Neural Computing and Applications (2021) 33:5111–5133
https://doi.org/10.1007/s00521-020-05304-9

ORIGINAL ARTICLE

Execution examination of chaotic S-box dependent on improved PSO algorithm

Nafiseh Hematpour[1] · Sodeif Ahadpour[1]

Received: 21 April 2020 / Accepted: 18 August 2020 / Published online: 31 August 2020

Abstract

Achieving proper nonlinear properties and autocorrelation in the S-box structure is an open challenge in cryptography. Besides, there have been numerous articles on the optimization of S-box, using two types of fitness functions for optimization. This study investigated both types of functions and compares their performance. In addition, this study used ergodic chaotic maps. First, the performance of particle swarm optimization (PSO) was improved using these maps. Then, the new chaotic S-boxes were designed based on the ergodic maps. After that, the improved PSO was used for optimization to obtain the best S-boxes. This optimization was performed once by selecting nonlinearity as a fitness function. At the second optimization, the entropy source was selected as a fitness function for optimization by examining the *P*-value of the mono-test frequency. Finally, the related results for the introduced chaotic S-boxes were compared to the optimized chaotic S-boxes with two types of fitness functions. The introduced S-boxes were safe due to the use of ergodic maps with high keyspace length. Furthermore, the simulation performance was analyzed and compared with other relevant approaches.

Keywords Substitution boxes (S-boxes) · Chaotic maps · Particle swarm optimization · Nonlinearity · Mono-test frequency · Performance analysis

1 Introduction

Maintaining the security of information transmission is important given the significant advances in network communication technologies. A serious challenge in cryptosystems is designing of substitution boxes (S-boxes) [1]. S-box designs are proposed based on mathematical structures, a group of theoretic approach [2]. S-box, performing confusion, has been widely employed in traditional cryptographic standards such as the Data Encryption Standard (DES) [3] and the Advanced Encryption Standard (AES) [4] for encryption and decryption process. Considering that the previous designs still do not have the highest score for the good S-box criteria, it is necessary to create a new S-box structure. In order to determine how efficient the S-box is, it has been examined in the form of a series of criteria such as nonlinearity (NL), linear approximation (LP), differential approximation (DP), strictly avalanche criteria (SAC), and bit independence criteria (BIC) [5, 6]. Therefore, if it reaches 120 in nonlinearity attacks [7], it will be one of the most ideal possible arrangements against nonlinearity attacks. The best-obtained value of SAC is 0.500 [8]. The best values so far obtained for BIC, LP, and DP are 112, 0.062, and 4, respectively [4]. Researchers have recently developed some algorithms for S-box design using the nonlinear properties of chaos. To use the chaotic space of nonlinear dynamic systems, a logistic map was used to design the S-box [9]. Subsequent studies have used maps such as three-dimensional Baker [8] or a combination of Baker and logistics [10] to develop the keyspace. Since hyper-chaotic systems have unique features, these systems were used in this article [11]. Studies on the use of chaos theory in the design of S-boxes have focused on the development of keyspace, and the use of chaotic boxes in nonlinear systems has not been considered in previous studies [12–18]. In ref. [19], the initial S-box was made with a spatiotemporal chaotic system, and then, its

✉ Sodeif Ahadpour
ahadpour@uma.ac.ir

Nafiseh Hematpour
n_hematpour@uma.ac.ir

[1] Department of Physics, University of Mohaghegh Ardabili, Ardabil, Iran

elements were randomly changed. The use of fractional derivatives as a superset of integer-order derivatives provides a more accurate description of natural phenomena [20–23]. Various applications of these derivatives have been found to solve problems such as mathematical biology [24–26], RC-electrical circuits [27], heat [28, 29], and fractal boundary value [30]. The fractional derivative of known equations such as the Fokker–Plank has better described the velocity of a particle in Brownian motion [31]. Solving nonlinear fractional models has been conducted even when the chaotic behavior has been studied [32–34]. Further, some attempts have been made to use the capabilities of fractional models combined with the chaotic properties of nonlinear systems to improve the designed S-boxes [35].

Optimization has been used to achieve stronger S-box. Optimization methods such as ant colony optimization [36], genetic algorithm optimization [37], firefly (FA) optimization [38], and teaching–learning-based optimization (TLBO) [39] have been used so far to increase the safety of designed S-boxes.

Russell Eberhart and James Kennedy (1995) devised particle swarm optimization [40]. PSO is a method based on swarm intelligence and solves optimization problems using optimized social behavior. The simplicity and simple implementation of PSO make it useful for signal processing and machine learning, neural networks, heat transfer, and so on [41–43]. In Ref. [44], one-dimensional and two-dimensional logistics and particle swarm optimization (PSO) were used to form the S-box. The random distribution capabilities of nonlinear dynamic systems can be used to reduce the required steps to achieve better results by the PSO method. Such a merger will improve the used PSO method [45].

Due to the increasing development of computational tools, it is necessary to review cryptographic methods and tools. The daily development of virtual network users has been accompanied by the need to increase keyspace. Considering the above-mentioned necessities, attention is paid to the forming and achieving of chaotic systems to meet these needs. Encryption algorithms with the help of S-box based on a strong chaotic system solve these problems while maintaining security. Increasing the complexity of the chaotic system used in the S-box has a significant effect on improving its criteria [46]. In order to improve the keyspace and the usable intervals of the map, another discrete map should be used to design the S-box.

In this study, the examples of Chebyshev polynomial of type one, two, their coupling, and hierarchy of rational-order maps were mentioned. Behavioral differences in these maps were used to improve PSO. Such differences can be seen again in the formation of the chaotic S-box. Finally, the performance of the produced chaotic S-boxes was optimized using the best results of improved PSO with two types of fitness. This study aimed to look for the best results by reviewing the available fitness (objective) functions.

This study is organized as follows: Section 2 presents the model. Section 3 provides the improved PSO. Section 4 includes the design of the S-box. The S-box analysis is indicated in Sect. 5. The final section presents the conclusion.

2 The model

In the previous studies, the polynomial of chaotic maps is proposed [47–49]. The coupling of these maps will also increase the keyspace [50]. Such studies continued with the introduction of the hierarchy of rational-order chaotic maps from the same family of maps [51]. The above-mentioned maps are used in subsequent sections in the generation of improved PSO, chaotic S-box, and optimized chaotic S-box. The use of these ergodic maps increases the speed of achieving the desired model in the optimization.

2.1 One-parameter families of chaotic maps

The ratio of polynomials of degree N is considered as follows:

$$\Phi(x,\alpha)=\frac{\alpha^2F}{1+(\alpha^2-1)F},$$

where F can be substituted with the Chebyshev polynomial of type one for $\Phi_N^{(1)}$ and Chebyshev polynomial of type two for $\Phi_N^{(2)}$. It should be noted that for certain value, the formed polynomial of type two is reduced to the logistic map. Using the first and second types of Chebyshev functions will lead to different types by this polynomial. In addition, the sentences of the odd and even present distinct behaviors. The chaotic interval is for even N and the first type in $\alpha\in(0,N)$, while the second-type chaotic interval occurs in $\alpha\in(1/N,\infty)$ [47]. Besides, for odd N, the chaotic interval is observed in $\alpha\in(1/N,N)$.

The Kolmogorov–Sinai entropy and invariant measures of these maps are calculated in [47]. Three of these maps are given below:

$$\Phi_2^{(1)}=\frac{\alpha^2(2x-1)^2}{4x(1-x)+\alpha^2(2x-1)^2}, \tag{1}$$

$$\Phi_2^{(2)}=\frac{4\alpha^2x(1-x)}{1+4(\alpha^2-1)x(1-x)}, \tag{2}$$

$$\Phi_3^{(1,2)}=\frac{\alpha^2x(4x-3)^2}{\alpha^2x(4x-3)^2+(1-x)(4x-1)^2}. \tag{3}$$

Figure 1 indicates the time series diagram of Chebyshev polynomial of type one and Chebyshev polynomial of type two. The sensitivity of the maps to the partial change of their initial values is seen in these figures. In all of three figures, the control parameter $\alpha = 0.75$ is set, and the initial condition changes. Figure 1.1 is plotted for $x_0 = 0.705$ and $x_0 = 0.706$, indicating that despite a slight change in the initial value of x_0, time series diagrams are obtained by a completely different way. Figure 1.2 is drawn for $x_0 = 0.307$ and $x_0 = 0.308$, and again with a slight change in the initial value of x_0, we see two different time series charts. We plotted Fig. 1.3 for $x_0 = 0.88899$ and $x_0 = 0.88891$,

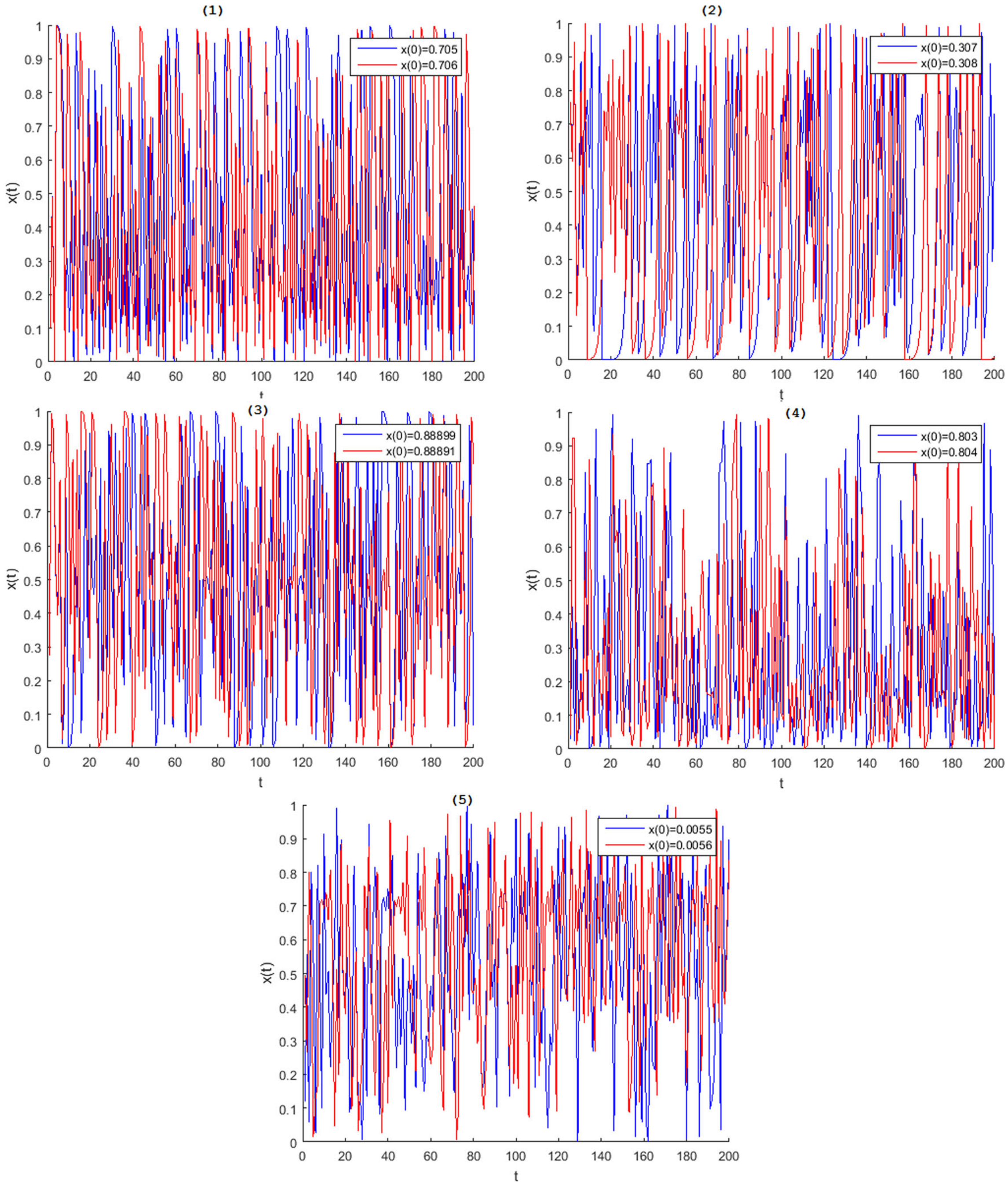

Fig. 1 The sensitivity of the chaotic map to initial conditions (1) $\Phi_2^{(1)}(x,\alpha)$ for $x_0 = 0.705$ and $x_0 = 0.706$, (2) $\Phi_2^{(2)}(x,\alpha)$ for $x_0 = 0.307$ and $x_0 = 0.308$, (3) $\Phi_3^{(1,2)}(x,\alpha)$ for $x_0 = 0.88899$ and $x_0 = 0.88891$ where the control parameter $\alpha = 0.75$, (4) the chaotic coupled map lattices for $x_0 = 0.803$ and $x_0 = 0.804$ where the control parameter $a_1 = 1.5$, $a_2 = 2.4$, $N_1 = 6$, $N_2 = 10$, $\varepsilon = 0.4$ and $y_0 = 0.756$, (5) hierarchy of rational-order chaotic maps for $x_0 = 0.0055$ and $x_0 = 0.0056$ where the control parameter $a_1 = 2.61$ and $a_2 = 3.168$

where the difference between the time series charts for minor changes in x_0 is evident.

Dependence on control parameter is quantified by Lyapunov exponent. In other words, the Lyapunov exponent of the chaotic system shows that neighboring orbits are exponentially separated. Positive Lyapunov exponent shows the chaotic area of the parameter and the measurability of system. In addition, a possible way to describe the keyspace may be in terms of positive Lyapunov exponents [52]. Lyapunov exponent curves for the maps of Eqs. 1, 2, 3 are shown in Figs. 2(1,2,3), respectively. It was obvious that in the various maps, the chaotic domains were different.

2.2 The chaotic coupled map lattices

The chaotic coupled map lattices were introduced and used in cryptography [50]. The form of these maps is as follows:

$$\Phi_{coupled}(X_{n+1}, Y_{n+1}) = \begin{cases} X_{n+1} = (1-\varepsilon)f_1(x_n) + \varepsilon f_2(y_n), \\ Y_{n+1} = (1-\varepsilon)f_1(y_n) + \varepsilon f_2(x_n). \end{cases} \tag{4}$$

Here, the strength of the coupling is shown with ε. f_1 and f_2 are the chaotic trigonometric maps:

$$\begin{cases} f_1(x_n, a_1) = \frac{1}{a_1^2}\tan^2\left(N_1 \arctan\left(x_{n-1}^{\frac{1}{2}}\right)\right), \\ f_2(x_n, a_2) = \frac{1}{a_2^2}\cot^2\left(N_2 \arctan\left(x_{n-1}^{-\frac{1}{2}}\right)\right), \end{cases}$$

where a_1 and a_2 represent control parameters. Time series and Lyapunov exponent curve are shown in Figs. 1.4, 2.4, respectively. Figure 1.4 is plotted for constant parameter $a_1 = 1.5$, $a_2 = 2.4$, $N_1 = 6$, $N_2 = 10$, $\varepsilon = 0.4$ and $y_0 = 0.756$ as well as variable initial value $x_0 = 0.803$ and $x_0 = 0.804$.

2.3 Hierarchy of rational-order chaotic maps

Hierarchy of rational-order chaotic maps was introduced in [51]. Numerical results and analytical calculations for Lyapunov exponent and Kolmogorov–Sinai entropy were obtained [51]. An example of this maps is:

$$x_{n+1} = \frac{a_1}{a_2} \times \frac{1-3x_n^2}{3x_n - x_n^3} \pm \sqrt{1 + \left(\frac{a_1}{a_2} \times \frac{1-3x_n^2}{3x_n - x_n^3}\right)^2}, \tag{5}$$

where a_1 and a_2 represent control parameters. Figure 1.5 shows the corresponding time series diagram of dynamical system Eq. 5. Figure 1.5 is plotted for constant control parameter $a_1 = 2.61$ and $a_2 = 3.168$ as well as variable initial value $x_0 = 0.03$ and $x_0 = 0.030001$. Lyapunov exponent curve is shown in Fig. 2.5.

3 Improved PSO

Difficulty in well adjusting the global and local search capabilities and the possibility of being locked into stagnation are some of the problems that conventional PSO faces [53]. In order to solve this problem, the PSO algorithm was improved by chaotic behaviors. These algorithms included chaos initialization or updating step by chaotic map [45]. This section describes the PSO steps and how to use chaotic maps to determine its initial population and update its particle location and velocity. The best result of this section was used as an improved PSO for later designing of the S-box in sect. 4.

3.1 PSO algorithm

In PSO, the swarm comprises particles with position and speed, each one showing a possible solution in the optimization problems. PSO algorithms are divided into types such as the linearly decreasing inertia weight PSO (LDwPSO) [54], the nonlinearly decreasing inertia weight PSO [55], and chaotic inertia weight PSO [56], in terms of how the w parameter changes. This study used the linearly decreasing inertia weight PSO (LDwPSO). The steps of the PSO algorithm are as follows:

Step 1 Defining the problem parameters
Cost function, number of decision variables (for example:nVar = 10), size of decision variables matrix (for example:VarSize = [1 nVar]), lower bound of variables (for example:VarMin = −10;) and upper bound of variables (for example:VarMax = 10) is determined in this step.

Step 2 Number the PSO Parameters
Maximum number of iterations (for example:MaxIt = 1000), population size (for example:nPop = 100), inertia weight (for example:w = 1), inertia weight damping ratio (wdamp=0.99), personal learning coefficient (for example:c1 = 1.5), global learning coefficient (for example:c2 = 2.0) is determined.

Step 3 Having initial population production using unifrnd function and call cost function for all primary particles and calculation of personal and global best for this population.

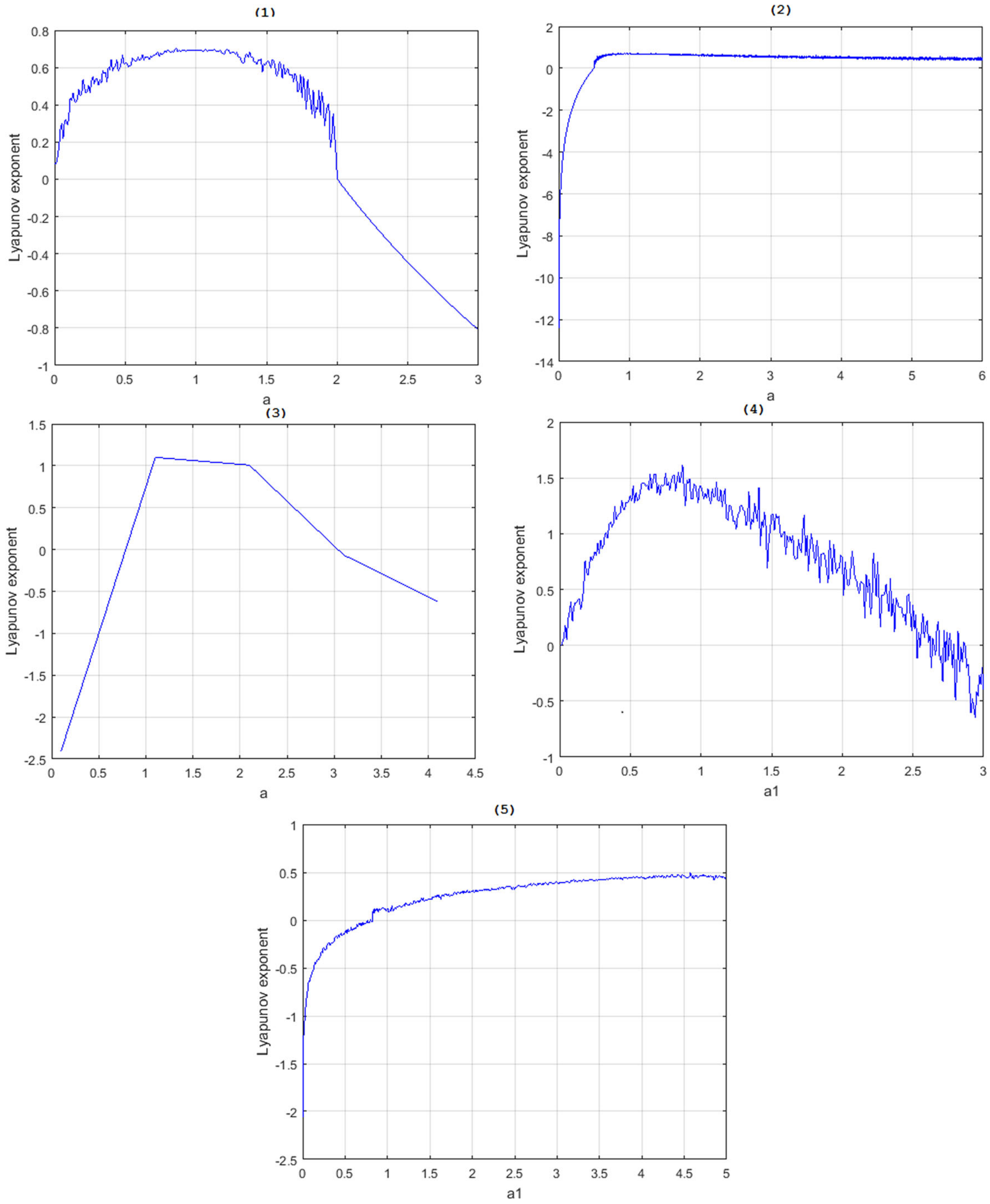

Fig. 2 The variation of the Lyapunov characteristic exponent (1) $\Phi_2^{(1)}(x, \alpha)$ in terms of parameters $a = \alpha$, (2) $\Phi_2^{(2)}(x, \alpha)$ in terms of parameters $a = \alpha$, (3) $\Phi_3^{(1,2)}(x, \alpha)$ in terms of parameters $a = \alpha$, (4) the chaotic coupled map lattices in terms of parameters $a1$, (5) hierarchy of rational-order chaotic maps in terms of parameters $a1$

Step 4 Updating the speed and position of its jth dimension at iteration t of each particle i using the following relationships and local and global search to achieve the best solution.

$$\begin{aligned} V_{i,j}(t+1) =& wV_{i,j}(t) + (c1)(r1)(BestX_{i,j}(t) - X_{i,j}(t)) \\ & + (c2)(r2)(GlobalBest(t) - X_{i,j}(t)), \end{aligned} \quad (6)$$

$$X_{i,j}(t+1) = X_{i,j}(t) + V_{i,j}(t+1), \quad (7)$$

where $V_{i,j}(t)$ represents a velocity of particle i at iteration t; $X_{i,j}(t)$ represents a position of i particle at iteration t; r1 and r2 indicate two random number between (0,1); $BestX_{i,j}(t)$ indicates the local best particle i in all swarm; and $GlobalBest(t)$ indicates the leader of the swarm or global best position of all population.

Well-known benchmark function (minimization) was used to test the effectiveness of the given algorithm. Sphere function is:

$$F_{\text{Sphere}} = \sum_{i=1}^{n} x_i^2,$$

where $-10 \le x_i \le 10$. The minimum of this function was zero. In optimization, the goal was to get a minimum or maximum of a function. Applying the PSO algorithm with 1000 repetitions reached the number 10^{-270}, which was very close to zero. The result for sphere cost function is shown in Fig. 3. As can be seen, cost function approached the minimum value of the sphere function as the iteration increased. In the next subsections, this study attempted to get a better number in the same repetitions. Since the minimum of the sphere function was zero, the better the optimization result, the closer it was to this minimum. PSO works with the production of the initial population to make the S-box perform better. In other words, it improves the S-box.

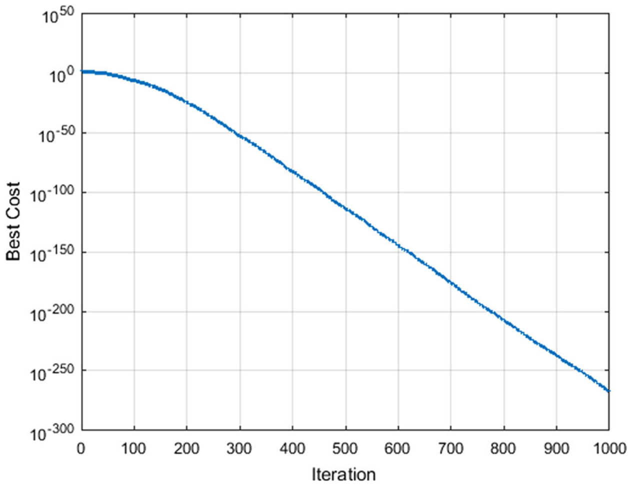

Fig. 3 The variation of the cost function (sphere) for PSO. This curve is plotted with number of decision variables(nVar = 10), lower bound of variables(VarMin = −10), upper bound of variables(VarMax = 10), maximum number of iterations(MaxIt = 1000), population size (for example:nPop = 100), inertia weight (for example:w = 1), inertia weight damping ratio(wdamp = 0.99), personal learning coefficient(c1 = 1.5), and global learning coefficient(c2 = 2.0)

3.2 PSO initial population determination with chaotic map

All of the families of chaotic maps in Sect. 2 were used separately for the initialization value of PSO. These values for each map were selected based on their chaotic domain (positive Lyapunov exponent). Due to the similarity in quasi-code, this study included only one of them in this section, while the recorded results were compared for all maps.

In this subsection, the values of X in Eqs. 1, 2, 3, 4 and 5 are normalized in the range [0,1] to each decision variable in the n-dimensional space of optimization problem. By considering Lyapunov exponent curves, the parameters used here were $\alpha = 0.75$ for Eqs. 1, 2, 3, $N_1 = 6, N_2 = 10, a_1 = 1.5, a_2 = 2.4, \varepsilon = 0.4$ for Eq. 4, and $a_1 = 2.61, a_2 = 3.168$ for Eq. 5. The function optimization problem with the continuous variable was defined as follows.

This study minimized $f(x)$ (cost function) by finding $X = [x_1, x_2, \ldots, x_n]$. X represents the decision solution vector consisting of n variables, x_i, bounded by lower (L_i) and upper limits (U_i). PSO initial population determination procedure with chaotic map can be illustrated:

Step 1 Entering the initial conditions chaotic map. Set $nPop = 100, x(1)$, for Eq. 1.

Step 2 Initializing the particle swarm position X by map function by Eq. 1:

```
Input:the initial conditions and control parameters
of the chaotic map and number of random numbers required (N)
Output:vector of random numbers required
function Xn=map(x,\alpha,N)
Create random initial conditions (x0)
Xn = [];
for i = 1:10000
 Iterate chaotic map
end
for i = 1:N

 Iterate chaotic map

 Converting the generated numbers from iterating map to
 interval 0 and 1(x=mod(x,1);).

 Converting the generated numbers to interval lower bound of variables
 (for example:VarMin=-10;) and upper bound of variables
 (for example:VarMax= 10).

  Xn = [Xn x]
end
end
```

The results for all the maps are shown in Fig. 4. As can be seen in all figures, the use of a chaotic map instead of unifrnd function had no significant effect on the cost function.

Step 2 Performing chaotic search by updating of the speed and position of each particle by using a subfunction derived from Eqs. 1–5 instead of $r1$ and $r2$:

```
Input:the initial conditions and control parameters
of the chaotic map and number of random numbers required (N)
Output:vector of random numbers required
function Xn=nafis(x,\alpha,N)
Create random initial conditions (x0)
Xn = [];
for i = 1:100
 Iterate chaotic map
end
for i = 1:N

 Iterate chaotic map

 Convert the generated numbers from iterating map to
 interval 0 and 1 (x=mod(x,1);).

 Xn = [Xn x]
end
end
```

3.3 Updating velocity with chaotic maps in PSO

As mentioned in the initialization value, all of the proposed maps in Sect. 2 were used to update velocity of PSO.

Update velocity with chaotic maps in PSO is as follows:

Step 1 Entering the initial conditions chaotic map. Set $nPop = 100, MaxIt = 1000, x(1)$, for Eq. 1.

The results for all the maps are shown in Fig. 5. All curves tended to a minimum with increasing repetition. Their difference speed toward zero corresponded to the maps. Graph 5.5 reached the number 10^{-300} in the same repetitions of 1000. Hierarchy of rational-order chaotic maps reached the minimum with the least number of steps. As mentioned, Fig. 3 represents the variation in the cost function (sphere) for PSO. Compared to Fig. 5, the created number for the cost function in the same repetitions of 1000 (graph 5) was better 10^{30} times.

4 The S-box design

Using chaos in the design of efficient S-boxes has been recently considered. This section describes the design of the S-box using the proposed chaotic maps. Then, these S-boxes are optimized by using improved PSO.

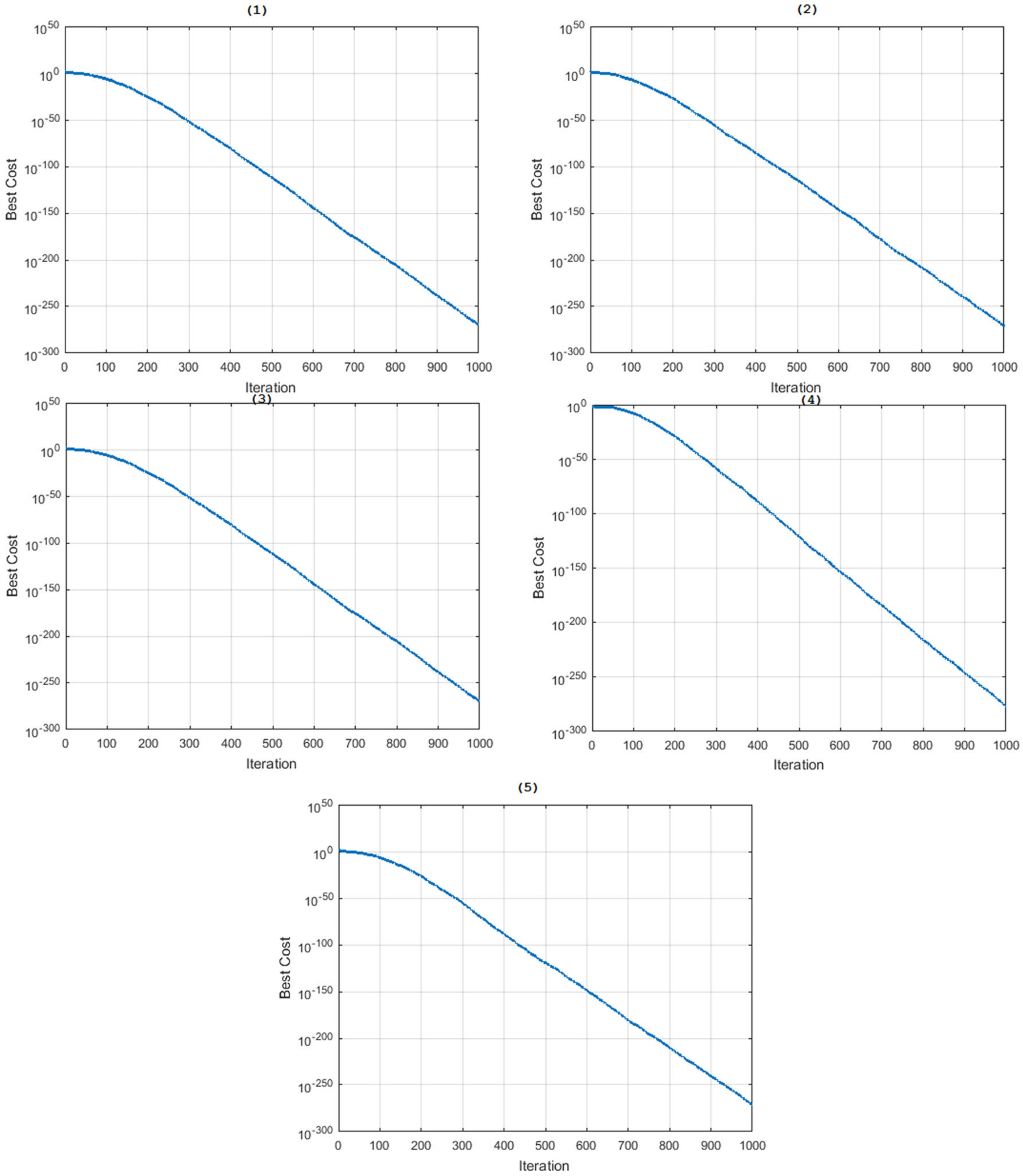

Fig. 4 The variation of the cost function (sphere) for (1) $\Phi_2^{(1)}(x, \alpha)$, (2) $\Phi_2^{(2)}(x, \alpha)$, (3) $\Phi_3^{(1,2)}(x, \alpha)$, (4) the chaotic coupled map lattices, (5) hierarchy of rational-order chaotic maps. These curves are plotted with number of decision variables(nVar $= 10$), lower bound of variables(VarMin $= -10$), upper bound of variables(VarMax $= 10$), Maximum number of iterations(MaxIt $= 1000$), population size (for example:nPop $= 100$), inertia weight (for example:w $= 1$), inertia weight damping ratio(wdamp $= 0.99$), personal learning coefficient(c1 $= 1.5$) and global learning coefficient(c2 $= 2.0$)

4.1 The chaotic S-box

Key is one of the fundamental pillars of any cryptography. Every algorithm is as secure as its key. For high levels of security, the key space size should not be less than 2^{100} [57, 58]. The keyspace provided by the introduced maps to produce the S-box was vast. The order of complexity for decoding in our proposed maps was calculated as follows:

1- for the Chebyshev polynomial

$$T(\alpha, x_0) = \theta(\alpha \times x_0),$$

2- for the maps of Eq. 4

$$T(N_1, N_2, a_1, a_2, \varepsilon, x_0, y_0) = \theta(N_1 \times N_2 \times a_1 \times a_2 \times \varepsilon \times x_0 \times y_0),$$

3- for the maps of Eq. 5

$$T(a_1, a_2, x_0) = \theta(a_1 \times a_2 \times x_0).$$

As an example in the Chebyshev polynomial of type one to determine the keyspace, the keyspace will be over than 10^{32} if the accuracy is 10^{-16} by considering the map interval (0, 1) and chaotic region of the control parameter (0,2).

This keyspace was affected by the chaotic range of control parameters and their number. This space is more than 10^{112} in the chaotic coupled map (Eq. 4) due to the increase in the number of control parameters $(x_0, y_0 \in (0,1), \varepsilon \in (0,1), a_1, a_2 \in (0.5, 2), N_1$ and $N_2)$. Hierarchy of rational-order chaotic maps was examined in the same way.

These spaces could resist all types of brute-force attacks. The best keyspace was for the chaotic coupled map lattices (Eq. 4).

From the mathematical perspective, an $n * m$ S-box was a nonlinear mapping $S : V_n \rightarrow V_m$, where V_n and V_m represent the vector spaces of n, m elements from $GF(2)$. The steps for creating S-box are shown as follows:

Step 1 Entering the initial conditions and control parameters by considering Lyapunov exponent curves (Fig. 2).

Step 2 Repeating the map to 1000000 to pass the transition state.

Step 3 Repeating the map to 700000 times and selecting the last number *x*(*f*).

Step 4 Getting the first S-box from the following equation.

$$S(1) = x(f) * 10^5 mod 256$$

Step 5 Repeating the map and selecting the last number *x*(*f*).

Step 6 The next S-box numbers were obtained from the following equation.

$$S(i) = x(f) * 10^5 mod 256$$

Step 7 If *S*(*i*) is in the S-box, the process continues from step 5.

Step 8 Putting the obtained S(i) in the S-box table.

Step 9 Repeating steps 5 through 8 until all 256 number S-boxes are filled.

The flowchart for creating S-box is shown in Fig. 6. Table 1 shows the best S-boxes generated from the maps of Eqs. 1, 2, 3, 4 and 5 presented in the model. This S-box is produced by the map of Eq. 3.

4.2 The chaotic S-box design with improved PSO

By considering the result of Sect. 3, this study improved PSO with the hierarchy of rational-order chaotic maps to optimize S-box. Various objective (fitness or cost) functions can be introduced in S-box optimization, including nonlinearity of the box and greater randomness of the box numbers. In this section, both cases are discussed in separate subsections.

4.2.1 The chaotic S-box design with improved PSO and nonlinearity fitness function

Many studies have considered nonlinearity as an objective function of optimization [37]. By considering this, the steps of the algorithm are as follows:

Step 1 Entering initial conditions and control parameters related to the chaotic map of the objective function.

Step 2 Entering improved PSO parameters.

Step 3 Having initial population production using unifrnd function.

Step 4 Creating of S-box with chaotic map (Sect. 4.1) and calculation of nonlinearity for all primary particles and calculation of personal and global best for this population.

Step 5 Updating the speed and position of each particle using the hierarchy of rational-order chaotic maps and local and global search to achieve the best solution (subsection 3.3).

Step 6 Saving the best nonlinearity and related S-box.

The flowchart for the optimized S-box design with improved PSO and nonlinearity fitness function is shown in Fig. 7. The results of all improved PSO algorithms for all

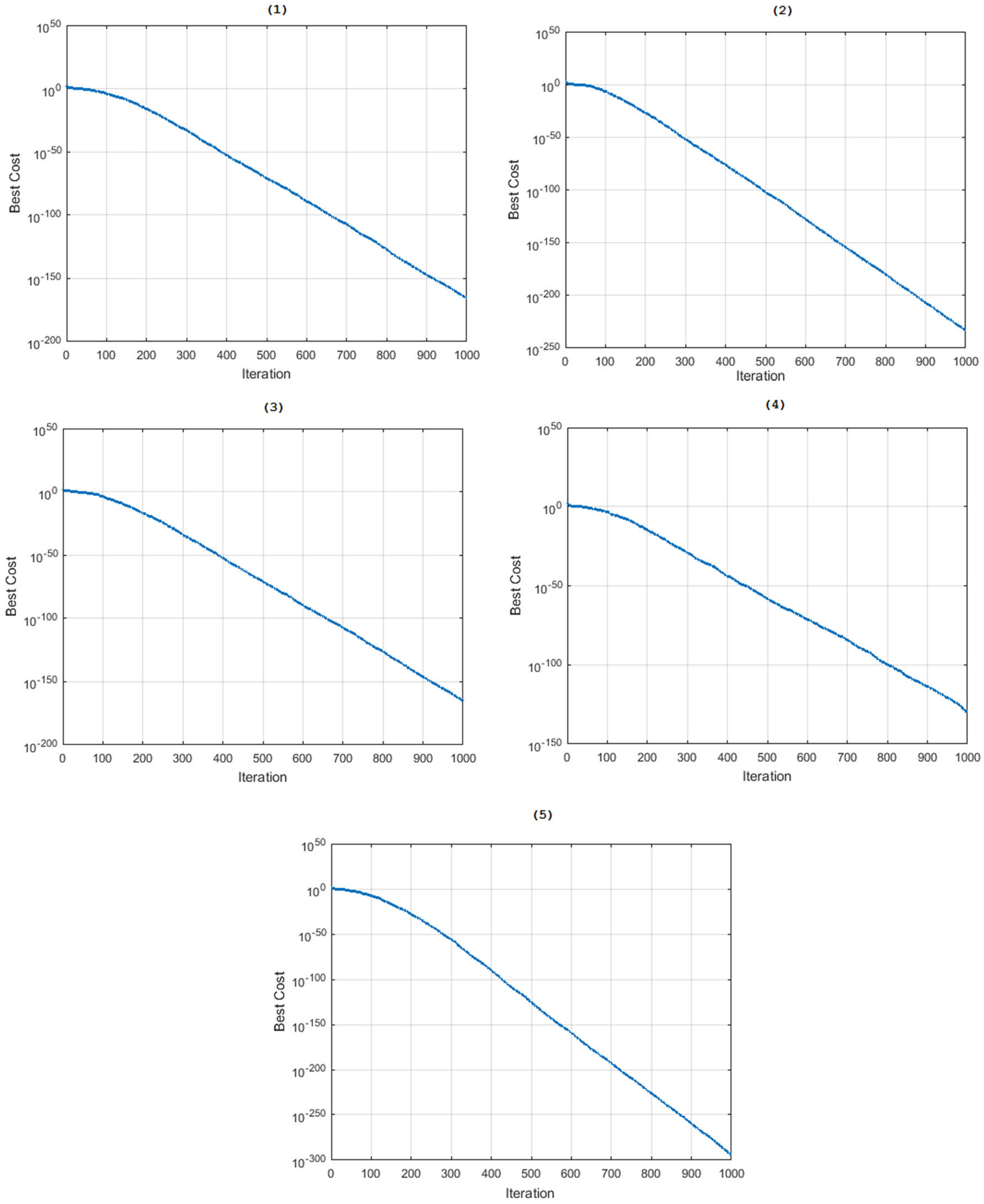

Fig. 5 The variation of the cost function (sphere) for (1) $\Phi_2^{(1)}(x,\alpha)$, (2) $\Phi_2^{(2)}(x,\alpha)$, (3) $\Phi_3^{(1,2)}(x,\alpha)$, (4) the chaotic coupled map lattices, (5) hierarchy of rational-order chaotic maps. These curves are plotted with number of decision variables(nVar $= 10$), lower bound of variables(VarMin $= -10$), upper bound of variables(VarMax $= 10$), maximum number of iterations(MaxIt $= 1000$), population size (for example:nPop $= 100$), inertia weight (for example:w $= 1$), inertia weight damping ratio (wdamp $= 0.99$), personal learning coefficient(c1 $= 1.5$), and global learning coefficient(c2 $= 2.0$)

the formed S-boxes in Sect. 4.1 are given in Fig. 8. The best-created S-boxes are shown in Table 2. This S-box is produced by the map of Eq. 4.

Fig. 6 S-box creation algorithm

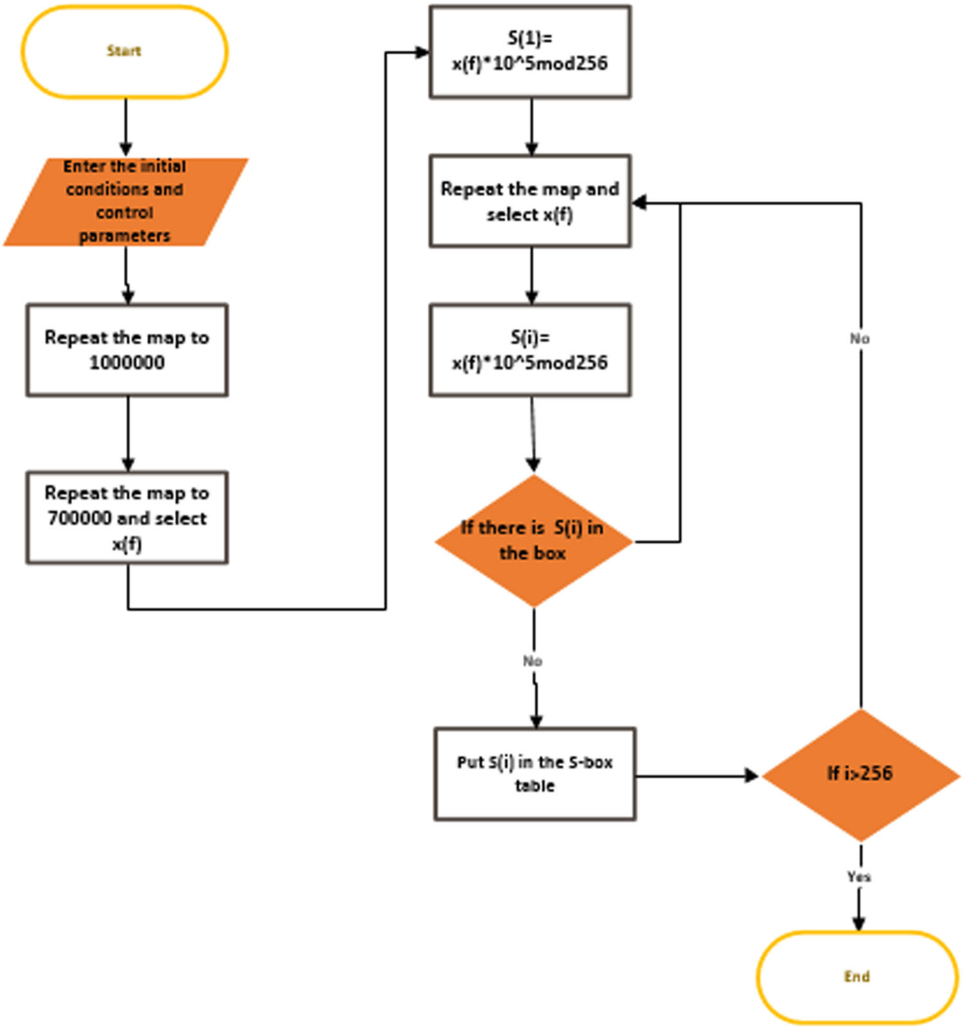
Start
Enter the initial conditions and control parameters
Repeat the map to 1000000
Repeat the map to 700000 and select x(f)
S(1)= x(f)*10^5mod256
Repeat the map and select x(f)
S(i)= x(f)*10^5mod256
If there is S(i) in the box
No
Put S(i) in the S-box table
If i>256
No
Yes
End

Table 1 Proposed S-box (16*16 matrix) from $\Phi_3^{(1,2)}(x, \alpha)$ in Eq. 3 for $\alpha = 0.75$,

50	18	92	216	167	118	57	64	103	235	27	86	72	197	213	232
46	204	153	14	199	246	194	130	47	44	79	24	211	196	220	20
171	137	143	222	203	248	239	243	138	125	156	17	255	174	185	132
73	100	116	84	215	12	173	210	37	177	91	74	60	198	52	139
5	66	161	96	122	111	81	136	172	170	49	42	113	146	99	48
183	95	1	151	140	227	160	195	226	157	85	71	206	148	128	182
256	230	11	39	207	9	225	158	212	97	15	168	217	114	119	62
93	205	36	34	51	188	35	228	250	251	26	56	190	134	112	75
218	124	149	117	135	223	141	200	166	101	238	90	89	120	147	76
159	55	192	25	187	245	229	63	176	202	88	152	67	6	247	244
19	186	241	32	53	221	165	191	214	13	123	3	10	115	110	129
163	133	87	219	175	107	83	208	145	164	82	7	94	142	69	68
45	240	109	254	249	155	121	16	40	54	169	2	233	224	65	231
8	150	189	80	236	98	162	179	105	58	178	181	4	180	21	30
77	234	127	43	59	22	78	29	144	104	33	41	23	70	201	193
184	108	106	31	131	126	252	154	237	38	209	242	253	61	102	28

4.2.2 The chaotic S-box design with improved PSO and mono-bit frequency test fitness function

Converting the outputs of the chaotic system to zero-one strings is one of the chaotic-based randomness production methods (entropy source) [59]. Here, six introduced maps are used to generate random numbers. The simplest and most effective ideals for statistical random measurement are the Chi-square test. The following formula was used to calculate:

$$\chi^2 = \frac{(\mathbf{n}_0 - \mathbf{n}_1)^2}{\mathbf{n}},$$

where n_0, n_1 represent the number of 0's and 1's in bit sequences, respectively [60]. This test is mono-bit frequency test. P-value was computed from the bit sequence for this test. If the P-value is greater than a predefined threshold 0.01, then sequence will pass the test successfully. Here, like Reference [61], the mono-bit test was selected as the optimization objective function. The steps of the algorithm are as follows:

Step 1 Entering initial conditions and control parameters related to the chaotic map of the objective (fitness) function.
Step 2 Entering improved PSO parameters.
Step 3 Having initial population production using unifrnd function.
Step 4 Creating S-box with chaotic map (Sect. 4.1).
Step 5 Having the generated S-box as a 16 ∗ 16 matrix. The rows were put together, and a 1 ∗ 256 matrix was obtained.

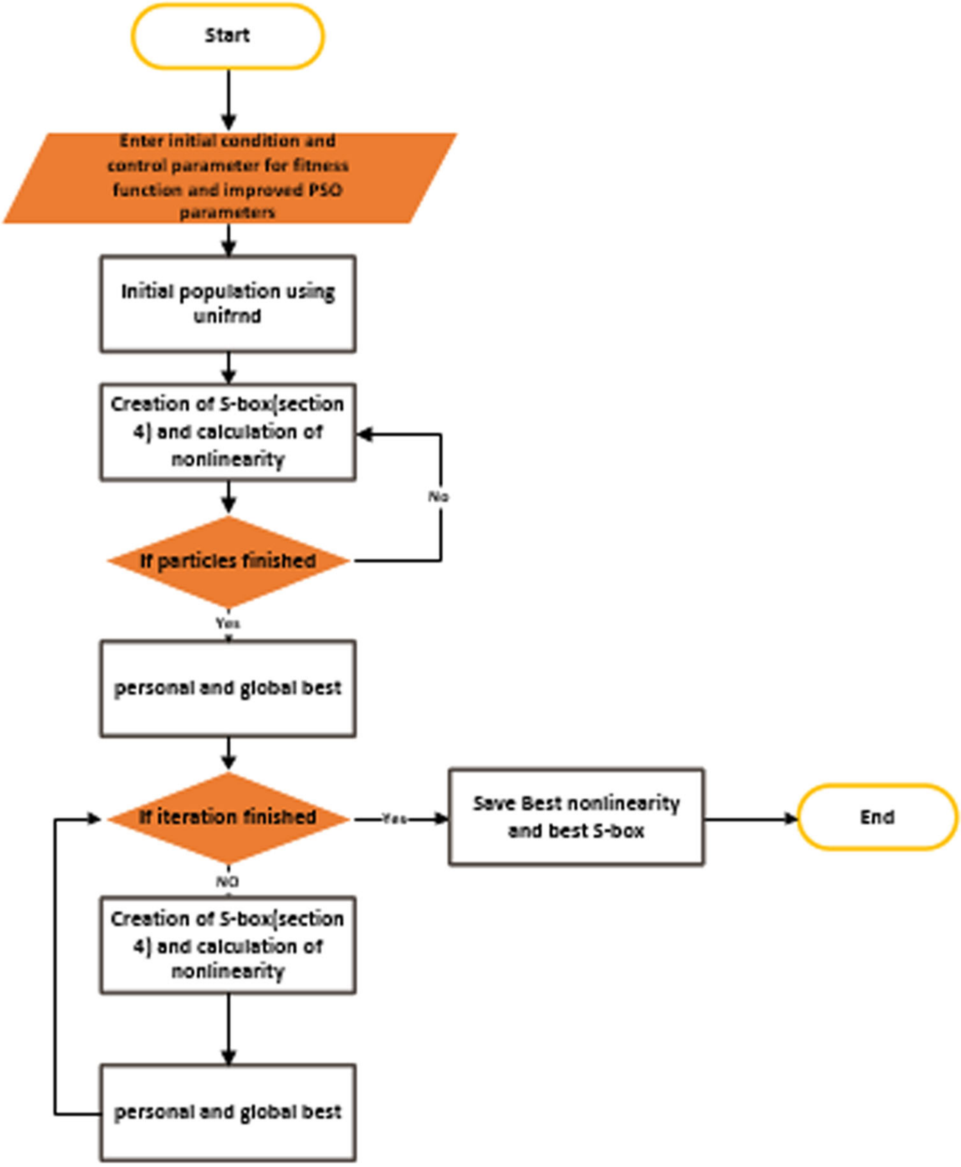
Start
Enter initial condition and control parameter for fitness function and improved PSO parameters
Initial population using unifrnd
Creation of S-box(section 4) and calculation of nonlinearity
No
If particles finished
Yes
personal and global best
If iteration finished
Yes
Save Best nonlinearity and best S-box
End
NO
Creation of S-box(section 4) and calculation of nonlinearity
personal and global best

Fig. 7 Optimized S-box creation algorithm with nonlinearity fitness function

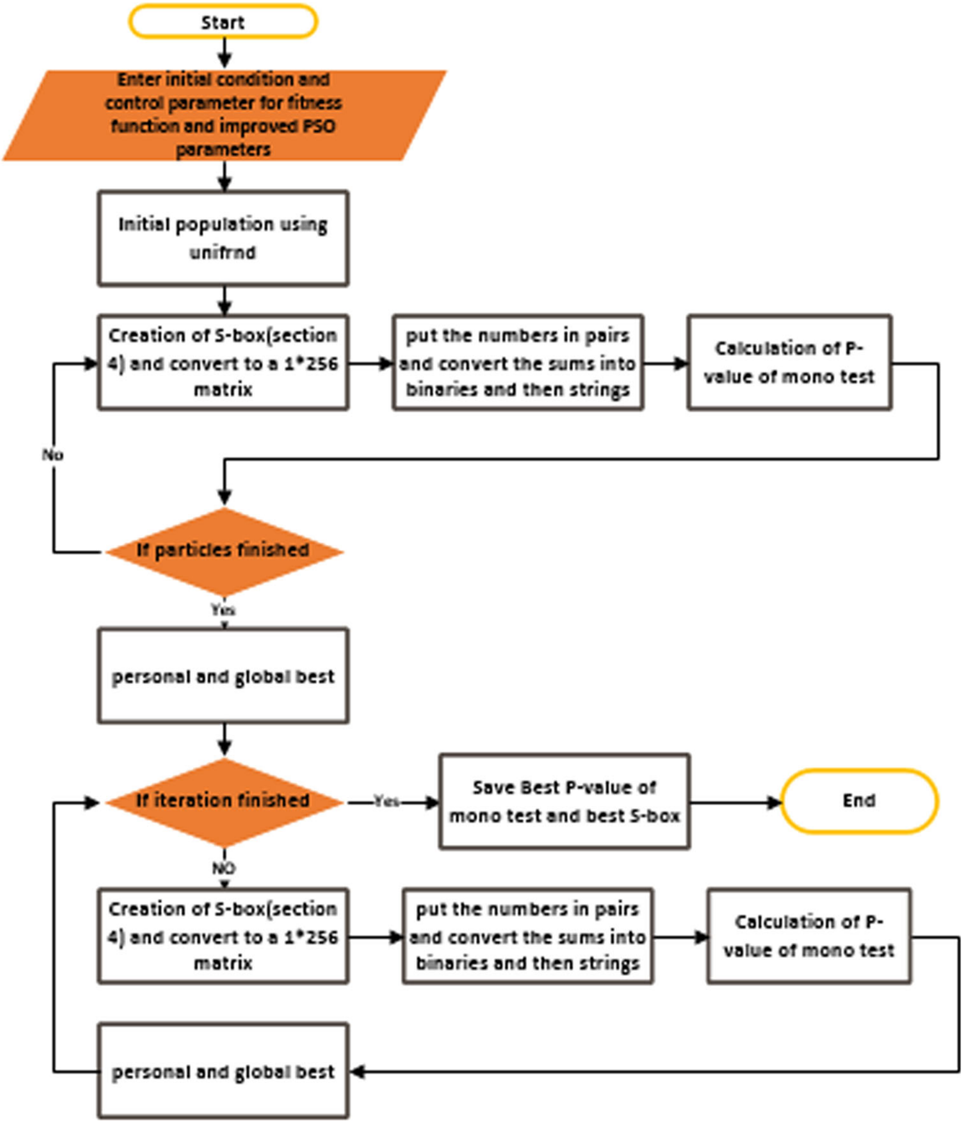
Start
Enter initial condition and control parameter for fitness function and improved PSO parameters
Initial population using unifrnd
Creation of S-box(section 4) and convert to a 1*256 matrix
put the numbers in pairs and convert the sums into binaries and then strings
Calculation of P-value of mono test
No
If particles finished
Yes
personal and global best
If iteration finished
Yes
Save Best P-value of mono test and best S-box
End
NO
Creation of S-box(section 4) and convert to a 1*256 matrix
put the numbers in pairs and convert the sums into binaries and then strings
Calculation of P-value of mono test
personal and global best

Fig. 8 Optimized S-box creation algorithm with *P*-value of mono-test fitness function

Step 6 Putting the numbers in pairs and converting the sums into binaries and then strings.

Step 7 Calculating mono-bit frequency test for all primary particles and calculation of personal and global best for this population.

Step 8 Updating the speed and position of each particle using the hierarchy of rational-order chaotic maps and local and global search to achieve the best solution (subsection 3.3).

Step 9 Saving the best *P*-value of mono-bit frequency test and related S-box.

The flowchart for the optimized S-box design with improved PSO and *P*-value of mono-bit frequency test (fitness function) is shown in Fig. 9. The results of all improved PSO optimizations for all the formed S-boxes in Sect. 4.1 are given in Fig. 10. The best-created S-box is shown in Table 3. This S-box is produced by the map of Eq. 2.

5 The S-box analysis

Subsequently, the following important tests were applied to the generated S-box in Tables 1, 2, 3:

nonlinearity (NL), strict avalanche criterion (SAC), bit independence criterion (BIC), linear approximation probability (LP), and differential approximation probability (DP).

Table 2 Offered optimized S-box (16*16 matrix) from the chaotic coupled map lattices (Eq. 4) with nonlinearity fitness for $N_1 = 6, N_2 = 10, a_1 = 1.5, a_2 = 2.4, \varepsilon = 0.4$

162	179	253	51	43	256	236	93	230	221	227	56	67	63	4	60
176	155	90	52	42	97	50	166	27	147	152	29	45	233	83	31
18	13	40	111	239	199	214	119	238	153	118	91	224	55	150	177
114	151	106	66	127	87	128	196	8	164	172	124	209	6	149	48
250	144	30	20	223	16	235	245	163	255	69	3	158	184	92	193
173	178	213	34	129	187	9	215	12	17	80	38	204	243	191	241
248	108	154	32	64	109	217	156	19	142	170	136	226	62	251	82
123	203	88	202	195	103	102	110	2	14	161	37	249	26	240	130
33	116	89	229	115	59	122	21	107	200	120	133	145	49	72	212
141	7	254	185	125	5	207	206	197	171	53	79	126	134	190	1
218	225	183	39	15	68	169	70	85	10	11	105	231	139	232	188
222	100	132	186	112	138	47	84	180	71	131	46	194	73	65	101
77	148	205	44	41	137	198	244	104	146	220	135	247	94	86	181
242	23	113	182	117	210	24	157	35	121	61	57	143	192	96	54
160	78	98	167	175	234	189	174	252	74	22	246	95	165	208	168
58	36	159	211	216	25	99	81	201	76	75	237	140	28	228	219

5.1 Nonlinearity

The degree of linearity of the S-box was given by nonlinearity test. Since the affine functions were weak in terms of cryptography, the similarity of the Boolean function variable of S-box was measured with the affine variable. The nonlinearity value was calculated using the following equation:

$$N = 2^{n-1} - \frac{1}{2}\max_{a\in B^n}\left|\sum_{x\in B^n}(-1)^{f(x)+a.x}\right|,$$

where $B = \{0,1\}$, $f : B^n \to B$, $a \in B^n$ and $a.x$ represents the dot product between a and x (see [62], for example). The highest theoretical limit of nonlinearity is 120 [7]. The best received value is 112 for the AES S-box [4]. For example, the nonlinearity of eight Boolean functions of offered optimized S-box for the map of Eq. 4 with nonlinearity fitness was 108, 104, 108, 106, 104, 108, 106, and 108. Therefore, the maximum and minimum and average values were 108, 104, and 106.5, respectively. This average was better than all the averages obtained with the chaotic S-box and their optimized S-box. The maximum and minimum and average values and comparing it with the results of previous work are given in Table 4.

5.2 Strict avalanche criterion (SAC)

Webster and Tavares introduced another important measure (as strict avalanche criterion) that describing when one bit in the input of Boolean function changed, half of the output bits should be changed [1]. The dependence matrix for all the proposed S-boxes was calculated based on the reference [1] method. Table 5 shows that the minimum, maximum, and average values of dependence matrices. Further, comparing its results with the results of previous work is presented in this table. The best-obtained average value for SAC was 0.5. All obtained values for the proposed S-boxes were appropriate and very close to 0.500. The received value for the offered optimized S-box generated by the map of Eq. 5 with the *P*-value fitness(0.499512) was the best-obtained value. The dependence matrix of this S-box is given in Table 6. The results were confirmed and improved compared to previous method.

5.3 Bit independence criterion (BIC)

Webster and Tavares defined a desirable feature for any encryption transformation for S-box analysis, called the output bits independence criterion (BIC)[1]. The independence of the avalanche vectors sets was measured by the BIC. If one changed the inverse of input single bits, these sets would be created [63]. BIC-nonlinearity and BIC-SAC for all the proposed S-boxes were calculated based on the reference [1] method. Table 7 indicates the average values of BIC-nonlinearity and BIC-SAC. The numerical results of this test and comparing it with the results of previous work are depicted in Table 7. The amounts of BIC-nonlinearity and BIC-SAC were 112 and 0.5 for the AES S-box [4], which were the best-achieved values. The

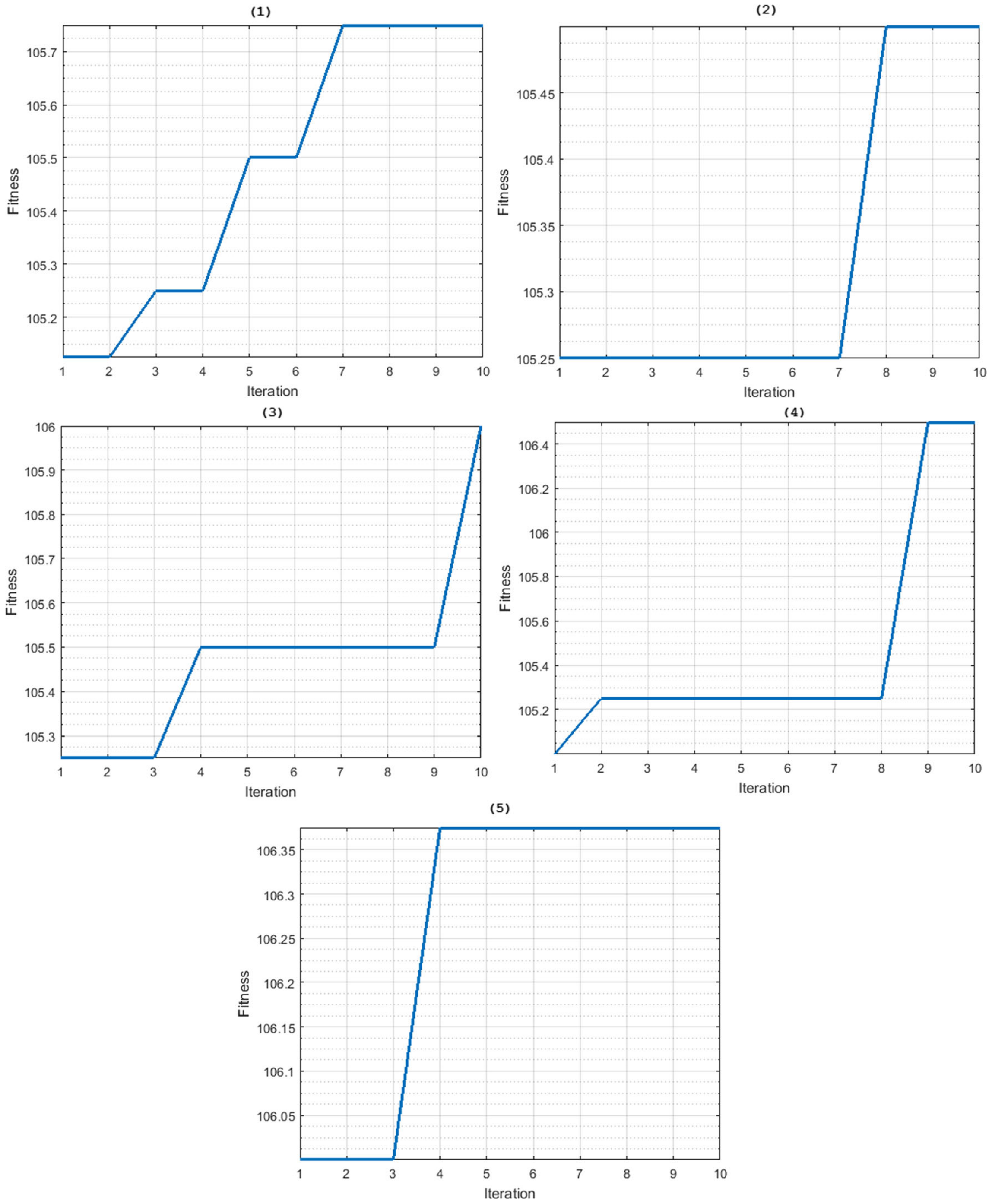

Fig. 9 Best nonlinearity of optimized S-box with nonlinearity fitness function for (1) $\Phi_2^{(1)}(x,\alpha)$ where $\alpha = 0.75$,, (2) $\Phi_2^{(2)}(x,\alpha)$ for $\alpha = 0.75$,, (3) $\Phi_3^{(1,2)}(x,\alpha)$ for $\alpha = 0.75$,, (4) the chaotic coupled map lattices for $N_1 = 6, N_2 = 10, a_1 = 1.5, a_2 = 2.4, \varepsilon = 0.4$, (5) hierarchy of rational-order chaotic maps for $a_1 = 2.61, a_2 = 3.168$. Improved PSO based on hierarchy of rational-order chaotic maps for $a_1 = 2.61, a_2 = 3.168$ is used. Other optimization conditions are: number of decision variables (nVar = 9), lower bound of variables (VarMin = 90), upper bound of variables (VarMax = 120), maximum number of iterations (MaxIt = 10), population size (nPop = 9), inertia weight (w = 1), inertia weight damping ratio (wdamp = 0.99), personal learning coefficient (c1 = 1.5), and global learning coefficient (c2 = 2.0)

obtained average values of BIC-nonlinearity for the proposed S-boxes were between 102.071 and 104.429. The best was for the chaotic coupled map lattices (Eq. 4), being more than all of the obtained values in Table 7 references except for reference [39]. The best-obtained average value of BIC-SAC is 0.49986 for optimized S-box Chebyshev

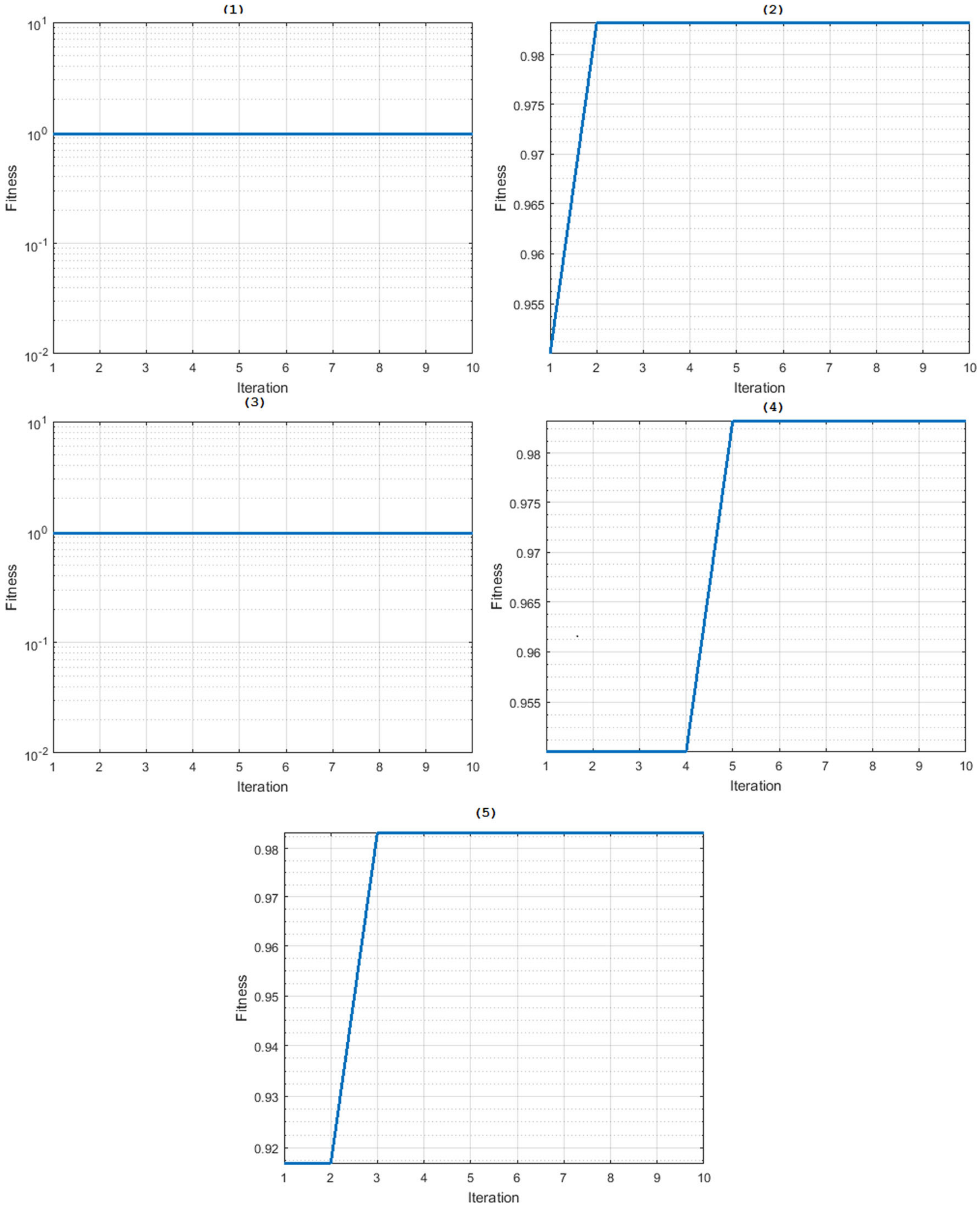

Fig. 10 Best *P*-value of optimized S-box with *P*-value of mono-test fitness function for (1) $\Phi_2^{(1)}(x, \alpha)$ where $\alpha = 0.75,$, (2) $\Phi_2^{(2)}(x, \alpha)$ for $\alpha = 0.75,$, (3) $\Phi_3^{(1,2)}(x, \alpha)$ for $\alpha = 0.75,$, (4) the chaotic coupled map lattices for $N_1 = 6, N_2 = 10, a_1 = 1.5, a_2 = 2.4, \varepsilon = 0.4$, (5) hierarchy of rational-order chaotic maps for $a_1 = 2.61, a_2 = 3.168$. Improved PSO based on hierarchy of rational-order chaotic maps for $a_1 = 2.61, a_2 = 3.168$ is used. Other optimization conditions are: number of decision variables (nVar = 9), lower bound of variables (VarMin = 0.01), upper bound of variables (VarMax = 1), maximum number of iterations (MaxIt = 10), population size (nPop = 9), inertia weight (w = 1), inertia weight damping ratio (wdamp = 0.99), personal learning coefficient(c1 = 1.5), and global learning coefficient (c2 = 2.0)

polynomial of type odd (Eq. 3) with nonlinearity fitness. This value was more than all of the obtained values in Table 7 references. BIC-nonlinearity and BIC-SAC of best proposed S-boxes are given in Tables 8 and 9.

Table 3 Offered optimized S-box (16*16 matrix) from $\Phi_2^{(2)}(x, \alpha)$ in Eq. 2 with *P*-value of mono-test fitness function for $\alpha = 0.75$,

40	19	218	52	58	85	242	110	82	8	84	192	243	122	211	125
43	244	193	185	215	155	92	17	177	234	164	60	201	181	39	150
250	75	123	120	206	183	158	24	121	69	89	106	74	94	37	21
9	255	180	56	169	111	15	129	99	157	46	212	227	135	223	100
30	182	214	33	167	204	104	73	36	118	90	31	160	55	114	45
29	241	109	134	102	221	10	159	146	79	38	113	48	200	213	101
203	42	124	162	117	209	6	252	62	153	11	78	59	139	238	67
186	253	248	108	116	107	237	32	127	1	98	245	63	147	26	189
208	2	198	138	91	14	54	44	136	47	35	178	83	137	5	53
220	128	131	132	191	199	3	7	88	190	231	142	161	86	65	77
27	148	230	210	188	20	168	251	61	96	176	232	225	173	68	228
25	50	184	239	247	81	175	194	156	229	152	72	70	143	49	216
187	222	240	97	249	166	197	71	144	80	76	105	112	28	140	235
151	41	170	154	18	51	12	64	119	256	174	34	126	224	254	133
57	4	163	202	149	196	16	95	141	171	87	233	13	130	103	236
23	226	217	179	207	22	205	246	66	195	165	219	172	145	93	115

Table 4 Nonlinearity values of offered S-box. The nonlinearity of eight Boolean functions of each offered S-box is calculated, and minimum, maximum, and average of them are shown in the table

Nonlinearity	Min	Max	Avg
Offered S-box with Eq. 1	99	106	103.375
Offered S-box with Eq. 2	98	106	102.5
Offered S-box with Eq. 3	96	108	104.75
Offered S-box with Eq. 4	100	108	103.5
Offered S-box with Eq. 5	94	104	101.25
Offered optimized S-box with Eq. 1 (nonlinearity)	102	108	105.75
Offered optimized S-box with Eq. 2 (nonlinearity)	104	107	105.5
Offered optimized S-box with Eq. 3 (nonlinearity)	102	110	105.75
Offered optimized S-box with Eq. 4 (nonlinearity)	104	108	106.5
Offered optimized S-box with Eq. 5 (nonlinearity)	102	111	106.375
Offered optimized S-box with Eq. 1 (*P*-value)	96	108	104
Offered optimized S-box with Eq. 2 (*P*-value)	102	106	104.25
Offered optimized S-box with Eq. 3 (*P*-value)	92	109	102.625
Offered optimized S-box with Eq. 4 (*P*-value)	98	106	102.25
Offered optimized S-box with Eq. 5 (*P*-value)	100	108	103.25
In [9]	98	108	103.2
In [10]	99	106	103.4
In [8]	100	106	103
In [17]	96	106	103
In [39]	104	110	106.5
In [18]	100	109	104.2
In [66]	100	106	103.2
In [73]	102	108	105.2
In [67]	104	110	106
In [69]	104	110	106
In [70]	106	108	106.75
In [61]			106.75
In [71]			106
AES S-box ([4])	112	112	112

Table 5 SAC values of offered S-boxes. Minimum, maximum, and average values of dependence matrices are shown

SAC	Min	Max	Avg
Offered S-box with Eq. 1	0.40625	0.625	0.505859
Offered S-box with Eq. 2	0.390625	0.609375	0.49707
Offered S-box with Eq. 3	0.40625	0.609375	0.499023
Offered S-box with Eq. 4	0.40625	0.609375	0.509277
Offered S-box with Eq. 5	0.375	0.625	0.503906
Offered optimized S-box with Eq. 1 (nonlinearity)	0.4375	0.625	0.512451
Offered optimized S-box with Eq. 2 (nonlinearity)	0.414063	0.601563	0.496094
Offered optimized S-box with Eq. 3 (nonlinearity)	0.421875	0.578125	0.497559
Offered optimized S-box with Eq. 4 (nonlinearity)	0.40625	0.640625	0.503662
Offered optimized S-box with Eq. 5 (nonlinearity)	0.40625	0.601563	0.498291
Offered optimized S-box with Eq. 1 (*P*-value)	0.390625	0.578125	0.504395
Offered optimized S-box with Eq. 2 (*P*-value)	0.390625	0.578125	0.504639
Offered optimized S-box with Eq. 3 (*P*-value)	0.375	0.632813	0.49707
Offered optimized S-box with Eq. 4 (*P*-value)	0.421875	0.59375	0.497559
Offered optimized S-box with Eq. 5 (*P*-value)	0.390625	0.601563	0.499512
In [9]	0.3761	0.5975	0.5058
In [10]	0.4140	0.6015	0.4987
In [8]	0.4218	0.6093	0.5000
In [17]	0.3906	0.6250	0.5039
In [39]	0.4375	0.6406	0.5120
In [18]	0.3906	0.5703	0.4931
In [66]	0.4218	0.5938	0.5048
In [73]	0.4080	0.5894	0.5050
In [67]	0.4218	0.5937	0.5039
In [69]	0.4062	0.6093	0.5012
In [70]			0.4941
In [61]			0.5015
In [71]			0.52881
AES S-box ([4])	0.4531	0.5625	0.5048

Table 6 Dependence matrix of the offered optimized S-box for hierarchy of rational-order chaotic maps (Eq. 5) with the *P*-value fitness

0.484375	0.468750	0.546875	0.546875	0.578125	0.421875	0.546875	0.468750
0.453125	0.515625	0.437500	0.500000	0.484375	0.468750	0.515625	0.468750
0.453125	0.515625	0.562500	0.500000	0.484375	0.593750	0.390625	0.468750
0.562500	0.515625	0.562500	0.421875	0.500000	0.453125	0.484375	0.500000
0.453125	0.578125	0.500000	0.484375	0.515625	0.484375	0.515625	0.562500
0.500000	0.484375	0.546875	0.546875	0.500000	0.484375	0.531250	0.453125
0.468750	0.453125	0.453125	0.593750	0.515625	0.546875	0.500000	0.562500
0.453125	0.515625	0.500000	0.531250	0.562500	0.437500	0.437500	0.421875

5.4 Linear approximation probability (LP)

The maximum value of imbalance in the event between input and output bits is called the linear approximation probability (LP). Mathematical definition of LP [64] is:

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

where a, b represent the input and output masks, and the set x contains all the possible inputs, and 2^n represents the number of its elements. If the S-box has low LP, it can resist linear attacks. The best LP results were from the AES S-box [4]. The LP obtained values were suitable for all suggested S-boxes. The most suitable value was for Chebyshev polynomial of type two (Eq. 2) and the hierarchy of rational-order chaotic maps (Eq. 5) (in optimized with *P*-value fitness). In addition, compared to previous works, it

Table 7 BIC values of offered S-boxes. These values are average

BIC	BIC-SAC	BIC-nonlinearity
Offered S-box with Eq. 1	0.503767	103.679
Offered S-box with Eq. 2	0.500767	103.5
Offered S-box with Eq. 3	0.498326	103.286
Offered S-box with Eq. 4	0.508022	104.429
Offered S-box with Eq. 5	0.501186	102.714
Offered optimized S-box with Eq. 1 (nonlinearity)	0.502651	102.857
Offered optimized S-box with Eq. 2 (nonlinearity)	0.504674	103.5
Offered optimized S-box with Eq. 3 (nonlinearity)	0.49986	103.214
Offered optimized S-box with Eq. 4 (nonlinearity)	0.499512	102.857
Offered optimized S-box with Eq. 5 (nonlinearity)	0.499721	103.893
Offered optimized S-box with Eq. 1 (*P*-value)	0.498954	103.857
Offered optimized S-box with Eq. 2 (*P*-value)	0.501046	102.071
Offered optimized S-box with Eq. 3 (*P*-value)	0.495466	102.964
Offered optimized S-box with Eq. 4 (*P*-value)	0.506766	103.643
Offered optimized S-box with Eq. 5 (*P*-value)	0.503976	103.786
In [9]	0.5031	104.2
In [10]	0.4995	103.3
In [8]	0.5024	103.1
In [17]	0.5010	100.3
In [39]	0.4983	104.57
In [18]	0.4988	103.3
In [66]	0.5009	103.7
In [73]	0.5053	104.2
In [67]	0.5058	103.4
In [69]	0.5003	103.5
In [70]	0.4957	103.5
In [61]	0.5029	104.07
In [71]		100
AES S-box ([4])		112

Table 8 BIC-nonlinearity criterion for the offered S-box for the chaotic coupled map lattices (Eq. 4)

–	106	104	106	108	108	104	104
106	–	96	108	102	102	104	98
104	96	–	108	106	108	106	106
106	108	108	–	102	100	104	104
108	102	106	102	–	102	106	106
108	102	108	100	102	–	106	106
104	104	106	104	106	106	–	104
104	98	106	104	106	106	104	–

was less than references [8, 10, 18, 39, 65, 66], and [67] and similar to reference [17]. Table 10 shows the numerical results and compares the results of previous work.

5.5 Differential approximation probability (DP)

Biham and Shamir introduced a differential cryptanalysis method [68]. This method calculated XOR distribution between input and output bits of S-box was called DP. If this distribution is close between the input and output bits, S-box will be resistant to differential attacks. DP is defined as follows:

$$DP = \max_{\Delta_x \neq 0, \Delta_y} (\#x \in X, f_x \oplus f(x + \Delta_x) = \Delta_y / 2^n),$$

where X shows the set of all possible input values, and 2^n represents the number of its elements. The DP value for a strong S-box should be close to zero. The best result ($DP = 4$) was for AES S-box [4]. All of the obtained values were suitable for the introduced S-boxes. The best DP (10) was for the chaotic coupled map lattices (Eq. 4) (in all recommended S-boxes) and Chebyshev polynomial of type two and hierarchy of rational-order chaotic maps (Eq. 5) (in both optimized modes) and Chebyshev

Table 9 BIC-SAC criteria for the offered optimized S-box for $\Phi_3^{(1,2)}(x, \alpha)$ in Eq. 3 with nonlinearity fitness

–	0.488281	0.511719	0.500000	0.507813	0.500000	0.482422	0.498047
0.488281	–	0.507813	0.519531	0.484375	0.498047	0.488281	0.513672
0.511719	0.507813	–	0.515625	0.521484	0.478516	0.507813	0.482422
0.500000	0.519531	0.515625	–	0.470703	0.521484	0.517578	0.511719
0.507813	0.484375	0.521484	0.470703	–	0.507813	0.490234	0.478516
0.500000	0.498047	0.478516	0.521484	0.507813	–	0.492188	0.498047
0.482422	0.488281	0.507813	0.517578	0.490234	0.492188	–	0.501953
0.498047	0.513672	0.482422	0.511719	0.478516	0.498047	0.501953	–

Table 10 LP values of offered S-boxes

LP	Max
Offered S-box with Eq. 1	0.128906
Offered S-box with Eq. 2	0.140625
Offered S-box with Eq. 3	0.15625
Offered S-box with Eq. 4	0.132813
Offered S-box with Eq. 5	0.132813
Offered optimized S-box with Eq. 1 (nonlinearity)	0.15625
Offered optimized S-box with Eq. 2 (nonlinearity)	0.132813
Offered optimized S-box with Eq. 3 (nonlinearity)	0.132813
Offered optimized S-box with Eq. 4 (nonlinearity)	0.140625
Offered optimized S-box with Eq. 5 (nonlinearity)	0.140625
Offered optimized S-box with Eq. 1 (*P*-value)	0.140625
Offered optimized S-box with Eq. 2 (*P*-value)	0.125
Offered optimized S-box with Eq. 3 (*P*-value)	0.140625
Offered optimized S-box with Eq. 4 (*P*-value)	0.140625
Offered optimized S-box with Eq. 5 (*P*-value)	0.125
In [65]	0.1328
In [10]	0.1328
In [39]	0.132813
In [66]	0.1289
In [17]	0.1250
In [18]	0.1563
In [8]	0.1289
In [73]	0.1172
In [67]	0.1406
In [70]	0.1172
AES S-box ([4])	0.062

Table 11 DP values of offered S-boxes

	DP
Offered S-box with Eq. 1	12
Offered S-box with Eq. 2	12
Offered S-box with Eq. 3	12
Offered S-box with Eq. 4	10
Offered S-box with Eq. 5	14
Offered optimized S-box with Eq. 1 (nonlinearity)	12
Offered optimized S-box with Eq. 2 (nonlinearity)	10
Offered optimized S-box with Eq. 3 (nonlinearity)	12
Offered optimized S-box with Eq. 4 (nonlinearity)	10
Offered optimized S-box with Eq. 5 (nonlinearity)	10
Offered optimized S-box with Eq. 1 (*P*-value)	10
Offered optimized S-box with Eq. 2 (*P*-value)	10
Offered optimized S-box with Eq. 3 (*P*-value)	12
Offered optimized S-box with Eq. 4 (*P*-value)	10
Offered optimized S-box with Eq. 5 (*P*-value)	10
In [9]	12
In [10]	10
In [8]	14
In [17]	12
In [39]	10
In [18]	12
In [66]	10
In [73]	12
In [67]	10
In [69]	10
In [70]	10
In [61]	10
In [71]	10
AES S-box ([4])	4

polynomial of type one (Eq. 1) (in optimized with *P*-value fitness). Compared to the previous work, this result was similar to references [10, 39, 61, 66, 67, 69, 70] and [71]. Table 11 represents DP results for proposed S-boxes and compares the results of previous work.

6 Concluding remarks

S-boxes aim to provide the necessary confusion that Shannon is declared as the foundations of any cipher system. This study proposed a new methodology for designing S-box by introducing strong chaotic maps. In addition, it improved PSO with this map. Improved PSO results have been used to optimize chaotic S-boxes. This optimization was conducted with two fitness functions (nonlinearity and *P*-value of mono-test). This study considered the hierarchy of trigonometric maps with their composition. This family of chaotic maps has ergodic properties. Ergodicity is equivalent to the confusion property. The study incorporated in this study is enabled to open new ways in the construction S-boxes based on strong chaotic maps and improved PSO. The best chaotic S-box (Table 1) was obtained using Eq. 3 maps. Considering the results of Tables 4, 5, 6, 7, 8 for the S-boxes of Tables 2 and 3, it is better to use nonlinearity fitness function optimization. Testing the *P*-value was related to the randomness of the box, perhaps with considering reference [72], it needed to reexamine the chaotic property of the S-box. As future work, multi-objective particle swarm optimization (MOPSO) can be used instead of PSO for improved optimization, so that in addition to nonlinearity, all S-box analysis criteria can be optimized. Future studies can even use other optimizations, such as harmony search (HS) algorithm. Furthermore, the use of a series of Julia sets based on generalized Chebyshev polynomial of type two or quantum maps from well-known quantum systems such as the Dicke model leads to the production of S-boxes with various performances.

In future, we intend to examine the effect of this dynamic behavior on the generation of the S-box and will compare it with classical maps. Failure to meet the S-boxes ideal criteria indicates the need to create new S-boxes.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Webster AF, Tavares SE (1985) On the design of s-boxes. In: Conference on the theory and application of cryptographic techniques, pp 523–534
2. Hussain I, Shah T, Gondal MA, Khan WA, Mahmood H (2013) A group theoretic approach to construct cryptographically strong substitution boxes. Neural Comput Appl 23(1):97–104
3. National Institute of Standards and Technology, FIPS PUB 46-3: Data Encryption Standard (DES), (Oct. 1999), super-sedes FIPS, 46-2
4. Advanced Encryption Standard (AES), (2001) Federal Information Processing Standards Publication 197 Std
5. Picek S, Batina L, Jakobović D, Ege B, Golub M (2014) S-box, SET, match: a toolbox for S-box analysis. In: Naccache D, Sauveron D (eds) Information security theory and practice. Securing the internet of things, vol 8501. Springer, Berlin, pp 140–149
6. Wang Y, Xie Q,Wu Y, Du B (2009) A software for S-box performance analysis and test. In: 2009 International Conference on Electronic Commerce and Business Intelligence, IEEE, pp 125-128
7. Aboytes-González JA, Murguía JS, Mejía-Carlos M, González-Aguilar H, Ramírez-Torres MT (2018) Design of a strong S-box based on a matrix approach. Nonlinear Dyn 94(3):2003–2012
8. Chen G, Chen Y, Liao X (2007) An extended method for obtaining S-boxes based on three-dimensional chaotic Baker maps. Chaos Solitons Fractals 31(3):571–579
9. Jakimoski G, Kocarev L (2001) Chaos and cryptography: block encryption ciphers based on chaotic maps. IEEE Trans Circuits Syst I Fundam Theory Appl 48(2):163–169
10. Tang G, Liao X, Chen Y (2005) A novel method for designing S-boxes based on chaotic maps. Chaos Solitons Fractals 23(2):413–419
11. Peng J, Zhang D, Liao X (2011) A method for designing dynamical S-boxes based on hyperchaotic Lorenz system. In: IEEE 10th International Conference on Cognitive Informatics and Cognitive Computing (ICCI-CC'11), IEEE, pp 304-309
12. Wang Y, Wong KW, Liao X, Xiang T (2009) A block cipher with dynamic S-boxes based on tent map. Commun Nonlinear Sci Numer Simul 14(7):3089–3099
13. Liu H, Kadir A, Gong P (2015) A fast color image encryption scheme using one-time S-Boxes based on complex chaotic system and random noise. Optics Commun 338:340–347
14. Lambić D (2018) S-box design method based on improved one-dimensional discrete chaotic map. J Inf Telecommun 2(2):181–191
15. Wang X, Akgul A, Cavusoglu U, Pham VT, Vo Hoang D, Nguyen XQ (2018) A chaotic system with infinite equilibria and its S-box constructing application. Appl Sci 8(11):2132
16. Ahmad M, Alam S (2014) A novel approach for efficient S-box design using multiple high-dimensional chaos. In: 2014 Fourth International Conference on Advanced Computing and Communication Technologies, IEEE, pp 95-99
17. Khan M, Shah T, Mahmood H, Gondal MA, Hussain I (2012) A novel technique for the construction of strong S-boxes based on chaotic Lorenz systems. Nonlinear Dyn 70(3):2303–2311
18. Özkaynak F, Yavuz S (2013) Designing chaotic S-boxes based on time-delay chaotic system. Nonlinear Dyn 74(3):551–557
19. Liu L, Zhang Y, Wang X (2018) A novel method for constructing the s-box based on spatiotemporal chaotic dynamics. Appl Sci 8(12):2650
20. Yang XJ (2019) General fractional derivatives: theory, methods and applications. Chapman and Hall/CRC, New York
21. Yang XJ, Baleanu D, Srivastava HM (2015) Local fractional integral transforms and their applications. Academic Press, New York
22. Yang XJ, Gao F, Ju Y (2019) General fractional derivatives with applications in viscoelasticity. Academic Press, New York
23. Yang XJ, Gao F, Ju Y, Zhou HW (2018) Fundamental solutions of the general fractional-order diffusion equations. Math Methods Appl Sci 41(18):9312–9320
24. Ghanbari B, Kumar S, Kumar R (2020) A study of behaviour for immune and tumor cells in immunogenetic tumour model with non-singular fractional derivative. Chaos Solitons Fractals 133:109619
25. Kumar S, Ahmadian A, Kumar R, Kumar D, Singh J, Baleanu D, Salimi M (2020) An efficient numerical method for fractional SIR epidemic model of infectious disease by using Bernstein wavelets. Mathematics 8(4):558
26. Kumar S, Kumar R, Agarwal RP, Samet B (2020) A study of fractional Lotka-Volterra population model using Haar wavelet and Adams–Bashforth–Moulton methods. Math Methods Appl Sci 43(8):5564–5578

27. Alshabanat A, Jleli M, Kumar S, Samet B (2020) Generalization of Caputo-Fabrizio fractional derivative and applications to electrical circuits. Front Phys 8:64
28. Baleanu D, Jleli M, Kumar S, Samet B (2020) A fractional derivative with two singular kernels and application to a heat conduction problem. Adv Differ Equ 2020(1):1–19
29. Kumar S, Ghosh S, Samet B, Goufo EFD (2020) An analysis for heat equations arises in diffusion process using new Yang-Abdel-Aty-Cattani fractional operator. Math Methods Appl Sci 43(9):6062–6080
30. Yang XJ, Baleanu D, Lazaveric MP, Cajic MS (2015) Fractal boundary value problems for integral and differential equations with local fractional operators. Thermal Sci 19(3):959–966
31. Kumar S, Ghosh S, Lotayif MS, Samet B (2020) A model for describing the velocity of a particle in Brownian motion by Robotnov function based fractional operator. Alex Eng J 59(3):1435–1449
32. Kumar S, Kumar R, Cattani C, Samet B (2020) Chaotic behaviour of fractional predator-prey dynamical system. Chaos Solitons Fractals 135:109811
33. Goufo EFD, Kumar S, Mugisha SB (2020) Similarities in a fifth-order evolution equation with and with no singular kernel. Chaos Solitons Fractals 130:109467
34. Yang XJ, Gao F, Srivastava HM (2017) Non-differentiable exact solutions for the nonlinear ODEs defined on fractal sets. Fractals 25(04):1740002
35. Ye T, Zhimao L (2018) Chaotic S-box: six-dimensional fractional Lorenz-Duffing chaotic system and O-shaped path scrambling. Nonlinear Dyn 94(3):2115–2126
36. Ahmad M, Bhatia D, Hassan Y (2015) A novel ant colony optimization based scheme for substitution box design. Proc Comput Sci 57(2015):572–580
37. Wang Y, Wong KW, Li C, Li Y (2012) A novel method to design S-box based on chaotic map and genetic algorithm. Phys Lett A 376(6–7):827–833
38. Ahmed HA, Zolkipli MF, Ahmad M (2019) A novel efficient substitution-box design based on firefly algorithm and discrete chaotic map. Neural Comput Appl 31(11):7201–7210
39. Farah T, Rhouma R, Belghith S (2017) A novel method for designing S-box based on chaotic map and teaching-learning-based Optimization. Nonlinear Dyn 88(2):1059–1074
40. Eberhart R, Kennedy J (1995) A new optimizer using particle swarm theory. In: MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science, IEEE, pp 39–43
41. Eberhart RC, Shi Y, Kennedy J (2001) Swarm intelligence. Elsevier, Amsterdam
42. Poli R, Kennedy J, Blackwell T (2007) Particle swarm optimization. Swarm intelligence 1(1):33–57
43. Parsopoulos KE, Vrahatis MN (2010) Particle swarm optimization and intelligence: advances and applications. IGI global, Hershey
44. Kamal ZA, Kadhim AF (2018) Generating dynamic S-BOX based on Particle Swarm Optimization and Chaos Theory for AES. Iraqi J Sci 59:1733–1745
45. Wang C, Yu T, Shao G, Nguyen TT, Bui TQ (2019) Shape optimization of structures with cutouts by an efficient approach based on XIGA and chaotic particle swarm optimization. Eur J Mech A Solids 74:176–187
46. Ye G, Zhou J (2014) A block chaotic image encryption scheme based on self-adaptive modelling. Appl Soft Comput 22:351–357
47. Jafarizadeh MA, Behnia S, Khorram S, Naghshara H (2001) Hierarchy of chaotic maps with an invariant measure. J Statist Phys 104(5–6):1013–1028
48. Jafarizadeh MA, Behnia S (2001) Hierarchy of chaotic maps with an invariant measure and their coupling. Phys D Nonlinear Phen 159(1–2):1–21
49. Jafarizadeh MA, Behnia S (2003) Hierarchy of one-and many-parameter families of elliptic chaotic maps of cn and sn types. Phys Lett A 310(2–3):168–176
50. Ahadpour S, Sadra Y (2012) A chaos-based image encryption scheme using chaotic coupled map lattices. Int J Comput Appl 49(2):15–18
51. Jafarizadeh MA, Foroutan M, Ahadpour S (2006) Hierarchy of rational order families of chaotic maps with an invariant measure. Pramana 67(6):1073–1086
52. Strogatz SH (2000) Nonlinear dynamics and chaos: with applications to physics, biology, chemistry, and engineering. Westview Press, Cambridge, p 478
53. Hasanipanah M, Armaghani DJ, Amnieh HB, Abd Majid MZ, Tahir MMD (2017) Application of PSO to develop a powerful equation for prediction of flyrock due to blasting. Neural Comput Appl 28(1):1043–1050
54. Shi Y, Eberhart RC (1998) Parameter selection in particle swarm optimization. In: International conference on evolutionary programming. Springer, Berlin, Heidelberg, pp 591–600
55. Chatterjee A, Siarry P (2006) Nonlinear inertia weight variation for dynamic adaptation in particle swarm optimization. Comput Oper Res 33(3):859–871
56. Feng Y, Teng GF, Wang AX, Yao YM (2007) Chaotic inertia weight in particle swarm optimization. In: Second International Conference on Innovative Computing, Information and Control (ICICIC 2007), IEEE, pp 475-475
57. Schneier B (2007) Applied cryptography: protocols, algorithms, and source code in C. Wiley, New York
58. Mollaeefar M, Sharif A, Nazari M (2017) A novel encryption scheme for colored image based on high level chaotic maps. Multimed Tools Appl 76(1):607–629
59. Schindler W (2009) Random number generators for cryptographic applications. In: Koç ÇK (ed) Cryptographic Engineering. Springer, Boston, pp 5–23
60. Pareek NK, Patidar V, Sud KK (2010) A random bit generator using chaotic maps. Int J Netw Secur 10(1):32–38
61. Tanyildizi E, Özkaynak F (2019) A new chaotic S-box generation method using parameter optimization of one dimensional chaotic maps. IEEE Access 7:117829–117838
62. Cusick TW, Stanica P (2017) Cryptographic Boolean functions and applications. Academic Press, Cambridge
63. Zhang H, Ma T, Huang GB, Wang Z (2009) Robust global exponential synchronization of uncertain chaotic delayed neural networks via dual-stage impulsive control. IEEE Trans Syst Man Cybern, Part B Cybern 40(3):831–844
64. Matsui M (1994) Linear cryptanalysis method for DES cipher, advances in cryptology–Eurocrypt'93. Lecture Notes Comput Sci 765:386–397
65. Lambić D (2017) A novel method of S-box design based on discrete chaotic map. Nonlinear Dyn 87(4):2407–2413
66. Özkaynak F, Özer AB (2010) A method for designing strong S-Boxes based on chaotic Lorenz system. Phys Lett A 374(36):3733–3738
67. Çavuşoğlu Ü, Zengin A, Pehlivan I, Kaçar S (2017) A novel approach for strong S-Box generation algorithm design based on chaotic scaled Zhongtang system. Nonlinear Dyn 87(2):1081–1094
68. Biham E, Shamir A (1991) Differential cryptanalysis of DES-like cryptosystems. J CRYPTOL 4(1):3–72
69. Çavuşoğlu Ü, Kaçar S, Zengin A, Pehlivan I (2018) A novel hybrid encryption algorithm based on chaos and S-AES algorithm. Nonlinear Dyn 92(4):1745–1759

70. Özkaynak F (2019) Construction of robust substitution boxes based on chaotic systems. Neural Comput Appl 31(8):3317–3326
71. Lambić D (2020) A new discrete-space chaotic map based on the multiplication of integer numbers and its application in S-box design. Nonlinear Dyn 100:699–711
72. Özkaynak F (2020) On the effect of chaotic system in performance characteristics of chaos based S-box designs. Phys A Statist Mech Appl 124072
73. Hussain I, Shah T, Gondal MA (2012) A novel approach for designing substitution-boxes based on nonlinear chaotic algorithm. Nonlinear Dyn 70(3):1791–1794

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