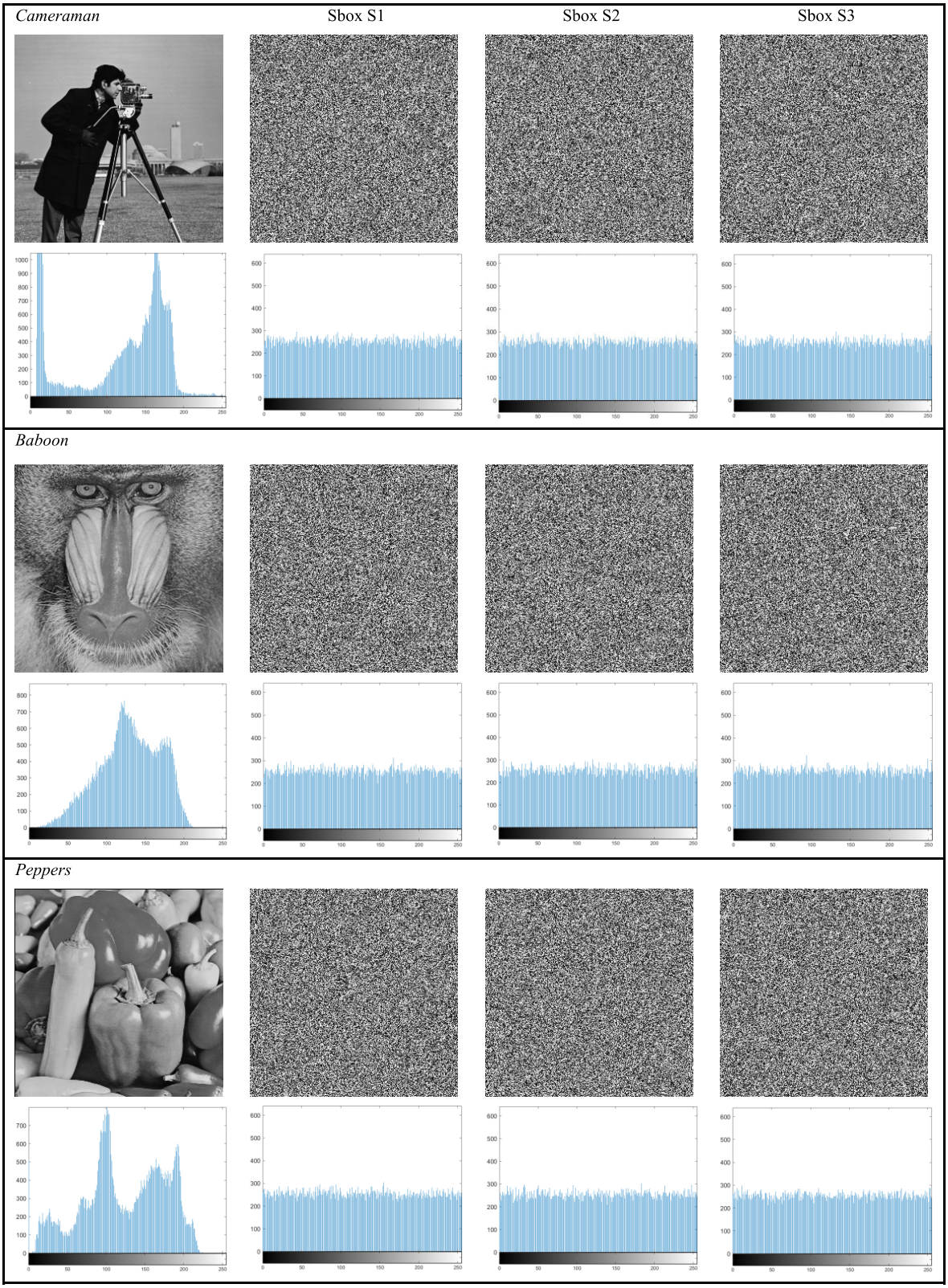



**FIGURE 7.** Plain-images and encrypted images using S-boxes S1, S2 and S3 with their corresponding histograms.

The results of MLC analysis (entropy, correlation, contrast, energy, homogeneity) to evaluate the encryption quality offered by proposed encryption scheme in images shown in Figure 7 are provided in Table 6. The obtained MLC analysis results are also compared with some contemporary image encryption schemes based on S-boxes substitution

**TABLE 6.** MLC results of entropy, correlation, contrast, energy, homogeneity from proposed encryption scheme and their comparison.

| Image | Entropy | Correlation | Contrast | Energy | Homogeneity |
|---|---|---|---|---|---|
| ***Cameraman*** | | | | | |
| Plain-image | 7.0097 | 0.92273 | 0.5872 | 0.18053 | 0.8953 |
| Proposed (S1) | 7.9971 | 0.00077 | 10.4909 | 0.01564 | 0.3887 |
| Proposed (S2) | 7.9974 | 0.00319 | 10.4993 | 0.01564 | 0.3912 |
| Proposed (S3) | 7.9969 | -0.00276 | 10.5248 | 0.01563 | 0.3889 |
| Ref. [55] | 7.9829 | 0.0023 | 8.5483 | 0.0174 | 0.4115 |
| Ref. [56] | 7.9431 | 0.0155 | 8.2113 | 0.0219 | 0.4248 |
| Ref. [57] | 7.9812 | -0.0045 | 8.3154 | 0.0177 | 0.4091 |
| Ref. [58] | 7.9591 | -0.0441 | 8.2314 | 0.0202 | 0.4151 |
| ***Baboon*** | | | | | |
| Plain-image | 7.264 | 0.79834 | 0.63265 | 0.09438 | 0.78209 |
| Proposed (S1) | 7.9969 | -0.00284 | 10.4806 | 0.01564 | 0.3893 |
| Proposed (S2) | 7.9964 | 0.00379 | 10.4267 | 0.01563 | 0.3898 |
| Proposed (S3) | 7.9969 | -0.00098 | 10.478 | 0.01564 | 0.3899 |
| Ref. [55] | 7.9851 | -0.0050 | 8.5792 | 0.0175 | 0.4076 |
| Ref. [56] | 7.9252 | 0.0119 | 8.0391 | 0.0222 | 0.4428 |
| Ref. [57] | 7.9824 | -0.0043 | 8.7348 | 0.0172 | 0.4074 |
| Ref. [58] | 7.9612 | -0.0512 | 8.1213 | 0.0210 | 0.4011 |
| ***Peppers*** | | | | | |
| Plain-image | 7.5327 | 0.93124 | 0.38498 | 0.1096 | 0.88806 |
| Proposed (S1) | 7.9973 | -0.00589 | 10.5348 | 0.01564 | 0.3884 |
| Proposed (S2) | 7.9969 | -0.00743 | 10.6008 | 0.01565 | 0.3879 |
| Proposed (S3) | 7.9975 | -0.00647 | 10.5738 | 0.01564 | 0.3884 |
| Ref. [55] | 7.9840 | -0.0017 | 8.4985 | 0.0175 | 0.4103 |
| Ref. [56] | 7.9233 | -0.0112 | 8.1423 | 0.0286 | 0.4648 |
| Ref. [57] | 7.9824 | -0.0043 | 8.7348 | 0.0172 | 0.4074 |
| Ref. [58] | 7.9562 | 0.0103 | 8.3129 | 0.0180 | 0.4219 |

investigated in [55]–[58]. The comparative analysis indicates that the proposed encryption schemes offer better encryption and security than S-box based encryption schemes as MLC individual metrics such as entropy, correlation coefficient, contrast, energy, and homogeneity have improved scores than schemes in [55]–[58].

### F. NPCR AND UACI

A robust cryptosystem should be highly sensitive to any minor alterations in in secret key or plain-image data. To measure the sensitivity of the system, we resort to net pixel change rate (NPCR) and unified average change intensity (UACI) tests. These two tests facilitate to perform sensitivity

**TABLE 7.** NPCR and UACI results from proposed encryption scheme and their comparison.

| Method | Cameraman | | Baboon | | Peppers | |
|---|---|---|---|---|---|---|
| | NPCR | UACI | NPCR | UACI | NPCR | UACI |
| Proposed (S1) | 99.60 | 33.61 | 99.54 | 33.34 | 99.63 | 33.54 |
| Proposed (S2) | 99.62 | 33.45 | 99.57 | 33.36 | 99.62 | 33.34 |
| Proposed (S3) | 99.60 | 33.55 | 99.62 | 33.56 | 99.65 | 33.59 |
| Ref. [55] | 99.64 | 33.56 | 99.59 | 33.48 | - | - |
| Ref. [56] | 99.56 | 34.03 | 99.56 | 33.03 | - | - |
| Ref. [57] | 99.62 | 33.7 | 99.62 | 33.7 | - | - |
| Ref. [58] | 99.61 | 33.48 | 99.61 | 33.49 | - | - |

analysis of the anticipated encryption scheme. It gives the pixel change rate between two encrypted images $C_1$ and $C_2$ which have minor difference in the input parameters like key or plain-image. Mathematically, they are calculated as:

$$NPCR = \frac{\sum_{i,j} D(i,j)}{M \times N} \times 100$$

$$D(i,j) = \begin{cases} 1 & if\ C_1(i,j) \neq C_2(i,j) \\ 0 & if\ C_1(i,j) = C_2(i,j) \end{cases}$$

$$UACI = \frac{1}{M \times N}\left[\sum \frac{|C_1(i,j) - C_2(i,j)|}{2^8 - 1}\right] \times 100$$

The NPCR score close to 99.6% and UACI close to 33.6% represent the optimal sensitivity performance of an image encryption scheme [59]. The anticipated results obtained to assess the sensitivity of proposed encryption scheme are shown in Tables 7. We can see that the NPCR and UACI scores are quite close to the optimal values which reflect the excellent sensitivity of proposed S-boxes based encryption scheme against any minor alterations. The NPCR and UACI performance of our encryption scheme is acceptable and comparable to the recently investigated S-box based encryption algorithms [55]–[58].

### G. SPEED ANALYSIS

The time taken by an encryption scheme to encrypt an image is one of the crucial metric for practical application of scheme for real-time application scenarios. An image encryption scheme is deemed to be better if takes less time and still holds strong encryption quality. To do the speed analysis of our encryption scheme, the MATLAB is used for implementation and simulation on Windows 8 having 4GB RAM and Intel core i7 CPU which operates at 2.2 GHz. The encryption time of our scheme for images of size 256 × 256, 512 × 512, and 1024 × 1024 comes out to be 0.2634s, 1.10136, and 3.9449s, respectively. The encryption speed performance is compared with other image encryption schemes in Table 8. The proposed encryption scheme offers great encryption quality and encryption time is considerably shorter than many of its contemporary schemes investigated in [56], [60]–[65]. Thus, the proposed S-boxes based encryption scheme is faster compared to many recent competitor schemes which make it suitable for real-time image encryption applications.

**TABLE 8.** Speed analysis of some encryption schemes (time in *secs*).

| | | | |
|---|---|---|---|
| Proposed | CPU Core i7 2.2 GHz, 4GB RAM on Windows 8 | 0.2634 | 1.10136 |
| Ref. [56] | CPU Core i3 1.9 GHz, 4GB RAM on Windows 7 | 3.0658 | - |
| Ref. [60] | CPU Core i3 2.53 GHz, 3GB RAM on Windows 7 | 1.1204 | - |
| Ref. [61] | CPU Core i3 2.4 GHz, 4GB RAM on Windows 8 | 0.631 | - |
| Ref. [62] | CPU Core i3, 4GB RAM on Windows 7 | - | 22.43 |
| Ref.[63] | CPU 3.3 GHz, 4GB RAM on Windows 7 | 0.382 | 1.489 |
| Ref. [64] | CPU Core i7 3.8 GHz, 16GB RAM on Linux | 1.7 | - |
| Ref. [65] | CPU Core i7 2.7 GHz, 4GB RAM on Windows 7 | 7.32 | - |

## VII. CONCLUSION

This paper suggested an efficient S-box method based on optimization technique, as an alternative to random, chaos and algebraic based methods, is proposed. Particle swarm optimization is approached to evolve the S-boxes for high nonlinearity as fitness value. The proposed method makes use of chaotic Renyi map for the generation of initial population and other required random values. The method has been analyzed for different scenarios by varying the parameters of PSO. Performance assessment indicated the proposed method is capable to generate strong S-boxes with good cryptographic features. When compared with many recent optimizations based S-boxes, it is found that the proposed S-boxes are sufficiently upright than many of its competitors. Thus, the proposed method is proficient in generating highly nonlinear S-boxes. Moreover, it has been also demonstrated that the proposed S-boxes are suitable and appropriate for usage in security applications to realize secure image based communication.

## ACKNOWLEDGMENT

The authors would like to thank the Deanship of Scientific Research at Umm Al-Qura University for supporting this work by grant code 19-COM-1-01-0015.

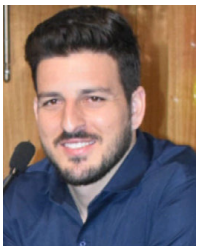
## REFERENCES

[1] B. Schneier, *Applied Cryptography: Protocols, Algorithms, and Source Code in C*. New York, NY, USA: Wiley, 1996.

[2] A. Rafiq and M. Khan, "Construction of new S-boxes based on triangle groups and its applications in copyright protection," *Multimedia Tools Appl.*, vol. 78, no. 11, pp. 15527–15544, Jun. 2019.
[3] L. R. Knudsen and M. J. B. Robshaw, *The Block Cipher Companion*. Berlin, Germany: Springer-Verlag, 2011.
[4] A. Razaq, H. Alolaiyan, M. Ahmad, M. A. Yousaf, U. Shuaib, W. Aslam, and M. Alawida, "A novel method for generation of strong substitution-boxes based on coset graphs and symmetric groups," *IEEE Access*, vol. 8, pp. 75473–75490, 2020.
[5] Y. Wang, P. Lei, and K.-W. Wong, "A method for constructing bijective S-box with high nonlinearity based on chaos and optimization," *Int. J. Bifurcation Chaos*, vol. 25, no. 10, Sep. 2015, Art. no. 1550127.
[6] A. M. Youssef and S. E. Tavares, "Resistance of balanced S-boxes to linear and differential cryptanalysis," *Inf. Process. Lett.*, vol. 56, no. 5, pp. 249–252, Dec. 1995.
[7] M. Khan, T. Shah, H. Mahmood, and M. A. Gondal, "An efficient method for the construction of block cipher with multi-chaotic systems," *Nonlinear Dyn.*, vol. 71, no. 3, pp. 489–492, Feb. 2013.
[8] M. Ahmad, F. Ahmad, Z. Nasim, Z. Bano, and S. Zafar, "Designing chaos based strong substitution box," in *Proc. 8th Int. Conf. Contemp. Comput. (IC)*, Aug. 2015, pp. 97–100.
[9] F. Özkaynak, V. Çelik, and A. B. Özer, "A new S-box construction method based on the fractional-order chaotic chen system," *Signal, Image Video Process.*, vol. 11, no. 4, pp. 659–664, May 2017.
[10] X. Wang, A. Akgul, U. Cavusoglu, V.-T. Pham, D. Vo Hoang, and X. Nguyen, "A chaotic system with infinite equilibria and its S-box constructing application," *Appl. Sci.*, vol. 8, no. 11, p. 2132, Nov. 2018.
[11] L. Liu, Y. Zhang, and X. Wang, "A novel method for constructing the S-box based on spatiotemporal chaotic dynamics," *Appl. Sci.*, vol. 8, no. 12, p. 2650, Dec. 2018.
[12] D. Lambić, "A novel method of S-box design based on chaotic map and composition method," *Chaos, Solitons Fractals*, vol. 58, pp. 16–21, Jan. 2014.
[13] M. Asim and V. Jeoti, "Efficient and simple method for designing chaotic S-boxes," *ETRI J.*, vol. 30, no. 1, pp. 170–172, Feb. 2008.
[14] G. Chen, "A novel heuristic method for obtaining S-boxes," *Chaos, Solitons Fractals*, vol. 36, no. 4, pp. 1028–1036, May 2008.
[15] Y. Tian and Z. Lu, "Chaotic S-box: Intertwining logistic map and bacterial foraging optimization," *Math. Problems Eng.*, vol. 2017, pp. 1–11, Nov. 2017.
[16] I. Shatheesh Sam, P. Devaraj, and R. S. Bhuvaneswaran, "An intertwining chaotic maps based image encryption scheme," *Nonlinear Dyn.*, vol. 69, no. 4, pp. 1995–2007, Sep. 2012.
[17] K. M. Passino, "Biomimicry of bacterial foraging for distributed optimization and control," *IEEE Control Syst. Mag.*, vol. 22, no. 3, pp. 52–67, Mar. 2002.
[18] G. Zaibi, A. Kachouri, F. Peyrard, and D. Fournier-Prunaret, "On dynamic chaotic S-box," in *Proc. Global Inf. Infrastruct. Symp.*, Jun. 2009, pp. 1–5.
[19] M. Ahmad, H. Chugh, A. Goel, and P. Singla, "A chaos based method for efficient cryptographic S-box design," in *Proc. Int. Symp. Secur. Comput. Commun.* Berlin, Germany: Springer, 2013, pp. 130–137.
[20] A. Belazi and A. A. A. El-Latif, "A simple yet efficient S-box method based on chaotic sine map," *Optik*, vol. 130, pp. 1438–1444, Feb. 2017.
[21] E. Tanyildizi and F. Ozkaynak, "A new chaotic S-box generation method using parameter optimization of one dimensional chaotic maps," *IEEE Access*, vol. 7, pp. 117829–117838, 2019.
[22] F. Özkaynak, "On the effect of chaotic system in performance characteristics of chaos based S-box designs," *Phys. A, Stat. Mech. Appl.*, vol. 550, Jul. 2020, Art. no. 124072.
[23] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Proc. 6th Int. Symp. Micro Mach. Hum. Sci.* Piscataway, NJ, USA: IEEE Service Center, 1995, pp. 39–43.
[24] Eberhart and Y. Shi, "Particle swarm optimization: Developments, applications and resources," in *Proc. Congr. Evol. Comput.*, vol. 1, May 2001, pp. 81–86.
[25] R. Poli, J. Kennedy, and T. Blackwell, "Particle swarm optimization," *Swarm Intell.*, vol. 1, no. 1, pp. 33–57, Jun. 2007.
[26] A. F. Webster and S. E. Tavares, *On the Design of S-boxes*. Berlin, Germany: Springer, 1986.
[27] M. Ahmad, N. Mittal, P. Garg, and M. Maftab Khan, "Efficient cryptographic substitution box design using travelling salesman problem and chaos," *Perspect. Sci.*, vol. 8, pp. 465–468, Sep. 2016.
[28] T. W. Cusick and P. Stanica, *Cryptographic Boolean Functions and Applications*. Amsterdam, The Netherlands: Elsevier, 2009.
[29] M. A. Yousaf, H. Alolaiyan, M. Ahmad, M. Dilbar, and A. Razaq, "Comparison of pre and post-action of a finite abelian group over certain nonlinear schemes," *IEEE Access*, vol. 8, pp. 39781–39792, 2020.
[30] C. Adams and S. Tavares, "The structured design of cryptographically good S-boxes," *J. Cryptol.*, vol. 3, no. 1, pp. 27–41, Jan. 1990.
[31] E. Biham and A. Shamir, "Differential cryptanalysis of Des. Like cryptosystems," *J. Cryptol.*, vol. 4, no. 1, pp. 3–72, Jan. 1991.
[32] E. Al Solami, M. Ahmad, C. Volos, M. Doja, and M. Beg, "A new hyperchaotic system-based design for efficient bijective substitution-boxes," *Entropy*, vol. 20, no. 7, p. 525, Jul. 2018.
[33] M. Matsui, "Linear cryptanalysis method for DES cipher," in *Proc. Adv. Cryptol.*, 1994, pp. 386–397.
[34] B. Alatas, E. Akin, and A. B. Ozer, 'Chaos embedded particle swarm optimization algorithms," *Chaos, Solitons Fractals*, vol. 40, pp. 1715–1734, May 2009, doi: 10.1016/j.chaos.2007.09.063.
[35] J. Chuanwen and E. Bompard, "A self-adaptive chaotic particle swarm algorithm for short term hydroelectric system scheduling in deregulated environment," *Energy Convers. Manage.*, vol. 46, no. 17, pp. 2689–2696, Oct. 2005.
[36] L. D. S. Coelho and V. C. Mariani, "A novel chaotic particle swarm optimization approach using Hénon map and implicit filtering local search for economic load dispatch," *Chaos, Solitons Fractals*, vol. 39, no. 2, pp. 510–518, Jan. 2009.
[37] T. Addabbo, M. Alioto, A. Fort, A. Pasini, S. Rocchi, and V. Vignoli, "A class of maximum-period nonlinear congruential generators derived from the Rényi chaotic map," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 54, no. 4, pp. 816–828, Apr. 2007.
[38] M. Ibnkahla, *Signal Processing for Mobile Communications Handbook*. Boca Raton, FL, USA: CRC Press, 2005, ch. 27.
[39] Q. Bai, "Analysis of particle swarm optimization algorithm," *Comput. Inf. Sci.*, vol. 3, no. 1, p. 180, Jan. 2010.
[40] P. Umapathy, C. Venkataseshaiah, and M. S. Arumugam, "Particle swarm optimization with various inertia weight variants for optimal power flow solution," *Discrete Dyn. Nature Soc.*, vol. 2010, pp. 1–15, Aug. 2010.
[41] J. Daemen and V. Rijmen, *The Design of RIJNDAEL: AES-The Advanced Encryption Standard*. Berlin, Germany: Springer-Verlag, 2002.
[42] 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.*, vol. 99, no. 4, pp. 3041–3064, Mar. 2020.
[43] T. Akhtar, N. Din, and J. Uddin, "Substitution box design based on chaotic maps and cuckoo search algorithm," in *Proc. Int. Conf. Adv. Commun. Technol. Netw. (CommNet)*, Apr. 2019, pp. 1–7.
[44] 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. A*, vol. 376, nos. 6–7, pp. 827–833, Jan. 2012.
[45] A. A. Alzaidi, M. Ahmad, H. S. Ahmed, and E. A. Solami, "Sine-cosine optimization-based bijective substitution-boxes construction using enhanced dynamics of chaotic map," *Complexity*, vol. 2018, pp. 1–16, Dec. 2018.
[46] A. A. Alzaidi, M. Ahmad, M. N. Doja, E. A. Solami, and M. M. S. Beg, "A new 1D chaotic map and $\beta$ -Hill climbing for generating substitution-boxes," *IEEE Access*, vol. 6, pp. 55405–55418, 2018.
[47] H. A. Ahmed, M. F. Zolkipli, and M. Ahmad, "A novel efficient substitution-box design based on firefly algorithm and discrete chaotic map," *Neural Comput. Appl.*, vol. 31, no. 11, pp. 7201–7210, Nov. 2019.
[48] M. Ahmad, M. N. Doja, and M. M. S. Beg, "ABC optimization based construction of strong substitution-boxes," *Wireless Pers. Commun.*, vol. 101, no. 3, pp. 1715–1729, Aug. 2018.
[49] 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.
[50] T. Zhang, C. L. P. Chen, L. Chen, X. Xu, and B. Hu, "Design of highly nonlinear substitution boxes based on I-ching operators," *IEEE Trans. Cybern.*, vol. 48, no. 12, pp. 3349–3358, Dec. 2018.
[51] 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, Jan. 2015.
[52] Y. Tian and Z. Lu, "S-box: Six-dimensional compound hyperchaotic map and artificial bee colony algorithm," *J. Syst. Eng. Electron.*, vol. 27, no. 1, pp. 232–241, Feb. 2016.
[53] I. Hussain, T. Shah, M. A. Gondal, and H. Mahmood, "Generalized majority logic criterion to analyze the statistical strength of S-boxes," *Zeitschrift für Naturforschung A*, vol. 67, no. 5, pp. 282–288, May 2012.

[54] S. S. Jamal, A. Anees, M. Ahmad, M. F. Khan, and I. Hussain, "Construction of cryptographic S-boxes based on mobius transformation and chaotic tent-sine system," *IEEE Access*, vol. 7, pp. 173273–173285, 2019.
[55] A. Ullah, A. Javeed, and T. Shah, "A scheme based on algebraic and chaotic structures for the construction of substitution box," *Multimedia Tools Appl.*, vol. 78, no. 22, pp. 32467–32484, Nov. 2019.
[56] A. Belazi, M. Khan, A. A. A. El-Latif, and S. Belghith, "Efficient cryptosystem approaches: S-boxes and permutation-substitution-based encryption," *Nonlinear Dyn.*, vol. 87, no. 1, pp. 337–361, Jan. 2017.
[57] A. Ullah, S. S. Jamal, and T. Shah, "A novel construction of substitution box using a combination of chaotic maps with improved chaotic range," *Nonlinear Dyn.*, vol. 88, no. 4, pp. 2757–2769, Jun. 2017.
[58] M. Khan, T. Shah, and S. I. Batool, "Construction of S-box based on chaotic Boolean functions and its application in image encryption," *Neural Comput. Appl.*, vol. 27, no. 3, pp. 677–685, Apr. 2016.
[59] X. Wang, C. Liu, and H. Zhang, "An effective and fast image encryption algorithm based on chaos and interweaving of ranks," *Nonlinear Dyn.*, vol. 84, no. 3, pp. 1595–1607, May 2016.
[60] M. A. B. Farah, R. Guesmi, A. Kachouri, and M. Samet, "A new design of cryptosystem based on S-box and chaotic permutation," *Multimedia Tools Appl.*, Mar. 2020.
[61] T. S. Ali and R. Ali, "A novel medical image signcryption scheme using TLTS and henon chaotic map," *IEEE Access*, vol. 8, pp. 71974–71992, 2020.
[62] M. Samiullah, W. Aslam, H. Nazir, M. I. Lali, B. Shahzad, M. R. Mufti, and H. Afzal, "An image encryption scheme based on DNA computing and multiple chaotic systems," *IEEE Access*, vol. 8, pp. 25650–25663, 2020.
[63] Q. Lu, C. Zhu, and X. Deng, "An efficient image encryption scheme based on the LSS chaotic map and single S-box," *IEEE Access*, vol. 8, pp. 25664–25678, 2020.
[64] M. Asgari-Chenaghlu, M.-A. Balafar, and M.-R. Feizi-Derakhshi, "A novel image encryption algorithm based on polynomial combination of chaotic maps and dynamic function generation," *Signal Process.*, vol. 157, pp. 1–13, Apr. 2019.
[65] A. M. Ayoup, A. H. Hussein, and M. A. A. Attia, "Efficient selective image encryption," *Multimedia Tools Appl.*, vol. 75, no. 24, pp. 17171–17186, Dec. 2016.

**MUSHEER AHMAD** received the B.Tech. and M.Tech. degrees from the Department of Computer Engineering, Aligarh Muslim University, India, in 2004 and 2008, respectively. He has been an Assistant Professor with the Department of Computer Engineering, Jamia Millia Islamia, New Delhi, since 2011. He has published over 70 research papers in international reputed refereed journals and conference proceedings. His research interests include, but not limited to, multimedia security, chaos-based cryptography, cryptanalysis, image processing, and optimization techniques.



**ISHFAQ AHMAD KHAJA** received the B.Tech. degree from the University of Kashmir, Srinagar, India, in 2017. He is currently pursuing the M.Tech. degree with the Department of Computer Engineering, Jamia Millia Islamia, New Delhi, India. His research interests include machine learning, cryptography, optimization techniques, image processing, quantum computing, and so on.

**ABDULLAH BAZ** (Senior Member, IEEE) received the B.Sc. degree in electrical and computer engineering from UQU, in 2002, the M.Sc. degree in electrical and computer engineering from KAU, in 2007, and the second M.Sc. degree in communication and signal processing and the Ph.D. degree in computer system design from Newcastle University, in 2009 and 2014, respectively. He was the Vice-Dean and the Dean of the Deanship of Scientific Research, UQU, from 2014 to 2020. He is currently an Assistant Professor with the Computer Engineering Department, the Vice-Dean of DFMEA, the General Director of the Decision Support Center, and the Consultant of the University Vice Chancellor with UQU. His research interests include data science, ML, AI, VLSI design, EDA/CAD tools, intelligent transportation, computer system and architecture, smart systems, smart health. Since 2015, he has been serving as a Review Committee Member of the IEEE International Symposium on Circuits and Systems (ISCAS) and a member of the Technical Committee of the IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS AND APPLICATIONS. He has served as a Reviewer in a number of journals, including the IEEE INTERNET OF THINGS, *IET Computer Vision*, *Artificial Intelligence Review*, and *IET Circuits, Devices & Systems*.

**HOSAM ALHAKAMI** received the B.Sc. degree in computer science from King Abdulaziz University, Saudi Arabia, in 2004, the M.Sc. degree in Internet software systems from the University of Birmingham, Birmingham, U.K., in 2009, and the Ph.D. degree in software engineering from De Montfort University, in 2015. From 2004 to 2007, he worked at Software Development Industry, where he implemented several systems and solutions for a national academic institution. He was the Vice-Dean of the Deanship of Admission and Registration for Academic affairs with UQU, from 2015 to 2020. Currently, he is an Associate Professor of the Computer Science Department with UQU. His research interests include algorithms, semantic web, and optimization techniques. He focuses on enhancing real-world matching systems using machine learning and data analytics in a context of supporting decision-making.

**WAJDI ALHAKAMI** received the B.Sc. degree in computer science from Jeddah University, Saudi Arabia, and the M.Sc. degree in computer network and the Ph.D. degree in network security from the University of Bedfordshire, U.K. He is currently working as an Assistant Professor at the Department of Computer Science, Taif University.