IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received June 17, 2020, accepted June 21, 2020, date of publication June 24, 2020, date of current version July 3, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3004526

On the Design of Chaos-Based S-Boxes

MIROSLAV M. DIMITROV, (Graduate Student Member, IEEE)
Institute of Mathematics and Informatics, Bulgarian Academy of Sciences, 1113 Sofia, Bulgaria
e-mail: mirdim@math.bas.bg

This work was supported by the Bulgarian National Science Fund under Contract 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 $x_1, \cdots, x_n$ is a mapping $f : B^n \mapsto B$ from n binary inputs $x = (x_1, x_2, \cdots, x_n) \in B^n$ to one binary output $y = f(x) \in B$.

The associate editor coordinating the review of this manuscript and approving it for publication was Chao-Yang Chen.

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

$$ANF_f = a_0 \oplus a_1x_1 \oplus a_2x_2 \oplus a_{1,2,\cdots,n}x_1x_2\cdots x_n,$$

where the coefficients $a_{i\ldots j} \in B = \{0, 1\}$.

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:

$$l_w(x) =< w, x >= w_1x_1 \oplus w_2x_2 \oplus \cdots \oplus w_nx_n,$$

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 $y = (y_1, y_2, \cdots, y_m) \in B^m$ by $S(x) = y$ to each $x = (x_1, x_2, \cdots, x_n) \in B^n$, 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^m outputs are present. For example, the (n, n) 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 $S(n, n)$ 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.

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

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 is the consequent linear function with length n 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 $S_{NL} = 2^{n-1} - max\left(\{abs(w_i)\}\right)$, 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 β-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 $F_{LUT} = [f0, f1, f2, f3, f4]$. 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 $f_1 \oplus f_3 \oplus f_4$, 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

TABLE 1. Comparison of nonlinearity of some CF-based (8, 8) S-boxes.

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	108	106	110	108	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	106	104
Liu [19]	108	102	104	104	102	104	106	106
Lambic [16]	106	106	106	106	106	108	108	106
Peng [17]	102	102	104	104	102	100	106	102
Tian [21]	106	106	110	108	106	108	108	108

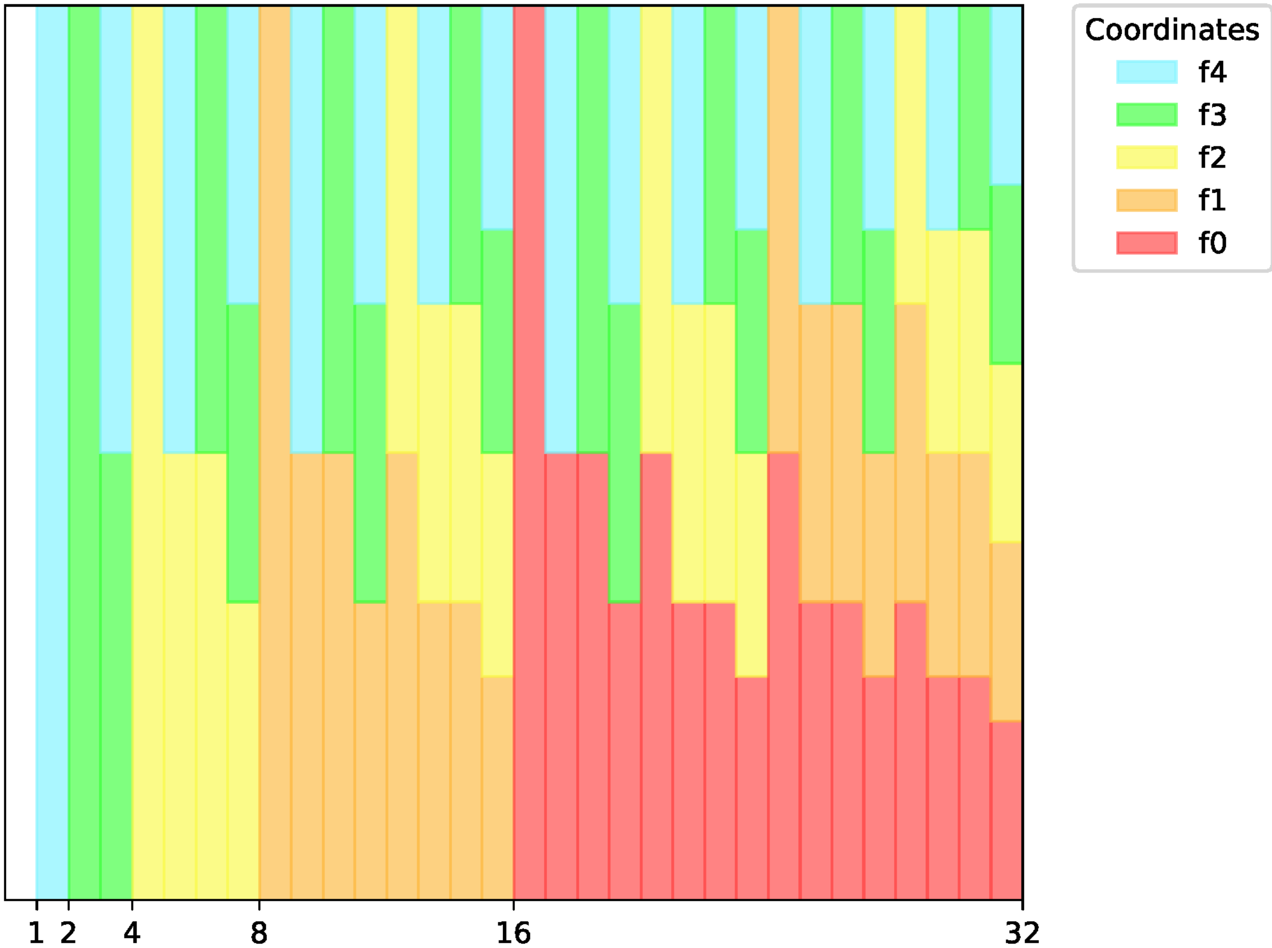

FIGURE 1. Coordinate decomposition of a (5, 5) S-box LAT.

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 [1], [2] [24].

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.

TABLE 2. Real nonlinearity values (NL) of the S-boxes given in Table 1.

Method	min	max	ACNV	NL
Ahmad [10]	106	110	107.5	90
Ahmad [11]	102	108	105.25	94
Al Solami [18]	106	110	108.5	94
Alzaidi [12]	110	112	110.25	96
Alzaidi [13]	108	110	109.5	94
Belazi [15]	102	108	105.25	88
Lai [14]	104	110	105.5	92
Liu [19]	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

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 [25]. 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

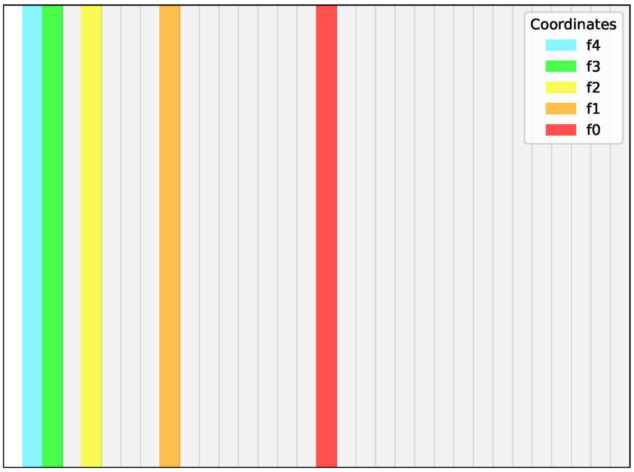

FIGURE 2. Columns of interest of a (5, 5) S-box LAT.

$E(S)$ is created, s.t. $E(S) = \sum_{p=0}^{n-1} \sum_{x \in C_{2^p}} |x|^M$, where M denotes a magnitude of our choice. The restriction $x \in C_{2^p}$ 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 function threats as significant the elements inside the colored columns of S_{LAT} illustrated in Figure 2.

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

Algorithm 1 Stochastic Hill Climbing Algorithm for an S-box ACNV Optimization

1: $s \leftarrow R(n)$ ▷ the function R(n) generates pseudo-random bijective S-box $S(n,n)$
2: **repeat**
3: $sdupl \leftarrow s$
4: RT(sdupl) ▷ the function RT(S) make a random transposition in S
5: **if** E($sdupl$) < E(s) **then**
6: $s \leftarrow sdupl$
7: **end if**
8: **until** STOP condition is reached ▷ 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. $W \in S^I \iff W \in S^T \wedge E(W) < E(S)$. In case $|S^I| = 1$, and we are allowed to randomly pick $\frac{|S^T|}{2}$ 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.

[1]hill climbing without neighborhood search

TABLE 3. Nonlinearities of S_c by coordinates.

Method	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8
this work	114	114	114	114	114	114	114	114

TABLE 4. SAC, Coordinate-average and final nonlinearity of S_c.

Method	min	max	ACNV	SAC	NL
this work	114	114	114	0.5000000	96

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

In Figure 7 the DLUT, in a hexadecimal format, of an optimized S-box $S_c(8,8)$ 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_c to **5d**. The characteristics of S_c are summarized in Tables 3 and 4.

In [26], 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 [27] and Skipjack [28] 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 [29].

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 256! ≈ 2^{1684} different bijective S-boxes. Despite algorithm 1 efficiency in

	00	01	02	03	04	05	06	07	08	09	0a	0b	0c	0d	0e	0f
00	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
10	c2	92	c9	7d	fb	59	47	f0	ad	c4	c0	ae	c8	a4	3a	a8
20	b7	fd	93	26	36	3f	fa	cc	34	a5	e5	f1	71	d8	b5	11
30	04	c7	33	c3	18	96	05	9c	07	12	88	e1	eb	2f	b2	35
40	09	82	2c	1b	19	6e	1a	a0	42	3b	d6	bb	29	e2	27	85
50	53	d1	20	ed	10	fc	31	4b	6a	cb	be	39	8e	4c	58	ef
60	d0	8f	a2	f7	43	0d	23	84	46	e9	03	7e	50	38	9f	2a
70	51	a3	40	cf	83	9d	28	f3	bd	b6	de	21	00	ff	f5	d2
80	cd	d4	13	ee	5d	97	44	17	0c	a7	7f	3d	64	5f	5b	73
90	41	a1	4f	dd	22	aa	90	80	45	ec	f8	14	db	5e	0b	9b
a0	e0	7a	6d	0a	49	06	24	5c	ca	d3	ac	66	91	95	e4	32
b0	e7	9a	37	79	8d	d5	4e	a9	6c	56	f4	ea	65	72	af	08
c0	ba	78	25	2e	1c	a6	b4	c6	e8	dc	74	1f	5a	bc	8b	8a
d0	70	3e	b1	e6	48	02	f6	0e	61	75	57	b9	86	c1	1d	9e
e0	e3	b8	98	15	69	d9	4a	94	da	1e	87	f9	ce	55	3c	df
f0	8c	81	89	4d	bf	62	52	68	60	99	2d	0f	b0	54	b3	16

FIGURE 3. An optimized AES S-box using Algorithm 1.

	00	01	02	03	04	05	06	07	08	09	0a	0b	0c	0d	0e	0f
00	18	23	c6	e8	87	b8	01	4f	36	a6	d2	f5	79	6f	91	52
10	f8	46	ae	1f	df	5d	0d	83	ca	28	e0	ec	22	6d	fe	2a
20	53	67	37	f4	3f	d0	2d	ce	38	59	2b	0c	c9	e3	6b	b1
30	9d	5b	10	26	cb	bc	da	73	e6	33	41	09	34	7d	fc	98
40	92	dd	74	35	bb	ab	5f	9e	c8	5c	a7	8d	af	4c	81	3b
50	19	32	0b	71	8f	89	13	63	c2	51	4a	21	9a	0a	06	04
60	93	0f	d5	82	ba	cd	1e	84	ff	7a	85	47	e1	c1	1a	e4
70	6c	54	c3	f3	60	ad	8c	5e	1c	44	02	dc	1b	a1	a3	3d
80	b5	00	ed	2f	78	d9	9f	49	17	69	6a	f7	31	77	30	ef
90	2e	d1	a2	ea	fb	9b	97	40	d8	9c	55	4d	f2	75	27	8a
a0	eb	86	5a	fa	42	ee	a8	b7	f1	57	29	80	90	12	39	65
b0	cf	3c	76	f0	7b	61	62	03	11	45	20	16	43	c7	bf	d6
c0	05	4e	4b	1d	db	b3	7e	f6	72	66	de	a5	e7	f9	7f	b2
d0	25	b0	be	50	68	70	bd	07	d3	6e	c4	e5	3e	b9	8e	a9
e0	8b	94	08	15	cc	aa	fd	b4	c5	58	a0	56	2c	0e	14	c0
f0	24	3a	ac	99	e9	b6	d4	95	48	a4	88	e2	d7	7c	64	96

FIGURE 4. An optimized Whirlpool S-box using Algorithm 1.

	00	01	02	03	04	05	06	07	08	09	0a	0b	0c	0d	0e	0f
00	1e	75	5f	e1	99	fc	89	2f	86	ee	f1	7b	23	52	10	94
10	4f	59	2c	8b	f8	42	30	00	6e	84	35	70	a0	c3	34	6f
20	4e	41	01	78	8f	a8	07	6c	62	af	7f	22	60	79	90	ec
30	68	f4	c4	32	1d	8c	0e	ce	de	3f	44	1f	40	98	43	d6
40	e7	cc	e0	e6	d1	9a	1a	b3	28	1c	7c	0c	b9	c0	71	21
50	cb	11	9e	e3	48	cd	e9	57	f5	63	36	1b	b8	bf	9d	a7
60	61	d7	f3	a9	12	fd	c1	b7	8e	a6	6b	66	72	64	85	d5
70	4b	7e	67	3c	65	17	ba	4a	97	29	83	6a	ae	f0	e4	2e
80	77	74	e8	2a	ac	95	3a	a2	3d	fa	50	58	ea	9f	93	33
90	b5	5c	06	51	a3	76	7a	80	bd	16	39	0a	03	73	d0	05
a0	f9	b0	55	2d	b2	49	f7	19	c6	45	d2	d8	5d	f2	87	ed
b0	da	eb	91	ca	3b	47	cf	fb	c7	dc	f6	a4	df	fe	b1	09
c0	0f	0d	2b	26	14	ff	4d	bc	02	81	b4	be	15	c5	d4	27
d0	88	04	82	c8	46	e5	24	c2	9b	7d	8d	d9	38	6d	ef	a1
e0	dd	69	5a	54	9c	53	25	20	5b	db	37	5e	ab	56	0b	4c
f0	13	3e	8a	d3	ad	31	08	96	a5	18	b6	e2	aa	92	c9	bb

FIGURE 5. An optimized Fantomas S-box using Algorithm 1.

finding S-boxes with better ACNV, we had never reached an ACNV greater than 114. However, we have found out that the multi-armed bandit problem [30], [31] [32], [33] 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

	00	01	02	03	04	05	06	07	08	09	0a	0b	0c	0d	0e	0f
00	a3	d7	09	83	f8	48	f6	f4	b3	21	15	78	99	b1	af	f9
10	5a	6b	69	0a	0f	27	ca	2f	52	95	c3	0d	4e	a0	c5	2c
20	8a	12	38	6e	bb	d9	c8	e2	cd	02	7e	5f	f1	87	19	8f
30	96	fe	2d	de	b2	6f	b6	ac	0c	ae	e5	7c	f7	43	aa	2a
40	b9	f3	7b	1e	eb	9a	cf	73	ee	61	1a	a9	50	9b	ff	b8
50	76	39	92	7f	3a	8c	bf	14	60	58	80	e1	66	0b	86	90
60	1f	91	62	ed	33	a6	65	e0	67	d4	82	d6	6d	98	e6	74
70	e8	44	93	c2	b0	fc	9d	6a	81	fa	56	42	4d	05	4a	5c
80	d8	5d	df	a4	49	1d	9e	16	4c	d2	be	00	ba	c6	47	53
90	84	c0	55	3f	1c	c7	d5	a2	88	34	dc	c9	7d	3c	31	20
a0	d3	2e	e9	28	9c	8e	23	ce	dd	94	85	a5	22	79	a8	40
b0	30	4b	e4	1b	d1	89	a7	3b	11	c1	fd	36	e7	cc	5b	64
c0	9f	04	a1	51	97	13	26	f0	29	db	cb	7a	75	8b	77	d0
d0	3d	ef	bc	70	71	63	37	2b	ec	41	da	24	ad	8d	10	18
e0	01	b5	54	07	35	4f	b7	c4	32	17	b4	fb	72	06	ab	0e
f0	5e	6c	68	f2	57	25	f5	e3	bd	08	3e	03	45	59	ea	46

FIGURE 6. An optimized Skipjack S-box using Algorithm 1.

	00	01	02	03	04	05	06	07	08	09	0a	0b	0c	0d	0e	0f
00	ab	f0	5e	3f	fa	e2	6f	8e	3c	36	30	db	29	73	da	45
10	87	f9	60	3b	bf	a4	c7	0c	a9	c0	f3	cb	68	ff	ee	a6
20	90	57	f2	77	ef	c2	78	b7	94	32	e6	4d	53	6d	26	98
30	c1	2c	2a	9a	12	2b	ea	e8	17	7c	5c	6e	50	d9	f6	88
40	83	69	5a	67	af	b9	1a	b8	8a	d4	b4	a0	cc	e1	24	c6
50	be	1f	a1	51	9f	64	4e	4f	2f	85	6b	76	86	35	4b	ed
60	81	84	39	13	62	c3	9e	dc	d0	66	5f	44	de	1c	bd	34
70	1d	1e	2d	6c	a2	46	97	c5	37	61	a3	56	fe	f7	d5	38
80	ce	05	09	18	aa	fc	91	28	9b	10	e9	0b	71	dd	e7	23
90	7f	72	59	6a	43	fd	d1	e4	f8	0d	55	74	c8	f5	27	65
a0	93	c4	19	49	00	20	3d	2e	a8	d3	01	7d	25	0e	f4	33
b0	02	04	0a	14	16	ae	31	11	cf	79	8f	d8	8b	d7	ca	b3
c0	bb	3e	0f	92	df	40	4c	cd	ac	22	5b	a5	bc	f1	75	89
d0	96	b1	e3	d2	7a	1b	70	58	03	47	80	9c	06	ba	c9	54
e0	ad	41	99	48	7e	3a	95	e0	ec	07	63	7b	b2	21	b0	4a
f0	8d	d6	15	fb	9d	5d	8c	42	08	b6	eb	a7	b5	e5	52	82

FIGURE 7. An optimized S-box $S_c(8, 8)$ using Algorithm 1.

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 $\binom{2^n}{2}$ bit transpositions in a given coordinate of S_{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_{LUT}, flipping the bit b_i will result in **some** other row $R = b_1 b_2 \cdots \overline{b_i} \cdots b_n$ in S_{LUT}. Otherwise, if R is not among

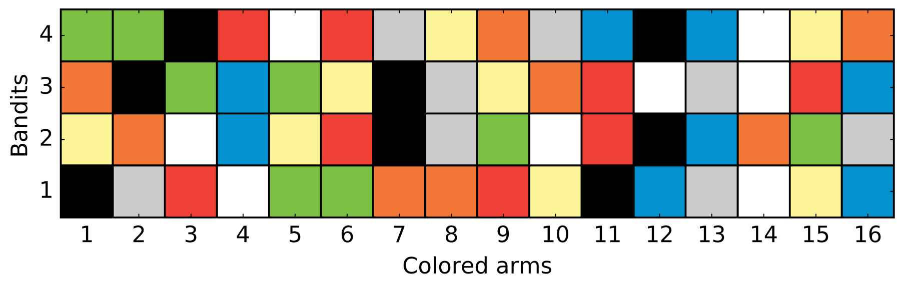

FIGURE 8. An example of 8-armed 4-bandit problem transformation of the S-box X.

the rows of S_{LUT}, S is not surjective, therefore not bijective, which contradicts our initial choice of S.

2) **One-to-one Maping:** If $b_1 b_2 \cdots b_i \cdots b_n$ is a row from S_{LUT}, flipping the bit b_i will result in **only one** row $R = b_1 b_2 \cdots \overline{b_i} \cdots b_n$ in S_{LUT}. Otherwise, if some other row R' of S_{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 $b_1 b_2 \cdots b_{i-1} b_{i+1} \cdots b_n$ 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_{LUT}, which differ only in bit position i.

For example, let's consider an S-box X(4,4), with $X_{DLUT} = [15, 14, 9, 2, 11, 3, 12, 4, 1, 13, 7, 8, 6, 10, 5, 0]$. 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_{DLUT} (starting from 1).

As an illustration, if we activate the white arm of bandit 1, we interchange the elements of X_{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 Multi-armed Bandit Algorithm for an S-box ACNV Optimization

$s \leftarrow R(n)$ ▷ the function R(n) generates pseudo-random bijective S-box $S(n, n)$
$\Omega \leftarrow MODEL(s)$ ▷ We transform the S-box s to a 2^{n-1}-armed n-bandit problem
repeat
  $bandit \leftarrow random(1, n, \Lambda)$ ▷ We choose a random bandit from $[1, n]$, based on some profit-maximizing strategy Λ
  $arm \leftarrow random(1, 2^{n-1})$ ▷ We choose a random arm from $\left[1, 2^{n-1}\right]$
  $oldBandit \leftarrow \mathrm{E}(bandit)$
  Activate($bandit$, arm)
  if $\mathrm{E}(bandit) < \mathrm{E}(oldBandit)$ **then**
    $\Omega \leftarrow MODEL(s)$ ▷ We update the model
  else
    Activate($bandit$, arm) ▷ We resume the original state of the bandit
  end if
until STOP condition is reached ▷ reaching $n2^{n-1}$ consequent unsuccessful attempts

	00	01	02	03	04	05	06	07	08	09	0a	0b	0c	0d	0e	0f
00	a9	7b	bb	c9	0a	55	d1	c1	a3	1a	24	26	bf	72	f6	0d
10	4a	73	e2	f1	3a	d3	35	b2	64	93	f5	d7	ff	dc	cb	4e
20	de	7a	a2	98	d9	87	b3	a5	28	ba	a0	45	56	67	61	0c
30	a7	33	53	2d	e7	58	7e	b6	37	71	1e	10	d0	e0	b7	49
40	96	91	6e	b0	f9	5b	fb	13	8a	db	ad	a1	8c	39	22	ee
50	89	4f	50	da	07	75	65	bd	9f	18	cd	17	41	be	2f	40
60	ca	05	ae	32	94	f7	a8	b1	aa	f8	e9	e4	82	54	01	69
70	27	81	5c	84	7f	b4	29	d2	fc	0e	a4	36	90	2e	15	00
80	4c	51	25	11	16	f3	3d	8d	9c	6c	95	ef	76	cc	8b	dd
90	2b	f4	ce	43	62	d4	74	fe	92	c2	7c	80	2a	21	68	bc
a0	c0	23	af	e3	78	6f	e1	eb	03	38	09	42	d6	ed	ec	02
b0	1d	fa	5e	b9	c8	c7	46	14	e6	99	9d	04	c4	d8	3f	9b
c0	e5	4d	31	63	79	3c	d5	f0	47	57	4b	c6	f2	2c	70	b5
d0	cf	9e	0b	3e	1f	5a	a6	6a	6b	12	1b	77	5f	48	ac	3b
e0	44	34	5d	ea	20	85	8f	30	9a	ab	1c	c3	59	8e	fd	08
f0	b8	e8	06	6d	66	7d	df	60	52	83	88	19	0f	97	c5	86

FIGURE 9. An optimized, by applying the composition of Algorithms 1 and 2, (8,8) S-box.

VII. RESULTS PART II

Our implementation of Algorithm 2 is based on a simple strategy Λ - we always choose a bandit, which posses the lowest nonlinearity. In case there are several bandits sharing the lowest value of nonlinearity, we choose one of them at random.

We launched Algorithm 2 as a stand-alone optimization routine, starting from pseudo-randomly generated S-boxes,

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
[34] [35]	84	106	100.0
[36]	98	108	102.3
[37]	96	106	102.5
[38]	100	106	103.0
[39]	96	106	103.0
[40]	98	108	103.0
[41]	98	108	103.2
[42]	100	106	103.2
[43]	99	106	103.3
[44]	96	108	103.5
[45]	101	108	103.8
[46]	101	106	103.8
[47]	102	106	104.0
[48]	98	108	104.0
[49]	100	106	104.0
[50]	102	106	104.0
[51]	98	108	104.0
[19]	102	108	104.5
[52] [53]	100	108	104.7
[54] [55]	102	108	104.7
[56]	100	108	104.75
[57]	100	107	104.8
[58]	104	106	105.0
[59]	102	108	105.2
[15]	102	108	105.3
[60]	100	110	105.5
[61]	98	110	105.5
[62]	102	110	105.5
[63]	104	108	105.7
[64]	102	108	106.0
[65] [66] [67]a	104	110	106.0
[67]c	106	108	106.0
[68]	104	110	106.2
[69]	104	110	106.5
[16]	106	108	106.5
[70] [71]	106	108	106.7
[72]	104	108	106.7
[73]	106	110	107.0
[67]b	104	108	107.0
[74]	106	108	107.5
[75]	106	110	107.75
[20]	108	108	108.0
[76]	104	110	108.0
[18]	106	110	108.5
[77]	108	112	109.0
[78] [79] [80]	112	112	112.0
Alg.2 (this work)	112	112	112.0
Alg.1 (this work)	114	114	114.0
Alg.1 $\bigcup$ Alg.2 (this work)	114	116	114.5

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 [6] 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 [81], 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 [82], [83], the stochastic optimization techniques [84], [85] [86], or the exploration-exploitation algorithms [87], [88] [89], could be exploited to further maximize the ACNV of a given S-box.

An interesting open question to be answered is to what extend the ACNV value of an (8,8) bijective S-box could be optimized? As summarized in [90], 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 posses 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 [91], 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 *Proc. Workshop Theory Appl. Cryptograph. Techn.* Springer, 1993, pp. 386–397.
[2] E. Biham, "On Matsui's linear cryptanalysis," in *Proc. Workshop Theory Appl. Cryptograph. Techn.* Berlin, Germany: Springer, 1994, pp. 341–355.

[3] E. Biham and A. Shamir, "Differential cryptanalysis of Des.like cryptosystems," *J. Cryptol.*, vol. 4, no. 1, pp. 3–72, Jan. 1991.
[4] D. Wagner, "The boomerang attack," in *Proc. Int. Workshop Fast Softw. Encryption*. Berlin, Germany: Springer, 1999, pp. 156–170.
[5] T. Jakobsen and L. R. Knudsen, "The interpolation attack on block ciphers," in *Proc. Int. Workshop Fast Softw. Encryption*. Berlin, Germany: Springer, 1997, pp. 28–40.
[6] J. Daemen and V. Rijmen, *The Design of Rijndael: AES-The Advanced Encryption Standard*. Berlin, Germany: Springer-Verlag, 2002.
[7] P. Barreto and V. Rijmen, "The Whirlpool hashing function," in *1st open NESSIE Workshop*, vol. 13. Leuven, Belgium: 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—Design andanalysis," in *Proc. Int. Workshop Sel. Areas Cryptogr.* Springer, 2000, pp. 39–56.
[9] F. Özkaynak, "On the effect of chaotic system in performance characteristics of chaos based s-box designs," *Phys. A, Stat. Mech. Appl.*, vol. 550, Jul. 2020, Art. no. 124072.
[10] M. Ahmad, N. Mittal, P. Garg, and M. Maftab Khan, "Efficient cryptographic substitution box design using travelling salesman problem and chaos," *Perspect. Sci.*, vol. 8, pp. 465–468, Sep. 2016.
[11] M. Ahmad, H. Haleem, and P. M. Khan, "A new chaotic substitution box design for block ciphers," in *Proc. Int. Conf. Signal Process. Integr. Netw. (SPIN)*, Feb. 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, pp. 1–16, Dec. 2018.
[14] Q. Lai, A. Akgul, C. Li, G. Xu, and Ü. Çavuşoğlu, "A new chaotic system with multiple attractors: Dynamic analysis, circuit realization and S-Box design," *Entropy*, vol. 20, no. 1, p. 12, Dec. 2017.
[15] A. Belazi, M. Khan, A. A. A. El-Latif, and S. Belghith, "Efficient cryptosystem approaches: S-boxes and permutation–substitution-based encryption," *Nonlinear Dyn.*, vol. 87, no. 1, pp. 337–361, 2017.
[16] D. Lambić, "S-box design method based on improved one-dimensional discrete chaotic map," *J. Inf. Telecommun.*, vol. 2, no. 2, pp. 181–191, Apr. 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. Advancements Comput. Technol.*, vol. 4, no. 18, pp. 282–289, Oct. 2012.
[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, Jul. 2018.
[19] L. Liu, Y. Zhang, and X. Wang, "A novel method for constructing the S-Box based on spatiotemporal chaotic dynamics," *Appl. Sci.*, vol. 8, no. 12, p. 2650, Dec. 2018.
[20] Y. Wang, K.-W. Wong, C. Li, and Y. Li, "A novel method to design S-box based on chaotic map and genetic algorithm," *Phys. Lett. A*, vol. 376, nos. 6–7, pp. 827–833, Jan. 2012.
[21] Y. Tian and Z. Lu, "Chaotic S-box: Intertwining logistic map and bacterial foraging optimization," *Math. Problems Eng.*, vol. 2017, Nov. 2017, Art. no. 6969312.
[22] C. Carlet, "Vectorial Boolean functions for cryptography," in *Boolean Models and Methods in Mathematics, Computer Science, and Engineering*, vol. 134. 2010, pp. 398–469.
[23] R. Forrie, "The strict avalanche criterion: Spectral properties of Boolean functions and an extended definition," in *Proc. Conf. Theory Appl. Cryptogr.* New York, NY, USA: Springer, 1988, pp. 450–468.
[24] H. M. Heys, "A tutorial on linear and differential cryptanalysis," *Cryptologia*, vol. 26, no. 3, pp. 189–221, Jul. 2002.
[25] W. Meier and O. Staffelbach, "Nonlinearity criteria for cryptographic functions," in *Proc. Workshop Theory Appl. Cryptograph. Techn.* Berlin, Germany: Springer, 1989, pp. 549–562.
[26] E. Tanyildizi and F. Özkaynak, "A new chaotic S-box generation method using parameter optimization of one dimensional chaotic maps," *IEEE Access*, vol. 7, pp. 117829–117838, 2019.
[27] V. Grosso, G. Leurent, F.-X. Standaert, and K. Varici, "LS-designs: Bitslice encryption for efficient masked software implementations," in *Proc. Int. Workshop Fast Softw. Encryption*. Springer, 2014, pp. 18–37.
[28] *Skipjack and KEA Algorithms Specifications, v2.0*, U.S. Dept. Commerce/Nat. Inst. Standards Technol., Gaithersburg, MD, USA, 1998.
[29] T Sage Developers "SageMath, version 7.1," The Sage Mathematics Software System, 2016.
[30] D. A. Berry and B. Fristedt, *Bandit Problems: Sequential Allocation of Experiments (Monographs on Statistics and Applied Probability)*, vol. 5. London, U.K.: Chapman Hall, 1985, pp. 71–87.
[31] M. N. Katehakis and A. F. Veinott, "The multi-armed bandit problem: Decomposition and computation," *Math. Oper. Res.*, vol. 12, no. 2, pp. 262–268, May 1987.
[32] S. Li, "The art of clustering bandits," Ph.D. dissertation, Università degli Studi dell'Insubria, Varese, Italy, 2016. [Online]. Available: http://insubriaspace.cineca.it/handle/10277/729
[33] S. Li, A. Karatzoglou, and C. Gentile, "Collaborative filtering bandits," in *Proc. 39th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR)*, 2016, pp. 539–548.
[34] M. Khan, "A novel image encryption scheme based on multiple chaotic S-boxes," *Nonlinear Dyn.*, vol. 82, nos. 1–2, pp. 527–533, Oct. 2015.
[35] M. Khan, T. Shah, and S. I. Batool, "Construction of S-box based on chaotic Boolean functions and its application in image encryption," *Neural Comput. Appl.*, vol. 27, no. 3, pp. 677–685, Apr. 2016.
[36] S. S. Jamal, M. U. Khan, and T. Shah, "A watermarking technique with chaotic fractional S-Box transformation," *Wireless Pers. Commun.*, vol. 90, no. 4, pp. 2033–2049, Oct. 2016.
[37] M. Khan and Z. Asghar, "A novel construction of substitution box for image encryption applications with Gingerbreadman chaotic map and S_8 permutation," *Neural Comput. Appl.*, vol. 29, no. 4, pp. 993–999, 2018.
[38] G. Chen, Y. Chen, and X. Liao, "An extended method for obtaining S-boxes based on three-dimensional chaotic baker maps," *Chaos, Solitons Fractals*, vol. 31, no. 3, pp. 571–579, Feb. 2007.
[39] M. Khan, T. Shah, H. Mahmood, M. A. Gondal, and I. Hussain, "A novel technique for the construction of strong S-boxes based on chaotic lorenz systems," *Nonlinear Dyn.*, vol. 70, no. 3, pp. 2303–2311, Nov. 2012.
[40] M. Khan, T. Shah, H. Mahmood, and M. A. Gondal, "An efficient method for the construction of block cipher with multi-chaotic systems," *Nonlinear Dyn.*, vol. 71, no. 3, pp. 489–492, Feb. 2013.
[41] G. Jakimoski and L. Kocarev, "Chaos and cryptography: Block encryption ciphers based on chaotic maps," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 48, no. 2, pp. 163–169, 2001.
[42] F. Özkaynak and A. B. Özer, "A method for designing strong S-Boxes based on chaotic lorenz system," *Phys. Lett. A*, vol. 374, no. 36, pp. 3733–3738, Aug. 2010.
[43] G. Tang, X. Liao, and Y. Chen, "A novel method for designing S-boxes based on chaotic maps," *Chaos, Solitons Fractals*, vol. 23, no. 2, pp. 413–419, Jan. 2005.
[44] M. Asim and V. Jeoti, "Efficient and simple method for designing chaotic S-Boxes," *ETRI J.*, vol. 30, no. 1, pp. 170–172, Feb. 2008.
[45] G. Tang and X. Liao, "A method for designing dynamical S-boxes based on discretized chaotic map," *Chaos, Solitons Fractals*, vol. 23, no. 5, pp. 1901–1909, Mar. 2005.
[46] F. Özkaynak and S. Yavuz, "Designing chaotic S-boxes based on time-delay chaotic system," *Nonlinear Dyn.*, vol. 74, no. 3, pp. 551–557, Nov. 2013.
[47] G. Chen, "A novel heuristic method for obtaining S-boxes," *Chaos, Solitons Fractals*, vol. 36, no. 4, pp. 1028–1036, May 2008.
[48] M. Khan, T. Shah, and M. A. Gondal, "An efficient technique for the construction of substitution box with chaotic partial differential equation," *Nonlinear Dyn.*, vol. 73, no. 3, pp. 1795–1801, Aug. 2013.
[49] M. Khan and T. Shah, "A construction of novel chaos base nonlinear component of block cipher," *Nonlinear Dyn.*, vol. 76, no. 1, pp. 377–382, Apr. 2014.
[50] H. Liu, A. Kadir, and Y. Niu, "Chaos-based color image block encryption scheme using S-box," *AEU-Int. J. Electron. Commun.*, vol. 68, no. 7, pp. 676–686, Jul. 2014.
[51] M. Khan, T. Shah, and S. I. Batool, "A new implementation of chaotic S-boxes in CAPTCHA," *Signal, Image Video Process.*, vol. 10, no. 2, pp. 293–300, Feb. 2016.
[52] I. Hussain, T. Shah, M. A. Gondal, W. A. Khan, and H. Mahmood, "A group theoretic approach to construct cryptographically strong substitution boxes," *Neural Comput. Appl.*, vol. 23, no. 1, pp. 97–104, Jul. 2013.
[53] F. Özkaynak, V. Çelik, and A. B. Özer, "A new S-box construction method based on the fractional-order chaotic chen system," *Signal, Image Video Process.*, vol. 11, no. 4, pp. 659–664, May 2017.
[54] I. Hussain, T. Shah, and M. A. Gondal, "A novel approach for designing substitution-boxes based on nonlinear chaotic algorithm," *Nonlinear Dyn.*, vol. 70, no. 3, pp. 1791–1794, Nov. 2012.

[55] F. Özkaynak, "An analysis and generation toolbox for chaotic substitution boxes: A case study based on chaotic labyrinth rene Thomas system," *Iranian J. Sci. Technol., Trans. Elect. Eng.*, vol. 44, pp. 89–98, Jun. 2019.

[56] M. Khan and T. Shah, "An efficient construction of substitution box with fractional chaotic system," *Signal, Image Video Process.*, vol. 9, no. 6, pp. 1335–1338, Sep. 2015.

[57] I. Hussain, T. Shah, H. Mahmood, and M. A. Gondal, "Construction of s8 liu j S-boxes and their applications," *Comput. Math. Appl.*, vol. 64, no. 8, pp. 2450–2458, Oct. 2012.

[58] F. Özkaynak, "From biometric data to cryptographic primitives: A new method for generation of substitution boxes," in *Proc. Int. Conf. Biomed. Eng. Bioinf.*, 2017, pp. 27–33.

[59] I. Hussain, T. Shah, M. A. Gondal, and H. Mahmood, "A novel method for designing nonlinear component for block cipher based on TD-ERCS chaotic sequence," *Nonlinear Dyn.*, vol. 73, nos. 1–2, pp. 633–637, Jul. 2013.

[60] I. Hussain, T. Shah, H. Mahmood, and M. A. Gondal, "A projective general linear group based algorithm for the construction of substitution box for block ciphers," *Neural Comput. Appl.*, vol. 22, no. 6, pp. 1085–1093, May 2013.

[61] M. Khan and T. Shah, "A novel image encryption technique based on Hénon chaotic map and S_8 symmetric group," *Neural Comput. Appl.*, vol. 25, nos. 7–8, pp. 1717–1722, 2014.

[62] A. Belazi and A. A. A. El-Latif, "A simple yet efficient S-box method based on chaotic sine map," *Optik*, vol. 130, pp. 1438–1444, Feb. 2017.

[63] G. Liu, W. Yang, W. Liu, and Y. Dai, "Designing S-boxes based on 3-D four-wing autonomous chaotic system," *Nonlinear Dyn.*, vol. 82, no. 4, pp. 1867–1877, Dec. 2015.

[64] F. U. Islam and G. Liu, "Designing S-Box based on 4D-4Wing hyperchaotic system," *3D Res.*, vol. 8, no. 1, Mar. 2017.

[65] Ü. Çavuşo˘glu, S. Kaçar, A. Zengin, and I. Pehlivan, "A novel hybrid encryption algorithm based on chaos and S-AES algorithm," *Nonlinear Dyn.*, vol. 92, no. 4, pp. 1745–1759, Jun. 2018.

[66] X. Wang, A. Akgul, Ü. Çavuşoğlu, V.-T. Pham, D. Vo Hoang, and X. Nguyen, "A chaotic system with infinite equilibria and its S-Box constructing application," *Appl. Sci.*, vol. 8, no. 11, p. 2132, Nov. 2018.

[67] X. Wang, Ü. Çavuşo˘glu, S. Kacar, A. Akgul, V.-T. Pham, S. Jafari, F. Alsaadi, and X. Nguyen, "S-box based image encryption application using a chaotic system without equilibrium," *Appl. Sci.*, vol. 9, no. 4, p. 781, Feb. 2019.

[68] Ü. Çavuşo˘glu, A. Zengin, I. Pehlivan, and S. Kaçar, "A novel approach for strong S-Box generation algorithm design based on chaotic scaled Zhongtang system," *Nonlinear Dyn.*, vol. 87, no. 2, pp. 1081–1094, Jan. 2017.

[69] T. Farah, R. Rhouma, and S. Belghith, "A novel method for designing S-box based on chaotic map and teaching–learning-based optimization," *Nonlinear Dyn.*, vol. 88, no. 2, pp. 1059–1074, 2017.

[70] F. Özkaynak, "Construction of robust substitution boxes based on chaotic systems," *Neural Comput. Appl.*, vol. 31, no. 8, pp. 3317–3326, Aug. 2019.

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

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

[73] M. Ahmad, D. Bhatia, and Y. Hassan, "A novel ant colony optimization based scheme for substitution box design," *Procedia Comput. Sci.*, vol. 57, pp. 572–580, 2015.

[74] H. A. Ahmed, M. F. Zolkipli, and M. Ahmad, "A novel efficient substitution-box design based on firefly algorithm and discrete chaotic map," *Neural Comput. Appl.*, vol. 31, no. 11, pp. 7201–7210, Nov. 2019.

[75] 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. 53079–53090, 2019.

[76] X. Zhang, Z. Zhao, and J. Wang, "Chaotic image encryption based on circular substitution box and key stream buffer," *Signal Process., Image Commun.*, vol. 29, no. 8, pp. 902–913, Sep. 2014.

[77] D. Lambić, "A novel method of S-box design based on chaotic map and composition method," *Chaos, Solitons Fractals*, vol. 58, pp. 16–21, Jan. 2014.

[78] I. Hussain, T. Shah, M. A. Gondal, and H. Mahmood, "An efficient approach for the construction of LFT S-boxes using chaotic logistic map," *Nonlinear Dyn.*, vol. 71, nos. 1–2, pp. 133–140, Jan. 2013.

[79] I. Hussain, T. Shah, M. A. Gondal, and H. Mahmood, "Efficient method for designing chaotic S-boxes based on generalized Baker's map and TDERC chaotic sequence," *Nonlinear Dyn.*, vol. 74, nos. 1–2, pp. 271–275, 2013.

[80] A. Belazi, A. A. Abd El-Latif, A.-V. Diaconu, R. Rhouma, and S. Belghith, "Chaos-based partial image encryption scheme based on linear fractional and lifting wavelet transforms," *Opt. Lasers Eng.*, vol. 88, pp. 37–50, Jan. 2017.

[81] M. S. Acikkapi, F. Ozkaynak, and A. B. Ozer, "Side-channel analysis of chaos-based substitution box structures," *IEEE Access*, vol. 7, pp. 79030–79043, 2019.

[82] A. Rosenfeld, "Fuzzy graphs," in *Fuzzy Sets and Their Applications to Cognitive and Decision Processes*. Amsterdam, The Netherlands: Elsevier, 1975, pp. 77–95.

[83] F. Hao, D.-S. Park, S. Li, and H. M. Lee, "Mining λ-maximal cliques from a fuzzy graph," *Sustainability*, vol. 8, no. 6, p. 553, 2016.

[84] Y. M. Ermoliev and R.-B. Wets, *Numerical Techniques for Stochastic Optimization*. Berlin, Germany: Springer-Verlag, 1988.

[85] M. A. Thathachar and P. S. Sastry, *Networks of Learning Automata: Techniques for Online Stochastic Optimization*. New York, NY, USA: Springer, 2011.

[86] H. Narasimhan, S. Li, P. Kar, S. Chawla, and F. Sebastiani, "Stochastic optimization techniques for quantification performance measures," *Stat*, vol. 1050, p. 13, May 2016.

[87] E. Alba and B. Dorronsoro, "The Exploration/Exploitation tradeoff in dynamic cellular genetic algorithms," *IEEE Trans. Evol. Comput.*, vol. 9, no. 2, pp. 126–142, Apr. 2005.

[88] W. G. Macready and D. H. Wolpert, "Bandit problems and the exploration/exploitation tradeoff," *IEEE Trans. Evol. Comput.*, vol. 2, no. 1, pp. 2–22, Apr. 1998.

[89] S. Li, C. Gentile, A. Karatzoglou, and G. Zappella, "Data-dependent clustering in exploration-exploitation algorithms," 2015, *arXiv:1502.03473*. [Online]. Available: https://arxiv.org/abs/1502.03473

[90] S. Picek, R. Santana, and D. Jakobovic, "Maximal nonlinearity in balanced Boolean functions with even number of inputs, revisited," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jul. 2016, pp. 3222–3229.

[91] P. Sarkar and S. Maitra, "Nonlinearity bounds and constructions of resilient Boolean functions," in *Proc. Annu. Int. Cryptol. Conf.* Berlin, Germany: Springer, 2000, pp. 515–532.

MIROSLAV M. DIMITROV (Graduate Student Member, IEEE) was born in Yambol, Bulgaria, in 1985. He received the B.S. degree in informatics and the M.S. degree in information security from the Faculty of Mathematics and Informatics, Sofia University. He is currently pursuing the Ph.D. degree in informatics with the Bulgarian Academy of Sciences. His research interests include cryptology, algorithms, and sequences.