On the Design of Chaos-Based S-boxes

MIROSLAV M. DIMITROV

1Bulgarian Academy of Sciences, Institute of Mathematics and Informatics, Sofia, 1113, Bulgaria (e-mail: mirdim@math.bas.bg)
Corresponding author: Miroslav M. Dimitrov (e-mail: mirdim@math.bas.bg).
This work has been supported by the Bulgarian National Science Fund under contract number DH 12/8, 15.12.2017.

ABSTRACT Substitution boxes (S-boxes) are critical nonlinear elements to achieve cryptanalytic resistance of modern block and stream ciphers. Given their importance, a rich variety of S-box construction strategies exists. In this paper, S-boxes generated by using chaotic functions (CF) are analyzed to measure their actual resistance to linear cryptanalysis. The aforementioned papers emphasize on the average nonlinearity of the S-box coordinates only, ignoring the rest of the S-box components in the process. Thus, the majority of those studies should be re-evaluated. Integrating such S-boxes in a given cryptosystem should be done with a considerable caution. Furthermore, we show that in the context of nonlinearity optimization problem the profit of using chaos structures is negligible. By using two heuristic methods and starting from pseudo-random S-boxes, we repeatedly reached S-boxes, which significantly outperform all previously published CF-based S-boxes, in those cryptographic terms, which the aforementioned papers utilize for comparison. Moreover, we have linked the multi-armed bandit problem to the problem of maximizing an S-box average coordinate nonlinearity value, which further allowed us to reach near-optimal average coordinate nonlinearity values significantly greater than those known in literature.

INDEX TERMS chaos, multi-armed bandit problem, nonlinearity, S-boxes

I. INTRODUCTION

The cryptographic properties of vector boolean functions, or S-boxes, are thoroughly examined by introducing a rich list of desirable parameters an S-box should have in order to guarantee an acceptable resistance to sophisticated cryptographic attacks such as, for example, linear cryptanalysis [1][2], differential cryptanalysis [3], boomerang attack [4] or interpolation attack [5]. Furthermore, S-boxes are widely used in modern cryptographic algorithms like AES [6], Whirlpool [7], Camellia [8] and many others.

Despite the rich variety of proposed methods for S-boxes generation, we mainly focus on S-box constructions benefiting from the study of chaos, to further analyze their actual resistance to linear cryptanalysis.

In Section II we introduce the definitions of some basic cryptographic characteristics used to measure the cryptographic strength of a given S-box.

In Section III we show that the actual nonlinearity value, or NL, of the majority of chaotic functions-based (CF-based) published S-boxes differs from the average nonlinearity value originally announced. This discrepancy is based on the fact that the aforementioned papers consider the average nonlinearity of the S-box coordinates only, or ACNV, ignoring the rest of the S-box components in the process. In Section IV we propose an algorithm, which significantly outperforms all previously published S-boxes in terms of ACNV. During our experiments, we repeatedly reached S-boxes with ACNV of 114. We want to emphasize, that ACNV greater than 112.0, to the best of our knowledge, was never achieved in the literature.

In Section V we demonstrate the efficiency of the proposed algorithm by optimizing the ACNV of some popular S-boxes. Thus, we show that the starting state of the optimization routines is negligible. Having this in mind, the competitiveness of S-boxes generated by exploiting chaos structures, at least in the context of S-box nonlinearity optimization problem, is arguable. The same observation was made in [9].

Then, in Section VI we translate the S-box ACNV optimization problem to the multi-armed bandit problem, which allow us to further improve our results by reaching an ACNV of 114.5 - a value significantly larger than those known in literature.

II. PRELIMINARIES

Let $B = \{0, 1\}$. A Boolean function $f(x)$ of $n$ variables $x_1, \ldots, x_n$ is a mapping $f : B^n \rightarrow B$ from $n$ binary inputs $x = (x_1, x_2, \ldots, x_n) \in B^n$ to one binary output.
y = f(x) ∈ B.

**Definition 1 (Algebraic Normal Form – ANF):** The algebraic normal form of an n-variable Boolean function \( f(x) \) is given by the following equation:

\[
ANF_f = a_0 + a_1 x_1 + a_2 x_2 + a_1,2,\ldots,n x_1 x_2 \cdots x_n,
\]

where the coefficients \( a_i \ldots \in B = \{0, 1\} \).

A linear Boolean function \( f \) is a function with specific algebraic normal form \( ANF_f \), s.t. no term with algebraic degree greater than 1 exists. A more formal definition follows:

**Definition 2 (Linear Boolean Function):** Any n-variable Boolean function of the form:

\[
l_w(x) = <w,x> = w_1 x_1 + w_2 x_2 + \cdots + w_n x_n,
\]

where \( w, x \in B^n \), is called a linear Boolean function.

An n-binary input into m-binary output mapping \( S : B^n \leftrightarrow B^m \), which assigns some \( y = (y_1, y_2, \ldots, y_m) \in B^m \) by \( S(x) = y \) to each \( x = (x_1, x_2, \ldots, x_n) \in B^n \), is called an \((n \times m)\) substitution table (S-box) and is denoted by \( S(n, m) \).

An S-box \( S(n, m) \) is said to be bijective, if it maps each input \( x \in B^n \) to a distinct output \( y = S(x) \in B^m \) and all possible \( 2^n \) outputs are present. For example, the \((n, m)\) bijective S-boxes are Boolean permutations on \( F_2^n \), where \( F_2 \) is a finite field with two elements.

An S-box \( S(n, m) \) can be bijective only when \( n = m \). Clearly, a bijective S-box \( S(n, m) \) represents a permutation of its \( 2^n \) inputs, since each input is mapped to a distinct output and all possible \( 2^n \) outputs are present. In this way, \( S(n, n) \) will be reversible, that is, there is a mapping from each distinct output to its corresponding input.

**Definition 3 (Look-up Table):** The look-up table LUT of an S-box \( S(n, m) \) is a \((2^n \times m)\) binary matrix \( S_{LUT} \), which rows consist of all outputs of \( S(n, m) \), corresponding to all possible \( 2^n \) inputs ordered lexicographically.

\[
S = \begin{bmatrix}
f_1(0, 0, \ldots, 0) & f_2(0, 0, \ldots, 0) & \cdots & f_m(0, 0, \ldots, 0) \\
f_1(0, 0, \ldots, 1) & f_2(0, 0, \ldots, 1) & \cdots & f_m(0, 0, \ldots, 1) \\
\vdots & \vdots & \ddots & \vdots \\
f_1(1, 1, 0, \ldots, 0) & f_2(1, 1, 0, \ldots, 0) & \cdots & f_m(1, 1, 0, \ldots, 0) \\
f_1(1, 1, 1, \ldots, 1) & f_2(1, 1, 1, \ldots, 1) & \cdots & f_m(1, 1, 1, \ldots, 1)
\end{bmatrix}
\]

We define each column of \( S_{LUT} \) as coordinate of \( S \). All linear combinations of coordinates of \( S \) are called components of \( S \).

**Definition 4 (Linear Approximation Table):** The linear approximation table of an S-box \( S(n, m) \), denoted by \( S_{LAT} \), is a \((2^n \times 2^m)\) table, which entries are given by:

\[
S_{LAT}[X][Y] = 2^{n-1} - d_H(X, Y),
\]

where \( Y \) is a linear combination of the coordinates of the current S-box \( X \), and \( d_H(X, Y) \) denotes the Hamming distance between \( X \) and \( Y \).

The linear approximation table of a given S-box \( S(n, n) \) reveals the actual correlation between the components of \( S \) and all linear Boolean functions sharing the same dimension \( n \).

**Definition 5 (S-box Nonlinearity):** The nonlinearity of an S-box \( S(n, m) \) is defined as \( S_{NL} = 2^{n-1} - \max \{ \| \text{abs}(w_i) \| \} \), where \( \{w_i\} \) is the set of all elements in the LAT, excluding the first row and the first column.

Lower values of nonlinearity could be exploited by the family of linear cryptanalysis attacks. Having this in mind, higher nonlinearity value is a desirable S-box property.

Each S-box is uniquely defined by its LUT. Therefore, if we translate each row of the LUT as decimal number, we can obtain a unique decimal representation of the S-box denoted by DLUT.

### III. CHAOS-BASED S-BOX CONSTRUCTIONS

The methods involved in CF S-box constructions are manifold. For example, chaos function combined with travelling salesman problem \([10]\), chaotic substitution box design \([11]\), 1D chaotic map combined with \( \beta \)-Hill climbing \([12]\), chaotic map combined with sine-cosine optimization \([13]\), chaotic system with multiple attractors \([14]\), chaotic map combined with heuristics \([15]\), one-dimensional discrete chaotic map \([16]\), hyperchaotic systems \([17]\), \([18]\), spatiotemporal chaotic dynamics \([19]\), chaotic map combined with genetic algorithms \([20]\), chaotic logistic maps combined with bacterial foraging optimization \([21]\) and many others (see Table 6).

Usually, the best candidate of each method is further compared to others in terms of important cryptographic properties like nonlinearity, differential uniformity \([22]\) and strict avalanche criterion (SAC) \([23]\). The majority of authors emphasize on the ACNV of their best candidate. In Table 1 the coordinate nonlinearities of several S-box candidates achieved by some CF-based methods are presented. A more detailed overview is given in Table 6.

The actual nonlinearity of an S-box is calculated by the minimum nonlinearity of all the components of the S-box. For example, let us take an arbitrary S-box \( F(5, 5) \) with \( F_{LUT} = \{f_0, f_1, f_2, f_3, f_4\} \). Each column of \( F_{LAT} \) is determined by some linear combination of coordinates of \( F \), sorted lexicographically, from left to right, by the binary representation of the column index, zero-filled to 5. Let \( F_{LAT}[i] \) denotes the \( i \)-th column of \( F_{LAT} \). Then, for example, the \( F_{LAT}[11] \) column holds the nonlinear characteristics of the Boolean function \( f_1 \oplus f_2 \oplus f_4 \), while \( F_{LAT}[4] \) holds the nonlinear characteristics of the Boolean function \( f_3 \). In Figure 1 the coordinate decomposition of \( F_{LAT} \) is visualized. Each coordinate is associated with distinct color. The number of segments in each column corresponds to the number of terms in the respective linear combination of coordinates. Since \( F_{LAT}[0] \) is the trivial linear combination (all coefficients are equal to zero), we leave the first column of Figure 1 colorless. For technical reasons and better illustration, the coordinate decomposition example is based on a \((5, 5)\) S-box. However, it is applicable to S-boxes of any dimension.

As defined in Definition 5 we seek the maximum absolute value \( v \) of all the elements in S-box \( S(n, n) \) LAT, to find the nonlinearity of \( S \), i.e. \( S_{NL} = 2^{n-1} - v \).
In Table 2 the actual nonlinearity of each S-box from Table 1 is calculated. The deviations observed are due to the fact that the designers consider the nonlinearity values of coordinates only (the non-segmented columns in the (8, 8) coordinate decomposition).

In the context of block ciphers, a low nonlinearity S-box value is associated with the cipher linear cryptanalysis resistance [24] [2] [25].

IV. ALTERNATIVE CONSTRUCTION

As we have shown in the previous section, the average value of the nonlinearities of the coordinates of a given S-box $S$ doesn’t correspond to the the actual nonlinearity of $S$. However, from the designer perspective, if a higher value of ACNV is desirable, a new heuristic construction is suggested.

In general, if we want to improve the nonlinearity of a given bijective S-box $S(n,n)$, a strategy of lowering the absolute value of coefficients in $S_{LAT}$ makes sense. Moreover, the elements of each column of $S_{LAT}$ are entangled by the Parceval’s theorem [26]. Let’s denote as $C_i$ the array composed of the elements of $S_{LAT}[i]$. Since we want to lower the nonlinearities of coordinates of $S$ only, an evaluating function $E(S)$ is created, s.t. $E(S) = \sum_{p=0}^{n-1} \sum_{x \in C_{2p}} |x|^M$, where $M$ denotes a magnitude of our choice. The restriction $x \in C_{2p}$ narrows down the set of possible columns of $S_{LAT}$ to be optimized, in terms of nonlinearity, to the set of coordinates of $S$. As example, in case of a $S(5, 5)$ S-box, the evaluation

| Method     | $f_1$ | $f_2$ | $f_3$ | $f_4$ | $f_5$ | $f_6$ | $f_7$ | $f_8$ |
|------------|-------|-------|-------|-------|-------|-------|-------|-------|
| Ahmad [10] | 108   | 110   | 110   | 108   | 106   | 106   | 106   | 106   |
| Ahmad [11] | 104   | 106   | 106   | 104   | 102   | 108   | 106   | 106   |
| Al Solami [18] | 108 | 110 | 108 | 106 | 110 | 110 | 110 | 110 |
| Alzaidi [12] | 110 | 112 | 110 | 110 | 110 | 110 | 110 | 110 |
| Alzaidi [13] | 110 | 110 | 110 | 110 | 110 | 108 | 110 | 108 |
| Belazi [15] | 106 | 106 | 106 | 104 | 108 | 102 | 106 | 104 |
| Lai [14] | 104 | 110 | 104 | 108 | 104 | 104 | 104 | 104 |
| Liu [19] | 108 | 102 | 104 | 104 | 102 | 104 | 106 | 106 |
| Lambic [16] | 106 | 106 | 106 | 106 | 106 | 106 | 106 | 106 |
| Peng [17] | 102 | 102 | 104 | 104 | 102 | 106 | 106 | 106 |
| Tian [21] | 106 | 106 | 110 | 108 | 106 | 108 | 108 | 108 |
function threats as significant the elements inside the colored columns of $S_{LAT}$ illustrated in Figure 2

By using stochastic hill climbing as heuristic function, starting from arbitrary pseudo-random S-box construction and by using $E(S)$, algorithm 1 is proposed.

**Algorithm 1**

1: $s \leftarrow R(n) \triangleright$ the function $R(n)$ generates pseudo-random bijective S-box $S(n,n)
2: \text{repeat}
3: \quad \text{sdupl} \leftarrow s
4: \quad \text{RT}(\text{sdupl}) \triangleright$ the function RT$(S)$ make a random transposition in $S$
5: \quad $\text{if } E(\text{sdupl}) < E(s) \text{ then}
6: \quad \quad s \leftarrow \text{sdupl}
7: \quad \text{end if}
8: \text{until STOP condition is reached} \triangleright$ reaching $\frac{n(n-1)}{4}$ cycles

Given an S-box $S(n,n)$ and by using just one transposition, we can reach a total of $\binom{n}{2}$ S-boxes. Let denote this set as $S^T$. We further define a set $S^I$, s.t. $W \in S^I \iff W \in S^T \land E(W) < E(S)$. In case $|S^I| = 1$, and we are allowed to randomly pick $|S^I|$ elements from $S^T$, the probability some of the picked elements to belong to $S^I$ is $\frac{1}{2}$. The threshold value of the stop condition in Algorithm 1 is constructed on this observation.

**V. RESULTS PART I**

By using a magnitude of 10, we repeatedly generated S-boxes with high coordinate nonlinearities. During our experiments, we have tried various magnitude values. However, larger or smaller values of the magnitude are respectively too aggressive or too tolerant to the largest elements of the S-box LAT.

In Figure 7 the DLUT, in a hexadecimal format, of an optimized S-box $S_e(8,8)$ is presented. The first row and column of the table correspond respectively to the first and second half of the input in hexadecimal format. For example, the input 11110101, equal to f5, is transformed by $S_e$ to 5d. The characteristics of $S_e$ are summarized in Tables 3 and 4.

| Author | Method | min | max | ACNV | NL |
|--------|--------|-----|-----|------|----|
| Ahmad  | 10     | 106 | 110 | 107.5| 90 |
| Ahmad  | 11     | 102 | 108 | 105.25| 94 |
| Al Solami | 13    | 106 | 110 | 108.5 | 94 |
| Alzaidi | 14     | 110 | 112 | 110.25| 96 |
| Alzaidi | 15     | 108 | 110 | 109.5 | 94 |
| Belazi | 15     | 102 | 108 | 105.25| 88 |
| Lai    | 17     | 104 | 110 | 105.5 | 92 |
| Liu    | 18     | 102 | 108 | 104.5 | 96 |
| Lambic | 16     | 106 | 108 | 106.5 | 94 |
| Peng   | 17     | 100 | 102 | 102.75| 88 |
| Tian   | 21     | 106 | 110 | 107.5 | 92 |

| Author | Method | f1 | f2 | f3 | f4 | f5 | f6 | f7 | f8 |
|--------|--------|----|----|----|----|----|----|----|----|
| this work | 114 | 114 | 114 | 114 | 114 | 114 | 114 | 114 | 114 |

In [27], Table 5, a summary on the CF-based S-box constructions found in the literature is presented (an updated version of it is to be found in Table 6). We significantly outperform all of them in terms of ACNV and SAC, reaching the optimal SAC value of 0.5.

We further launched the algorithm on some popular (8,8) S-box constructions. However, because of the non-deterministic nature of the optimization process, it is difficult to match a given S-box input $S_{start}$, which is to be optimized, with the final optimized S-box $S_{end}$. To achieve such matching, we have restricted the algorithm of changing the first 16 elements of $S_{start}$. This allows us to further demonstrate the flexibility of the optimization process. Furthermore, since the first 16 elements of $S_{start}$ and $S_{end}$ are always shared, $S_{end}$ can be successfully matched to $S_{start}$.

In Figures 3 and 4, an optimized by algorithm 1 versions of Rijndael [6] and Whirlpool [7] S-boxes are presented. The colored cells represent those elements of the corresponding S-box, which were not modified during the optimization process. Furthermore, in Figures 5 and 6, the optimized versions of Fantomas [28] and Skipjack [29] S-boxes are given.

All of the aforementioned S-boxes are optimized to the ACNV of 114.0. Algorithm 1 was implemented with the built-in tools provided by the open-source mathematical software system SageMath [30].

**VI. S-BOX AS MULTI-ARMED BANDITS**

The space of bijective S-boxes is vast. For example, in the case of (8,8) S-boxes, we have a total of $2^{256} \approx 2^{1684}$ different bijective S-boxes. Despite algorithm 1 efficiency in finding S-boxes with better ACNV, we had never reached an
An optimized Fantomas S-box using Algorithm 1.

Figure 3.

Method min max ACNV SAC NL
---
this work | 114 | 114 | 114 | 0.500000 | 96

ACNV greater than 114. However, we have found out that the multi-armed bandit problem [31] [32] [33] [34] is closely related to the nonlinearity optimization problem.

Each bijective S-box $S(n, n)$ can be represented as a collection of $n$ bandits, such that each bandit uniquely corresponds to some of the $n$ coordinates of $S$. The arms of each bandit could be associated with the operation of applying a single transposition in $S$, while the profit of our action could be measured with the fitness function presented in Algorithm 1.

Associating each one of the possible $\binom{n}{2}$ transposition of elements of $S$ DLUT to some distinct arm in each bandit is a trivial and non-working model - at the end, the bandits would be indistinguishable. Having this in mind, the following model is constructed:

- **Property I**: Since each bandit uniquely corresponds to some coordinate of $S$, each bandit arm is restricted to initiate a transposition of two bits inside a column of $S_{LUT}$ only (instead of a transposition of any two elements in $S_{DLUT}$).
- **Property II**: To keep the bijective property of $S$, in case
an arm of some bandit is activated, the set of all distinct \( \binom{n}{2} \) bit transpositions in a given coordinate of \( S_{\text{LUT}} \) is restricted to a subset of transpositions with a size of \( 2^{n-1} \).

The restriction introduced in Property II is motivated by the following observations:

1) **Existence:** If \( b_1 b_2 \cdots b_i \cdots b_n \) is a row from \( S_{\text{LUT}} \), flipping the bit \( b_i \) will result in some other row \( R = b_1 b_2 \cdots b_i \cdots b_n \) in \( S_{\text{LUT}} \). Otherwise, if \( R \) is not among the rows of \( S_{\text{LUT}} \), \( S \) is not surjective, therefore not bijective, which contradicts our initial choice of \( S \).

2) **One-to-one Mapping:** If \( b_1 b_2 \cdots b_i \cdots b_n \) is a row from \( S_{\text{LUT}} \), flipping the bit \( b_i \) will result in only one row \( R = b_1 b_2 \cdots b_i \cdots b_n \) in \( S_{\text{LUT}} \). Otherwise, if some other row \( R' \) of \( S_{\text{LUT}} \) exists, s.t. \( R \equiv R' \), \( S \) is not injective, therefore not bijective, which contradicts our initial choice of \( S \).

3) **Search space:** The total number of distinct bit sequences of the form \( b_1 b_2 \cdots b_i \cdots b_n \) is \( 2^{n-1} \).

Let’s denote as a bandit \( B_i \) the bandit, which corresponds to the \( i \)-th coordinate of \( S \). Each bandit consists of \( 2^{n-1} \) distinct arms, s.t. each arm of \( B_i \) corresponds to a distinct value of \( b_1 b_2 \cdots b_{i-1} b_{i+1} \cdots b_n \). Activating an arm of \( B_i \) will result of interchanging two rows of \( S_{\text{LUT}} \), which differ only in bit position \( i \).

For example, let’s consider an S-box \( X(4,4) \), with \( X_{\text{DLUT}} = [15, 14, 9, 2, 11, 3, 12, 4, 1, 13, 7, 8, 6, 10, 5, 0] \). \( X \) is a bijective S-box with dimension 4. Therefore, we can transform \( X \) as an 8-armed 4-bandit problem. In Figure 8, a visual interpretation of the bandits transformation of \( X \) is shown. Each row corresponds to a distinct bandit, while each pair of cells inside a given row, sharing the same color, corresponds to an arm of the given bandit. The x-axis represents the indexes of elements of \( X_{\text{DLUT}} \) (starting from 1).

As an illustration, if we activate the white arm of bandit 1, we interchange the elements of \( X_{\text{DLUT}} \) with indexes 14 and 4, i.e. 10 and 2. Their respective binary representations (with zero-fill of 4) are 1010 and 0010 (they differ only in bit position 1).

The profit (if any) of activating a bandit \( B_i \)'s arm is measured by the same function \( E \) presented in Algorithm 1.

The transformation of the \((n,n)\) bijective S-box ACNV optimization problem to the \(2^{n-1}\)-armed \( n \)-bandit problem allows us to focus on the optimization of the nonlinearity of single coordinates. Furthermore, by design, activating an arm of a given bandit doesn’t affect the states of other bandits. Having this in mind, Algorithm 2 is proposed.

**Algorithm 2**

1. \( s \leftarrow R(n) \) \( \triangleright \) the function \( R(n) \) generates pseudo-random bijective S-box \( S(n,n) \)
2. \( \Omega \leftarrow \text{MODEL}(s) \) \( \triangleright \) We transform the S-box \( s \) to a \( 2^{n-1} \)-armed \( n \)-bandit problem
3. **repeat**
   4. \( \text{bandit} \leftarrow \text{random}(1,n, \Lambda) \)
   5. \( \text{arm} \leftarrow \text{random}(1,2^{n-1}) \) \( \triangleright \) We choose a random arm from \([1, 2^{n-1}]\)
   6. \( \text{oldBandit} \leftarrow E(\text{bandit}) \)
   7. \( \text{ACTIVATE(bandit, arm)} \)
   8. **if** \( E(\text{bandit}) < E(\text{oldBandit}) \) **then**
   9. \( \Omega \leftarrow \text{MODEL}(s) \) \( \triangleright \) We update the model
   10. **else**
   11. \( \text{ACTIVATE(bandit, arm)} \) \( \triangleright \) We resume the original state of the bandit
12. **until** STOP condition is reached \( \triangleright \) reaching \( n2^{n-1} \) consequent unsuccessful attempts

**TABLE 5.** Nonlinearities of the S-box coordinates given in Figure 9.

| Method       | \( f_1 \) | \( f_2 \) | \( f_3 \) | \( f_4 \) | \( f_5 \) | \( f_6 \) | \( f_7 \) | \( f_8 \) |
|--------------|----------|----------|----------|----------|----------|----------|----------|----------|
| this work    | 116      | 114      | 116      | 114      | 114      | 114      | 114      | 114      |

We launched Algorithm 2 as a stand-alone optimization routine, starting from pseudo-randomly generated S-boxes, and in almost all of the instances we reached S-boxes with an average coordinate nonlinearity value of 112. However, when we initiated Algorithm 2 with S-boxes, which have been already optimized by Algorithm 1, we have reached an average coordinate nonlinearity value of 114.5. An example of such S-box is given in Figure 9. The corresponding nonlinearity by coordinates is given in Table 5.

In Table 6 an extended S-box comparison between the state-of-the-art methods is given. The entries are sorted, in increasing order, by ACNV (the last column).

**VIII. CONCLUSION AND FUTURE WORK**

CF-based S-box construction is a relatively new and interesting technique, which interconnects the tools provided by various academic disciplines with the problem of finding secure cryptographic primitives.

In this paper, we analyzed the actual linear cryptanalysis resistance of CF-based S-boxes, which differs from the average nonlinearity value announced by a great number of papers. Integrating such S-boxes in a cryptosystem should be done with a considerable caution. For example, if we interchange the Rijndael S-box in AES with some CF-based S-box with higher ACNV, but lower overall nonlinearity, the resulting modified block cipher will be significantly weaker in terms of resistance to linear cryptanalysis. Furthermore,
we show that exploiting chaos structures, in the context of nonlinearity optimization problem, is arguable. Thus, the benefits of using chaos structures in the design of S-boxes is unclear and yet to be determined. However, as stated in [82], the chaos-based designs may be an alternative to application attacks, such as side-channel analysis.

Nevertheless, from designer perspective, if the overall nonlinearity value of an S-box $S$ is negligible compared to the average nonlinearity value of all coordinates of $S$, two novel S-box constructions are suggested.

While Algorithm 1 yields better results than Algorithm 2, the latest could be used as an Algorithm 1 extension, to further improve the parameters of the resulting S-box. The methods presented in this paper significantly outperform all other state-of-the-art methods for designing S-boxes with high ACNV.

The linkage of the $n$-armed bandit problem to the problem of finding such S-boxes, opens an interesting area of future research - the investigation of how other state-of-the-art methods, such as the concept of fuzzy graphs [33] [34], the stochastic optimization techniques [85] [86] [87], or the exploration-exploitation algorithms [88] [89] [90], could be exploited to further maximize the ACNV of a given S-box.

An interesting open question to be answered is to what extent the ACNV value of an (8,8) bijective S-box could be

![Diagram](image1)

**FIGURE 8.** An example of 8-armed 4-bandit problem transformation of the S-box $X$

---

**TABLE 6.** A comparison of S-boxes, yielded by various methods to be found in the literature, with those S-boxes, reached by the algorithms presented in this work.

| Method          | Min NL | Max NL | ACNV |
|-----------------|--------|--------|------|
| Alg.1 (this work) | 112    | 112    | 12.0 |
| Alg.2 (this work) | 112    | 112    | 12.0 |
| Alg.1 U Alg.2 (this work) | 114    | 116    | 14.0 |

![Diagram](image2)

**FIGURE 9.** An optimized, by applying the composition of Algorithms 1 and 2, (8,8) S-box
optimized? As summarized in [91], the maximal nonlinearity value achieved in balanced Boolean functions with 8 variables is 116. Therefore, if an ACNV for an (8, 8) bijective S-box greater than 116.0 is ever found, at least one of its eight components will possess nonlinearity value 118, which will finally give an answer to the long-standing problem of the maximum possible nonlinearity value of an eight variable balanced Boolean functions. Furthermore, as shown in [92], the upper bound for eight variable balanced Boolean functions is less than 120. Thus, the maximum theoretical possible ACNV of (8, 8) bijective S-boxes is less or equal to 118.0, but most probably, considering the academic skepticism that eight variable balanced Boolean functions with nonlinearity value 118 really exist, less or equal to 116.0.

REFERENCES

[1] M. Matsui, “Linear cryptanalysis method for DES cipher,” in Workshop on the Theory and Application of Cryptographic Techniques. Springer, 1993, pp. 386–397.

[2] E. Biham, “On Matsui’s linear cryptanalysis,” in Workshop on the Theory and Application of Cryptographic Techniques. Springer, 1994, pp. 341–353.

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

[4] D. Wagner, “The boomerang attack,” in International Workshop on Fast Software Encryption. Springer, 1999, pp. 156–170.

[5] T. Jakobsen and L. R. Knudsen, “The interpolation attack on block ciphers,” in International Workshop on Fast Software Encryption. Springer, 1997, pp. 28–40.

[6] J. Daemen and V. Rijmen, The Design of Rijndael: AES—the Advanced Encryption Standard. Springer Science & Business Media, 2002.

[7] P. Barreto, V. Rijmen et al., “The Whirlpool hashing function,” in First open NESSIE Workshop, Leuven, Belgium, vol. 13. Citeseer, 2000, p. 14.

[8] K. Aoki, T. Ichikawa, M. Kanda, M. Matsui, S. Moriai, J. Nakajima, and T. Tokita, Camellia: A 128-bit block cipher suitable for multiple platforms and design analysis, in International Workshop on Selected Areas in Cryptography. Springer, 2000, pp. 39–56.

[9] F. Özkaynak, “On the effect of chaotic system in performance characteristics of chaos based s-box designs,” Physica A: Statistical Mechanics and its Applications, p. 124072, 2019.

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

[11] M. Ahmad, H. Haleem, and P. M. Khan, “A new chaotic substitution box design for block ciphers,” in 2014 International Conference on Signal Processing and Integrated Networks (SPIN). IEEE, 2014, pp. 255–258.

[12] A. A. Alzaidi, M. Ahmad, M. N. Doja, E. Al Solami, and M. S. Beg, “A New 1D Chaotic Map and β-Hill Climbing for Generating Substitution-Boxes,” IEEE Access, vol. 6, pp. 55405–55418, 2018.

[13] A. A. Alzaidi, M. Ahmad, H. S. Ahmed, and E. A. Solami, “Sine-Cosine Optimization-Based Bijective Substitution-Boxes Construction Using Enhanced Dynamics of Chaotic Map,” Complexity, vol. 2018, 2018.

[14] Q. Lai, A. Akgul, C. Li, G. Xu, and U. Çavuşoğlu, “A new chaotic system with multiple attractors: Dynamic analysis, circuit realization and S-Box design,” Entropy, vol. 20, no. 1, p. 12, 2018.

[15] A. Belazi, M. Khan, A. A. Al-Latif, and S. Belghith, “Efficient cryptosystem approaches: S-boxes and permutation–substitution-based encryption,” Nonlinear Dynamics, vol. 87, no. 1, pp. 337–361, 2017.

[16] D. Lambic, “S-box design method based on improved one-dimensional discrete chaotic map,” Journal of Information and Telecommunication, vol. 2, no. 2, pp. 181–191, 2018.

[17] J. Peng, S. Jin, L. Lei, and R. Jia, “A novel method for designing dynamical key-dependent s-boxes based on hyperchaotic system,” Int. J. Adv. Comput. Technol., vol. 4, pp. 282–289, 2016.

[18] E. Al Solami, M. Ahmad, C. Volos, M. Doja, and M. Beg, “A new hyperchaotic system-based design for efficient bijective substitution-boxes,” Entropy, vol. 20, no. 7, p. 525, 2018.
F. Özkaynak, “From biometric data to cryptographic primitives: A new

I. Hussain, T. Shah, H. Mahmood, and M. A. Gondal, “Construction of

G. Liu, W. Yang, W. Liu, and Y. Dai, “Designing S-boxes based on 3-

I. Hussain, T. Shah, H. Mahmood, and M. A. Gondal, “A projective general

X. Wang, A. Akgul, V. T. Pham, D. Vo Hoang, and X. Q. Nguyen, “A chaotic system with infinite equilibria and its S-box constructing application,” Applied Sciences, vol. 8, no. 11, p. 2132, 2018.

[61] X. Wang, U. Çavuşoğlu, S. Çağar, A. Akgul, V. T. Pham, S. Jafari, F. E. Alsaa’di, and X. Q. Nguyen, “S-box based image encryption application using a chaotic system without equilibrium,” Applied Sciences, vol. 9, no. 4, p. 781, 2019.

[68] X. Wang, Ü. Çavuş, S. Kacar, A. Akgul, V. T. Pham, S. Jafari, F. E. Alsaa’di, and X. Q. Nguyen, “S-box based image encryption application using a chaotic system with infinite equilibria,” Applied Sciences, vol. 9, no. 4, p. 781, 2019.

[72] D. Lambić, “A novel method of S-box design based on discrete chaotic map,” Nonlinear Dynamics, vol. 87, no. 4, pp. 2407–2413, 2017.

[71] F. Özkaynak, “Construction of robust substitution boxes based on chaotic systems,” Neural Computing and Applications, vol. 31, no. 8, pp. 3317–3326, 2019.

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

[74] M. Ahmad, D. Bhatia, and Y. Hassan, “A novel ant colony optimization based scheme for substitution box design,” Procedia Computer Science, vol. 57, no. 2015, pp. 572–580, 2015.

[75] H. A. Ahmed, M. F. Zolkipi, and M. Ahmad, “A novel efficient substitution-box design based on firefly algorithm and discrete chaotic map,” Neural Computing and Applications, vol. 31, no. 11, pp. 7201–7210, 2019.

[76] L. Yi, X. Tong, Z. Wang, M. Zhang, H. Zhu, and J. Liu, “A novel block encryption algorithm based on chaotic S-box for wireless sensor network,” IEEE Access, vol. 7, pp. 53 079–53 090, 2019.

[77] X. Zhang, Z. Zhao, and J. Wang, “Chaotic image encryption based on circular substitution box and key stream buffer,” Signal Processing: Image Communication, vol. 29, no. 8, pp. 902–913, 2014.

[78] D. Lambić, “A novel method of S-box design based on chaotic map and composition method,” Chaos, Solitons & Fractals, vol. 58, pp. 16–21, 2014.

[79] I. Hussain, T. Shah, M. A. Gondal, and H. Mahmood, “An efficient approach for the construction of LFT S-boxes using chaotic logmap,” Nonlinear Dynamics, vol. 71, no. 1-2, pp. 133–140, 2013.

[80] Hussain, Iqtadar and Shah, Tariq and Gondal, Muhammad Asif and Mahmood, Hasan, “Efficient method for designing chaotic S-boxes based on generalized Baker’s map and TDERC chaotic sequence,” Nonlinear Dynamics, vol. 74, no. 1-2, pp. 271–275, 2013.

[81] A. Belazi, A. A. A. El-Latif, A.-V. Diaconu, R. Rouhama, and S. Belghit, “Chaos-based partial image encryption scheme based on linear fractional and lifting wavelet transforms;” Optics and Lasers in Engineering, vol. 88, pp. 37–50, 2017.

[82] M. Ş. Açikkapı, F. Özkaynak, and A. B. Özer, “Side-channel analysis of chaos-based substitution box structures,” IEEE Access, vol. 7, pp. 79 030–79 043, 2019.

[83] A. Rosenfeld, “Fuzzy graphs,” in Fuzzy sets and their applications to cognitive and decision processes. Elsevier, 1975, pp. 77–95.

[84] F. Hao, D.-S. Park, S. Li, and H. M. Lee, “Mining A-Maximal Cliques from a Fuzzy Graph,” Sustainability, vol. 8, no. 6, p. 553, 2016.

[85] Y. M. Ermoliev and R.-B. Wets, Numerical techniques for stochastic optimization. Springer-Verlag, 1988.

[86] M. A. Thathachar and P. S. Sastry, Networks of learning automata: Techniques for online stochastic optimization. Springer Science & Business Media, 2011.

[87] H. Narasimhan, S. Li, P. Kar, S. Chawla, and F. Sebastiani, “Stochastic optimization techniques for quantification performance measures,” stat, vol. 1050, p. 13, 2016.

[88] E. Alba and B. Dorronsoro, “The exploration/exploitation tradeoff in dynamic cellular genetic algorithms;” IEEE transactions on evolutionary computation, vol. 9, no. 2, pp. 126–142, 2005.

[89] W. G. Macready and D. H. Wolpert, “Bandit problems and the exploitation/exploitation tradeoff,” IEEE Transactions on evolutionary computation, vol. 2, no. 1, pp. 2–22, 1998.

[90] S. Li, C. Gentile, A. Karatzoglou, and G. Zappella, “Data-dependent clustering in exploration-exploitation algorithms,” CoRR, abs/1502.03473, 2015.

[91] S. Piek, R. Santana, and D. Jakobovic, “Maximal nonlinearity in balanced boolean functions with even number of inputs, revisited,” in 2016 IEEE Congress on Evolutionary Computation (CEC). IEEE, 2016, pp. 3222–3229.

[92] P. Sarkar and S. Maitra, “Nonlinearity bounds and constructions of resilient boolean functions,” in Annual International Cryptology Conference. Springer, 2000, pp. 515–532.
MIROSLAV M. DIMITROV was born in Yambol, Bulgaria in 1985. He received the B.S. degree in Informatics and M.S. degree in Information Security, both from the Faculty of Mathematics and Informatics, Sofia University. He is currently pursuing the Ph.D. degree in Informatics at Bulgarian Academy of Sciences. His research interests include cryptology, algorithms, and sequences.