Hindawi
Complexity
Volume 2018, Article ID 9389065, 16 pages
https://doi.org/10.1155/2018/9389065

Research Article

Sine-Cosine Optimization-Based Bijective Substitution-Boxes Construction Using Enhanced Dynamics of Chaotic Map

Amer Awad Alzaidi,[1] Musheer Ahmad ,[2] Hussam S. Ahmed,[3] and Eesa Al Solami[4]

[1] *Department of Information Systems, University of Jeddah, Jeddah 21589, Saudi Arabia*
[2] *Department of Computer Engineering, Jamia Millia Islamia, New Delhi 110025, India*
[3] *Faculty of Computer Systems and Software Engineering, Universiti Malaysia Pahang, Malaysia*
[4] *Department of Information Technology, University of Jeddah, Jeddah 21589, Saudi Arabia*

Correspondence should be addressed to Musheer Ahmad; musheer.cse@gmail.com

Received 6 September 2018; Accepted 13 November 2018; Published 2 December 2018

Guest Editor: Iqtadar Hussain

This paper proposes a novel method of constructing strong substitution-boxes (S-boxes) of order n ($4 \leq n \leq 8$) based on a recent optimization algorithm known as sine-cosine algorithm (SCA). The paper also proposes a new 1D chaotic map, which owns enhanced dynamics compared to conventional chaotic map, for generating initial population of S-boxes and facilitating the optimization mechanism of SCA. The proposed method applies the SCA with enhanced chaotic map to explore and exploit the search space for obtaining optimized S-boxes on the basis of maximization of nonlinearity as fitness function. The S-box construction involves three phases such as initialization of population, optimization, and adjustment. The simulation and performance analyses are done using standard measures of nonlinearity, strict avalanche criterion, bits independence criterion, differential uniformity, linear approximation probability, and autocorrelation function. The obtained experimental results are compared with some immediate optimization-based and other S-boxes to show the strength of proposed method for constructing bijective S-boxes of salient cryptographic features.

1. Introduction

Due to recent technological advancements, the transfer of sensitive data ranging from text to images and video has made it exigent to employ effective cryptographic systems for their security. In the sphere of cryptographic systems, the properties of confusion and diffusion are the two significant aspects for a strong secure cipher [1]. The property of confusion enables cipher to make the connection between the encryption key and the encrypted message as complicated as could be, while diffusion aims to diminish the reliance between the plain message and the comparing encrypted message as much as possible. The process of substitution (a plain message supplanted by another) has been distinguished as an efficient way for fundamental confusion property. On the other hand, transposition (disturbing securely the order of message symbols) is a procedure meant for diffusion [2]. Further, the product cipher utilizes substitution and transposition processes alternatively to accomplish both confusion and diffusion for strong security of the system. The property of confusion is achieved through nonlinear vectorial Boolean functions termed as substitution-box (S-box). An S-box is an integral component of the cryptographic systems which performs substitution.

Fundamentally, a bijective $n \times n$ substitution-box is a mathematical one-to-one mapping S from $\{0,1\}^n$ to $\{0,1\}^n$, where n is its order denoting the inputs and outputs sizes. It is also regarding multi-input and multioutput Boolean function as $S(x) = [f_{n\text{-}1}(x) f_{n\text{-}2}(x) \ldots f_1(x) f_0(x)]$, where f_i ($0 \leq i \leq n$-1) is component n-variable function defined as f_i: $\{0,1\}^n \longrightarrow \{0,1\}$ [3]. In bijective S-box of order n, the number of preimages of each output is only one and can be represented through permutation of $\{0, 1, 2, \ldots, 2^n\text{-}1\}$. S-boxes are essential components of symmetric ciphers which are inherently used in block ciphers based on substitution-permutation networks (SPN) and Feistel architecture [4, 5]. The security of these cryptosystems majorly relies on the potency of nonlinear mapping of S-boxes transforming plaintext to ciphertext. Eventually, the power of symmetric cryptosystems heavily depends on the cryptographic strengths of substitution-boxes employed, which in turn weigh on their design methodology. Hence, designing strong S-boxes is really essential for good symmetric cryptosystems. The design methods and criteria required to engineer S-boxes include characteristics of simplicity, primarily large nonlinearity, and good balance of other performance criteria. The methods based on algebraic techniques are studied as they show strong properties such as high nonlinearity and resiliency to differential/linear cryptanalyses [6, 7]. But, recent algebraic cryptanalysis has uncovered that they are imperfect to algebraic attacks [8, 9]. Therefore, instead of targeting new S-boxes using formal algebraic approaches, we need to explore some alternative S-boxes design methods. As an option, a number of alternative methods are being investigated in recent past to yield S-boxes using various optimization techniques [10–25] and chaotic systems [26–32]. Theoretically, a strong S-box possesses the features like balancedness, high nonlinearity, small differential uniformity and linear probability, strict avalanche effect close to 0.5, output bits independence, low autocorrelation, etc. [27, 33].

1.1. Related Work. There have been a number of proposals investigated in literature which are based on metaheuristic techniques for evolving strong S-boxes. The reviews of recent optimization-based 8×8 S-box proposals are as follows: Wang et al. and Guesmi et al. in [10–12] explored the features genetic algorithm (GA) for evolving 8×8 S-box. The chaotic logistic map and tent map are adopted in [10, 11] for initialization of starting populations and parameters of GA. Wang et al. did an improvisation in the adjustment phase of same approach to construct more potent S-box in [11]. However, Guesmi et al. made use of logistic map for initial S-box design and a differential chaotic Lorenz system for performing the operations of crossover and mutation during GA optimization process in [12]. Ahmad et al. explored the metaheuristics Ant Colony Optimization (ACO) to optimize single initial S-box in [13]. Here, a combination of chaotic tent map and logistic map is put on to get initial S-box. The optimized S-box was exacted to have respectable cryptographic features compared to some S-boxes. In [14], the Artificial Bee Colony (ABC) algorithm and hyperchaotic map are sought to yield effective 8×8 S-box, wherein a 6D hyperchaos was applied for initial population of S-boxes. The same author engaged Bacteria Foraging Optimization (BFO) with intertwining logistic map with similar approach for S-box optimization in [15]. In [16], the well-known traveling salesman problem is attempted for generating a strong S-box. The weights of edges of subgraphs extracted from initial S-box are randomly decided by skew tent map. Farah et al. presented a chaotic map and Teaching-Learning Based Optimization (TLBO) based method for efficient S-box design in [17]. The TLBO is applied to find the optimized keys that ensued into an S-box which satisfied the given conditions as fitness function. Hussam et al. practiced the firefly algorithm (FA) for optimizing an initial S-box from a discrete-space chaotic map in [18]. Zhang et al. implemented the I-Ching Operators (ICO) evolved from Chinese I-Ching philosophy inventively for getting optimized 8×8 S-boxes in [19]. Recently, Alzaidi et al. investigated a β-hill climbing technique which is an individual-based optimization algorithm for generating an 8×8 S-box in [20], where the initial individual candidate S-box and parameters of improved hill climbing technique are rendered using a new discrete-chaotic map. Moreover, Ye et al. operated an O-shaped path shuffling mechanism to obtain a strong configuration S-box in [26], where the starting S-box is derived from chaotic sequences of 6D fractional Lorenz-Duffing system.

The optimization-based proposals which reported generic method to construct bijective S-boxes of order n for $5 \leq n \leq 8$ are reviewed as follows: Millan gave a design strategy based on hill climbing (HC) approach for evolving highly nonlinear S-boxes in 1998 [21]. In [22], a tweaking based heuristic technique was presented to generate optimized power mapping-based S-boxes by mutation operations. In [23], both the Particle Swarm Optimization (PSO) and Differential Evolution (DE) techniques are applied to construct many n×n S-boxes. Tesar suggested a particular genetic algorithm and total tree search to optimize smaller S-boxes in [24]. Picek et al. investigated an improvised cost function for getting better S-boxes using GA, GA with the feature of tree search (GaT), and Local Search Algorithm (LSA) in [25]. Recently, Solami et al. in [27] executed a random Heuristic Search (HS) approach using hyperchaotic system for synthesis bijective S-boxes.

1.2. Our Contribution. We explore a recent optimization mechanism of sine-cosine algorithm which is based on mathematical sine and cosine functions for synthesis of strong bijective S-boxes. The initialization of input parameters of proposed method is executed through 1D chaotic map. The improved chaotic map has enhanced dynamics compared to some widely used 1D chaotic maps. The contributions of our paper are as follows:

(i) A generic method capable of constructing strong bijective S-boxes of order n ($4 \leq n \leq 8$) is proposed using sine-cosine algorithm (SCA) and chaos. The maximization of nonlinearity of S-box is considered as the fitness function during optimization phase.

(ii) An improved discrete 1D chaotic map is proposed which is found to have considerably better chaotic dynamics and performance with respect to Lyapunov exponent, bifurcation, uniform distribution, chaotic range, approximate entropy, and randomness.

(iii) The parameters of SCA and initial population of bijective S-boxes are generated chaotically though our improved chaotic map.

(iv) We performed experiments to know the effectiveness of proposed generic method for constructing strong S-boxes using standard performance criteria.

(v) We compared the obtained results with most of the optimization-based S-box design works and other recent S-boxes works to justify the superlative performance of our method over most of the existing ones.

The remaining portion of this paper is developed as follows. The mathematical model and dynamics of new chaotic map are discussed in Section 2. A brief review and working of sine-cosine algorithm are introduced in Section 3. The proposed method of constructing bijective S-boxes based on SCA and new chaotic map is provided in Section 4. Section 5 is devoted to experimentation, performance evaluation, and comparison with recent S-box proposals. Section 6 provides the conclusion of works done in paper.

2. Proposed Chaotic Map and Its Dynamics

In this section, the enhanced dynamics and improved chaotic performance of proposed 1D chaotic map are discussed. It is quite known that most of the 1D chaotic maps are found to have either of the following limitations as far as their chaotic phenomenon is concerned.

(i) They have limited chaotic range for control parameter.

(ii) They have only one control parameter.

(iii) They have low value of largest Lyapunov exponent.

(iv) They exhibit nonuniform distribution in their own bounded region.

(v) Their return map shows nonuniform coverage of attractor in phase space.

The 1D chaotic maps which suffer from one or more of these demerits are the well-known logistic map, sine map, Chebyshev map, cubic map, tent map, etc. [36–38]. Therefore, a considerable research is devoted to designing improved chaotic maps which do not exhibit the above-mentioned problems [38–41]. In this paper, we modify the chaotic sine map to present a new chaotic map which is found to have enhanced dynamics and chaotic performance compared to conventional chaotic sine map. The chaotic sine map is given as

$$x_{n+1} = a \times \sin\left(\pi \times x_n\right) \quad (1)$$

Here, $a \in (0,1]$ is its control parameter and $x_n \in [0,1]$ is the chaotic variable for all $n \geq 0$. The diagrams representing Lyapunov exponent, bifurcation, and attractors plot are shown in Figure 1 for $x_0 = 0.1234567$, a =1. The proposed chaotic map governs the following equation.

$$x_{n+1} = \left(c^2 \times x_n + \left|\sin\left(\pi\left(x_n^3 - r \times x_n\right)\right)\right|\right) \mod (1) \quad (2)$$

Here, c and r are two control parameters of map (2); x_n is its chaotic sequence variable which is bounded within $(0,1]$. We analyze the dynamics of our proposed 1D chaotic map (2) using Lyapunov exponent, bifurcation, chaotic attractor' phase space, approximate entropy, and randomness behaviors as follows.

Table 1: Comparison of Lyapunov exponents for two chaotic maps.

Chaotic map	Parameter	Largest LE
Chaotic Sine map	$a = 1$	0.6889
Proposed Chaotic map	$c = 10, r \in [0,10]$	4.6523
	$r = 10, c \in [0,10]$	4.5955
	$r = 10, c \in [0,100]$	9.2107
	$r = 10, c \in [0,10^6]$	27.631
	$r = 10, c \in [0,10^8]$	36.733

2.1. Lyapunov Exponent. The Lyapunov exponent (LE) of a dynamical map characterises the velocity of separation between two close trajectories. It is computed using the following mathematical formula.

$$LE = \lim_{n\longrightarrow\infty}\left[\frac{1}{n}\sum_{i=1}^{n}\log_2\left|\frac{dx_{i+1}}{dx_i}\right|\right] \quad (3)$$

It indicates the degree of chaotic behavior of a function and a positive value indicates higher orbital divergence and existence of chaos. It means that function shows high dependence on initial condition, which is one of the characteristics of a chaotic map. A higher value of positive LE indicates more complicated dynamics showing stronger dependence on initial conditions and better divergence in phase space and [42]. By default, $x_0 = 0.1234567$, c=10, r=10 are taken as initial values for new map (2). The Lyapunov exponent diagrams of proposed map versus the two control parameters c and r are shown in Figures 2(a)–2(c). The diagram shows that map (2) exhibits chaotic phenomenon as the LEs are satisfactorily positive in all three diagrams except for the value of c near 5 as evident from Figure 2(b). Therefore, a value of c ≥ 6 is recommended for obtaining good chaotic dynamics of our map. We further analyze the LE for higher values of r and c, and it has been noted that LE is about 4.6 for any $r > 0$ and increases with increase in $c \geq 6$. Another LE diagram is provided in Figure 2(c) for $c \in [6,100]$ to demonstrate the exponent's increasing behavior based on parameter c. The LE values for two maps are compared in Table 1. The maximum values of largest LE proposed chaotic map for different parameter values are significantly higher than maximum LE of chaotic sine map. The LEs further show that the proposed chaotic map has got sufficiently extended chaotic range for wider range of control parameters. Hence, the proposed chaotic map provides better chaotic behavior and sensitivity to initial conditions.

2.2. Bifurcation. In chaos theory, the bifurcation diagram of a dynamical system shows the way its values approached asymptotically as a function of bifurcation parameters. It represents the qualitative changes of the system as a map of parameters. The bifurcation behavior of proposed map is analyzed and representative diagrams are shown in Figures 2(d)–2(f) for two parameters r and c. The diagrams show that output values of map have uniform distribution over the complete range of $(0,1]$ and it has been found that such uniformity persists even for larger values of control

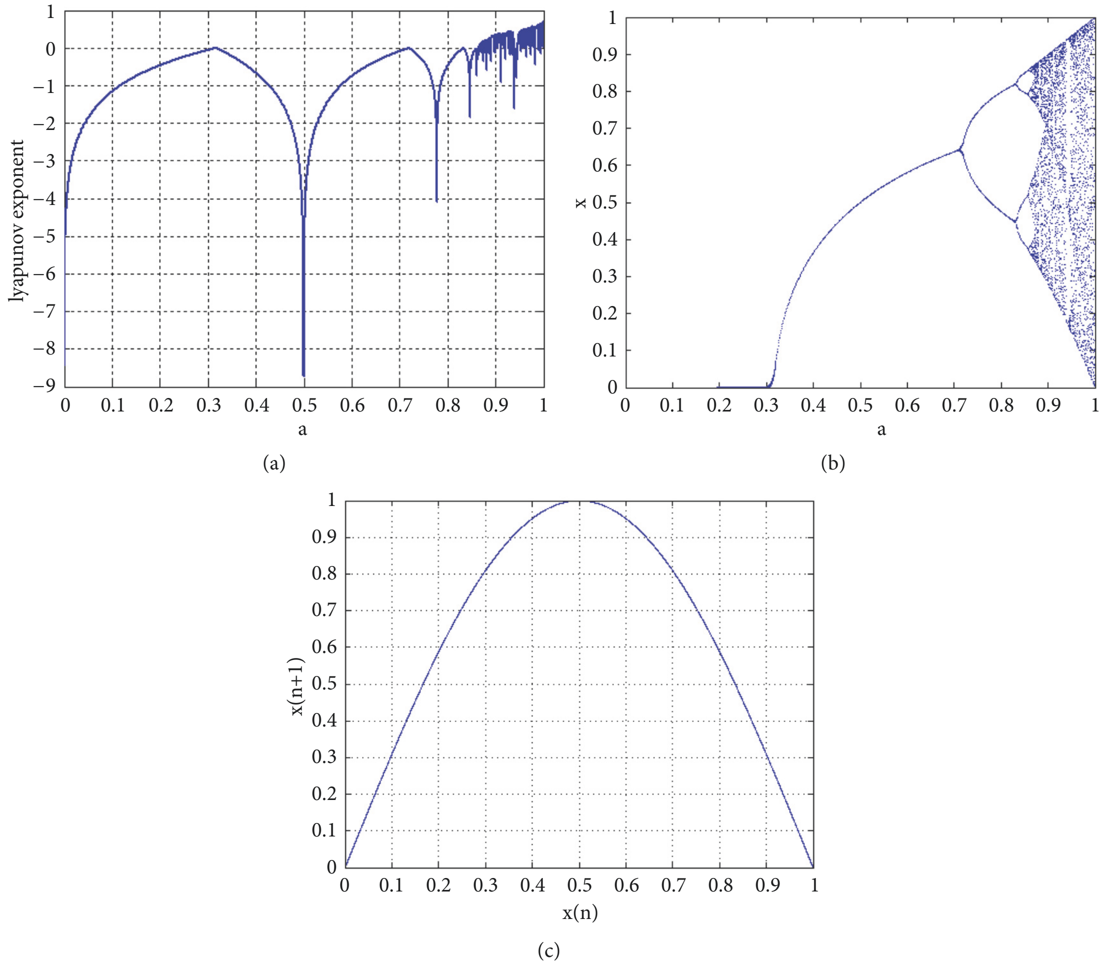


Figure 1: Dynamics of chaotic sine map: (a) Lyapunov exponent, (b) bifurcation, and (c) attractor's phase space for a=1.

parameters. Hence, the proposed map offers good bifurcation behavior for both the control parameters.

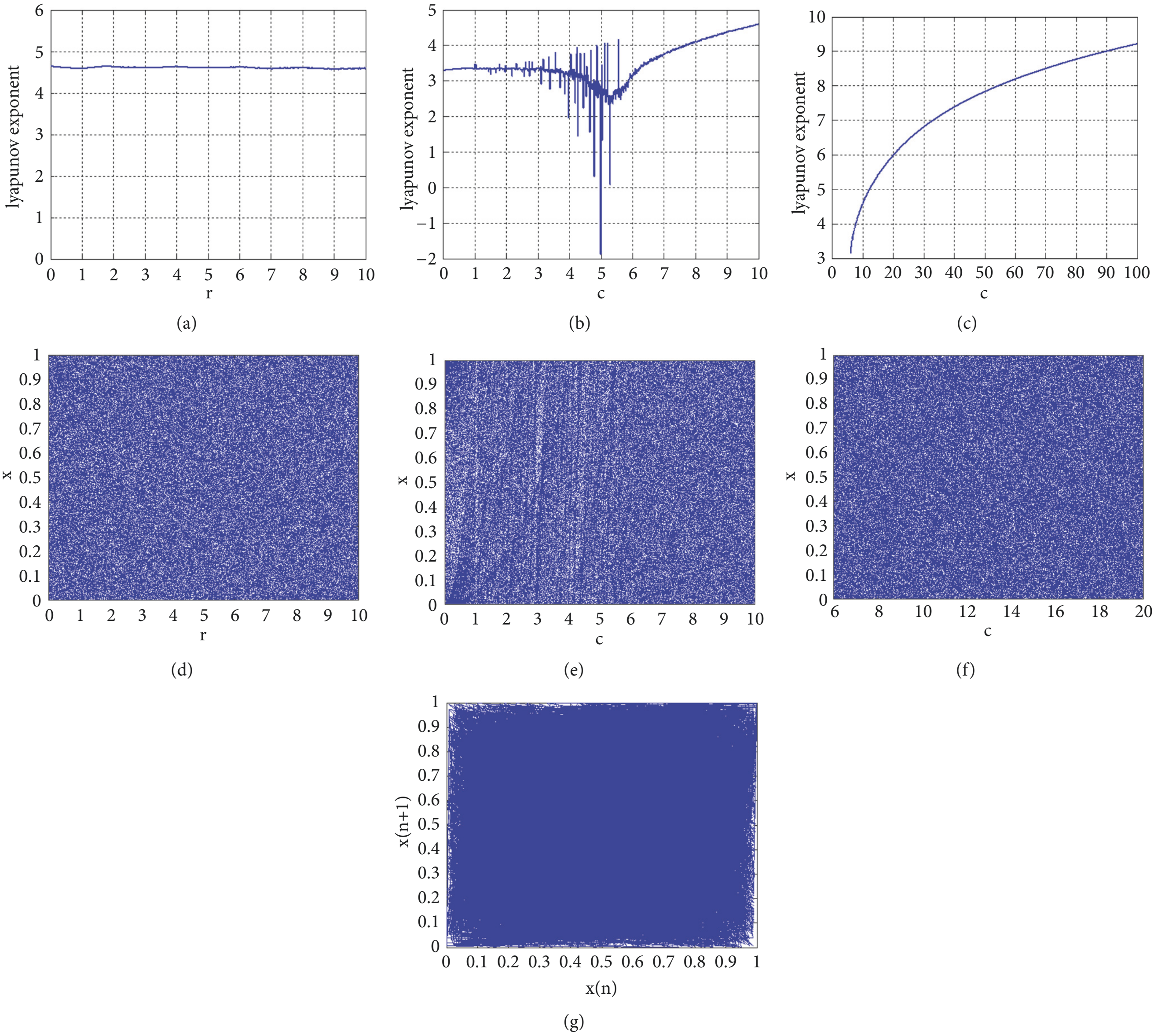
2.3. Chaotic Attractor's Phase Diagram. For the dynamical map $x_{n+1} = F(x_n, \textit{parameters})$, the plot of x_{n+1} versus x_n denotes the attractor's behavior in phase space; it is also referred to as the *return map*. It demonstrates the way the attractor visits its phase space. The phase diagram of proposed map is evaluated and provided in Figure 2(g). It is clear that chaotic attractor of our map visits the whole phase space and dispersed uniformly unlike the attractor of chaotic sine map, thereby, showing the enhanced dynamics of proposed chaotic map.

2.4. Approximate Entropy. Approximate entropy (ApEn) is a statistic that was introduced by Steve Pincus [43] in 1991 to quantify the irregularity in time series. It is one of the most exploited, nonlinear entropy measures that deal with the amount of complexity content in time series data. A small ApEn indicates that the time series is deterministic, whereas its higher value indicates randomness and complexity. This means a time series with lesser repeated patterns has a high ApEn and, thus, will be less predictable. ApEn is a model independent measure of sequence irregularity that is very less affected by noise, which makes it useful in distinguishing correlated stochastic processes and composite deterministic processes. Due to these properties, it is used to characterise complexity content in chaotic behavior of dynamical maps. The ApEn of sequences of floating-point values of size 10000 from two chaotic maps under analysis are computed and listed in Table 2. The calculated ApEn score for chaotic sine map (a = 1) is 0.60017, whereas the value of 1.31088 is obtained for proposed chaotic map (c=10, r=10). The higher ApEn score for proposed map verifies that the sequences from proposed chaotic map (2) have higher complexity, unpredictability, and irregularity compared to classical chaotic sine map.

Table 2: Comparison of approximate entropy for two chaotic maps.

Chaotic map	**Parameter**	**ApEn**
Chaotic Sine map	$a = 1$	0.60017
Proposed Chaotic map	$c = 10, r = 10$	1.31088

2.5. Randomness. NIST SP800-22 statistical test suite is standard and most complete random test suite. It is used to gain confidence that the anticipated scheme is potent



(a) (b) (c)

(d) (e) (f)

(g)

Figure 2: Dynamics of proposed chaotic map: (a) Lyapunov exponent vs $r \in [0, 10]$ for c=10, (b) Lyapunov exponent vs $c \in [0, 10]$ for r=10, (c) Lyapunov exponent vs $c \in [6, 100]$ for r=10, (d) bifurcation vs $r \in [0, 10]$ for c=10, (e) bifurcation vs $c \in [0, 10]$ for r=10, (f) bifurcation vs $c \in [6, 20]$ for r=10, and (g) attractor's phase space for c=r=10.

to generate sequences of high randomness. NIST defines a statistical package for testing randomness of sequences. The suite consists of 15 different statistical tests [44]. In each test, a *p_value* is computed from input sequence of length N. The *p_value* should be greater than significant level α to pass a test and if the sequence passes all tests it is considered random with confidence of 1- α. Otherwise, it is deemed to be nonrandom sequence. By default α is taken to be 0.01, which means that one would expect 1 sequence in 100 to be rejected. It is better for length of input sequences to be at least 10^6. In order to perform NIST randomness tests, the proposed chaotic map (2) is executed for 10^6 times and each chaotic value is transformed to binary for having a binary sequence of length 10^6. The obtained binary sequence is tested for randomness using NIST test suite whose results are listed in Table 3. All the *p_values* for 15 different tests are quite higher than α = 0.01 to show that the generated sequence passed all the tests. Hence, the proposed chaotic map has capability to generate sequences of high randomness.

We found that the proposed chaotic map does not suffer from the limitations discussed earlier as the new map has (1) extremely wider chaotic range for both control parameters, (2) two control parameters which augment the key space of security primitive based on this map, (3) high magnitude of largest Lyapunov exponent, (4) more uniform distribution of chaotic values over the

Table 3: NIST statistical randomness test results for chaotic sequence generated from the proposed chaotic 1D map for $x_0 = 0.1234567, c = 10$, and $r = 10$.

S. No.	Test Name	p-value	Results
1	Frequency Test	0.051895	Passed
2	Block Frequency Test	0.407276	Passed
3	Serial Test	0.018807	Passed
		0.196409	Passed
4	Cumulative Sums Test	0.061700 (forward)	Passed
		0.074320 (backward)	Passed
5	Runs Test	0.029389	Passed
6	Longest Runs of 1's Test	0.759296	Passed
7	Rank Test	0.024293	Passed
8	DFT (Spectral) Test	0.335274	Passed
9	Overlapping Template Matching Test	0.968453	Passed
10	Non-Overlapping Template Matching Test	0.696373	Passed
11	Maurer's Universal Test	0.276614	Passed
12	Approximate Entropy Test	0.575046	Passed
13	Linear Complexity Test	0.789477	Passed
14	Random Excursion Test		
	x = -4	0.832798	Passed
	x= -3	0.674635	Passed
	x= -2	0.731976	Passed
	x = -1	0.301989	Passed
	x = 1	0.465132	Passed
	x = 2	0.854617	Passed
	x = 3	0.981938	Passed
	x = 4	0.024384	Passed
15	Random Excursion Variant Test		
	x = -9	0.177800	Passed
	x = -8	0.188100	Passed
	x = -7	0.226119	Passed
	x = -6	0.188221	Passed
	x = -5	0.219228	Passed
	x = -4	0.441707	Passed
	x= -3	0.407524	Passed
	x= -2	0.289741	Passed
	x = -1	0.545097	Passed
	x = 1	0.322076	Passed
	x = 2	0.299495	Passed
	x = 3	0.571503	Passed
	x = 4	0.703061	Passed
	x = 5	0.594882	Passed
	x = 6	0.442186	Passed
	X = 7	0.398535	Passed
	x = 8	0.498375	Passed
	x = 9	0.563156	Passed

whole range of $(0, 1]$, and (5) high randomness in generated sequences. Hence, the dynamics of proposed chaotic map are better than some existing 1D chaotic maps and appropriate for utilization in any secure cryptographic primitive design.

3. Sine-Cosine Algorithm (SCA)

Sine-cosine algorithm is an effective population-based optimization algorithm which capitalizes the cyclic patterns of mathematical sine and cosine functions to figure out optimization problems. It is recently proposed by Seyedali Mirjalili and proved to have a considerably faster convergence than PSO, GA, FA, Gravitational Search Algorithm (GSA), Bat Algorithm (BA), and Flower Pollination Algorithm (FPA) [45]. Being a population-based technique, it begins with initial generation of random solutions X_i (search agents) in search space which get improved on iterations. An effective optimization technique should have a good balance of two phases: *exploration* and *exploitation* [46]. The feature of exploration is meant to explore the promising regions of search space that are not yet examined with high rate of randomness. In exploitation, incremental changes in random solutions are incorporated to look for neighbouring solutions. In SCA, several parameters are integrated to emphasize equally exploration and exploitation aspects of search space to avoid problem of local optima. Sine-cosine algorithm uses the following equation to update the position of search agent X_i according to parameter r_4.

$$X_i^{t+1} = \begin{cases} X_i^t + r_1 \times \sin(r_2) \times \left| r_3 \times P_i^t - X_i^t \right|, & if\ r_4 < 0.5 \\ X_i^t + r_1 \times \cos(r_2) \times \left| r_3 \times P_i^t - X_i^t \right|, & if\ r_4 \geq 0.5 \end{cases} \tag{4}$$

Here, X_i^t denotes the position of current solution at t^{th} iteration and i^{th} dimension; P_i indicates the best solution obtained so far in i^{th} dimension. It involves four parameters r_1, r_2, r_3, and r_4. The parameter r_1 guides the direction of next solution movement; r_2 controls the amount of random displacement of movement; parameter $r_3 \in [0, 2]$ offers random weight for best solution (so far) to emphasize ($r_3 > 1$) or deemphasize ($r_3 < 1$) its effect; the random parameter $r_4 \in [0, 1]$ helps to switch uniformly between the sine and cosine functions. This cyclic effect of sine and cosine functions enables a solution to reposition around another solution which ensures the exploitation of search space. However, the exploration is achieved by varying the range of two functions through random fixing of parameter r_2 in $[0, 2\pi]$. This way, both exploration and exploitation are guaranteed in during SCA iterations. To balance these two features for effective space search mechanism, it is recommended to update the range of sine and cosine adaptively using equation (5) in [45].

$$r_1 = s - t\frac{s}{T} \tag{5}$$

Here, t is current iteration number, T is maximum allowed iterations, and s is a constant. The primary steps of SCA operation are as follows:

```
initialize a set of search agents (solutions) $X_i$ ($i = 1, 2, \ldots, pop\_count$)
set maximum number of iterations $Max\_iterations$
repeat
    Evaluate each agent $X_i$ by fitness function
    Update the best solution achieved so far ($P = X'$)
    Update $r_1$ using equation (5)
    for each search agent $X_i$
        Update $r_2$, $r_3$, and $r_4$
        Update $X_i$ using equation (4)
while ($current\_iteration < Max\_iterations$)
return the best solution $P$
```

4. Proposed Bijective S-Box Construction

This section presents the proposed bijective S-boxes construction method using sine-cosine algorithm. For bijective S-boxes of sizes 4×4, 5×5, 6×6, 7×7, and 8×8, the respective dimensions are 16, 32, 64, 128, and 256 which correspond to approximated volume of search spaces as high as 10^{13}, 10^{35}, 10^{89}, 10^{215}, and 10^{506}, respectively. We can see that the total search space for S-box search method is quite vast. Practicing a total random (or chaos-based) search does not guarantee a strong S-box [25]. Therefore, it is persuasive to apply a systematic or metaheuristic search technique that can cogently explore and exploit the search space [18].

To begin, a number of random S-boxes of order n are generated using new chaotic map (2) by *initialization*( ) procedure. The steps of sine-cosine algorithm are followed to update all S-box candidates around best S-box obtained so far (termed as *dest_sbox_g*). The updating of S-boxes may violate the bijectivity property. Therefore, a modified *adjustment*() procedure suggested in [11] is executed to remove the duplicate entries and add the missing S-box elements so as to lessen the decrease in nonlinearity. The adjustment is done by scanning the S-box for redundant and missing values. For each redundant entry, the decrease in nonlinearity (d) is calculated corresponding to each missing value. The redundant entry is replaced by the missing value for which d is minimum. After the replacement of all redundant values, the bijectivity of S-box is restored. The *select_best*( ) procedure is intended to return an S-box of order n having highest nonlinearity score among all input S-boxes; i.e., it provides local best S-box for the current iteration. This local best is adjudged against the best S-box achieved so far for reconsideration of global best S-box revision. The fitness of an S-box is determined by computing its nonlinearity performance metric. It is to be noted that most perfect affine and linear approximation attacks reported in [47, 48] have justified the significance of synthesizing S-boxes with high nonlinearity scores. Consequently, the nonlinearity measure, in particular, has given much priority in most of the S-box works available in literature [49]. Therefore, the S-box having higher nonlinearity is preferred over the other S-box during local best selection and global best revision in the optimization iteration of proposed method. The nonlinearity measure of an S-box is discussed in Section 5. A highly nonlinear configuration of n×n S-box is obtained when maximum allowable iterations are reached. The steps of proposed S-box construction method are as follows:

```
**SCA-Optimization ( )**
  Input order of S-box as $n$ ($4 \leq n \leq 8$), $pop\_count$, $Max\_itreration$, $len = 2^n$
  Generate initial population of $n \times n$ S-boxes $X_i$ and store them in 3D matrix $X = X_1, X_2, \ldots, X_{pop\_count}$
      $X =$ **initialization** $(pop\_count, n)$
  Determine local best S-box $X_i$ among $X_1, X_2, \ldots, X_{pop\_count}$ having larger nonlinearity
      $dest\_sbox\_g =$ **select_best**$(X_1, X_2, \ldots, X_{pop\_count})$
  $Fitness\_g =$ **nonlinearity**$(dest\_sbox\_g)$
  $iteration = 1$
  while ($iteration \leq Max\_iteration$)
     Update $r_1$
     for $i = 1$ to $pop\_count$
        for $j = 1$ to $len$
           Update $r_2$, $r_3$ and $r_4$
           Update each element of $X$ according to $r_4$ as
              if ($r_4 < 0.5$)
                 $X_i(j) = (\lfloor (X_i(j) + r_1 \times \sin(r_2) \times |r_3 \times dest\_sbox\_g(j) - X_i(j)|) \times 10^{15} \rfloor) \mathrm{mod}(len)$
              else
                 $X_i(j) = (\lfloor (X_i(j) + r_1 \times \cos(r_2) \times |r_3 \times dest\_sbox\_g(j) - X_i(j)|) \times 10^{15} \rfloor) \mathrm{mod}(len)$
              endif
        endfor
     endfor
     for $t = 1$ to $pop\_count$
        $X[\,][\,][t] =$ **adjustment**$(X[\,][\,][t])$
     endfor
     $local\_sbox\_p =$ **select_best**$(X_1, X_2, \ldots, X_{pop\_count})$
     $Fitness\_p =$ **nonlinearity**$(local\_sbox\_p)$
     if($Fitness\_p \geq Fitness\_g$)
        Update $Fitness\_g$ and $dest\_sbox\_g$ with best solution
     endif
     $iteration = iteration + 1$
  endwhile
  return $dest\_sbox\_g$
```

```
**Initialization** ($pop\_count$, $n$)
  Initialize $x0, c, r, T, len = 2^n, m = \lfloor n/2 \rfloor, i = 2^{\mathrm{m}}, j = 2^{n-m}$
  Take an empty matrix $sbox[i][j][pop\_count]$
  Execute chaotic map (2) for $\beta$ times and dispose the values, except the last
  $t = 1$
  while ($t \leq pop\_count$)
     $j = 1$
     while ($j < len$)
           Further execute map (2) once and collect output $x$
           $q = (\lfloor x \times 10^{15} \rfloor) \mathrm{mod}(len)$
            if ($q \notin sbox[\,][\,][t]$ ))
```

Table 4: The proposed 8×8 substitution-box.

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

```
                append $q$ to $sbox[\ ][\ ][t]$
                $j = j + 1$
            endif
        endwhile
        $t = t + 1$
    endwhile
    return $sbox$
```

Adjustment (*S*)

```
Set $len$ = length($S$), and $n = \lfloor \log_2(len) \rfloor$
Scan $S$ to find all missing elements and store them in array $M$ whose length is $k$
for $i = 1$ to $len$
    Find first repeated value $w$ in $S$, break if no repeated value exists
    Calculate nonlinearities of $S$ as $N_1, N_2, N_3, \ldots, N_n$
    for $j = 1$ to $k$
        Replace $w$ by $M(j)$ in $S$ and $S_j$ is resulted
        Calculate nonlinearities of $S_j$ as $N_1^j, N_2^j, N_3^j, \ldots N_n^j$
        Calculate the decrease of nonlinearities as $d^j = \sum_{t=1}^{n} ( N_t^j - N_t ),\ if\ N_t^j \le N_t$
    endfor
    Denote $S_{new}$ as the S-box having decrease of nonlinearities not smaller than the other S-boxes.
    $S = S_{new}$
endfor
```

5. Performance Results

The experimental setting for performing experiment and simulation analysis includes order of S-box $n \in [4, 8]$, $x_0 = 0.1234567$, $c = 10$, $r = 10$, $\beta = 100$, *Max_iteration* = 500, *pop_count* = 3, and $s = 3$. Here, the SCA parameters r_2, r_3, and r_4 are randomly calculated every time they needed within their specified ranges using chaotic variable obtained by further iterating chaotic map (2). The parameter r_1 is adaptively updated using s as per (5). The five optimized n×n S-boxes (for $4 \le n \le 8$) shown in Tables 4–8 are obtained using proposed method. All floating-point operations are done according to IEEE-754 floating-point standard. All five proposed bijective S-boxes are assessed cryptographically against standard performance benchmarks metrics including nonlinearity, strict avalanche criterion, bit-independent criterion, differential uniformity, LAP, and autocorrelation function. The proposed S-boxes are bijective as the number

Table 5: The proposed 7×7 substitution-box.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	118	82	103	17	18	77	105	93	58	65	84	12	122	30	64	99
1	101	76	55	44	95	106	34	91	78	94	54	29	39	73	50	2
2	21	116	87	117	126	45	97	86	59	13	19	3	80	90	38	52
3	41	28	36	25	7	110	48	120	11	42	102	23	72	121	109	104
4	114	9	56	79	70	66	92	107	40	111	62	115	14	127	24	63
5	71	15	68	108	51	125	6	81	60	8	20	31	69	4	85	124
6	22	1	98	96	74	89	16	26	27	0	119	49	5	57	10	32
7	112	37	123	35	47	100	33	46	61	113	83	75	43	67	88	53

Table 6: The proposed 6×6 substitution-box.

	0	1	2	3	4	5	6	7
0	8	7	45	23	5	21	48	12
1	18	38	31	51	1	60	27	11
2	25	61	20	13	33	56	42	46
3	32	26	10	34	15	2	41	29
4	52	54	6	43	36	14	44	47
5	4	40	59	49	37	17	50	55
6	22	0	53	16	35	19	57	62
7	30	3	24	39	9	28	58	63

Table 7: The proposed 5×5 substitution-box.

	0	1	2	3	4	5	6	7
0	25	2	6	21	29	24	11	28
1	5	3	1	7	13	30	31	20
2	17	16	19	9	26	22	0	15
3	12	27	4	10	23	14	18	8

Table 8: The proposed 4×4 substitution-box.

	0	1	2	3
0	14	15	4	2
1	5	8	10	12
2	9	6	1	13
3	3	0	11	7

of preimages of each S-box output is only one and all 2^n elements of proposed $n{\times}n$ S-box are distinct and lie in $[0, 2^n - 1]$. The S-boxes of size 8×8 are particularly emphasized in literature. Therefore, the detailed performance outcomes and corresponding metric tables are provided exclusively for our 8×8 S-box. The performance results of all five bijective S-boxes are also compared with recent S-boxes which are majorly based on optimization techniques.

5.1. Nonlinearity. The S-boxes are typically vectorial Boolean functions; as a result their nonlinearity score is determined by finding the nonlinearity of its component Boolean functions. The measure of nonlinearity of S-boxes used in block ciphers is crucial as it puts restraint on the linear attacks. A higher nonlinearity provides a bad approximation by linear functions and makes the linear cryptanalysis difficult for attacker [12]. It is the sole purpose of S-boxes to offer highly nonlinear transformation from input data to the encoded data. Practically, the nonlinearity for any n-bit Boolean function $f(x)$ is figured out via Walsh spectrum as [49, 50].

$$nl(f) = 2^{n-1} - \frac{1}{2}\left(\max_{z\in\{0,1\}^n} \left|W_f(z)\right|\right) \tag{6}$$

Here $W_f(z)$ is the Walsh spectrum of Boolean function $f(x)$, which is defined as

$$W_f(z) = \sum_{x\in\{0,1\}^n} (-1)^{f(x)\oplus x.z} \tag{7}$$

Here, $x.z$ is the bitwise dot product and $z \in \{0,1\}^n$. We calculated the nonlinearity of component n Boolean functions of our proposed $n{\times}n$ S-boxes which are put together in Table 9. High nonlinearity of our all S-boxes proved the effectiveness of proposed S-box method in terms of generating highly nonlinear S-boxes. Specifically, the respective scores for proposed 8×8 S-box in Table 4 are 110, 110, 110, 110, 110, 108, 110, and 108, indicating that all nonlinearities are greater than or equal to 108 with a remarkable average of 109.5. Thus, the proposed method is able to offer high nonlinearity, strong confusion, and good defiance to approximation assaults.

5.2. Strict Avalanche Criterion. Webster and Tavares generalized the idea of avalanche effect and introduced its formal version which was named as the strict avalanche criterion (SAC). If f is a Boolean function such that it satisfies SAC, it entails that if we one alters any one of the input data bits, then all of the output bits should be modified with an ideal probability of 1/2. In order to verify SAC for S-boxes, the dependency matrix using an approach given in [51] is determined. Each value of the dependency matrix should be close to the ideal value of 0.5, whose average is identified as the SAC value. An ideal SAC is crucial to shrink the chances of correlation among all input/output combination and information leakage to attackers. To illustrate this, the estimated dependency matrix for proposed 8×8 S-box is shown in Table 10, whose average is 0.4985 which is reasonably close to 0.5. The average of dependency matrices for other proposed S-boxes is also obtained and their respective averages are listed in Table 11 to demonstrate the satisfactory performance of our S-boxes with respect to stated avalanche criterion.

Table 9: Nonlinearities of n Boolean functions f_i of the proposed bijective $n \times n$ S-boxes.

S-box	Nonlinearities								
	nl_0	nl_1	nl_2	nl_3	nl_4	nl_5	nl_6	nl_7	average
4×4	4	4	4	4	-	-	-	-	4
5×5	12	10	12	10	12	-	-	-	11.2
6×6	22	24	24	24	24	24	-	-	23.666
7×7	50	52	52	52	52	52	52	-	51.714
8×8	110	110	110	110	110	108	110	108	109.5

Table 10: Dependency matrix for SAC of the proposed 8×8 S-box.

0.5	0.5156	0.5	0.4218	0.4531	0.5937	0.5	0.5156
0.5625	0.4218	0.4375	0.5	0.4531	0.5781	0.4531	0.5
0.5312	0.5312	0.5625	0.5468	0.4218	0.4843	0.4843	0.5312
0.5312	0.4375	0.5156	0.4218	0.4687	0.4687	0.4687	0.5937
0.4843	0.4531	0.5156	0.5468	0.5156	0.5468	0.5625	0.5468
0.5156	0.4531	0.5156	0.5156	0.5	0.5156	0.5156	0.4062
0.4843	0.5156	0.5156	0.5156	0.5312	0.5312	0.5	0.4531
0.5156	0.4843	0.5	0.4843	0.4531	0.5156	0.4062	0.4843

Table 11: SAC scores for the proposed bijective $n \times n$ S-boxes.

S-box	SAC
4×4	0.5781
5×5	0.505
6×6	0.5035
7×7	0.5019
8×8	0.4985

Table 12: BIC-NL results for the proposed 8×8 S-box.

	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7
f_0	-	106	104	100	100	104	106	104
f_1	106	-	106	102	106	102	106	102
f_2	104	106	-	102	104	104	106	104
f_3	100	102	102	-	108	106	102	106
f_4	100	106	104	108	-	104	98	106
f_5	104	102	104	106	104	-	106	106
f_6	106	106	106	102	98	106	-	104
f_7	104	102	104	106	106	106	104	-

5.3. Bits Independent Criterion. Another performance measure for S-boxes referred to as bits independent criterion (BIC) has been apprised by Adams and Tavares [52]. It entails that when any input bit is inverted the output bits should change independently for all pairs of avalanche vectors. This means that the output bits should be pairwise independent. Let $f_1, f_2, \ldots, f_n$ be the component n Boolean functions of a bijective S-box of order n, it is said to accomplish the criterion provided that the function $f_k \oplus f_j$ $(k \neq j, 0 \leq k, j \leq n\text{-}1)$ is highly nonlinear and also satisfies the SAC. The BIC results for all possible functions $f_k \oplus f_j$of proposed 8×8 S-box are calculated and shown in Tables 12 and 13 for both nonlinearity and SAC measures. The average value of BIC-nonlinearity matrix is 104.07, which is a pretty high and the average of BIC-SAC matrix is 0.5020, which is also close to 0.5. These two averages are also computed for other proposed S-boxes and listed in Table 14. The obtained BIC results designate that proposed S-boxes are quite proficient in fulfilling the BIC performance criterion.

5.4. Differential Uniformity. Measure of differential uniformity (DU) is associated with the change in the differential output detected for a change in input Δa. It is observed to quantify robustness of S-box to differential cryptanalysis practiced by Biham and Shamir [53]. DU denotes the upper limit of likelihoods of yielding an output differential $\Delta b = b_i \oplus b_j$ when the input differential is $\Delta a = a_i \oplus a_j$. For computation, the exclusive-OR distributions between the inputs and outputs of anticipated S-box are recorded using the following expression.

$$DU(S) = \max_{\Delta a \neq 0, \Delta b} \left(\# \{ a \in S \mid S(a) \oplus S(a \oplus \Delta a) = \Delta b \} \right) \quad (8)$$

Adopting the procedure explained in [53], the I/O exclusive-OR distribution for proposed 8×8 S-box is figured out and presented in Table 15. We can see that its largest entry is 10 which is the obtained differential uniformity of our S-box. The DU scores of our other S-boxes are provided in Table 16. The proposed S-boxes are found to have low differential uniformities and hence able to resist differential cryptanalysis.

5.5. Linear Approximation Probability. The notion of linear approximation probability to assess robustness of an S-box is introduced by Matsui [48]. It denotes the likelihood of getting a linear approximation of anticipated S-box. The LAP of an S-box relies on the concurrence of input bits with output bits. To compute LP, for two chosen masks m_a and m_b, the mask of

Table 13: BIC-SAC results for the proposed 8×8 S-box.

	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7
f_0	-	0.498	0.4843	0.4746	0.5156	0.5019	0.5136	0.5175
f_1	0.498	-	0.498	0.5	0.4902	0.5058	0.5058	0.5175
f_2	0.4843	0.498	-	0.4746	0.5273	0.5234	0.4902	0.4921
f_3	0.4746	0.5	0.4746	-	0.5078	0.5136	0.4648	0.5039
f_4	0.5156	0.4902	0.5273	0.5078	-	0.5078	0.498	0.5019
f_5	0.5019	0.5058	0.5234	0.5136	0.5078	-	0.5019	0.5273
f_6	0.5136	0.5058	0.4902	0.4648	0.498	0.5019	-	0.498
f_7	0.5175	0.5175	0.4921	0.5039	0.5019	0.5273	0.498	-

Table 14: BIC scores for the proposed bijective $n\times n$ S-boxes.

S-box	BIC-NL	BIC-SAC
4×4	3.667	0.5208
5×5	9.4	0.5325
6×6	22.13	0.5145
7×7	48	0.5068
8×8	104.07	0.5020

all possible values of input a is evaluated. Similarly, the masks of all output values $b = S(a)$ for given S-box S are determined. The highest occurrence of the results is considered as LAP. Accordingly, this probability is computed mathematically as

$$LAP(S) = \max_{m_a, m_b \neq 0} \left| \frac{\# \{ a \in S \mid a.m_a = b.m_b \}}{2^n} - 0.5 \right| \tag{9}$$

Here, m_a and m_b denoted mask values of inputs and outputs and there are 2^n possible inputs a for bijective $n\times n$ S-box. An S-box with high LAP score may lead to break under Matsui's linear cryptanalysis. We followed the procedure given in [48] to calculate the LAP for our proposed S-boxes; the obtained probabilities are provided in Table 16. In particular, the LAP score for proposed 8×8 S-box comes out as 0.1328 which is fairly low compared to several contemporary S-boxes.

5.6. Autocorrelation Function. Mathematically, the autocorrelation function of any Boolean function f is determined as [54]

$$r_f(d) = \sum_{\forall x, d \in \{0,1\}^n} (-1)^{f(x)} (-1)^{f(x \oplus d)} \tag{10}$$

Here, $r(d) = 2^n$ (when $d = 0$) for all Boolean functions, and for other possible inputs $r(d)$ lies in $[2^{-n}, 2^n]$. The highest value of ACF is designated as the absolute indicator of function f which ascertains good diffusion property as cryptographic quality of function [55]. It is expressed as

$$\left| ACF_f \right| = \max \left(\left| r_f(d) \right| \right) \quad for\ d \neq 0 \tag{11}$$

The ACF quality metric of Boolean functions is broadened to gauge an S-box S: $\{0,1\}^n \longrightarrow \{0,1\}^n$ by taking all 2^n -1 nonzero linear combinations F of all its n component functions. For an S-box, it is obtained using the following expression [56].

$$\left| ACF_S \right| = \max \left(\left| r_{F_i}(w) \right| \right) \qquad w = 1, \dots, 2^n \ i = 1, \dots, 2^n - 1 \tag{12}$$

For a cryptographically strong S-box, the score of ACF of S-box should be kept as low as possible. The largest of ACF score for our 8×8 S-box comes out as 96. The ACF scores for all proposed bijective S-boxes are shown in Table 16.

5.7. Comparison. The performance outcomes of proposed S-boxes obtained in previous sections are compared with optimization-based and other S-boxes recently suggested in literature. For cryptographic strength, the S-boxes, having larger nonlinearity, SAC closer to 0.5, maximum nonlinearity, and satisfaction of SAC while examining bits independent criterion, lower differential uniformity, LAP, and autocorrelation, are appreciated. Specifically, the performance comparison of proposed 8×8 S-box is done in Table 17; the following remarks are recognized:

(a) The proposed S-box is proficient to supply highest nonlinearity compared to all S-boxes investigated in Table 17. The statistics of minimum (108) is equal to S-boxes in [10, 11, 19], maximum (110) is similar to S-boxes in [11–17, 19], and these statistics are better than rest of the S-boxes of Table 17. It shows that proposed method can construct highly nonlinear S-boxes than most of the recent S-boxes.

(b) The SAC score of proposed 8×8 S-box is 0.4985 which indicates that S-box can satisfy the avalanche criterion quite well similar to other recent S-boxes and it is found slightly better than most of the listed S-boxes.

(c) The bits independent criterion for nonlinearity and SAC comes out as 104.07 and 0.5020 for proposed S-box. Our BIC-NL results are fairly higher than S-boxes investigated in [10, 12, 15, 16, 19, 26, 28–31, 34] and BIC-SAC satisfies the avalanche criterion with negligible margin like other S-boxes. Hence, satisfaction of bits independence criterion is consistent and better than most of the 8×8 S-boxes.

(d) The worst differential uniformity of 54 is exhibited by S-box investigated in [31]. Our differential uniformity is considerably improved compared with DU scores found in [28, 30–32, 34] and it is consistent with rest of S-boxes. This shows that our S-box can offer a decent resistance to Biham's cryptanalysis.

Table 15: I/O XOR distribution for the proposed 8×8 S-box.

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

Table 16: The results of DU, LAP, and ACF for proposed bijective $n \times n$ S-boxes.

S-box	DU	LAP	ACF
4×4	6	0.375	16
5×5	6	0.25	32
6×6	8	0.25	40
7×7	10	0.1875	64
8×8	10	0.1328	96

Table 17: Comparison of some recent optimization techniques based 8×8 S-boxes.

8×8 S-Box	Nonlinearity			SAC	BIC-NL	BIC-SAC	DU	LAP	ACF
	min	max	avg.						
Proposed	108	110	109.5	0.4985	104.07	0.5020	10	0.13281	96
Wang [10]	108	108	108	0.5068	103.36	0.5017	10	0.14062	96
Wang [11]	108	110	109	0.5026	104.79	0.5026	10	0.1406	104
Guesmi [12]	106	110	107.5	0.4971	103.857	0.5034	10	0.125	96
Ahmad [13]	106	110	107	0.5014	104.214	0.5016	10	0.1484	96
Tian [14]	106	110	108	0.5073	104	0.5020	10	0.15234	96
Tian [15]	106	110	107.5	0.5093	103.07	0.5025	10	0.1406	96
Ahmad [16]	106	110	107.5	0.5036	103.93	0.5040	10	0.1484	104
Farah [17]	104	110	106.5	0.4995	104.571	0.4983	10	0.1172	96
Hussam [18]	106	108	107.5	0.4943	104.35	0.4982	10	0.125	96
Zhang [19]	108	110	108.75	0.4946	102.785	0.5054	10	0.1328	104
Tian [26]	104	108	106.75	0.4976	103.57	0.5022	10	0.1328	96
Khan [28]	84	106	100	0.4812	101.9	0.4962	16	0.1797	192
Anees [29]	100	106	103	0.5020	102.93	0.4998	10	0.1406	112
Jamal [30]	98	108	102.25	0.4836	101.57	0.4948	14	0.1679	108
Khan [31]	96	106	103.25	0.5151	103.07	0.4864	54	0.1562	120
Ataullah [32]	106	108	106.75	0.4939	106.57	0.5040	16	0.125	168
Razzaq [34]	104	108	106.75	0.5031	103.64	0.5074	12	0.1484	104
Hayat [35]	100	108	105	0.5007	104.14	0.4965	10	0.1328	104

Table 18: Comparison of average nonlinearities of small-sized S-boxes.

S-box	Ref. [21]	Ref. [22]	Ref. [23]	Ref. [24]	Ref. [25]	Ref. [27]	Proposed
4×4	NR	NR	NR	NR	4	4	4
5×5	10	6	10	10	10	11.2	11.2
6×6	20	18	22	22	22	23.667	23.667
7×7	46	42	48	48	48	51.14	51.714
8×8	102	104	98	104	104	108.5	109.5

(e) The lowest linear approximation probability of 0.1172 is offered by Farah's S-box investigated in [18] among all 8×8 S-boxes of Table 17. We found a LAP value of 0.13281 and can deliver more resistance to linear cryptanalysis than S-boxes investigated in [10, 11, 13–16, 28–31, 34].

(f) In [55], it has been pointed out that measure of autocorrelation function relates to the cryptographic diffusion characteristics that an S-box can offer as a part of a cryptosystem. The calculated ACF for proposed 8×8 S-box is 96 and better than ACF scores of many S-boxes synthesized in [11, 16, 19, 28–32, 34, 35] and it is comparable with other S-boxes of the table. This means that our S-box can provide better ACF property and diffusion when incorporated in a cryptographic system.

We compared the nonlinear performance of other proposed bijective S-boxes with similar proposals available in literature in Table 18. The nonlinear measure of bijective S-boxes is consistently reported in all these proposals. It is worth noting that the best nonlinearity results are reported in Table 18 for all proposals. From the comparison of Table 18, it is easy to acknowledge that the proposed S-boxes can offer larger and better nonlinearity than the rest of the other contemporary S-boxes. Hence, the proposed S-box method is well competent for the construction of highly nonlinear bijective S-boxes.

6. Conclusion

Constructing cryptographically strong substitution-boxes is a problem which has been addressed with much attention. Due to vast search space of S-boxes, it is not advisable to follow a random search approach as it does not guarantee a quality S-box. On the other hand, metaheuristic techniques have been investigated to design a systematic search mechanism for synthesis of strong S-boxes. To generate more nonlinearly better bijective S-boxes, a recent sine-cosine algorithm as an optimization technique is applied in this paper. To augment the effective searching of SCA, an improved 1D chaotic map is proposed which owns some enhanced dynamics compared to conventional chaotic sine map. The good balance of exploration and exploitation of search space in SCA enables the generation of strong S-boxes with proposed method. In proposed SCA based S-box method, the initial population of S-boxes is generated randomly using new chaotic map. The same chaotic map is further utilized in continuation to assign fresh values to SCA parameters. During optimization mechanism of proposed method, the nonlinearity is taken as fitness function to obtain the optimized S-box. Example bijective S-boxes obtained from proposed method are provided and detailed performance outcomes for proposed 8×8 S-box are presented. The results demonstrate that the proposed S-boxes not only offer high nonlinearity but also fulfil other performance criteria pretty well. Moreover, the comparison analysis revealed that the proposed method outperforms many optimization-based S-boxes and is suitable for cryptographic applications.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

[1] R. A. Mollin, *An Introduction to Cryptography*, Chapman and Hall/CRC, Boca Raton, Fla, USA, 2007.

[2] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, Boca Raton, Fla, USA, 1997.

[3] M. Ahmad, M. N. Doja, and M. M. Beg, "ABC optimization based construction of strong substitution-boxes," *Wireless Personal Communications*, vol. 101, no. 3, pp. 1715–1729, 2018.

[4] C. M. Adams, *A formal and practical design procedure for substitution-permutation network cryptosystems [Doctoral Dissertation]*, Queen's University Kingston, Queen, Canada, 1990.

[5] H. Feistel, "Cryptography and computer privacy," *Scientific American*, vol. 228, no. 5, pp. 15–23, 1973.

[6] V. Rijmen, P. S. L. M. Barreto, and D. L. G. Filho, "Rotation symmetry in algebraically generated cryptographic substitution tables," *Information Processing Letters*, vol. 106, no. 6, pp. 246–250, 2008.

[7] S. D. Sinha and C. P. Arya, "Algebraic construction and cryptographic properties of Rijndael substitution box," *Defence Science Journal*, vol. 62, no. 1, pp. 32–37, 2012.

[8] V. B. Gregory, *Algebraic Cryptanalysis*, Springer, Berlin, Germany, 2009.

[9] A. M. Youssef, S. E. Tavares, and G. Gong, "On some probabilistic approximations for AES-like s-boxes," *Discrete Mathematics*, vol. 306, no. 16, pp. 2016–2020, 2006.

[10] Y. Wang, K.-W. Wong, C. Li, and Y. Li, "A novel method to design S-box based on chaotic map and genetic algorithm," *Physics Letters A*, vol. 376, no. 6-7, pp. 827–833, 2012.

[11] W. Yong and L. Peng, "An Improved Method to Obtaining S-Box Based on Chaos and Genetic Algorithm," *HKIE Transactions*, vol. 19, no. 4, pp. 53–58, 2012.

[12] R. Guesmi, M. A. B. Farah, A. Kachouri, and M. Samet, "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] M. Ahmad, D. Bhatia, and Y. Hassan, "A novel ant colony optimization based scheme for substitution box design," *Procedia Computer Science*, vol. 57, pp. 572–580, 2015.

[14] Y. Tian and Z. Lu, "S-box: Six-dimensional compound hyperchaotic map and artificial bee colony algorithm," *Journal of Systems Engineering and Electronics*, vol. 27, no. 1, pp. 232–241, 2016.

[15] Y. Tian and Z. Lu, "Chaotic s-box: intertwining logistic map and bacterial foraging optimization," *Mathematical Problems in Engineering*, vol. 2017, Article ID 6969312, 11 pages, 2017.

[16] M. Ahmad, N. Mittal, P. Garg, and M. Maftab Khan, "Efficient cryptographic substitution box design using travelling salesman problem and chaos," *Perspectives in Science*, vol. 8, pp. 465–468, 2016.

[17] T. Farah, R. Rhouma, and S. Belghith, "A novel method for designing S-box based on chaotic map and Teaching–Learning-Based Optimization," *Nonlinear Dynamics*, vol. 88, no. 2, pp. 1059–1074, 2017.

[18] H. A. Ahmed, M. F. Zolkipli, and M. Ahmad, "A novel efficient substitution-box design based on firefly algorithm and discrete chaotic map," *Neural Computing and Applications*, 2018.

[19] T. Zhang, C. L. Chen, L. Chen, X. Xu, and B. Hu, "Design of highly nonlinear substitution boxes based on I-ching operators," *IEEE Transactions on Cybernetics*, pp. 1–10, 2018.

[20] A. A. Alzaidi, M. Ahmad, M. N. Doja, E. A. Solami, and M. M. Beg, "A new 1D chaotic map and β-hill climbing for generating substitution-boxes," *IEEE Access*, vol. 6, no. 1, pp. 55405–55418, 2018.

[21] W. Millan, "How to improve the nonlinearity of bijective S-boxes," in *Information Security and Privacy*, vol. 1438 of *Lecture Notes in Computer Science*, pp. 181–192, Springer, Berlin, Germany, 1998.

[22] J. Fuller, W. Millan, and E. Dawson, "Multi-objective optimisation of bijective S-boxes," *New Generation Computing*, vol. 23, no. 3, pp. 201–218, 2005.

[23] E. C. Laskari, G. C. Meletiou, and M. N. Vrahatis, "Utilizing evolutionary computation methods for the design of S-boxes," in *Proceedings of the 2006 International Conference on Computational Intelligence and Security, ICCIAS 2006*, pp. 1299–1302, China, October 2006.

[24] P. Tesar, "A new method for generating high non-linearity s-boxes," *Radioengineering*, vol. 19, pp. 23–26, 2010.

[25] S. Picek, M. Cupic, and L. Rotim, "A new cost function for evolution of S-Boxes," *Evolutionary Computation*, vol. 24, no. 4, pp. 695–718, 2016.

[26] T. Ye and L. Zhimao, "Chaotic S-box: six-dimensional fractional Lorenz–Duffing chaotic system and O-shaped path scrambling," *Nonlinear Dynamics*, vol. 94, no. 3, pp. 2115–2126, 2018.

[27] 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, 2018.

[28] M. Khan, T. Shah, and S. I. Batool, "Construction of S-box based on chaotic Boolean functions and its application in image encryption," *Neural Computing and Applications*, vol. 27, no. 3, pp. 677–685, 2016.

[29] A. Anees and Z. Ahmed, "A technique for designing substitution box based on van der pol oscillator," *Wireless Personal Communications*, vol. 82, no. 3, pp. 1497–1503, 2015.

[30] S. S. Jamal, M. U. Khan, and T. Shah, "A watermarking technique with chaotic fractional S-box transformation," *Wireless Personal Communications*, vol. 90, no. 4, pp. 2033–2049, 2016.

[31] M. Khan and Z. Asghar, "A novel construction of substitution box for image encryption applications with Gingerbreadman chaotic map and S8 permutation," *Neural Computing and Applications*, vol. 29, no. 4, pp. 993–999, 2018.

[32] A. Attaullah, S. S. Jamal, and T. Shah, "A novel construction of substitution box using a combination of chaotic maps with improved chaotic range," *Nonlinear Dynamics*, vol. 88, pp. 2757–2769, 2017.

[33] D. Lambic, "A novel method of S-box design based on discrete chaotic map," *Nonlinear Dynamics*, vol. 87, no. 4, pp. 2407–2413, 2017.

[34] A. Razaq, A. Yousaf, U. Shuaib, N. Siddiqui, A. Ullah, and A. Waheed, "A novel construction of substitution box involving coset diagram and a bijective map," *Security and Communication Networks*, vol. 2017, Article ID 5101934, 16 pages, 2017.

[35] U. Hayat, N. A. Azam, and M. Asif, "A Method of Generating 8×8 Substitution Boxes Based on Elliptic Curves," *Wireless Personal Communications*, vol. 101, no. 1, pp. 439–451, 2018.

[36] D. Arroyo, G. Alvarez, and V. Fernandez, "On the inadequacy of the logistic map for cryptographic applications," 2008, https://arxiv.org/abs/0805.4355.

[37] S. Li, G. Chen, and X. Mou, "On the dynamical degradation of digital piecewise linear chaotic maps," *International Journal of Bifurcation and Chaos*, vol. 15, no. 10, pp. 3119–3151, 2005.

[38] C. Pak and L. Huang, "A new color image encryption using combination of the 1D chaotic map," *Signal Processing*, vol. 138, pp. 129–137, 2017.

[39] R. Parvaz and M. Zarebnia, "A combination chaotic system and application in color image encryption," *Optics & Laser Technology*, vol. 101, pp. 30–41, 2018.

[40] S. Behnia, A. Akhshani, H. Mahmodi, and A. Akhavan, "Chaotic cryptographic scheme based on composition maps," *International Journal of Bifurcation and Chaos*, vol. 18, no. 1, pp. 251–261, 2008.

[41] Y. Zhou, L. Bao, and C. L. P. Chen, "A new 1D chaotic system for image encryption," *Signal Processing*, vol. 97, pp. 172–182, 2014.

[42] L. Chen, S. Tang, Q. Li, and S. Zhong, "A new 4D hyperchaotic system with high complexity," *Mathematics and Computers in Simulation*, vol. 146, pp. 44–56, 2018.

[43] S. M. Pincus, "Approximate entropy as a measure of system complexity," *Proceedings of the National Acadamy of Sciences of the United States of America*, vol. 88, no. 6, pp. 2297–2301, 1991.

[44] A. Rukhin, J. Sota, J. Nechvatal et al., "A Statistical test suite for random and pseudorandom number generators for cryptographic applications," NIST special publication 800-22, 2001.

[45] S. Mirjalili, "SCA: a sine cosine algorithm for solving optimization problems," *Knowledge-Based Systems*, vol. 96, pp. 120–133, 2016.

[46] M. Črepinšek, S.-H. Liu, and M. Mernik, "Exploration and exploitation in evolutionary algorithms: a survey," *ACM Computing Surveys*, vol. 45, no. 3, article 35, 2013.

[47] C. Ding, G. Xiao, and W. Shan, *The stability theory of stream ciphers, LNCS 561*, Springer Science and Business Media, 1991.

[48] T. Helleseth, "Linear cryptanalysis method for des cipher," in *Advances in Cryptology—EUROCRYPT*, vol. 765 of *Lecture*

Notes in Computer Science, pp. 386–397, Springer, Berlin, Germany, 1993.

[49] C. Carlet and C. Ding, “Nonlinearities of S-boxes,” *Finite Fields and Their Applications*, vol. 13, no. 1, pp. 121–135, 2007.

[50] T. W. Cusick and P. Stanica, *Cryptographic Boolean Functions and Applications*, Elsevier, Amsterdam, Netherlands, 2009.

[51] A. F. Webster and S. E. Tavares, “On the design of S-boxes,” in *Advances in Cryptology*, vol. 218 of *Lecture Notes in Computer Science*, pp. 523–534, 1986.

[52] C. Adams and S. Tavares, “The structured design of cryptographically good S-boxes,” *Journal of Cryptology. The Journal of the International Association for Cryptologic Research*, vol. 3, no. 1, pp. 27–41, 1990.

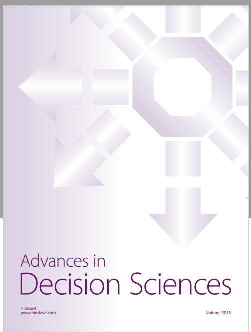
[53] E. Biham and A. Shamir, “Differential cryptanalysis of DES-like cryptosystems,” *Journal of Cryptology*, vol. 4, no. 1, pp. 3–72, 1991.

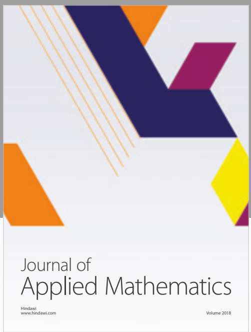
[54] L. D. Burnett, *Heuristic optimization of Boolean functions and substitution boxes for cryptography [Doctoral dissertation]*, Queensland University of Technology, 2005.

[55] S. Kavut, “Results on rotation-symmetric S-boxes,” *Information Sciences*, vol. 201, pp. 93–113, 2012.

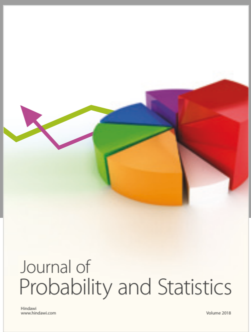
[56] X.-M. Zhang and Y. Zheng, “GAC — the criterion for global avalanche characteristics of cryptographic functions,” *J.UCS. The Journal of Universal Computer Science*, vol. 1, no. 5, pp. 320–337, 1996.

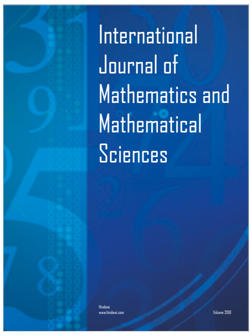
Advances in
Operations Research


Advances in
Decision Sciences

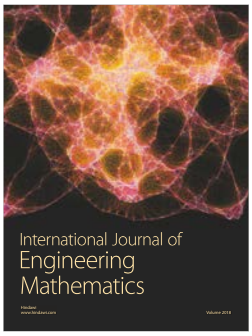

Journal of
Applied Mathematics

The Scientific
World Journal

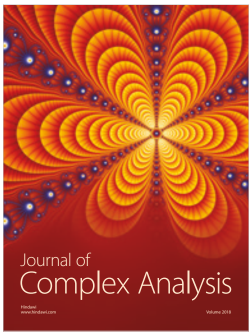

Journal of
Probability and Statistics

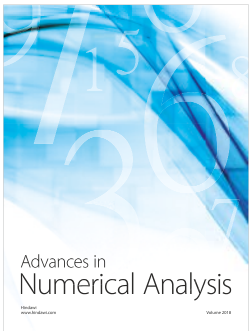

International
Journal of
Mathematics and
Mathematical
Sciences

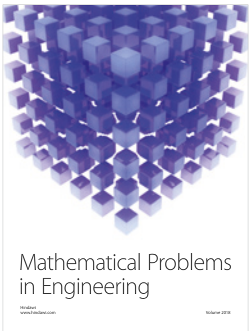
Journal of
Optimization

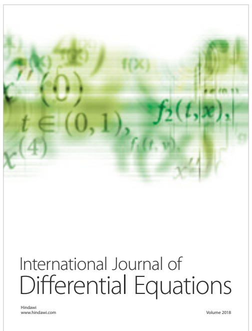

International Journal of
Engineering
Mathematics

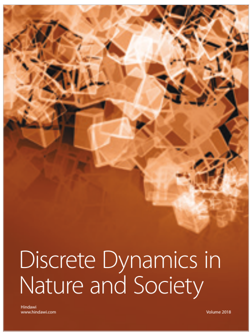
International Journal of
Analysis

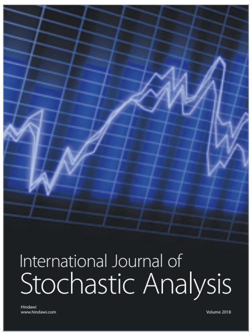

Journal of
Complex Analysis


Advances in
Numerical Analysis


Mathematical Problems
in Engineering

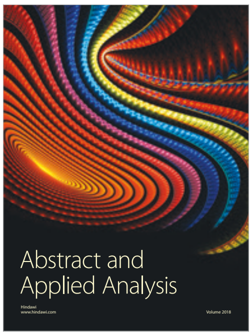

International Journal of
Differential Equations

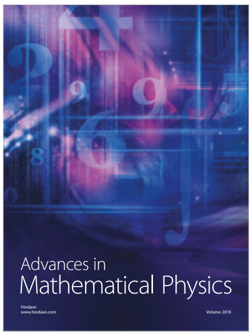

Discrete Dynamics in
Nature and Society


International Journal of
Stochastic Analysis

Journal of
Mathematics

Journal of
Function Spaces


Abstract and
Applied Analysis


Advances in
Mathematical Physics