mathematics

MDPI

Article

Stochastic Fractal Search Algorithm Improved with Opposition-Based Learning for Solving the Substitution Box Design Problem

Francisco Gonzalez *, Ricardo Soto and Broderick Crawford

Escuela de Ingeniería Informática, Pontificia Universidad Católica de Valparaíso, Avenida Brasil 2241, Valparaíso 2362807, Chile; ricardo.soto@pucv.cl (R.S.); broderick.crawford@pucv.cl (B.C.)
* Correspondence: francisco.gonzalez@pucv.cl

Abstract: The main component of a cryptographic system that allows us to ensure its strength against attacks, is the substitution box. The strength of this component can be validated by various metrics, one of them being the nonlinearity. To this end, it is essential to develop a design for substitution boxes that allows us to guarantee compliance with this metric. In this work, we implemented a hybrid between the stochastic fractal search algorithm in conjunction with opposition-based learning. This design is supported by sequential model algorithm configuration for the proper parameters configuration. We obtained substitution boxes of high nonlinearity in comparison with other works based on metaheuristics and chaotic schemes. The proposed substitution box is evaluated using bijectivity, the strict avalanche criterion, nonlinearity, linear probability, differential probability and bit-independence criterion, which demonstrate the excellent performance of the proposed approach.

Keywords: cryptography; substitution box; opposition-based learning; metaheuristics; stochastic fractal search

MSC: 37M99

check for updates

Citation: Gonzalez, F.; Soto, R.; Crawford, B. Stochastic Fractal Search Algorithm Improved with Opposition-Based Learning for Solving the Substitution Box Design Problem. *Mathematics* **2022**, *10*, 2172. https://doi.org/10.3390/math10132172

Academic Editor: Ximeng Liu

Received: 21 April 2022
Accepted: 17 June 2022
Published: 22 June 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The explosive increase in the use of communication channels through digital media, the use of automatic learning in the field of medicine, and the digitization of participatory democratic means in the exchange of goods and services are some areas in which it is strongly required that the concept of security is robustly associated. Cryptography allows us to guarantee this property. Taking into account the symmetric block cipher scheme, for example, advanced encryption standard, we observe that the main component of these ciphers is the substitution box, which allows us to incorporate the concept of confusion, *obscuring the link between the secret key and cipher text* [1] in our cipher system. The strength of the substitution box present in the encryption system allows us to ensure its quality. There are different metrics to analyze a substitution box: nonlinearity, strict avalanche criterion, balance, bit independent criterion, and transparency order [2], to name a few. These metrics are used to determine the weaknesses or strengths of the substitution box against cryptanalysis [3,4]. In this work, we maximize the property of nonlinearity of substitution boxes of 8 input bits and 8 output bits. Traditionally, substitution box design methods can be grouped into the following schemes: algebraic, random, chaotic and heuristic methods. Here, we use a population optimization algorithm based on fractal search, the stochastic fractal search algorithm, which is inspired by fractals to comply with the exploitation property and stochasticity to implement the exploration mechanism, allowing the algorithm to efficiently traverse the search space. This metaheuristic is integrated with opposition-based learning, whose main notion emerges from the concept of Yin-Yang, and allows us to increase the degree of exploration of the first algorithm, through the

generation of opposite populations and subsequent analysis of the best solutions that are then operated by the optimization algorithm. We can increase the performance of the proposed scheme by addressing the problem of the adequate selection of parameters through the use of sequential model-based optimization for general algorithm configuration, which gives us an optimal set of parameters for the execution of stochastic fractal search. We have contrasted the execution of the hybrid algorithm with the classic version of the algorithm, showing the best performance of the hybrid proposal. Experiments show that the substitution boxes found have excellent cryptographic properties, which is compared with the results of other types of substitution box design. We evaluated the proposed substitution box based on bijectivity [5,6], nonlinearity, strict avalanche criteria, differential uniformity, linear approximation probability, justifying being a correct scheme for the generation of substitution boxes.

The organization of the work is defined as follows: the state of the art is introduced in the next section. The substitution box problem is described in Section 3. Section 4 provides an explanation of the optimization algorithm and opposition-based learning. Experimental findings are presented in Section 5. Finally, conclusions and future works can be found in Section 6.

2. State of the Art

During the last years, two research areas concerning the development of methods for the generation of substitution boxes have been strongly promoted: optimization algorithms and chaotic systems. In the set of works related to solving the substitution box design problem, we can mention the following works: [7] proposed a construction method based on a linear fractional transform, using the Box–Muller Transform, polarization decision, and central limit algorithm. The results reported good values for nonlinearity and other common metrics of security criteria. In [8], the authors reported a design combining the cuckoo search algorithm with chaotic maps. The latter was used to generate the initial population of the substitution box. Experiments showed that the substitution boxes were found to have good qualities to resist linear and differential attacks. A design that occupies the Mobius transformation was reported in [9]. The transformation was applied into random values that were generated by a nonlinear combination of chaotic tent map and sine map. The statistical analysis carried out showed good results, comparing the proposed substitution boxes with others of public knowledge, such as AES or Skipjack. In [10], the authors constructed a scheme utilizing the firefly algorithm. A discrete chaotic map fulfills the function of initializing the population. The use of the latter does not contain fixed points, which promotes the generation of chaotic sequences. The performance of this approach is checked against general criteria: bijectivity, bit-independence criteria, strict avalanche criteria, differential uniformity, linear approximation probability. The nonlinearity reaches the value of 107.5 on average. An algebraic technique for building a promising substitution box was proposed in [11]. The proposed substitution box is evaluated using standard metrics such as the bit-independent criterion, strict avalanche criterion, and nonlinearity, among others. Additionally, it was compared with another substitution box, including Skipjack, Xyi, AES, Gray, APA and Prime. The work in [12] implemented an improved one-dimensional chaotic logistic map that exhibits a strong chaotic behavior. The results were exposed through statistical and algebraic analysis. The proposed substitution box obtained an average nonlinearity of 108.13. Particle swarm optimization integrated with the chaotic Renyi map for the initial population was presented in [13]. In the experiments, various configurations for the parameters of the algorithm were considered, for example, setting the number of iterations equal to 1000. On the other hand, it was proposed an image encryption scheme that is based on the suggested substitution box. The analysis of results was carried out using the majority logic criterion and other metrics. In [14], the authors considered a design applying biometrics. In this case, characteristics of the fingerprints were used for the construction of substitution boxes. The results were compared against chaotic and biometric design schemes. The standard comparison metrics were used and, in addition, randomness,

confidence interval and time consumption were also evaluated. A design that uses chaotic logistic map for the generation of dynamic key-based substitution boxes was explained in [15]. This work counteracted the repercussions of algebraic attacks. The results were analyzed based on the conventional strength metrics of a substitution box. A random-restart hill-climbing algorithm for the construction of substitution box was described in [16]. Their objective function was nonlinearity. The results showed a reduction in the construction time and also added an algorithm to optimize linear approximation probability without affecting negatively the nonlinearity and biyectivity. In [17], the authors combined sine map and logistic functions to form a new chaotic map that holds excellent dynamic and complex properties, and this design was hybridized with algebraic techniques for the construction of substitution boxes. Standard metrics, such as nonlinearity, strict avalanche criteria, and linear approximation probability, were compared to other designs. The proposed substitution box was applied to image encryption. The results were analyzed using majority logic criterion. The utilization of piecewise linear chaotic map and quantum chaos for a key-based approach was presented in [18]. The proposed substitution box was evaluated against bijectivity, nonlinearity, strict avalanche criteria, bit independence criterion, differential approximation probability, and key sensitivity. The Leaders and Followers combined with hill-climbing was presented in [19]. An objective function was implemented that considers the whole Walsh–Hadamard spectrum of substitution boxes. The experiments reported that the method is resistant to classical cryptanalysis and side-channel attacks. In [20], the authors proposed a modified firefly algorithm, that performs a random movement based on the best firefly utilizing discrete chaotic maps. The experimental results revealed that the proposed method complies with properties that guarantee the strength of a substitution box. The work described in [21] is a genetic algorithm whose operators aim to maintain bijectivity and improve the nonlinearity of the solutions. The initial population is generated through a chaotic logistic map. The results show that the solutions are balanced and comply with high nonlinearity. A composition between logistic, tent and sine map builds a new chaotic map as explained in [22]. The qualities of the system were determined using the Lyapunov exponents and entropy variation. They proposed an image encryption scheme that showed good performance in complexity and execution times and that is also resilient to types of attacks, which allows it to be used in private network security. In [23], the authors presented the use of the leaders and followers algorithm in conjunction with hill climbing. This hybrid is also improved by means of machine learning, whose purpose is to determine the optimal moment of transition between exploration and exploitation. The scheme used an objective function that incorporates nonlinearity and transparency order in a weighted way. The results of this scheme are competitive with the results presented in several publications and leave room for possible improvements, for example, the use of deep learning, the formation of more complex objective functions using more than two properties. An improved version of the work established in [24] was proposed in [25]. This work proposed the cryptanalysis of an image encryption scheme, in which the formation of the substitution boxes was performed using a combination of chaotic maps, involving Lorenz [26] and Rossler [27]. The proposal in [28] utilized a heuristic evolution strategy based on affine transformation and permutation process. The experiments were evaluated according to standard criteria, fixed point analysis, and computational time. The proposed substitution box was tested to encrypt images, and the suitability was assessed using a majority logic criterion. It was concluded that the proposal is a feasible candidate to be applied in the context of image security. The introduction of a Markov model was given in [29]. This approach consists in stacking bitwise operations and solving them with reinforcement learning. This approach generated results are comparable to the state of the art. On the other hand, it included improvements to the SKINNY S-box implementation. The training of the model took about a month of computation. Results are expressed in terms of differential uniformity, linearity and number of nonlinear operations. The method in [30] to build substitution boxes uses particle swarm optimization, with random population. The proposed substitution box in combination with the chaotic Rossler map

is employed to establish security in the storage and communication of images. Standard metrics were used to demonstrate the quality of the proposed substitution boxes. The images resulting from the encryption process were analyzed based on the histogram, correlation coefficient, and entropy, among other metrics. In [31], the authors exposed a design, implementing the Tiki-Taka optimization algorithm in combination with a chaotic map selection method. Five chaotic maps are included for population generation. The selection of the map is based on a reward–penalty mechanism. Common metrics are used to evaluate the strength of the proposed substitution box, and the transparency order is also included. It was shown that, thanks to the properties of chaotic systems—ergodicity, pseudo randomness, and unpredictability—the performance of the original version of the optimization algorithm is improved. A modified Pascal's triangle in combination with the equation of the elliptic curve was presented in [32]. The experiments demonstrate the characteristics (differential approximation probability and linear approximation probability, among others) of the proposed substitution box, which was also used to implement image encryption and noise removal. For the comparison, common substitution boxes in the literature were used, such as AES, Skipjack, Xyi, along with other works. The work in [33] implemented the cuckoo search algorithm, which was enhanced in its search capacity and convergence speed, using a discrete chaotic map for the initial population generation. The results include nonlinearity, bijectivity, the bit-independence criterion, strict avalanche criterion, differential uniformity, and linear probability. The proposed scheme generated strong substitution boxes that meet the majority of cryptographic requirements.

One of the optimization algorithms that has emerged in recent times, and that has been applied to solve various problems, is stochastic fractal search. In the original paper [34], the proposed algorithm was evaluated using classic benchmark functions. The results were compared with particle swarm optimization and artificial bee colony, among others. The set of experiments also addressed the resolution of three different engineering design problems: tension/compression of a spring, welded beam and pressure vessel. The results of these experiments were compared against different algorithms: mathematical programming, genetic algorithm, coevolutionary particle swarm optimization, nonlinear integer and discrete programming, to name a few. These first results showed that the algorithm is capable of being used in various types of problems. For example, the problem of measuring the similarity between two overlapping sets of images is known as template matching. The work in [35] used stochastic fractal search to solve this problem, comparing the experimental results with algorithms, such as artificial bee colony and imperialist competitive algorithm, among others. The results showed that the algorithm obtained a better performance in contrast to other works present in the literature. The work in [36] used stochastic fractal search to address the problem of the environmental–economic dispatch problem in power systems operations, considering factors, such as physical restrictions, pollution, and transmission losses. The results included a comparison with various optimization algorithms. For example, the genetic algorithm, gravitational search algorithm, and six other algorithms. The work confirmed the ability of the algorithm to achieve values close to the global optimum within a short time frame. Stochastic fractal search was used to address the problem of unmanned aerial vehicle path planning, finding good results in acceptable times [37]. The problem of modeling photovoltaic systems includes the estimation of parameters with the available values of voltage and current. In the work of [38], the stochastic fractal search was incorporated as a mechanism for estimating the parameters of the previously mentioned problem in order to obtain efficient models. The results demonstrated the effectiveness of the algorithm in improving the capacity of the models, showing better performance against other recently published algorithms. A solution to the permutation flowshop scheduling problem was presented in [39], using stochastic fractal search. To demonstrate the ability of the algorithm to solve this problem, several types of instances were used. The algorithm was able to find solutions close to the known optima, according to the results presented. The work presented in [40] used stochastic fractal search for the parameter estimation of the support vector regression algorithm. The latter is

used in solving the bearing life prediction problem. The results obtained were consistent with those provided by other works in the literature. In the work of [41], the parameters of a control algorithm for automatic voltage regulator were estimated using stochastic fractal search. The results were compared with six other algorithms, showing an excellent ability to solve this problem. In [42], stochastic fractal search was proposed as a solution to the problem of visual tracking. The algorithm exhibited satisfactory results in difficult instances of this problem in comparison with other state-of-the-art algorithms. Another important aspect worth mentioning is the fact that stochastic fractal search was subject to modifications in some works. For example, the work in [43], for the first time, used the stochastic fractal search algorithm to solve complex multi-objective optimization problems. The incorporation of the differential evolution as a stochastic fractal search operator was proposed in [44]. The work in [45] used the chaotic maps of Chebyshev and Gauss/Mouse as modifiers of the equations in the diffusion and the first update processes. The focus on [46] incorporated Lévy flight and internal feedback information in the stochastic fractal search.

In the work carried out by [47], a series of integrations between optimization algorithms and machine learning techniques were exposed. One type of integration is where machine learning techniques act as low-level components in metaheuristics. There is a component category that performs the generation of initial solutions. In this type of component, several examples of the use of opposition-based learning hybridized with metaheuristics can be found in the literature. These works solve benchmark-type problems [48–52]; high-dimensional continuous optimization problems [53]; the reactive power dispatch problem [54]; symbolic regression [55]; and load frequency control [56]. The central idea in the use of opposition-based learning is that it provides complementary solutions to improve the convergence of the search process, increase search space coverage and increase the diversity of the population. To the best of our knowledge, there are no works in the literature that perform the integration between metaheuristics and opposition-based learning to solve the substitution box design problem.

Works that use stochastic fractal search and opposition-based learning allow us to affirm that the mentioned techniques are good candidates to present an integration between both. Our contribution is to present a new hybrid scheme, composed of the stochastic fractal search algorithm and opposition-based learning to solve the substitution box design problem, which presents excellent results in terms of performance. We also incorporated a tool that allows us to find an optimal algorithm configuration and thus establish a high quality of the experiments performed, achieving consistency in the performance of the algorithm. The resulting implementation, in conjunction with the formality of the experiments, led to obtaining competitive results in terms of nonlinearity of the substitution boxes found.

3. Substitution Box

Substitution boxes are the main component of an encryption system that allows us to guarantee its strength. Ensuring that a substitution box has a high nonlinearity [57], we can assert that the cryptographic system of which it is part will be resistant to various types of attack [58].

A substitution box S is defined as a mapping function, that takes n input bits and returns m output bits, $S : Z_2^n \Rightarrow Z_2^m$. It can be implemented as a lookup table or it can be dynamically generated. Next, we present the mathematical basis of the substitution boxes and the definition of the objective function to use.

Preliminaries

- Let f be a Boolean function defined as $f : F_2^n \rightarrow F_2^m$, where n is the input bits and m. We can mention several ways to represent a Boolean function: algebraic form, truth table, and hexadecimal form, to name a few.

- The representation in its algebraic normal form is built on the basis of the operations of sum and product of the input variables. We can write f in the following way:

$$f(x_1, \ldots, x_n) = a_0 \oplus a_1 x_1 \oplus \cdots \oplus a_n x_n \oplus \cdots \oplus a_{n-1,n} x_{n-1} x_n \oplus a_{1,2\ldots n} x_1 x_2 \ldots x_n \quad (1)$$

  where $a_0, a_1, a_{1,2,\ldots,n} \in \{0,1\}^*$.
- The decimal representation of a Boolean function is based on a vector of length 2^n, where n indicates the number of input bits. The elements of the vector are in decimal form.
- The number of 1 present in the truth table representation of a Boolean function is defined as the Hamming weight.
- If the elements present in the decimal representation are unique or the hamming weight is equal to 2^{n-1}, the Boolean function satisfies the balance property.
- The Hamming distance is calculated by counting all the differences in the output bits of two Boolean functions.
- The maximum of the degrees of the monomials of the algebraic normal form is called the algebraic degree.
- If the algebraic degree of a Boolean function is equal to one, the function is defined as an affine Boolean function. An affine Boolean function can be described as

$$f_{aff(x_1,x_2,\ldots,x_n)} = a_n x_n + a_{n-1} x_{n-1} + \cdots + a_2 x_2 + a_1 x_1 + a_0 \quad (2)$$

  where $a_i \in \{0,1\}$. When a_0 is zero, we get a linear Boolean function. A Boolean function of n input bits, can have 2^{n+1} affine Boolean functions.
- The Walsh–Hadamard transformation $\hat{F}_f(w)$ of Boolean function f with n variables is defined as:

$$\begin{aligned} \hat{F}_f(w) &= \sum_{x \in \mathbb{B}^n} \hat{f}(x)(-1)^{<w,x>} \\ &= \sum_{x \in \mathbb{B}^n} (-1)^{f(x) \oplus <w,x>} \\ &= \sum_{x \in \mathbb{B}^n} \hat{f}(x)\hat{l}_w(x) \end{aligned} \quad (3)$$

  where $\hat{f}$ is the polarity representation of a Boolean function, $\hat{l}_w(x)$ is the signed function of the linear function $l_w(x) =< w, x >$, $\hat{F}_f(w) \in [-2^n, 2^n], \forall w \in \mathbb{B}^n$ and $\hat{F}_f(w)$ is known as a *spectral walsh coefficient*. The real-value vector of all 2^n spectral coefficients is referred to as the Walsh–Hadamard transformation spectrum. The maximum absolute value, taken by $\hat{F}_f$, is given by: $WHT_{max}(f) = max_{(w \in \mathbb{B}^n)} |\hat{F}_f(w)|$.
- The Hadamard matrix, is a binary matrix of dimensions $2^n \times 2^n$. For the elements $W(i,j)$, where i represents the rows and j represents the columns,

$$H_{2^n} = \begin{bmatrix} H_{2^{n-1}} & H_{2^{n-1}} \\ H_{2^{n-1}} & -H_{2^{n-1}} \end{bmatrix} \quad (4)$$

  where $2 \leq n \in N$.
- In cryptography, one of the critical properties of a Boolean function is the nonlinearity. The nonlinearity is the minimum distance between a Boolean function and any affine Boolean function. A low nonlinearity value implies that a cryptographic algorithm may be weak against linear or differential attacks. The distance between a Boolean function $f(x)$ and any affine Boolean function $a(x) = a_0 + l(x)$, can be calculated as follows:

$$\begin{aligned} d(f(x), a(x)) &= wt(f(x) \oplus a(x)) \\ &= \sum_{x=0}^{2^n-1} (f(x) \oplus a(x)) \\ &= \frac{1}{2} \sum_{x=0}^{2^n-1} (1 - (-1)^{f(x)+a(x)}) \\ &= 2^{n-1} - \frac{1}{2}(-1)^{a_0} \sum_{x=0}^{2^n-1} (-1)^{f(x)+l(x)} \end{aligned} \tag{5}$$

taking $l(x) = \langle w, x \rangle = w_1 x_1 \oplus \cdots \oplus w_n x_n$, where w is the coefficient vector; we can rewrite the above result as

$$\begin{aligned} d(f(x), a(x)) &= 2^{n-1} - \frac{1}{2}(-1)^{a_0} \sum_{x=0}^{2^n-1} (-1)^{f(x)+\langle w,x \rangle} \\ &= 2^{n-1} - \frac{1}{2}(-1)^{a_0} S_{(f)}(w) \end{aligned} \tag{6}$$

where $S_{(f)}(w)$ is the Walsh–Hadamard transform of $f(x)$ on w, $\langle w, x \rangle$ is the representation of all affine boolean functions and $a_0 \in \{0, 1\}$. Finally, the nonlinearity is defined as follows:

$$nl(f) = 2^{n-1} - \frac{1}{2} max|S_{(f)}(w)| \tag{7}$$

The objective function to be used by the optimization algorithm is the nonlinearity.

4. Stochastic Fractal Search Algorithm with Opposition Based Learning

4.1. Stochastic Fractal Search Algorithm

The fractal concept helps us to describe the shape of an object, which, regardless of the level of visual distance we have from it, appears geometrically similar to the whole. Random fractals can be generated by various iterative methods, for example, Gaussian walks, trajectories of Brownian motion, and diffusion limited aggregation, among others.

In nature, there is a phenomenon known as dielectric breakdown (for example, lightning bolts and frost crystals), whose properties show that the branches that are generated can be modeled as stochastic patterns and also include fractal properties [59]. This phenomenon can be seen as a process of diffusion limited aggregation (DLA. Figure 1). The steps in the DLA process can be described as follows: consider an initial particle, then other particles are randomly generated around the initial particle by random walk, this process is repeated until a cluster is formed. This cluster will have a fractal shape.

The fractal search algorithm is inspired by this phenomenon. It takes the concept of diffusion limited aggregation as a search algorithm.

In [34], two metaheuristic algorithms are presented. The first one, fractal search, which is based on the properties of fractals, aims to use a few iterations to achieve a good level of efficiency and rapid convergence with adequate performance, but there are edges that allow improvements, for example, the high number of parameters that must be properly configured and the lack of communication between the solutions during the search process. For these reasons, a second algorithm was developed called stochastic fractal search, whose operators can be distinguished between two groups: diffusion and update.

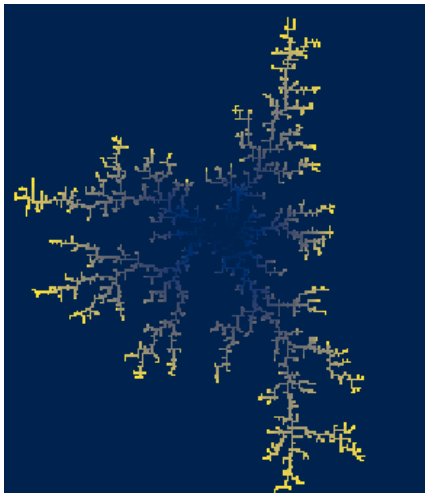

Figure 1. Fractal example using DLA.

The objective of the diffusion operator is exploitation, where each solution is diffused, in order to find a global minimum and also prevent the algorithm from stagnating at a local optimum. This operator generates new solutions by Gaussian walk from a particular solution. Figure 2 represents this operator. To keep the number of solutions constant, only the best solution generated by this process will be considered, and it will be compared with the original solution of the diffusion process, and in the event that the new solution has a better value of the objective function, the original solution will be replaced.

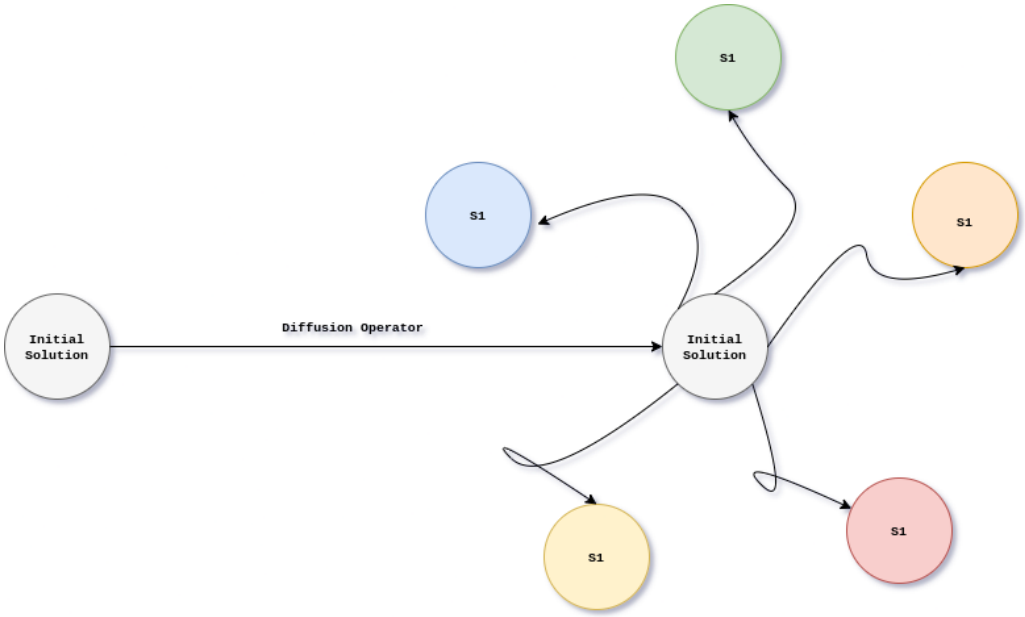

Figure 2. Diffusion operator.

In this process, there are two parameters. *Walk* is a random number between 0 and 1 uniformly distributed, and establishes which formula will be used to generate a new solution, which can be (8) or (9). The second parameter involved is the maximum number of solutions that are generated from a solution. We call this parameter the *maximum number of diffusion*.

$$GW_1 = Gaussian(\mu_{BP}, \sigma) + (\epsilon \times BP - \epsilon' \times P_i) \quad (8)$$

$$GW_2 = Gaussian(\mu_P, \sigma) \quad (9)$$

where ϵ and ϵ' are random numbers uniformly distributed in the $[0, 1]$ interval. BP and P_i are the best solution and the ith solution in the population, respectively. μ_P is exactly equal to P_i. The parameter μ_{BP} is exactly equal to the best solution, BP. σ is the standard deviation which is calculated as follows:

$$\sigma = \left| \frac{log(j)}{j} \times (P_i - BP) \right| \tag{10}$$

where j is the current iteration. To the extent that iterations increase, the term $\frac{log(j)}{j}$ is oriented to reduce the size of the Gaussian jumps.

The update operator is oriented so that the solutions modify their position based on the position of other solutions on the population. This operator allows us to comply with the quality of exploitation, which is determined by random walks using Gaussian distribution. Suppose we have a combinatorial optimization problem whose solutions can be represented by a vector of d dimensions, and the domain of the solutions is in the interval $[LB, UB]$.

$$P_j = LB + \epsilon \times (UB - LB) \tag{11}$$

where LB is the lower bound, and UB is the upper bound of the domain. ϵ is a uniformly distributed random number defined in the range $[0, 1]$. After generating the initial population and calculating the objective function of each of them, we proceed to identify the best solution BP. In the diffusion process, all the solutions have made a move either around their current position or the position of the best solution. The two updating processes are aimed at improving the exploratory capacity of the algorithm. The first update procedure is performed on each index of the solution vector, and the second procedure is aimed at modifying a solution based on other solutions of the population. This leads to an improvement in the exploration capacity, which increases the diversity of the population. The first task of the updating processes is to order the solutions based on a ranking formed by the value of the objective function (nonlinearity). Then, to each solution a probability is assigned, calculated by

$$Prob_i = \frac{rank(P_i)}{N} \tag{12}$$

where $rank(P_i)$ is the ranking of the solution P_i in relation to the population and N is the total number of solutions. Then for each solution P_i, the jth index is updated if the following condition is met:

$$Prob_i < \xi \tag{13}$$

where ξ is a random number, with uniform distribution, in the range $[0, 1]$. To update the respective index, the following formula is applied:

$$P'_i(j) = P_r(j) - \epsilon \times (P_t(j) - P_i(j)) \tag{14}$$

where $P'_i(j)$ is the new modified index of solution P_i, $P_r(j)$ is j index of a random solution r, ϵ is a random number from an uniform distribution in the interval $[0, 1]$, $P_t(j)$ is the j index of a random solution t, and $P_i(j)$ is the current j index of the analyzed solution.

At the beginning of the second update process, the solutions are ordered following the same logic as in the first process. Then, if the condition $Prob_i < \kappa$ is satisfied for a solution P'_i, where κ is a uniform distributed random number in the interval $[0, 1]$, the following equations can be applied:

$$P''_i = \begin{cases} P'_i - \hat{\epsilon} \times (P'_t - BP) & \kappa \leq 0.5 \\ P'_i + \hat{\epsilon} \times (P'_t - P'_r) & \kappa > 0.5 \end{cases} \tag{15}$$

where P_i'' is the modified solution, P_t' and P_r' are randomly selected solutions from the population, and $\hat{\epsilon}$ are random numbers generated by the Gaussian normal distribution. The modified solution P_i'' will replace the original solution P_i' only in the case of improvement of the value of the target function.

In this work, we investigated the generation of the initial population under five schemes:

- Using a combination of chaotic maps. The first map is the logistic map (16) which uses the parameter $\mu = [3.57, 4.0]$. This map will iterate 1000 times, and the output is used as input for the tent map (17). The tent map uses $b = [0.5, 1.5]$ and also iterates over 1000 times. The result of this operation allows us to form a vector of 256 unique elements in the interval $[0, 255]$ that will be our representation of a Boolean function.

$$x_{n+1} = \mu x_n (1 - x_n) \tag{16}$$

$$x_{n+1} = \begin{cases} \frac{x_n}{b}, & \text{for } x_n \in [0, b] \\ \frac{(1-x_n)}{1-b}, & \text{for } x_n \in [b, 1] \end{cases} \tag{17}$$

- Standard C++ functions for generating random numbers.
- Generation of random solutions together with the respective opposite solutions and extracting half of each set, whose solutions present the best fitness to form the initial population.
- Generation of random solutions in conjunction with the respective opposite solutions, combining this set and using the best solutions.
- Generating random solutions in conjunction with the respective opposite solutions, combining this set, selecting a subset of good solutions, and extracting a random set from the latter to form the initial population.

The last scheme was the one that presented the best results in terms of average nonlinearity of the initial population, so this scheme is used by the hybrid algorithm.

4.2. Opposition Based Learning

The idea of opposition [60] has been present since ancient times. We can see this in different cultures, such as China, using the concept of Yin-Yang. This notion of the opposite is considered to generate a learning scheme which allows us to describe the perceived reality. These ideas are taken to the field of computing and are concretized with the definition of opposite number [61] as follows:

Let $x(x_1, x_2, \ldots, x_d)$ be a point in with d dimensions and $x \in [a_i, b_i], i = 1, 2, 3, \ldots, d$. The opposite of x is defined by $\breve{x}(\breve{x_1}, \breve{x_2}, \ldots, \breve{x_d})$ as follows:

$$\breve{x} = a_i + b_i - x_i \tag{18}$$

A two-dimensional representation can be seen in Figure 3. Opposition-based learning follows the reasoning that it is beneficial to explore the search space using random directions in conjunction with the opposite directions simultaneously, which could raise the probability of finding promising regions of the search space.

4.3. Integration

The use of opposition-based learning in our scheme occurs at the moment when the last operator of the optimization algorithm ends. After completing the update II process, the opposite population of the current population is generated. These two sets are combined and ordered according to the fitness function. Then, the best m solutions are selected for the next iteration. m is a parameter of stochastic fractal search that determines the number of solutions to be generated. Algorithm 1 shows the pseudocode of the hybrid implementation. At Lines 1–2, the parameters of stochastic fractal search are set, and the initial population is generated. At Line 3 begins the main iteration. Lines 4–6 apply the diffusion process for each solution in the population. Lines 7–9 execute the first update process for all the

population, and then Lines 10–12 run the second update process for all solutions. Line 13 represents the process of opposition-based learning. In Figure 4, we can see a diagram of the proposed implementation. In the upper part of the image, we can see the mechanism for generating the initial population that will be processed by the optimization algorithm. The three operators of the optimization algorithm that operate sequentially are also expressed. Upon completion of the last operator, we determine whether the termination criteria have been met. If it is negative, the generation process of the opposite population is continued to later determine the solutions that will be incorporated again into the optimization process.

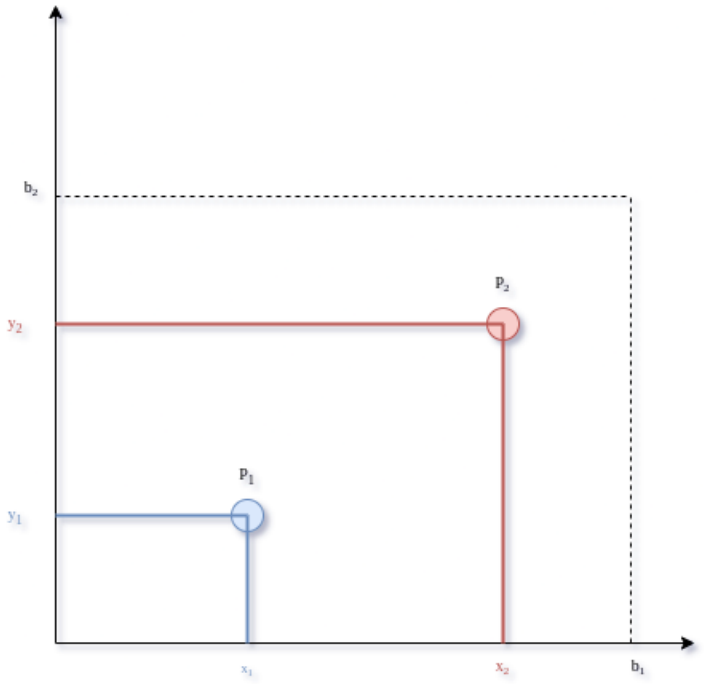

Figure 3. Two-dimensional opposition.

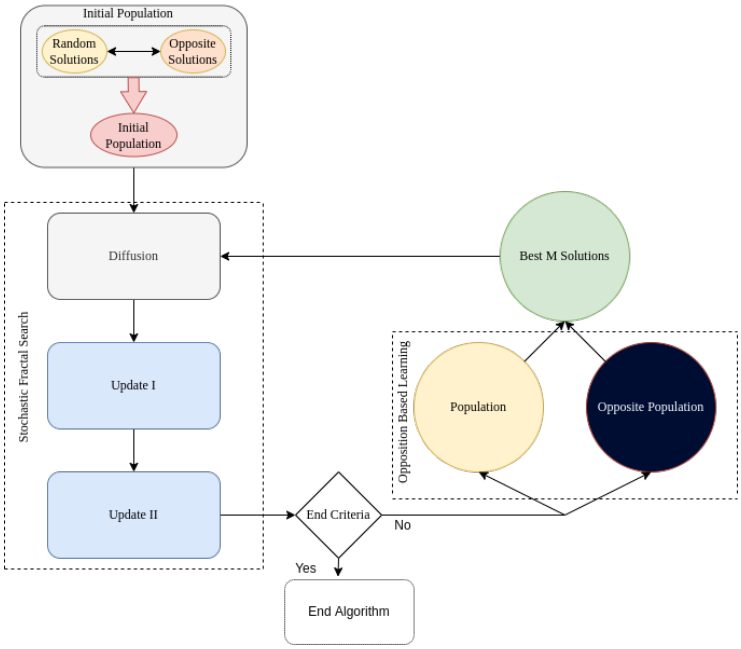

Figure 4. Stochastic fractal search with opposition-based learning.

Algorithm 1 *SFS OBL.*

Set parameters stochastic fractal search
Generate initial population
while $(i \leq MaximumIteration)$ **do**
  for $i = 1 : m$ (*m number of solutions*) **do**
    Diffusion$(solution_i)$
  end for
  for $i = 1 : m$ (*m number of solutions*) **do**
    Update1$(solution_i)$
  end for
  for $i = 1 : m$ (*m number of solutions*) **do**
    Update1$(solution_i)$
  end for
  Generate Oppossite Population and select ***m*** *best solutions*
end while

4.4. Sequential Model-Based Algorithm Configuration

The no-free-lunch theorem [62] states that no optimization algorithm is capable of tackling all existing problems optimally. This theorem can be applied to the parameter configuration of optimization algorithms. For this reason, the present work deals with this problem using a tool to establish an adequate parameter configuration of the stochastic fractal search algorithm. The selected tool is sequential model-based optimization for general algorithm configuration, SMAC [63], whose strength lies in the use of Bayesian optimization and racing mechanisms. SMAC is an iterative procedure that uses a surrogate model to describe the way in which the optimization problem is related to the parameters of the algorithm and analyzes its performance. The substitute model is used to propose a good parameter setting. Table 1 shows the four parameters delivered to SMAC for the search for an optimal configuration. The objective of the execution was set at the value of the objective function (the alternative is to use the execution time of the optimization algorithm), and a budget of 30 SMAC executions were established. The optimal configuration of the stochastic fractal search is shown in Table 2.

Table 1. Scenario delivered to SMAC for execution.

Parameter	Range	Default
iterations	[100, 400]	150
number population	[10, 50]	25
walk	[0.1, 1.0]	0.3
max number diffusion	[1, 5]	1

Table 2. Results of SMAC execution.

Parameter	Value
iterations	183
number population	36
walk	0.583946
max number diffusion	5

5. Results

A substitution box must meet certain requirements to be classified as a strong substitution box. In the literature, a set of metrics is considered standard to evaluate the competence of the proposed substitution box. The metrics used are bijectivity, algebraic degree, the strict avalanche criterion, nonlinearity, the bit-independence criterion, differential approximation probability and linear approximation probability. The present scheme operates with 8-bit input and output substitution boxes. The implementation of this work was done in C++ with the support of a library [64], which allowed analyzing the results of the algorithm. The

experiments were executed using a dual Intel Xeon E5-2690 with 32 GB of RAM running on Debian 10. The source code of this project is available in [65]. Regarding the results, we can mention that the proposed substitution box achieves a better nonlinearity compared to some works proposed in the literature. All generated substitution boxes satisfy the property of bijectivity. The resistance to algebraic attacks is demonstrated based on the value of algebraic immunity equal to 4. The strict avalanche criterion of the proposed substitution box is 0.5005, whose difference is insignificant with respect to the ideal value. The resistance to differential attacks of the proposed substitution box is based on a low value of differential approximation probability equal to 0.046. Regarding the tolerance of the proposed substitution box against linear attacks: this condition is fulfilled based on a low value of linear approximation probability equal to 0.125.

5.1. Nonlinearity

The proposed S-box an its inverse representation are shown in Table 3 and 4 respectively. In Table 5 shows a comparison of different works using the average nonlinearity of the S-box coordinates only [66].

Table 3. Proposed S-box .

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

Table 4. Inverse S-box.

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

Table 5. Experimental Results.

Method	Min NL	Max NL	ACNV
[24,67]	84	106	100.0
[68]	100	106	103.0
[69]	98	108	103.2
[70]	100	106	103.2
[71]	96	108	103.5
[72]	101	108	103.8
[73]	100	106	104.0
[74]	100	108	104.75
[75]	104	108	105.7
[76]	102	108	106.0
[77]	106	108	106.0
[78]	104	110	106.2
[79]	104	110	106.5
[80]	106	108	106.5
[81]	102	110	106.5
[82,83]	106	108	106.7
[84]	104	108	106.7
[85]	106	110	107.0
[10]	106	108	107.5
[86]	106	110	107.75
[87]	108	108	108.0
[88]	104	110	108.0
[89]	106	110	108.5
[90]	108	112	109.0
This Work	**106**	**112**	**109.25**
[91–93]	112	112	112.0
[66]	112	112	112.0

5.2. Bijectivity

The bijectivity of a substitution box is guaranteed when the following equation is satisfied:

$$wt\left(\sum_{i=1}^{n} a_i f_i\right) = 2^{n-1} \tag{19}$$

where wt is the Hamming weight, $a_i \in \{0,1\}$ and $(a_1, a_2, \ldots, a_n) \neq (0,0,\ldots,0)$. All the results obtained comply with this property.

5.3. Algebraic Degree

The resistance to various types of attack of a substitution box can be analyzed in terms of the algebraic degree [94,95]. The proposed substitution box has an algebraic degree of $deg(f) = 7$, which is a good value for this metric. On the other hand, the criterion of algebraic immunity, which indicates the resistance to algebraic attacks [96,97] of the proposed substitution box, has a value of 4, which is the maximum value for a substitution box of the dimensions used.

5.4. Strict Avalanche Criterion

Webster and Tavares [98] introduced the definition of the strict avalanche criterion. This concept aims at the idea that if we change a single input bit, the output bits would change with a probability of $\frac{1}{2}$. The proposed substitution box is analyzed using the dependency matrix shown in Table 6. The strict avalanche criterion of the proposed substitution box is 0.5005, which has a very low deviation from 0.5, the ideal value. This allows us to establish that the proposed substitution box exhibits a good avalanche effect and meets the aforementioned criteria.

Table 6. Dependency matrix for avalanche effect.

0.5000	0.4219	0.5156	0.4688	0.4844	0.4531	0.5469	0.4531
0.4844	0.5000	0.4844	0.5625	0.4688	0.5313	0.5313	0.4531
0.4844	0.5313	0.4844	0.4844	0.5469	0.5313	0.5000	0.4531
0.4531	0.4531	0.4688	0.5781	0.4844	0.5313	0.5469	0.5156
0.4844	0.4688	0.5156	0.4688	0.4844	0.5625	0.4375	0.5156
0.5000	0.5000	0.5469	0.5000	0.5000	0.4688	0.5625	0.4844
0.5000	0.5000	0.5000	0.5781	0.4531	0.5000	0.4688	0.4688
0.5313	0.5156	0.5781	0.5156	0.4844	0.5000	0.4688	0.5625

5.5. Bit-Independent Criterion

The bit-independent criterion, BIC, specifies that when any single input bit i is changed, the output bits j and k should change independently for all i, j and k. In Table 7, the results of this metric are presented.

Table 7. Bit-independent criterion.

0	104	104	104	104	108	100	106
104	0	102	102	98	104	102	100
104	102	0	102	106	106	106	106
104	102	102	0	100	96	106	108
104	98	106	100	0	108	106	106
108	104	106	96	108	0	104	104
100	102	106	106	106	104	0	102
106	100	106	108	106	104	102	0

5.6. Differential Approximation Probability

An S-box, having differential uniformity, establishes a unique correspondence between an input differential Δx to an output differential Δy. A low value of maximum differential approximation probability determines that the substitution box is immune against differential cryptanalysis. The differential uniformity of a substitution box is measured with differential approximation probability (DAP), which can be described as

$$DAP(\Delta x \rightarrow \Delta y) = \frac{\#\{x \in X | S(x) \oplus S(x \oplus \Delta x) = y\}}{2^m}$$

where X is the collection of all input values, and 2^m is the number of elements. The maximum DAP for the proposed S-box is 0.046. The low value of this property confirms that the proposed substitution box is tolerant against differential attacks. Table 8 shows comparative results with other works.

Table 8. Comparison of max differential probability of some S-boxes.

S-Box	**DAP**
This work	0.046
[85]	0.039
[69]	0.046
[68]	0.046
[71]	0.039
[72]	0.054
[70]	0.039
[74]	0.046
[99]	0.046
[100]	0.046
[13]	0.031
[101]	0.047

Table 8. *Cont.*

S-Box	DAP
[102]	0.039
[103]	0.031
[104]	0.055
[91]	0.015
[92]	0.015
[93]	0.031
[88]	0.046
[87]	0.039
[79]	0.039
[76]	0.039
[75]	0.039
[84]	0.039

5.7. Linear Approximation Probability

The maximum value of the imbalance of an event can be expressed with linear approximation probability *LP*. Two masks are applied to the parity of the input and output bits: Γx, Γy, respectively. In the work [4], the definition of *LP* is described by

$$LP = \max_{\Gamma x, \Gamma y \neq 0} \left| \frac{\#\{x \in X \mid x \cdot \Gamma x = S(x) \cdot \Gamma y\}}{2^n} - \frac{1}{2} \right| \tag{20}$$

where X is the set of all possible inputs, and 2^n is the number of elements. A small value of this property confirms that the proposed substitution box is resistant to linear attacks. The maximum value of *LP* for our proposed S-box is 0.125, and a comparison is shown in Table 9.

Table 9. Comparison of linear approximation probability of some S-boxes.

S-Box	LP
This work	0.125
[90]	0.093
[83]	0.132
[80]	0.132
[73]	0.132
[11]	0.132
[105]	0.140
[106]	0.125
[10]	0.125
[107]	0.132
[108]	0.125
[13]	0.132
[101]	0.148
[102]	0.137
[103]	0.113
[104]	0.132
[91]	0.062
[92]	0.142
[93]	0.102
[88]	0.139
[87]	0.140
[79]	0.117
[84]	0.132

5.8. Comparison without Opposition Based Learning

Using the same parameters provided by SMAC, we carried out the execution of experiments, considering only the optimization algorithm.

We consider an effective operator as the one that has managed to find a better solution in a given iteration. The optimization algorithm is composed of three movements: diffusion, update I and update II. These three movements are represented in Figure 5. We observe the number of times an operator manages to find an improved solution in the respective iteration. We can see that the diffusion operator in the SFS version achieves a higher percentage of effectiveness in relation to the total number of operations performed in that version of the algorithm. In general, the update II operator has a low percentage of effectiveness in both algorithms, so it can be a study to improve in future work.

In Figure 6, we observe a comparison between the two versions of the algorithm in relation to the iterations carried out, and the best fitness found for each iteration. It can be seen that the hybrid algorithm has a better performance compared to the standard algorithm. We can attribute this behavior to the fact that the hybrid algorithm has a stronger diversification component, which allows us to explore the search space with greater efficiency. This feature ensures that the hybrid algorithm does not get caught in a local optimum.

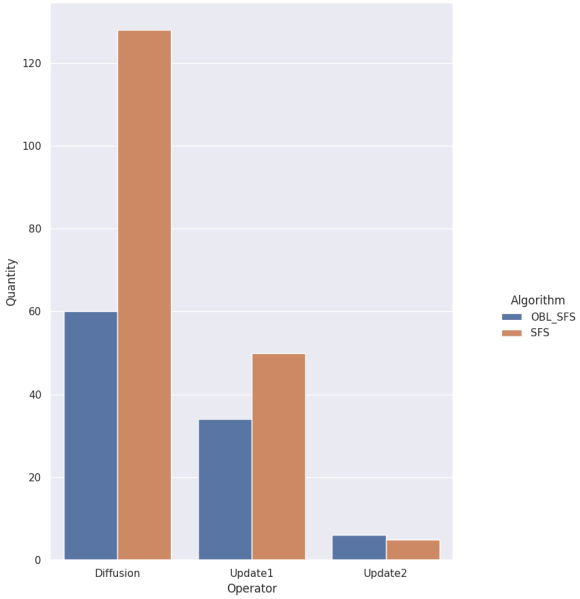

Figure 5. Effective operators.

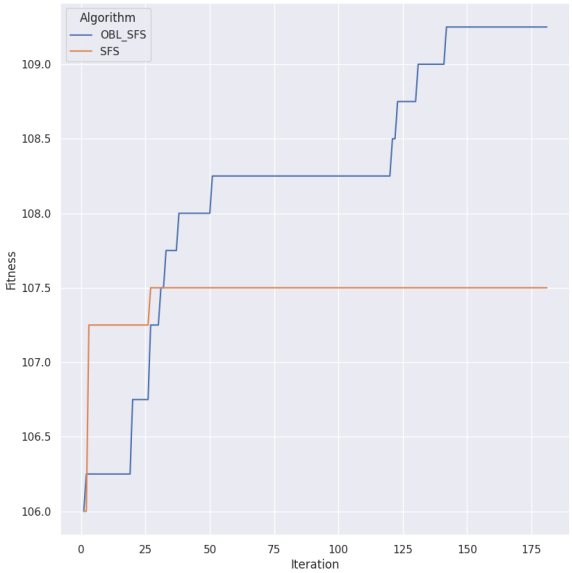

Figure 6. Fitness per iteration.

During the execution of both algorithms, the population is stored for each iteration. With both sets, five solution clusters are generated for each one of them. Figure 7 shows the five clusters with the solutions generated by the hybrid version of the algorithm. Figure 8 displays the five clusters with the version of the algorithm that only includes SFS.

This comparison is useful to verify that the hybrid version has a better performance than the standard version. It is observed that the standard version does not generate solutions whose fitness is greater than 108. It is also established that there is a smaller number of inferior solutions in terms of fitness in the hybrid version.The SFS version generates solutions that are under the nonlinearity of 104, which does not happen in the hybrid version.

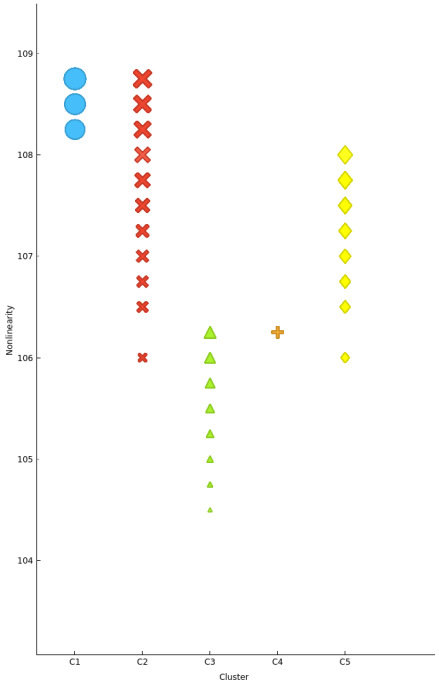

Figure 7. SFS OBL clusters of solutions.

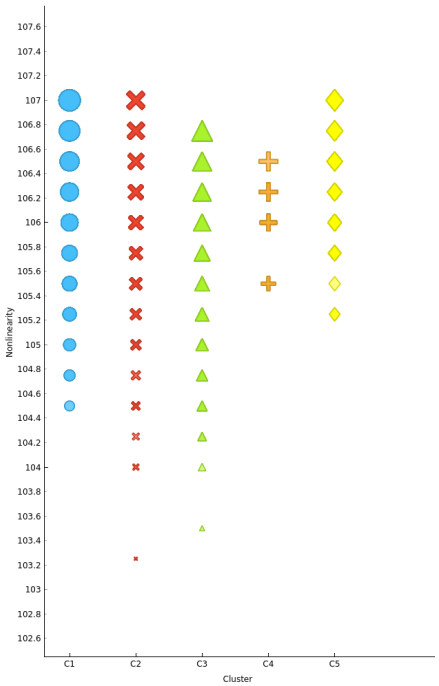

Figure 8. SFS clusters of solutions.

5.9. Brief Image Analysis

The proposed substitution box was used in a simple image encryption application. This consists of a substitution of the pixels of an image by the values present in the substitution box. We used four common images that can be seen in Table 10, along with the respective encryption. The proposed substitution box was compared with other well-known substitution boxes, such as AES, Camellia, Safer and Skipjack. To analyze the results, a set of metrics known as the majority logic criterion was used. This set includes entropy, correlation, contrast, energy, and homogeneity. We also included the analysis using the number of pixel change rate and the unified average changing intensity. We can observe that the performance of the proposed substitution box, in this brief analysis, is similar to that of substitution boxes known in the literature, even though in some cases, it has better attributes than the latter ones. The results in Table 11 show that the proposed substitution box is feasible to be included in the design of image encryption algorithms.

- The randomness of the information present in the encrypted image is measured with entropy. Values close to 8 are preferable.

$$Entropy = \sum_i p(x_i) log_2 \left(\frac{1}{p(x_i)} \right) \tag{21}$$

- Comparing the values between neighboring pixels determines the degree of similarity between them, which is known as correlation.

$$Correlation = \sum \frac{(i - \mu i)(j - \mu j)}{\sigma_i \sigma_j} \tag{22}$$

- The energy is a measure of the localized change of the image.

$$Energy = \sum p(i,j)^2 \tag{23}$$

- Contrast is a measure of luminance that allows one object to be distinguished from another.

$$Contrast = \sum |i - j|^2 p(i,j) \tag{24}$$

- Homogeneity determines the relationship between the elements of the gray level co-occurrence matrix with respect to its diagonal.

$$Homogeneity = \sum \frac{p(i,j)}{1 + |i - j|} \tag{25}$$

- Number of pixel change rate are designed to test the number of changing pixels between two encrypted images.

$$NPCR = \frac{\sum_{i=1}^{M} \sum_{j=1}^{N} d(i,j)}{M \times N} \quad d(i,j) = \begin{cases} 0 & C1(i,j) = C2(i,j) \\ 1 & C1(i,j) \neq C2(i,j) \end{cases} \tag{26}$$

- Unified average changing intensity is designed to test the number of mean intensities modified between two encrypted images.

$$UACI = \frac{1}{M \times N} \left[\frac{\sum_{i=1}^{M} \sum_{j=1}^{N} C1(i,j) - C2(i,j)}{255} \right] \tag{27}$$

Table 10. Original image and encrypted versions using different substitution boxes.

Image	Proposed	AES	Camellia	Safer	Skipjack

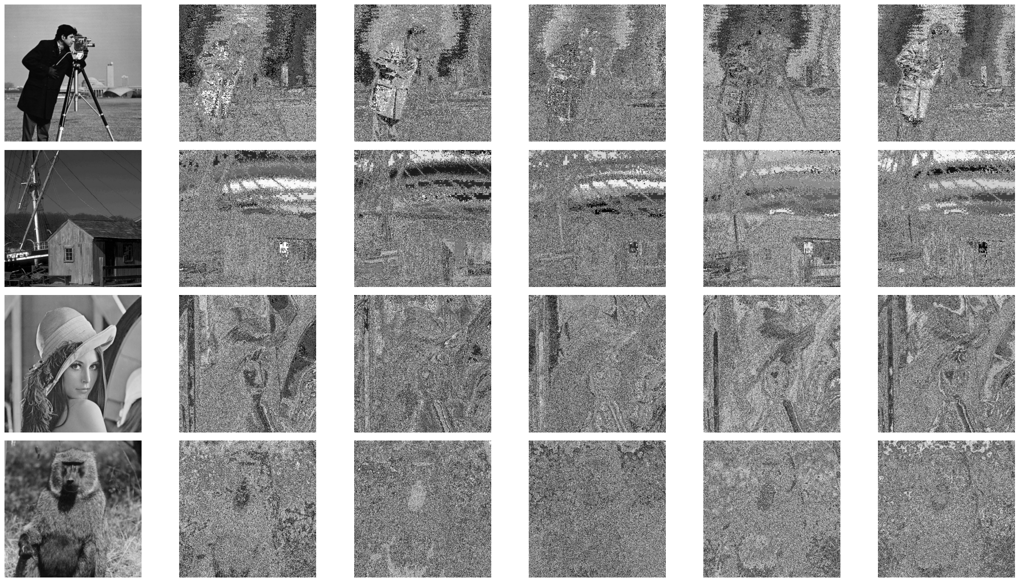

Table 11. Image analysis.

Sbox	Image	Entropy	Correlation	Contrast	Homogeneity	Energy	NPCR	UACI
Proposed	cameraman	7.8963	0.0756	14.3905	0.0304	0.0052	0.9957	0.3509
	house	7.8023	0.3008	37.7356	0.0798	0.0084	0.9944	0.3257
	lena	7.9233	0.2116	25.5224	0.0990	0.0212	0.9946	0.3089
	baboon	7.9107	0.1242	31.3474	0.0412	0.0055	0.9949	0.3009
Aes	cameraman	7.8604	0.0475	14.1664	0.0309	0.0053	0.9982	0.3343
	house	7.7912	0.2642	40.3050	0.0794	0.0086	0.9982	0.3301
	lena	7.9289	0.2023	29.0630	0.1038	0.0222	0.9979	0.2956
	baboon	7.9108	0.0776	32.6460	0.0408	0.0054	0.9984	0.3039
Camellia	cameraman	7.8734	0.0440	12.3571	0.0313	0.0052	0.9948	0.2968
	house	7.8284	0.1495	36.7644	0.0782	0.0083	0.9980	0.3209
	lena	7.9014	0.1687	27.5230	0.0986	0.0214	0.9983	0.2754
	baboon	7.9059	0.0777	28.8319	0.0406	0.0056	0.9978	0.2715
Safer	cameraman	7.8686	0.0992	12.2388	0.0317	0.0053	0.9959	0.3177
	house	7.7647	0.3108	32.5408	0.0767	0.0082	0.9878	0.3494
	lena	7.9338	0.1927	26.0448	0.0999	0.0216	0.9954	0.3185
	baboon	7.8983	0.1541	27.1982	0.0414	0.0054	0.9919	0.3018
Skipjack	cameraman	7.8624	0.0590	9.6959	0.0332	0.0055	0.9976	0.3309
	house	7.7418	0.2369	37.4516	0.0822	0.0090	0.9961	0.3553
	lena	7.9186	0.1660	23.1028	0.1038	0.0219	0.9946	0.2955
	baboon	7.9046	0.1171	27.8579	0.0438	0.0057	0.9944	0.2995

6. Conclusions

A strong substitution box allows us to establish a good degree of security for a symmetric block cipher. It is the only nonlinear component that contributes to the confounding property established by Shannon. For these reasons, a robust and efficient replacement box design is of vital importance.

In this work, we implemented a hybrid scheme using stochastic fractal search and opposition-based Learning, maximizing the nonlinearity property of the substitution boxes. Opposition-based learning allowed us to perform a much more comprehensive search space exploration in contrast to just using stochastic fractal search. The hybrid proposal establishes a clear improvement over the version that only uses stochastic fractal search, either in the greater quantity of good solutions found as well as in the superior quality of

these. The use of sequential model-based algorithm configuration allowed us to establish a set of optimal parameters of the stochastic fractal search algorithm and carry out the execution of experiments in a standardized way.

The results obtained in this work prove to be competitive with other techniques present in the literature. Even so, there is room to incorporate improvements. For instance, using other opposition-based learning schemes, for example, quasi opposition, fitness-based opposition, reflected extended opposition, quasi-reflection, partial opposition, to name a few. Using several opposition-based learning schemes at the same time, we could establish a reward and punishment system that can identify which opposition algorithm to use at a given moment in the execution of the metaheuristic. We could also generate a parallel work scheme, where each thread or core works with a different opposition system and the resulting populations can be communicated between the different threads in order to generate an increase in the diversity of the total population and enhance the exploratory capacity of the scheme.

Author Contributions: All the authors of this work have collaborated equally in the development of this paper. All authors have read and agreed to the published version of the manuscript.

Funding: Francisco González is supported by Postgraduate Grant Pontificia Universidad Católica de Valparaso, Chile, 2021. Ricardo Soto is supported by Grant CONICYT/FONDECYT/REGULAR/1190129. Broderick Crawford is supported by Grant CONICYT/FONDECYT/REGULAR/1210810.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Shannon, C.E. Communication theory of secrecy systems. *Bell Syst. Tech. J.* **1949**, *28*, 656–715. [CrossRef]
2. Picek, S.; Cupic, M.; Rotim, L. A new cost function for evolution of s-boxes. *Evol. Comput.* **2016**, *24*, 695–718. [CrossRef]
3. Biham, E.; Shamir, A.; Cryptol, J. Differential cryptanalysis of des. *Like Cryptosyst.* **1991**, *4*, 3.
4. Matsui, M. Linear cryptanalysis method for des cipher. In *Advances in Cryptology—EUROCRYPT'93*; Springer: Berlin/Heidelberg, Germany, 1994; pp. 386–397.
5. Zahid, A.H.; Arshad, M.J. An innovative design of substitution-boxes using cubic polynomial mapping. *Symmetry* **2019**, *11*, 437. [CrossRef]
6. Zahid, A.H.; Arshad, M.J.; Ahmad, M. A novel construction of efficient substitution-boxes using cubic fractional transformation. *Entropy* **2019**, *21*, 245. [CrossRef]
7. Khan, M.F.; Ahmed, A.; Saleem, K. A novel cryptographic substitution box design using gaussian distribution. *IEEE Access* **2019**, *7*, 15999–16007. [CrossRef]
8. Akhtar, T.; Din, N.; Uddin, J. Substitution box design based on chaotic maps and cuckoo search algorithm. In Proceedings of the 2019 International Conference on Advanced Communication Technologies and Networking (CommNet), Rabat, Morocco, 12–14 April 2019; pp. 1–7.
9. Jamal, S.S.; Anees, A.; Ahmad, M.; Khan, M.F.; Hussain, I. Construction of cryptographic s-boxes based on mobius transformation and chaotic tent-sine system. *IEEE Access* **2019**, *7*, 173273–173285. [CrossRef]
10. Ahmed, H.A.; Zolkipli, M.F.; Ahmad, M. A novel efficient substitution-box design based on fire fly algorithm and discrete chaotic map. *Neural Comput. Appl.* **2019**, *31*, 7201–7210. [CrossRef]
11. Jamal, S.S.; Shah, T. A novel algebraic technique for the construction of strong substitution box. *Wirel. Pers. Commun.* **2018**, *99*, 213–226.
12. Ullah, A.; Javeed, A.; Shah, T. A scheme based on algebraic and chaotic structures for the construction of substitution box. *Multimed. Tools Appl.* **2019**, *78*, 32467–32484. [CrossRef]
13. Ahmad, M.; Khaja, I.A.; Baz, A.; Alhakami, H.; Alhakami, W. Particle swarm optimization based highly nonlinear substitution-boxes generation for security applications. *IEEE Access* **2020**, *8*, 116132–116147. [CrossRef]
14. Şengel, Ö.; Aydın, M.A.; Sertbaş, A. An efficient generation and security analysis of substitution box using fingerprint patterns. *IEEE Access* **2020**, *8*, 160158–160176. [CrossRef]
15. Malik, M.S.M.; Ali, M.A.; Khan, M.A.; Ehatisham-Ul-Haq, M.; Shah, S.N.M.; Rehman, M.; Ahmad, W. Generation of highly nonlinear and dynamic aes substitution-boxes (s-boxes) using chaos-based rotational matrices. *IEEE Access* **2020**, *8*, 35682–35695. [CrossRef]

16. Ibrahim, S.; Abbas, A.M. A novel optimization method for constructing cryptographically strong dynamic s-boxes. *IEEE Access* **2020**, *8*, 225004–225017. [CrossRef]
17. Ahmad, M.; Al-Solami, E.; Alghamdi, A.M.; Yousaf, M.A. Bijective S-Boxes Method Using Improved Chaotic Map-Based Heuristic Search and Algebraic Group Structures. *IEEE Access* **2020**, *8*, 110397–110411. [CrossRef]
18. Peng, J.; Pang, S.; Zhang, D.; Jin, S.; Feng, L.; Li, Z. S-boxes construction based on quantum chaos and pwlcm chaotic mapping. In Proceedings of the 2019 IEEE 18th International Conference on Cognitive Informatics Cognitive Computing (ICCI*CC), Milan, Italy, 23–25 July 2019; pp. 1–6.
19. Freyre-Echevarría, A.; Martínez-Díaz, I.; Pérez, C.M.L.; Sosa-Gómez, G.; Rojas, O. Evolving nonlinear s-boxes with improved theoretical resilience to power attacks. *IEEE Access* **2020**, *8*, 202728–202737. [CrossRef]
20. Alhadawi, H.S.; Lambić, D.; Zolkipli, M.F.; Ahmad, M. Globalized firefly algorithm and chaos for designing substitution box. *J. Inf. Secur. Appl.* **2020**, *55*, 102671. [CrossRef]
21. Wang, Y.; Zhang, Z.; Zhang, L.Y.; Feng, J.; Gao, J.; Lei, P. A genetic algorithm for constructing bijective substitution boxes with high nonlinearity. *Inf. Sci.* **2020**, *523*, 152–166. [CrossRef]
22. Farah, M.A.; Farah, A.; Farah, T. An image encryption scheme based on a new hybrid chaotic map and optimized substitution box. *Nonlinear Dyn.* **2020**, *99*, 3041–3064. [CrossRef]
23. Bolufé-Röhler, A.; Tamayo-Vera, D. Machine learning based metaheuristic hybrids for s-box optimization. *J. Ambient. Intell. Humaniz. Comput.* **2020**, *11*, 5139–5152. [CrossRef]
24. Khan, M. A novel image encryption scheme based on multiple chaotic s-boxes. *Nonlinear Dyn.* **2015**, *82*, 527–533. [CrossRef]
25. Alanazi, A.S.; Munir, N.; Khan, M.; Asif, M.; Hussain, I. Cryptanalysis of novel image encryption scheme based on multiple chaotic substitution boxes. *IEEE Access* **2021**, *9*, 93795–93802. [CrossRef]
26. Lorenz, E.N. Deterministic nonperiodic flow. *J. Atmos. Sci.* **1963**, *20*, 130–141. [CrossRef]
27. Rössler, O.E. An equation for continuous chaos. *Phys. Lett. A* **1976**, *57*, 397–398. [CrossRef]
28. Zahid, A.H.; Iliyasu, A.M.; Ahmad, M.; Shaban, M.M.U.; Arshad, M.J.; Alhadawi, H.S.; Abd El-Latif, A.A. A novel construction of dynamic s-box with high nonlinearity using heuristic evolution. *IEEE Access* **2021**, *9*, 67797–67812. [CrossRef]
29. Kim, G.; Kim, H.; Heo, Y.; Jeon, Y.; Kim, J. Generating cryptographic s-boxes using the reinforcement learning. *IEEE Access* **2021**, *9*, 83092–83104. [CrossRef]
30. Khan, L.S.; Hazzazi, M.M.; Khan, M.; Jamal, S.S. A novel image encryption based on rossler map diffusion and particle swarm optimization generated highly non-linear substitution boxes. *Chin. J. Phys.* **2021**, *72*, 558–574. [CrossRef]
31. Zamli, K.Z.; Kader, A.; Din, F.; Alhadawi, H.S. Selective chaotic maps tiki-taka algorithm for the s-box generation and optimization. *Neural Comput. Appl.* **2021**, *33*, 16641–16658. [CrossRef]
32. Siddiqui, N.; Naseer, A.; Ehatisham-ul-Haq, M. A novel scheme of substitution-box design based on modified pascal's triangle and elliptic curve. *Wirel. Pers. Commun.* **2021**, *116*, 3015–3030. [CrossRef]
33. Alhadawi, H.S.; Majid, M.A.; Lambić, D.; Ahmad, M. A novel method of s-box design based on discrete chaotic maps and cuckoo search algorithm. *Multimed. Tools Appl.* **2021**, *80*, 7333–7350. [CrossRef]
34. Salimi, H. Stochastic fractal search: A powerful metaheuristic algorithm. *Knowl.-Based Syst.* **2015**, *75*, 1–18. [CrossRef]
35. Luo, Q.; Zhang, S.; Zhou, Y. Stochastic fractal search algorithm for template matching with lateral inhibition. *Sci. Program.* **2017**, *2017*, 1803934. [CrossRef]
36. Alomoush, M.I.; Oweis, Z.B. Environmental-economic dispatch using stochastic fractal search algorithm. *Int. Trans. Electr. Energy Syst.* **2018**, *28*, e2530. [CrossRef]
37. Li, W.; Sun, S.; Li, J.; Hu, Y. Stochastic fractal search algorithm and its application in path planning. In Proceedings of the 2018 IEEE CSAA Guidance, Navigation and Control Conference (CGNCC), Xiamen, China, 10–12 August 2018; pp. 1–5.
38. Rezk, H.; Babu, T.S.; Al-Dhaifallah, M.; Ziedan, H.A. A robust parameter estimation approach based on stochastic fractal search optimization algorithm applied to solar pv parameters. *Energy Rep.* **2021**, *7*, 620–640. [CrossRef]
39. Sasmito, A.; Pratiwi, A.B. Stochastic fractal search algorithm in permutation flowshop scheduling problem. *Aip Conf. Proc.* **2021**, *2329*, 050003.
40. Li, Y.; Huang, X.; Zhao, C.; Ding, P. Stochastic fractal search-optimized multi-support vector regression for remaining useful life prediction of bearings. *J. Braz. Soc. Mech. Sci. Eng.* **2021**, *43*, 414. [CrossRef]
41. Çelik, E. Incorporation of stochastic fractal search algorithm into efficient design of pid controller for an automatic voltage regulator system. *Neural Comput. Appl.* **2018**, *30*, 1991–2002. [CrossRef]
42. Charef-Khodja, D.; Toumi, A.; Medouakh, S.; Sbaa, S. A novel visual tracking method using stochastic fractal search algorithm. *Signal Image Video Process.* **2021**, *15*, 331–339. [CrossRef]
43. Khalilpourazari, S.; Naderi, B.; Khalilpourazary, S. Multi-objective stochastic fractal search: a powerful algorithm for solving complex multi-objective optimization problems. *Soft Comput.* **2020**, *24*, 3037–3066. [CrossRef]
44. Awad, N.H.; Ali, M.Z.; Suganthan, P.N.; Jaser, E. Differential evolution with stochastic fractal search algorithm for global numerical optimization. In Proceedings of the 2016 IEEE Congress on Evolutionary Computation (CEC), Vancouver, BC, Canada, 24–29 July 2016; pp. 3154–3161.
45. Rahman, T.A.; Jalil, N.A.; As'Arry, A.; Ahmad, R.R. Chaos-enhanced stochastic fractal search algorithm for global optimization with application to fault diagnosis. In *Materials Science and Engineering Conference Series*; IOP Publishing: Bristol, UK, 2017; Volume 210, p. 012060.

46. Zhou, C.; Sun, C.; Wang, B.; Wang, X. An improved stochastic fractal search algorithm for 3D protein structure prediction. *J. Mol. Model.* **2018**, *24*, 125. [CrossRef]
47. Talbi, E.-G. Machine learning into metaheuristics: A survey and taxonomy. *ACM Comput. Surv.* **2021**, *54*, 1–32. [CrossRef]
48. Ahandani, M.A. Opposition-based learning in the shuffled bidirectional differential evolution algorithm. *Swarm Evol. Comput.* **2016**, *26*, 64–85. [CrossRef]
49. Rahnamayan, S.; Tizhoosh, H.R.; Salama, M.M. Opposition-based differential evolution. *IEEE Trans. Evol. Comput.* **2008**, *12*, 64–79. [CrossRef]
50. Wang, H.; Wu, Z.; Rahnamayan, S.; Liu, Y.; Ventresca, M. Enhancing particle swarm optimization using generalized opposition-based learning. *Inf. Sci.* **2011**, *181*, 4699–4714. [CrossRef]
51. Wang, H.; Li, H.; Liu, Y.; Li, C.; Zeng, S. Opposition-based particle swarm algorithm with cauchy mutation. In Proceedings of the 2007 IEEE Congress on Evolutionary Computation, Singapore, 25–28 September 2007; pp. 4750–4756.
52. Si, T.; De, A.; Bhattacharjee, A.K. Particle swarm optimization with generalized opposition based learning in particle's pbest position. In Proceedings of the 2014 International Conference on Circuits, Power and Computing Technologies [ICCPCT-2014], Nagercoil, India, 20–21 March 2014.
53. Wang, H.; Wu, Z.; Rahnamayan, S. Enhanced opposition-based differential evolution for solving high-dimensional continuous optimization problems. *Soft Comput.* **2011**, *15*, 2127–2140. [CrossRef]
54. Basu, M. Quasi-oppositional differential evolution for optimal reactive power dispatch. *Int. J. Electr. Power Energy Syst.* **2016**, *78*, 29–40. [CrossRef]
55. Yazdani, S.; Shanbehzadeh, J. Balanced cartesian genetic programming via migration and opposition-based learning: Application to symbolic regression. *Genet. Program. Evolvable Mach.* **2015**, *16*, 133–150. [CrossRef]
56. Shankar, G.; Mukherjee, V. Quasi oppositional harmony search algorithm based controller tuning for load frequency control of multi-source multi-area power system. *Int. J. Electr. Power Energy Syst.* **2016**, *75*, 289–302. [CrossRef]
57. Carlet, C.; Dalai, D.K.; Gupta, K.C.; Maitra, S. Algebraic immunity for cryptographically significant boolean functions: Analysis and construction. *IEEE Trans. Inf. Theory* **2006**, *52*, 3105–3121. [CrossRef]
58. Rodríguez-Henríquez, F.; Saqib, N.A.; Pérez, A.D.; Koc, C.K. *Cryptographic Algorithms on Reconfigurable Hardware*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2007 .
59. Niemeyer, L.; Pietronero, L.; Wiesmann, H. Fractal Dimension of Dielectric Breakdown. *Phys. Rev. Lett.* **1984**, *52*, 1033–1036. [CrossRef]
60. Mahdavi, S.; Rahnamayan, S.; Deb, K. Opposition based learning: A literature review. *Swarm Evol. Comput.* **2018**, *39*, 1–23. [CrossRef]
61. Tizhoosh, H.R. Opposition-based learning: A new scheme for machine intelligence. In Proceedings of the International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC'06), Vienna, Austria, 28–30 November 2005; Volume 1, pp. 695–701.
62. Wolpert, D.H.; Macready, W.G. No free lunch theorems for optimization. *IEEE Trans. Evol. Comput.* **1997**, *1*, 67–82. [CrossRef]
63. Hutter, F.; Hoos, H.H.; Leyton-Brown, K. Sequential model-based optimization for general algorithm configuration. In *Learning and Intelligent Optimization*, Coello, C.A.C., Ed.; Springer: Berlin/Heidelberg, Germany, 2011; pp. 507–523.
64. Álvarez-Cubero, J.A.; Zufiria, P.J. Algorithm 959: Vbf: A library of c++ classes for vector boolean functions in cryptography. *ACM Trans. Math. Softw.* **2016**, *42*, 1–22. [CrossRef]
65. Molina, F.G. Stochastic Fractal Search Algorithm improved with Opposition-Based Learning for solving the substitution box design problem. *Figshare* **2021**. [CrossRef]
66. Dimitrov, M.M. On the design of chaos-based s-boxes. *IEEE Access* **2020**, *8*, 117173–117181. [CrossRef]
67. Khan, M.; Shah, T.; Batool, S.I. Construction of s-box based on chaotic boolean functions and its application in image encryption. *Neural Comput. Appl.* **2016**, *27*, 677–685. [CrossRef]
68. Chen, G.; Chen, Y.; Liao, X. An extended method for obtaining sboxes based on three-dimensional chaotic baker maps. *Chaos Solitons Fractals* **2007**, *31*, 571–579. [CrossRef]
69. Jakimoski, G.; Kocarev, L. Chaos and cryptography: Block encryption ciphers based on chaotic maps. *IEEE Trans. Circuits Syst. I Fundam. Theory Appl.* **2001**, *48*, 163–169. [CrossRef]
70. Özkaynak, F.; Özer, A.B. A method for designing strong sboxes based on chaotic lorenz system. *Phys. Lett. A* **2010**, *374*, 3733–3738. [CrossRef]
71. Asim, M.; Jeoti, V. Efficient and simple method for designing chaotic s-boxes. *ETRI J.* **2008**, *30*, 170–172. [CrossRef]
72. Tang, G.; Liao, X. A method for designing dynamical s-boxes based on discretized chaotic map. *Chaos Solitons Fractals* **2005**, *23*, 1901–1909. [CrossRef]
73. Khan, M.; Shah, T. A construction of novel chaos base nonlinear component of block cipher. *Nonlinear Dyn.* **2014**, 76, 377–382. [CrossRef]
74. Khan, M.; Shah, T. An efficient construction of substitution box with fractional chaotic system. *Signal Image Video Process* **2015**, *9*, 1335–1338. [CrossRef]
75. Liu, G.; Yang, W.; Liu, W.; Dai, Y. Designing s-boxes based on 3-d four-wing autonomous chaotic system. *Nonlinear Dyn.* **2015**, *82*, 1867–1877. [CrossRef]

76. Islam, F.U.; Liu, G. Designing sbox based on 4d-4wing hyperchaotic system. *3D Res.* **2017**, *8*, 9. [CrossRef]
77. Wang, X.; Çavuşoğlu, Ü.; Kacar, S.; Akgul, A.; Pham, V.T.; Jafari, S.; Alsaadi, F.E.; Nguyen, X.Q. S-box based image encryption application using a chaotic system without equilibrium. *Appl. Sci.* **2019**, *9*, 781. [CrossRef]
78. Zengin, A.; Pehlivan, I.; Kaçar, S. A novel approach for strong s-box generation algorithm design based on chaotic scaled zhongtang system. *Nonlinear Dyn.* **2017**, *87*, 1081–1094. [CrossRef]
79. Farah, T.; Rhouma, R.; Belghith, S. A novel method for designing s-box based on chaotic map and teaching learning-based optimization. *Nonlinear Dyn.* **2017**, *88*, 1059–1074. [CrossRef]
80. Lambi, D. S-box design method based on improved one-dimensional discrete chaotic map. *J. Inf. Telecommun.* **2018**, *2*, 181–191. [CrossRef]
81. Soto, R.; Crawford, B.; Molina, F.G.; Olivares, R. Human behaviour based optimization supported with self-organizing maps for solving the s-box design problem. *IEEE Access* **2021**, *9*, 84605–84618. [CrossRef]
82. Özkaynak, F. Construction of robust substitution boxes based on chaotic systems. *Neural Comput. Appl.* **2019**, *31*, 3317–3326. [CrossRef]
83. Lambi, D. A novel method of s-box design based on discrete chaotic map. *Nonlinear Dyn.* **2017**, *87*, 2407–2413. [CrossRef]
84. Ye, T.; Zhimao, L. Chaotic s-box: Six-dimensional fractional lorenz duffing chaotic system and o-shaped path scrambling. *Nonlinear Dyn.* **2018**, *94*, 2115–2126. [CrossRef]
85. Ahmad, M.; Bhatia, D.; Hassan, Y. A novel ant colony optimization based scheme for substitution box design. *Procedia Comput. Sci.* **2015**, *57*, 572–580. [CrossRef]
86. Yi, L.; Tong, X.; Wang, Z.; Zhang, M.; Zhu, H.; Liu, J. A novel block encryption algorithm based on chaotic s-box for wireless sensor network. *IEEE Access* **2019**, *7*, 53079–53090. [CrossRef]
87. Wang, K.-W.; Wong, Y.; Li, C.; Li, Y. A novel method to design sbox based on chaotic map and genetic algorithm. *Phys. Lett. A* **2012**, *376*, 827–833. [CrossRef]
88. Zhang, X.; Zhao, Z.; Wang, J. Chaotic image encryption based on circular substitution box and key stream buffer. *Signal Process. Image Commun.* **2014**, *29*, 902–913. [CrossRef]
89. Al Solami, E.; Ahmad, M.; Volos, C.; Doja, M.N.; Beg, M.M.S. A new hyperchaotic system-based design for efficient bijective substitution-boxes. *Entropy* **2018**, *20*, 525. [CrossRef]
90. Lambi, D. A novel method of s-box design based on chaotic map and composition method. *Chaos Solitons Fractals* **2014**, *58*, 16–21. [CrossRef]
91. Hussain, I.; Shah, T.; Gondal, M.A.; Mahmood, H. An efficient approach for the construction of lft s-boxes using chaotic logistic map. *Nonlinear Dyn.* **2013**, *71*, 133–140. [CrossRef]
92. Hussain, I.; Shah, T.; Gondal, M.A.; Mahmood, H. Efficient method for designing chaotic s-boxes based on generalized baker's map and tderc chaotic sequence. *Nonlinear Dyn.* **2013**, *74*, 271–275. [CrossRef]
93. Belazi, A.; Abd El-Latif, A.A.; Diaconu, A.V.; Rhouma, R.; Belghith, S. Chaos-based partial image encryption scheme based on linear fractional and lifting wavelet transforms. *Opt. Lasers Eng.* **2017**, *88*, 37–50. [CrossRef]
94. Boura, C.; Canteaut, A. On the influence of the algebraic degree of f^{-1} on the algebraic degree of $g \circ f$. *IEEE Trans. Inf. Theory* **2013**, *59*, 691–702. [CrossRef]
95. Boura, C.; Canteaut, A.; Cannière, C.D. Higher-order differential properties of keccak and luffa. In *Fast Software Encryption*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 252–269.
96. Courtois, N.T. Fast algebraic attacks on stream ciphers with linear feedback. In *Advances in Cryptology—CRYPTO 2003*; Boneh, D., Ed.; Springer: Berlin/Heidelberg, Germany, 2003; pp. 176–194.
97. Courtois, N.T.; Meier, W. Algebraic attacks on stream ciphers with linear feedback. In *Advances in Cryptology—EUROCRYPT 2003, International Conference on the Theory and Applications of Cryptographic Techniques, Warsaw, Poland, 4–8 May 2003*; Springer: Berlin/Heidelberg, Germany, 2003; Volume 2656, pp. 345–359.
98. Webster, A.F.; Tavares, S.E. On the design of s-boxes. In *Advances in Cryptology—CRYPTO '85 Proceedings*; Springer: Berlin/Heidelberg, Germany, 1986; pp. 523–534.
99. Ahmad, M.; Khan, P.M.; Ansari, M.Z. A simple and efficient key-dependent s-box design using fisher-yates shuffle technique. In *Recent Trends in Computer Networks and Distributed Systems Security*; Martínez Pérez, G., Thampi, S.M., Ko, R., Shu, L., Eds.; Springer: Berlin/Heidelberg, Germany, 2014; pp. 540–550.
100. Gondal, M.A.; Raheem, A.; Hussain, I. A scheme for obtaining secure s-boxes based on chaotic baker's map. *3D Res.* **2014**, *5*, 17. [CrossRef]
101. Zhang, Y.Q.; Hao, J.L.; Wang, X.Y. An efficient image encryption scheme based on s-boxes and fractional-order differential logistic map. *IEEE Access* **2020**, *8*, 54175–54188. [CrossRef]
102. Bin Faheem, Z.; Ali, A.; Khan, M.A.; Ul-Haq, M.E.; Ahmad, W. Highly dispersive substitution box (s-box) design using chaos. *ETRI J.* **2020**, *42*, 619–632. [CrossRef]
103. Hussain, S.; Jamal, S.S.; Shah, T.; Hussain, I. A power associative loop structure for the construction of non-linear components of block cipher. *IEEE Access* **2020**, *8*, 123492–123506. [CrossRef]
104. El-Latif, A.; Ahmed, A.; Abd-El-Atty, B.; Amin, M.; Iliyasu, A.M. Quantum-inspired cascaded discrete-time quantum walks with induced chaotic dynamics and cryptographic applications. *Sci. Rep.* **2020**, *10*, 1930. [CrossRef]

105. Yong, W.; Peng, L. An improved method to obtaining s-box based on chaos and genetic algorithm. *HKIE Trans.* **2012**, *19*, 53–58. [CrossRef]
106. Ahmad, M.; Doja, M.N.; Beg, M.M. Abc optimization based construction of strong substitution-boxes. *Wirel. Pers. Commun.* **2018**, *101*, 1715–1729. [CrossRef]
107. Zhang, T.; Chen, C.P.; Chen, L.; Xu, X.; Hu, B. Design of highly nonlinear substitution boxes based on i-ching operators. *IEEE Trans. Cybern.* **2018**, *48*, 3349–3358. [CrossRef] [PubMed]
108. Alzaidi, A.A.; Ahmad, M.; Ahmed, H.S.; Solami, E.A. Sine-cosine optimization-based bijective substitution-boxes construction using enhanced dynamics of chaotic map. *Complexity* **2018**, *2018*, 9389065. [CrossRef]