FIRAT UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
TÜRKİYE

GENERATION OF SUBSTITUTION BOX STRUCTURES BASED ON RANDOM SELECTION AND IMPLEMENTATION OF ITS PRACTICAL APPLICATIONS

Habib IBRAHIM

Master's Thesis

DEPARTMENT OF SOFTWARE ENGINEERING

JULY 2021

FIRAT UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
TÜRKİYE

Department of Software Engineering

Master's Thesis

GENERATION OF SUBSTITUTION BOX STRUCTURES BASED ON RANDOM SELECTION AND IMPLEMENTATION OF ITS PRACTICAL APPLICATIONS

Author
Habib IBRAHIM

Supervisor
Assoc. Prof. Dr. Fatih ÖZKAYNAK

JULY 2021
ELAZIG

FIRAT UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
TÜRKİYE

Department of Software Engineering

Master's Thesis

Title:	Generation of Substitution Box Structures Based on Random Selection and Implementation of Its Practical Applications
Author:	Habib IBRAHIM
Submission Date:	07 June 2021
Defense Date:	07 July 2021

THESIS APPROVAL

This thesis, which was prepared according to the thesis writing rules of the Graduate School of Natural and Applied Sciences, Fırat University, was evaluated by the committee members who have signed the following signatures and was unanimously approved after the defense exam made open to the academic audience.

		Signature
Supervisor:	Assoc. Prof. Dr. Fatih ÖZKAYNAK Fırat University, Faculty of Technology	Approved
Chair:	Assist Prof. Dr. Fatih TOPALOĞLU Turgut Özal University, Faculty of Engineering and Natural Sciences	Approved
Member:	Assoc. Prof. Dr. Fatih ÖZYURT Fırat University, Faculty of Engineering	Approved

This thesis was approved by the Administrative Board of the Graduate School on

..... / ..... / 20 .....

Signature

Associated Prof. Dr. Kürşat Esat ALYAMAÇ
Director of the Graduate School

DECLARATION

I hereby declare that I wrote this Master's Thesis titled “Generation of Substitution Box Structures Based on Random Selection and Implementation of Its Practical Applications” in consistent with the thesis writing guide of the Graduate School of Natural and Applied Sciences, Firat University. I also declare that all information in it is correct, that I acted according to scientific ethics in producing and presenting the findings, cited all the references I used, express all institutions or organizations or persons who supported the thesis financially. I have never used the data and information I provide here in order to get a degree in any way.

07 July 2021

Habib IBRAHIM

PREFACE

The cryptology science has gradually gained importance with our digitalized lives. Ensuring the security of data transmitted, processed and stored across digital channels is a major challenge. One of the frequently used components in cryptological algorithms to ensure security is substitution-box structures. Random selection-based substitution-box structures have become increasingly important lately, especially because of their advantages to prevent side channel attacks. However, the low nonlinearity value of these designs is a problem. In this study, different datasets for substitution-box structures based on random selection have been publicly presented to the researchers. The fact that the proposed dataset has high nonlinearity values will allow it to be used in many practical applications in the future studies.

This thesis was supported by the project number **TEKF.20.21** by Firat University Scientific Research Projects Coordination Unit (FÜBAP).

Habib IBRAHIM
ELAZIG, 2021

TABLE OF CONTENTS

Page

ABSTRACT

Generation of Substitution Box Structures Based on Random Selection and Implementation of Its Practical Applications

Habib IBRAHIM

Master's Thesis

FIRAT UNIVERSITY
Graduate School of Natural and Applied Sciences

Department of Software Engineering

July 2021, Page: x + 35

Practical applications in cryptology science have a very wide spectrum. One of the important primitives based on many of these applications is the substitution-box structures. Since substitution-box structures are the main component targeted by many attacks, developing new substitution-box structures resistant to new attack is a hot research topic. Random selection-based designs have become increasingly popular recently, as they are more resistant to side channel attacks than mathematically based designs. In this thesis study, four different random selection-based substitution-box generator approach has been proposed. The proposed substitution-boxes are designed based on pseudo random number generator algorithm, true random number generator hardware. noise entropy source and chaotic systems. More than two hundred new substitution-box structures generated within the scope of the thesis are presented to researchers as a public dataset. Analysis results showed that the proposed substitution-boxes have better performance criteria than many designs in the literature. These successful results indicate that the outputs of study can be used successfully in many practical applications that will be designed in the future.

.

Keywords: Cryptology, Chaos, Random selection, Substitution box,

ÖZET

Rasgele Seçim Tabanlı Yer Değiştirme Kutularının Üretilmesi ve Pratik Uygulamalarının Gerçekleştirimi

Habib IBRAHIM

Yüksek Lisans Tezi

FIRAT ÜNİVERSİTESİ
Fen Bilimleri Enstitüsü

Yazılım Mühendisliği Anabilim Dalı

Temmuz 2021, Sayfa: x + 35

Kriptoloji bilimindeki pratik uygulamalar çok geniş bir yelpazeye sahiptir. Bu uygulamaların birçoğunda temel alınan önemli temel yapıtaşlarından biri yer değiştirme kutusu yapılarıdır. Yer değiştirme kutusu yapıları birçok saldırının hedef aldığı temel bileşen olduğu için yeni saldırılara dirençli yer değiştirme kutusu yapılarının geliştirilmesi sıcak bir araştırma konusudur. Yan kanal saldırılarına karşı matematiksel temelli tasarımlara göre daha dirençli olduğu için rasgele seçim tabanlı tasarımlar son zamanda giderek popüler olmaya başlamıştır. Bu tez çalışmasında rasgele seçim tabanlı dört farklı yer değiştirme kutusu üreteç yaklaşımı önerilmiştir. Önerilen yaklaşımlar sözde rasgele sayı üreteç algoritması, gerçek rasgele sayı üreteç donanımı, gürültü entropi kaynağı ve kaotik sistemleri temel alarak tasarlanmıştır. Tez kapsamında üretilen iki yüzden fazla yer değiştirme kutusu yapısı araştırmacıların erişimine açık bir veriseti olarak sunulmuştur. Analiz sonuçları önerilen yer değiştirme kutusu yapılarının literatürdeki birçok tasarıma göre daha iyi performans ölçütlerine sahip olduğu göstermiştir. Bu başarılı sonuçlar ileride tasarlanacak birçok pratik uygulamada elde edilen çıktıların başarılı şekilde kullanılabileceğine işaret etmektedir.

Anahtar Kelimeler: Kriptoloji, Kaos, Rasgele seçim, Yer değiştirme kutusu

LIST OF FIGURES

Page

LIST OF TABLES

Page

ABBREVIATIONS

Abbreviations

S-BOX : Substitution Box
NL : Nonlinearity
SAC : Strict Avalanche Criterion
BIC : Bit Independence Criterion
XOR : Xclusive OR
RNG : Random Number Generator
TRNG : True Random Number Generator
PRNG : Pseudo Random Number Generator

1. INTRODUCTION

“Cryptology is a science that aims to provide secure data communication in the simplest sense. In order for a cryptographic protocol to be considered secure, it is expected to be guaranteed to meet two basic requirements. These requirements are confusion and diffusion properties. The diffusion feature aims to prevent the attacker from obtaining the secret key of the algorithm using statistical analysis.” Generally, methods are used in which the original data is shifted, displaced, or mapped with the help of a function to provide propagation. Since these methods used in the exchange of the original data have a linear structure, they can be easily estimated by the attacker. Therefore, it is necessary to provide a strong confusion process with the need for spreading. Because the confusion feature makes it difficult to express the relationship between the secret key of the algorithm and the encrypted data.

One of the most effective approaches to achieve confusion feature is to obtain a strong entropy source. However, this is a difficult task in cryptographic applications. A flaw in the design will affect the security of the entire algorithm. Especially for cryptographic applications, designs based on physical resources are needed. In the literature, one of the popular approaches based on these physical resources is chaos-based randomness generators.

Chaotic systems are a strong candidate to achieve cryptographic randomness. Therefore, designing various cryptographic protocols using the dynamics of chaotic systems is an active research area. One of the most successful examples of using two disciplines in this research area was the design of chaos-based substitution box structures.

“An S-box structure based on random selection is one of the successful examples of chaos-based cryptology design. The analysis toolbox checks whether the s-box data provide the bijective property. Hence, it displayed a warning “selected s-box is not bijective” whenever the bijective property is not provided. Else, the s-box source file is displayed in a 16 x 16 matrix when the bijective property is provided. Also, the nonlinearity property of the s-box structure is also analyzed and it very essential since s-box structures are the only nonlinear element in various cryptographic algorithms.

Again, the strict avalanche criterion (SAC) property is also measured. This measures the amount of nonlinearity in substitution boxes (s-boxes) and analyzes the rate of change in the output bits when only one-bit change occurs in the input bits [5]. Hence, the optimal value of SAC is 0.5. This implies that the output bits change when a one-bit change occurs in the input bit values. In

addition, the bit independence criterion (BIC) is also examined. Here, the effects of both inputs and outputs on the nonlinearity and avalanche criterion are tested. Moreover, the difference distribution table (XOR table) is used to show how resistant the S-box structure is to differential cryptanalysis.

Chaos theory offers researchers various opportunities in many areas of science. [1] The rich dynamics that it contains have always made chaotic systems in the center of attention. In addition to its use in modeling and control areas, its random behavior has led cryptography experts to focus on this field [2]. The basic idea behind this interest is that confusion and diffusion requirements can be met with the principle of sensitive dependence on initial conditions and control parameters. Confusion and diffusion requirements are two important properties of encryption protocols. These requirements are identified by Claude Shannon in 1945. “Confusion makes it difficult to find the key from the ciphertext and if a single bit in a key is changed, most or all the bits in the ciphertext will be affected. Diffusion means that if we change a single bit of the plaintext, then (statistically) half of the bits in the ciphertext should change”. It has been suggested that these requirements can be met using chaotic systems since chaotic outputs are extremely sensitive to changes in initial conditions and control parameters and have a nonlinear characteristic. Researchers have used chaotic systems as an entropy source in cryptographic designs. They used the initial condition and control parameters of chaotic systems as the secret key of cryptographic protocols. It has been suggested that different entropy sources can be produced by using different initial conditions and control parameters as they will produce different outputs with small changes that may occur in the initial conditions and control parameters. Many cryptographic protocols such as image encryption algorithms [3-10], key generators [1, 5] and s-box designs [1-15] have been proposed using this design idea.

Although this design approach has been widely studied, the security analysis of some proposals has not been done according to certain criteria and has caused various security problems. Chaos-based s-box designs stand out as the design class that is not affected by these problems. Because the requirements for s-box performance analysis are almost standardized [15]. Bijective, nonlinearity, bit independence criterion (BIC), strict avalanche criterion (SAC) and input/output XOR distribution criteria are known as standard measurements used in this analysis process of s-boxes. nonlinearity criterion can be associated with the confusion criterion, which is one of the general characteristics of encryption algorithms. The ideal value to be achieved for this criterion is 112. The ideal value for the strict avalanche criterion is 0.5. This value indicates the difficulty of making statistical inferences. Values smaller or greater than 0.5 increase the success of statistical analysis. BIC measurement is related to nonlinearity and SAC measurements through the relationship between input and output bits. Input/output XOR distribution is related to differential

cryptanalysis. To show its resistance against differential attacks, the maximum value that can be calculated. The expected value is 4. Larger values indicate that differential attacks can be more successful [1].

"S-box structure is a bijective function that converts values in a certain range to values in a certain range. The AES s-box structure is a nonlinear function that maps 256 values between 0 and 255 to 256 values between 0 and 255". Therefore, in the literature, different s-box structures have been tried to be obtained by converting the chaotic system outputs to 256 different values. Many different s-box structures have been generated by changing the initial conditions and control parameters. Also, different chaotic system classes or different conversion algorithms have been used to improve s-box performance criteria.

When design studies are classified in terms of chaotic system types, there are two general classes: discrete and continuous-time chaotic systems. Discrete-time systems have become one of the most preferred systems for researchers in the design process [1]. The main reason for this selection is shown that the systems can produce very fast results due to their simple mathematical models. The biggest advantage of continuous-time systems is that they have more complex mathematical models than discrete-time systems [7, 13, 14]. It is thought that this complexity will positively affect the quality of the entropy source. To use this advantage of continuous-time systems [6, 8] most effectively, special chaotic systems such as hyperchaotic [10, 11], time-delay [12] and fractional-order systems [3, 4, 8, 9] have also been used in the design process.

Purpose of Thesis: The thesis aims to search the new substitution box structures based on random selection that can be used to AES block encryption architecture. There are several metrics to evaluate the success of substitution box designs. Analyzes show that nonlinearity is more suitable for performance comparison in these metrics. Therefore, the aim of the thesis study is that to search maximum nonlinearity substitution box structures based on random selection and to implementation of its practical applications in information security

Definition of the Problem: Chaos based substitution box structures are obtained by using a design approach based on the random selection principle. Recent studies have shown that these structures can be more successful in preventing cryptanalysis scenarios especially for applications attacks such as side channel analyses. However, although random selection-based designs are more resistant to application attacks, random substitution box is worse than math-based designs in terms of performance criteria.

Hypotheses: If performance improvements of chaos based substitution box structures can be made, the usability of these structures in practical applications will be shown. In this case, both a precaution will be developed for cryptanalysis techniques for application attacks such as side channel analysis, and new designs that are alternative to mathematical designs can be paved.

Unique Aspect of Thesis Study: As part of the seminar studies, Habib Ibrahim generated many substitution box structures based on the random selection principle. As a result of these trials, it has been shown that 20 substitution box structures that can be used directly in practical applications are more successful than studies in the literature for nonlinear criterion and XOR distribution which are substitution box design metrics. These results show that physical entropy sources can be obtained as a source of randomness in thesis studies, basically new substitution box structures. Both these successful designs and practical applications in the field of information security based on these designs stand out as the original aspect of the thesis.

2. A RANDOM SELECTION BASED SUBSTITUTION-BOX STRUCTURE DATASET FOR CRYPTOLOGY APPLICATIONS

The reason why the concept of the Internet of Things is so popular is that it brings many advantages, but a serious problem that needs to be solved is how to address security concerns [1, 2]. Although there are many critical issues that need to be addressed in order to guarantee security, the first solution that comes to mind is the use of encryption algorithms [3]. In particular, block encryption algorithms are widely used in many practical applications to ensure data confidentiality. Block cipher algorithms consist of various basic components. Among these components, one of the important factors affecting security is the substitution boxes (s-boxes) [4, 5].

In the simplest terms, the s-box structure is a nonlinear transformation. This transformation aims to map the relationship between input and output into a form that the attacker cannot understand [6, 7]. In this way, statistical inferences will be prevented and precautions will be taken against strong attack techniques such as known/chosen plaintext/ciphertext attacks. However, there are various design criteria that a s-box structure must meet in order to meet these requirements [4, 5]. Various approaches have been proposed in the literature to meet these design criteria. Each of these design approaches has several advantages and disadvantages [8]. In this study, a dataset that can be used as a random selection-based s-box design is generated.

The most important problem of random selection-based s-box structures is that nonlinearity criterion is lower than mathematical designs [8]. In this study, a dataset with higher nonlinearity value compared to many designs in the literature is shared. The shared dataset is made available to researchers. It is thought that this dataset will be widely used in many practical applications such as block cipher systems, random number generators and image encryption.

2.1. Details of Substitution Box Generator Program

There are various programs that can be used as s-box generator. However, the preliminary knowledge and field experience required to use these programs is a serious problem. An efficient program for generating s-box structures has been developed in Ref. [9]. In order to increase the widespread usability of the developed program, outputs were generated for 16x16 s-box structures. 16x16 sized s-box structures have various advantages especially for software implementations. Since many programming languages work on a byte basis (1 byte = 8-bit), they convert 2^8-length inputs into 2^8-length outputs. The interface of the program is shown in Figure 2.1.

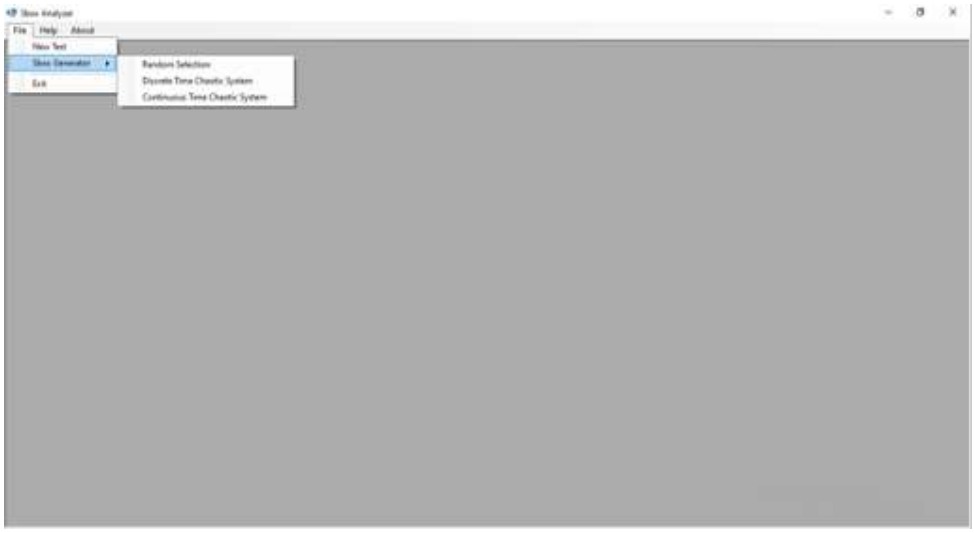

Figure 2.1. Analysis and generation program for s-box structures

In the s-box generation phase of the program, the following steps are operated in order.

- The entropy source is determined.
- Continuous time chaotic systems, discrete time chaotic systems or random number generator library of programming language can be used as the source of entropy.
- The state variables of chaotic systems are rationally valued. Therefore, they are converted between 0-255 using the mode function.
- Since values between 0-255 will be generated directly with the random function, any additional function such as is not required.
- By checking whether the generated values are in the s-box before, the requirement to be bijective is provided from the s-box design criteria.
- Until the entire s-box table is filled, the operations continue by taking a new value from the entropy source.

For more detailed information about the research on the effect of the chaotic system class on success, Ref. [20] can be examined. In this study, the randomness library has been used as an entropy source. Visual Studio C# programming language has been used as the programming language. It contains different alternatives within the random library. Ref. [9] can be examined for these options and other details.

2.2. Properties of Generated S-Box Dataset

Thousands of s-box structures have been generated using the proposed program. Among these s-box structures, those with a nonlinearity value of 105 and above were additionally saved.

These s-boxes shared as an public dataset. Since the nonlinearity value is interpreted as a measure of the complexity of the relationship between input/output, it is directly associated with linear and differential cryptanalysis.

Five basic measurements are used as s-box performance criteria. These criteria are called bijective, nonlinearity, strict avalanche criterion (SAC), bit independence criterion (BIC), and XOR distribution [4, 5]. In Table 2.1, the performance analysis results of twenty different s-box structures obtained as a result of this research study are publicly shared.

Table 2.1. Performance analysis results of twenty different s-box structures

s-box	Nonlinearity (NL)			Bit Independence Criterion		Strict Avalanche Criterion			Maximum I/O XOR
	min	max	avg	SAC	NL.	min	max	avg	
s-box 1	98	108	105	0.4969	103	0.4219	0.6094	0.5044	10
s-box 2	102	108	105.5	0.5042	104.36	0.4219	0.5038	0.5007	10
s-box 3	104	108	105.75	0.4989	103.36	0.4219	0.625	0.5132	10
s-box 4	100	108	105	0.5011	103.86	0.375	0.5781	0.4917	10
s-box 5	100	110	105	0.5028	104	0.4219	0.6406	0.5005	10
s-box 6	102	108	105	0.5051	102.36	0.4062	0.5625	0.5037	10
s-box 7	104	108	105.25	0.501	103.93	0.4062	0.5938	0.5049	10
s-box 8	100	108	105	0.4984	103.14	0.4062	0.5781	0.4973	10
s-box 9	100	108	105	0.4996	103.36	0.4062	0.6094	0.5073	10
s-box 10	100	108	105	0.4954	103.71	0.4062	0.5938	0.5061	10
s-box 11	102	108	105	0.4952	103.36	0.4062	0.5938	0.5	10
s-box 12	102	108	105	0.5026	103.29	0.4219	0.5938	0.5027	10
s-box 13	102	108	105.5	0.5004	104.21	0.4062	0.6094	0.5022	10
s-box 14	102	108	105	0.4999	103.21	0.4375	0.6094	0.5049	10
s-box 15	104	108	105.75	0.5035	102.71	0.3906	0.625	0.5022	10
s-box 16	102	108	105.5	0.4974	103.79	0.3906	0.6406	0.4956	10
s-box 17	100	108	105.25	0.4946	102.64	0.3906	0.5938	0.5063	10
s-box 18	102	108	105.25	0.4997	104.21	0.4219	0.5938	0.5039	10
s-box 19	102	108	105	0.498	103.57	0.4219	0.6094	0.499	10
s-box 20	102	108	105	0.4981	102.29	0.4531	05938	0.5056	10

The approach used to generate s-box provides the feature of being bijective during the design phase. A value of 0.5 is accepted as the ideal value for the SAC criterion. For nonlinearity measurement, the value 112 is the highest value that can be reached. The BIC value is reconsidering the input-output relationship for the SAC and nonlinearity criteria. In the XOR distribution table, the highest value among the calculated values is desired to be as small as possible. Ref. [4, 5] can be examined for mathematical equations, how to make calculations and other details about these criteria.

2.3. Performance Comparisons

The s-box structure of the highest nonlinearity value in the dataset presented in the second section is given in Table 2.2. Performance comparisons of this s-box structure with some studies previously published in the literature are given in Table 2.3.

Table 2.2. A sample s-box structure with highest nonlinearity value

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

Table 2.3. Comparison results of the proposed approach with similar studies in the literature

S-box	Strict Avalanche Criterion			Nonlinearity (NL)			Maximum I/O XOR	Bit Independence Criterion	
	avg	max	min	min	max	avg		SAC	NL
Ref. [10]	0.5022	0.5781	0.4063	100	110	105.5	32	0.4983	107
Ref. [11]	0.4926	0.5937	0.4062	98	110	105.5	32	0.4994	105.7
Ref. [12]	0.5010	0.6094	0.4063	102	110	105.5	12	0.4988	104.3
Ref. [13]	0.5056	0.5781	0.4375	102	108	105.3	10	0.4971	104
Ref. [14]	0.5059	0.5781	0.4063	102	108	105.2	12	0.5013	104.3
Ref. [15]	0.4987	0.5469	0.4531	104	108	105.25	10	0.4990	102.6
Ref. [16]	0.5037	0.5625	0.4375	102	108	105.25	10	0.4994	102.6
Ref. [17]	0.5073	0.6094	0.4062	98	108	105.25	10	0.4986	103,86
Ref. [18]	0.5012	0.5938	0.4063	104	106	105	10	0.4994	103.4
Ref. [19]	0.5046	0.6093	0.4750	102	106	105	10	0.5004	103.6
Ref. [20]	0.4990	0.5850	0.4290	100	107	104.8	12	0.4890	104.7
Ref. [21]	0.4037	0.5938	0.3906	100	108	104.7	32	0.4965	105
Ref. [22]	0.5056	0.5937	0.3906	102	108	104.7	12	0.5021	104.1
Ref. [23]	0.4978	0.6093	0.4218	100	108	104.75	12	0.5009	103,6
Ref. [24]	0.4982	0.5781	0.4218	100	108	104.7	10	0.4942	103.1
Ref. [25]	0.5034	0.5938	0.3906	102	108	104.7	10	0.4972	103.3
Ref. [26]	0.498	0.6406	0.4219	102	108	104.5	12	0.5013	104.6
Ref. [27]	0.4980	0.6093	0.3750	102	106	104	10	0.4971	103.2
Ref. [28]	0.5026	0.5781	0.3906	100	106	104	10	0.5033	103.2
Ref. [29]	0.5	-	-	-	-	104	10	0.498	102.8
Ref. [30]	0.4954	0.6094	0.2813	98	108	104	12	0.4967	102
Ref. [31]	0.4946	0.6250	0.3750	100	106	104	10	0.4990	102.5
Ref. [32]	0.5018	0.5175	0.4825	102	106	104	10	0.5019	103.5
Ref. [33]	0.5039	0.6093	0.4218	98	108	104	12	0.5078	104
Ref. [34]	0.5058	0.5781	0.3906	101	108	103.8	14	0.4958	102.6
Ref. [35]	0.5036	0.6328	0.4140	101	106	103.8	10	0.5037	103.4
Ref. [36]	0.4987	0.6015	0.4140	99	106	103.3	10	0.4995	103.3
Ref. [37]	0.5058	0.625	0.4062	99	106	103.3	12	0.5037	103.6
Ref. [38]	0.5058	0.5975	0.3671	98	108	103.2	12	0.5031	104.2
Ref. [39]	0.5048	0.5937	0.4218	100	106	103.2	10	0.5009	103.7
Ref. [40]	0.5039	0.625	0.3906	96	106	103	12	0.5010	100.3
Ref. [41]	0.5	0.6093	0.4218	100	106	103	14	0.5024	103.1
Ref. [42]	0.5012	0.5937	0.4062	98	108	103	12	0.4988	104.1
Ref. [43]	0.5178	0.6719	0.3906	96	106	102.5	54	0.4026	102.5
Ref. [44]	0.4836	0.6016	0.3281	98	108	102.3	14	0.4992	100
Ref. [45]	0.5059	0.6094	0.4219	96	108	102.25	16	0.5050	103.5
Ref. [46]	-	-	-	-	-	102	8	-	-

Ref. [47]	0.4812	0.625	0.125	84	106	100	16	0.4962	101.9
Ref. [48]	0.4812	0.625	0.125	84	106	100	16	0.4962	101.9

2.4. Discussion

S-box structures are a critical cryptographic component that is at the focal point of many known basic attacks. It is known that the approaches based on mathematical transformations, which are one of the known design techniques, are not resistant to side channel analysis, which are practical attacks, and that random selection-based designs can be used as an alternative. However, it has been shown in previous studies that the nonlinearity value of random selection-based designs is low. The aim of this study is to generate s-box structures with high nonlinearity value produced according to the random selection principle.

In the study, different s-box structure with high nonlinearity value has been publicly presented to the access of all researchers. It has been observed that the twenty different s-box structures, which is publicly presented to the researchers, is better than the studies in the literature. It is thought that these results can be used successfully in many practical applications, especially in future block cipher systems, random number generators and random number generators.

3. A SUBSTITUTION-BOX STRUCTURE BASED ON CROWD SUPPLY INFINITE NOISE TRNG

With the COVID pandemic, we have begun to experience how critical digital processes will be in the future. In this period of experience, it has been once again observed with painful experiences how resistant our data in digital environments is against attacks. These painful experiences have brought the question of how cryptographic protocols can be designed to guarantee security [1].

One of the most important reasons why cryptology science is considered to be a difficult discipline is that the threat of attack continues for existing systems [2]. Because, attacker capabilities are constantly diversifying and developing with technological developments. One of the types of attacks that have evolved has been side channel attacks. Side channel attacks have shown that even strong encryption algorithm such as AES (Advanced Encryption Standard) [3] and RSA (Rivest–Shamir–Adleman public-key cryptosystem) [4], which are mathematically impossible to break, can show various vulnerabilities or even break when implemented on a hardware. For this reason, it has become a necessity to search for new cryptographic components that are more resistant to side channel attacks [5]. Recent studies have shown that random selection-based substitution-box (s-box) structures can be used as a countermeasure against these attacks.

In this study, a new s-box structure has been proposed as an alternative to mathematically based designs. The original aspect that differentiates the proposed s-box structure from similar structures in the literature is that it is based on a physically unclonable function (PUF) hardware as the entropy source. It has been observed that the unpredictable nature of the PUF hardware, which has a plug-and-play structure over the USB port, generates a strong entropy source [6]. Then, with the help of an effective algorithm, random values created on the basis of this entropy source have been transformed into s-box structures. Successful analysis results indicated that the proposed s-box structure could be used effectively in various cryptographic applications in the future.

The rest of the study is organized as follows. In the second section, details of the PUF hardware that constitutes the background of the study are given. In the third section, the operation steps of the algorithm that transforms PUF outputs into s-box structures are explained. Experimental analysis and performance comparisons of the proposed s-box structure are given in the fourth section. In the last section, the results have been discussed and the study has been summarized.

3.1. Details of Crowd Supply Infinite Noise TRNG

Two different options are on the designers' agenda to meet the randomness requirements. These options are DRNG (Deterministic Random Number Generator) and TRNG (True Random Number Generator) structures. Both options have various advantages and disadvantages. Ref. [7, 8] can be examined for a detailed literature analysis on cryptographic randomness and detailed reviews for both design RNG classes. TRNG structures are the best option to address the unpredictability requirement, as many designs require a strong entropy source. Since the TRNG constructs are based on natural processes and/or the unclonable properties of the device, it is not possible to obtain pre/post random values using any seed value. Because of such critical importance, many commercial and academic TRNG/PUF hardware are available in the literature. One of these PUF hardware is Crowd Supply Infinite Noise TRNG. The general view of the hardware is given in Figure 3.1.

The working logic of the Crowd Supply Infinite Noise TRNG hardware is explained step by step below [6]:

- The hardware is plugged into the computer using the USB port.
- It calculates the temperature value using the heat sensors on the hardware. This calculated value constitutes the source of unpredictable entropy.
- Calculated values are used as sources of noise
- The most important problem of TRNG is that the entropy source contains statistically weakness. Using the SHA3 [9] cryptographic hash function, this problem is addressed.

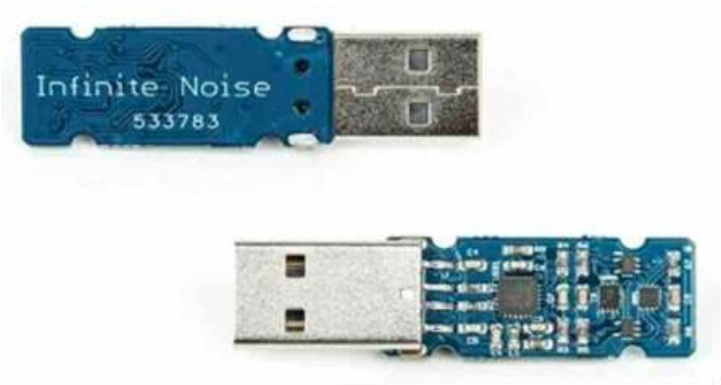

Figure 3.1. Crowd Supply Infinite Noise TRNG [6]

The structure representing the working architecture of the Crowd Supply Infinite Noise TRNG device is shown in Figure 3.2.

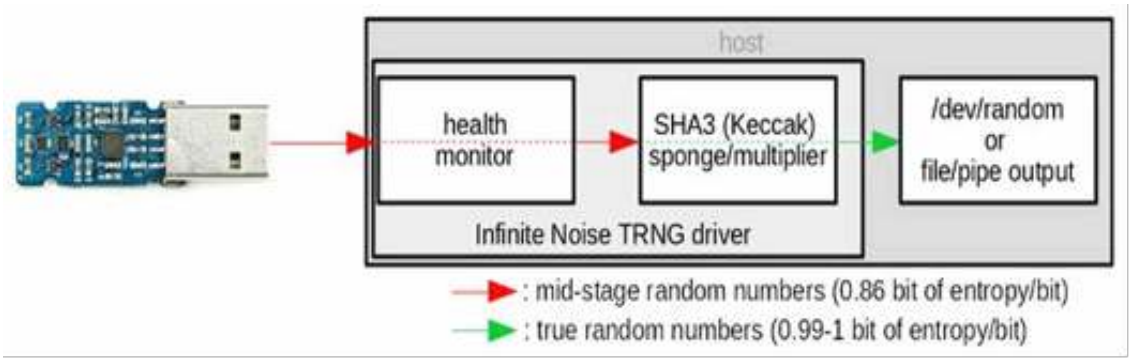

Figure 3.2. The working architecture of Crowd Supply Infinite Noise TRNG [6]

A practical application using this device has been demonstrated by Aina'u and Özkaynak. An image encryption algorithm is designed in Ref. [10]. Crowd Supply Infinite Noise TRNG device is used in the proposed algorithm to prevent possible threats against selected/chosen plaintext attack. Ref. [10] can be used for statistical properties and other details of the TRNG structure.

3.2. Proposed S-Box Generator Method

The general structure of the proposed method to generate s-box is shown in Figure 3.3. The operation steps of the algorithm are detailed below.

Figure 3.3. General steps of s-box generator approach

Step 1. The Crowd Supply Infinite Noise TRNG device is operated using the computer's USB port.

Step 2. Random values obtained using the device interface are saved in a .txt file.

Step 3. Each value in the .txt file is rationally valuable. These values are multiplied by 1000 and rounded to the nearest integer.

Step 4. Random values are mapped to integer values between 0-255 by applying mode 256 function.

Step 5. If the generated value has not been added to the s-box before, it is added. Otherwise, the process is repeated for a new integer.

Step 6. These processes continue until the s-box is filled.

More than 500 s-box structures were produced with random values produced using the device. Among these s-box structures, s-box structures with a nonlinearity value of 105 and above are presented as an open dataset. The s-box table with the highest nonlinearity value among these 80 s-box structures is presented in Table 3.1.

Table 3.1. A s-box with highest nonlinearity value

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

The analysis results for Table 3.1 s-box according to the basic s-box design criteria are given in Figure 3.4. The analysis results shown in Figure 3.4 obtained using analysis program developed by Özkaynak. This program, which can be used for free, draws attention with its simple interface and the ability to present the results in a different way.

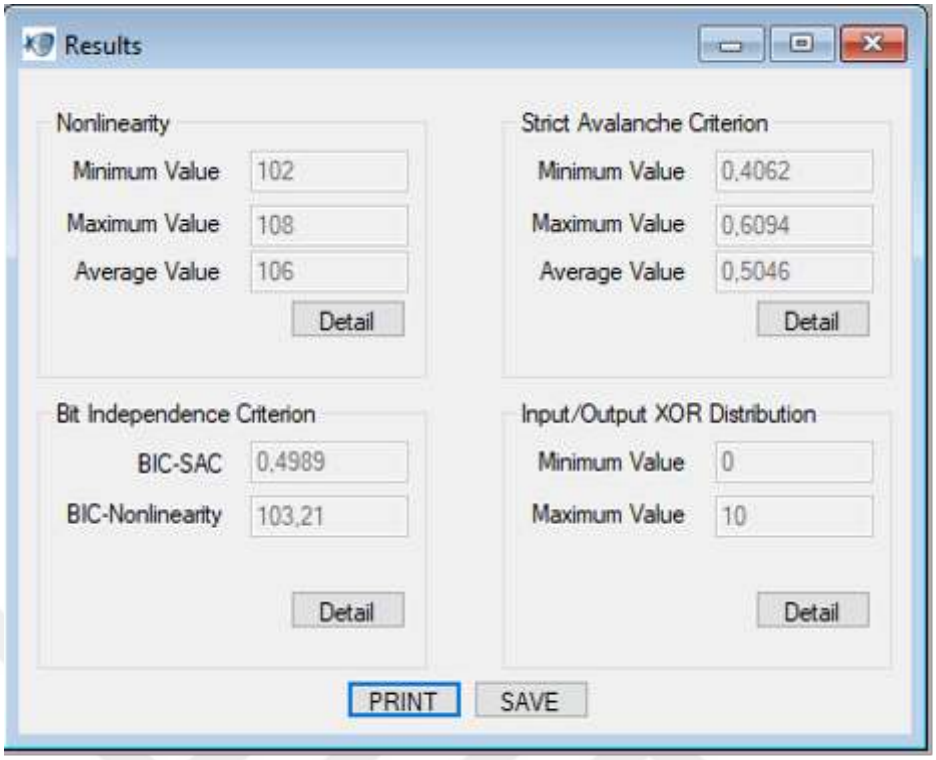

Figure 3.4. Performance metrics for generated s-box in Table 3.1

3.3. Performance Comparisons

There are various quantitative evaluation criteria that can be used in design process of s-box structures. Among these criteria, the most widely known and focused five basic criteria are: bijective, strict avalanche criterion (SAC), bit independence criterion (BIC), nonlinearity and XOR distribution. Ref. [11, 12] can be viewed for details and mathematical expressions of these criteria. For these basic criteria, the performance comparisons of other s-box structures in the literature and the proposed s-box are given in Table 3.2.

Table 3.2. Comparison results of the proposed approach with similar studies in the literature

S-box	Strict Avalanche Criterion			Nonlinearity (NL)			Maximum I/O XOR	Bit Independence Criterion	
	avg	max	min	min	max	avg		SAC	NL
Ref. [10]	0.5022	0.5781	0.4063	100	110	105.5	32	0.4983	107
Ref. [11]	0.4926	0.5937	0.4062	98	110	105.5	32	0.4994	105.7
Ref. [12]	0.5010	0.6094	0.4063	102	110	105.5	12	0.4988	104.3
Ref. [13]	0.5056	0.5781	0.4375	102	108	105.3	10	0.4971	104
Ref. [14]	0.5059	0.5781	0.4063	102	108	105.2	12	0.5013	104.3
Ref. [15]	0.4987	0.5469	0.4531	104	108	105.25	10	0.4990	102.6

Ref. [16]	0.5037	0.5625	0.4375	102	108	105.25	10	0.4994	102.6
Ref. [17]	0.5073	0.6094	0.4062	98	108	105.25	10	0.4986	103,86
Ref. [18]	0.5012	0.5938	0.4063	104	106	105	10	0.4994	103.4
Ref. [19]	0.5046	0.6093	0.4750	102	106	105	10	0.5004	103.6
Ref. [20]	0.4990	0.5850	0.4290	100	107	104.8	12	0.4890	104.7
Ref. [21]	0.4037	0.5938	0.3906	100	108	104.7	32	0.4965	105
Ref. [22]	0.5056	0.5937	0.3906	102	108	104.7	12	0.5021	104.1
Ref. [23]	0.4978	0.6093	0.4218	100	108	104.75	12	0.5009	103,6
Ref. [24]	0.4982	0.5781	0.4218	100	108	104.7	10	0.4942	103.1
Ref. [25]	0.5034	0.5938	0.3906	102	108	104.7	10	0.4972	103.3
Ref. [26]	0.498	0.6406	0.4219	102	108	104.5	12	0.5013	104.6
Ref. [27]	0.4980	0.6093	0.3750	102	106	104	10	0.4971	103.2
Ref. [28]	0.5026	0.5781	0.3906	100	106	104	10	0.5033	103.2
Ref. [29]	0.5	-	-	-	-	104	10	0.498	102.8
Ref. [30]	0.4954	0.6094	0.2813	98	108	104	12	0.4967	102
Ref. [31]	0.4946	0.6250	0.3750	100	106	104	10	0.4990	102.5
Ref. [32]	0.5018	0.5175	0.4825	102	106	104	10	0.5019	103.5
Ref. [33]	0.5039	0.6093	0.4218	98	108	104	12	0.5078	104
Ref. [34]	0.5058	0.5781	0.3906	101	108	103.8	14	0.4958	102.6
Ref. [35]	0.5036	0.6328	0.4140	101	106	103.8	10	0.5037	103.4
Ref. [36]	0.4987	0.6015	0.4140	99	106	103.3	10	0.4995	103.3
Ref. [37]	0.5058	0.625	0.4062	99	106	103.3	12	0.5037	103.6
Ref. [38]	0.5058	0.5975	0.3671	98	108	103.2	12	0.5031	104.2
Ref. [39]	0.5048	0.5937	0.4218	100	106	103.2	10	0.5009	103.7
Ref. [40]	0.5039	0.625	0.3906	96	106	103	12	0.5010	100.3
Ref. [41]	0.5	0.6093	0.4218	100	106	103	14	0.5024	103.1
Ref. [42]	0.5012	0.5937	0.4062	98	108	103	12	0.4988	104.1
Ref. [43]	0.5178	0.6719	0.3906	96	106	102.5	54	0.4026	102.5
Ref. [44]	0.4836	0.6016	0.3281	98	108	102.3	14	0.4992	100
Ref. [45]	0.5059	0.6094	0.4219	96	108	102.25	16	0.5050	103.5
Ref. [46]	-	-	-	-	-	102	8	-	-
Ref. [47]	0.4812	0.625	0.125	84	106	100	16	0.4962	101.9
Ref. [48]	0.4812	0.625	0.125	84	106	100	16	0.4962	101.9

3.4. Discussions

In this study, various s-box structures are designed based on a physical source of entropy. The aim of the proposed study is to develop a general strategy that can be used to transform powerful entropy sources into cryptographic primitives. In the study, more than 500 s-box structures have been produced using the data obtained from the Crowd Supply Infinite Noise TRNG device. Among these s-boxes, the 80 s-box component, which has a high nonlinearity value, has

been presented to the researchers as an public dataset. It is thought that these components, which are the outputs of the study, can be used to guarantee information security in many practical applications in the future studies.

4. RANDOMLY GENERATED SUBSTITUTION BOX STRUCTURE DATASET

Recent studies have shown that the nonlinearity values of chaos-based s-box structures can be improved up to 110. Therefore, it is thought that a dataset that can be used in many practical information security applications in future studies will facilitate the work of researchers. In this section, it is aimed to create a dataset to serve this purpose.

4.1. Details of S-box Generator Approach

The flowchart of the approach used to construct the s-box structures that will serve as a basis for further research is shown in Figure 4.1. The interface of the program used to output s-box structures based on this flowchart is shown in Figure 4.2.

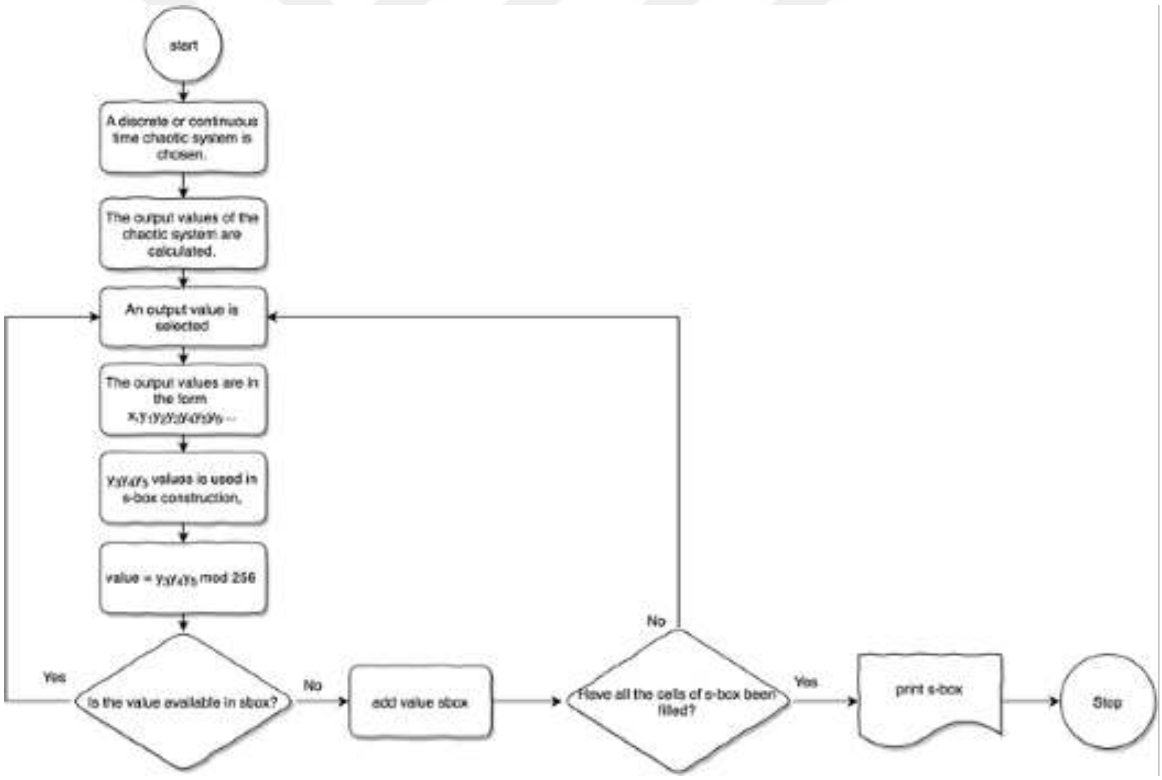

Figure 4.1. Flowchart of s-box generator approach

In this generator program, different designs can be realized by using discrete or continuous time chaotic systems. However, chaotic systems were not used in this study. The entropy source was created by using the rand() function of the programming language in which the interface program was designed. Visual Studio C# programming language was used as the program. For more details, the related study can be reviewed.

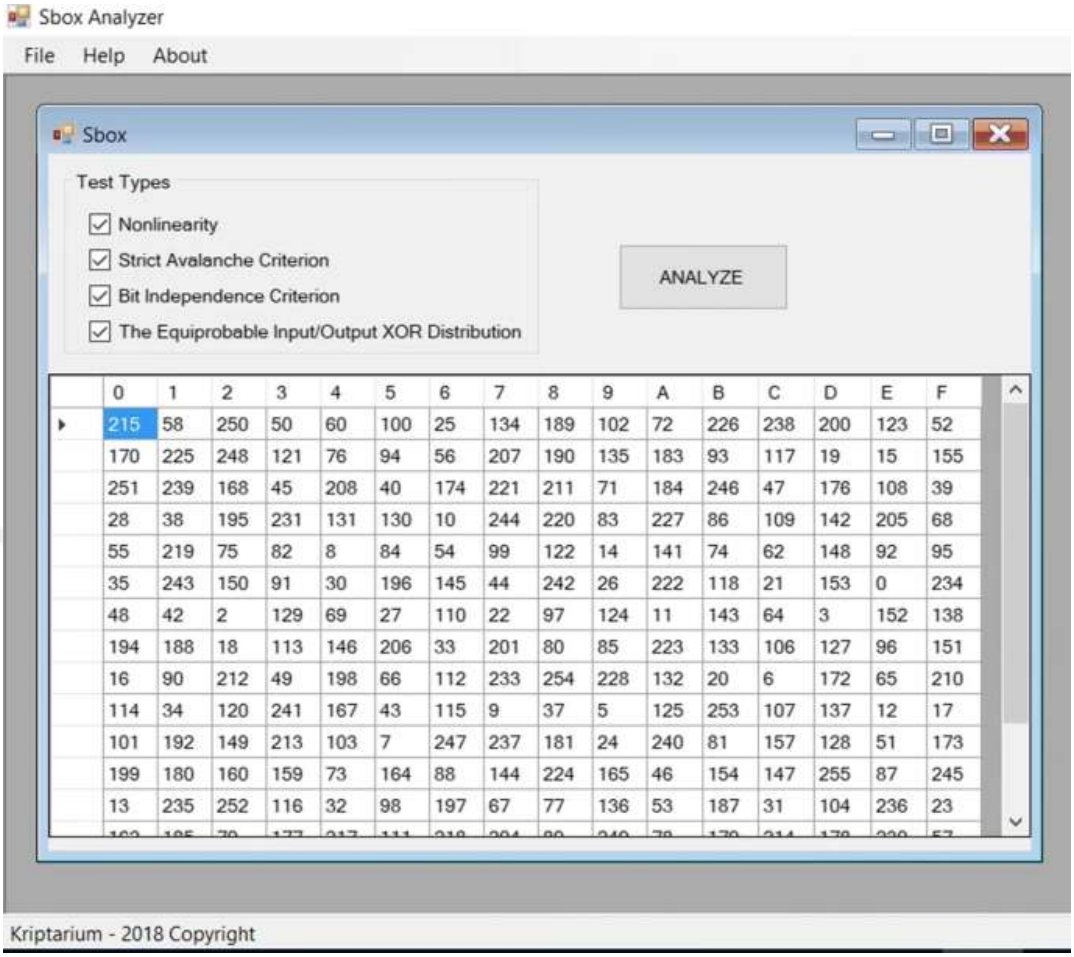

Figure 4.2. Analysis program screenshot

Hundreds of s-box structures were generated, but only those with a nonlinearity value of 105 and above have been made available to researchers in Table 4.1-Table 4.13.

Table 4.1. Generated s-box structure 1

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

Table 4.2. Generated s-box structure 2

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

Table 4.3. Generated s-box structure 3

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

Table 4.4. Generated s-box structure 4

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

Table 4.5. Generated s-box structure 5

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

Table 4.6. Generated s-box structure 6

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

Table 4.7. Generated s-box structure 7

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

Table 4.8. Generated s-box structure 8

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

Table 4.9. Generated s-box structure 9

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

Table 4.10. Generated s-box structure 10

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

Table 4.11. Generated s-box structure 11

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

Table 4.12. Generated s-box structure 12

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

Table 4.13. Generated s-box structure 13

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

4.2. Analysis Results of Generated S-box Dataset

Analysis results for five basic s-box success evaluation criteria are given in Table 4.14. A separate comparison table is not provided for these s-box builds. It has been observed that it is better than many designs in the literature listed in the second and third sections, both according to the nonlinearity criterion and the XOR distribution.

Table 4.14. Analysis results for s-box success evaluation criteria

S-box	Nonlinearity (NL)			Bit Independence Criterion		Strict Avalanche Criterion			Maximum I/O XOR
	min	max	avg	SAC	NL	min	max	avg	
Table 4.1	102	108	105.25	0.5019	103.64	0.375	0.6406	0.4998	10
Table 4.2	100	108	105.25	0.5024	102.36	0.4375	0.5938	0.5029	10
Table 4.3	104	108	105.5	0.4992	103.29	0.3438	0.5938	0.4966	10
Table 4.4	100	108	105	0.5046	103.86	0.4219	0.6406	0.51	10
Table 4.5	104	108	105	0.4976	104.14	0.4219	0.5938	0.501	10
Table 4.6	102	108	105	0.4997	103.57	0.3906	0.5781	0.5015	10
Table 4.7	102	108	105	0.4965	103.36	0.4062	0.5781	0.5007	10
Table 4.8	100	110	106.25	0.5044	102.71	0.3908	0.6094	0.5066	10
Table 4.9	102	108	105.25	0.4983	103.71	0.4062	0.5781	0.4978	10
Table 4.10	101	108	105.5	0.5024	103	0.3906	0.6094	0.4998	10
Table 4.11	102	108	105	0.4981	103.86	0.3906	0.5938	0.5029	10
Table 4.12	104	110	105.25	0.5053	104.79	0.3906	0.5781	0.4866	10
Table 4.13	102	108	105.25	0.4995	103.07	0.4219	0.5781	0.5039	10

5. A SUBSTITUTION BOX STRUCTURE BASED ON CHAOTIC MAY MAP

In this study, a cryptographic component is designed based on chaotic systems. The use of chaotic systems in the design of substitution-box structures as cryptological components is a common approach in the literature. The difference of this study from the similar ones is that the chaotic May map as a chaotic system has been used for the first time in the design of the substitution-box structures. The nonlinearity value of the substitution-box structure based on the chaotic May map has been measured as 106.75. This value is expressed as the highest value that can be achieved for chaos-based substitution-box structures. Considering the design criteria, it is thought that this successful substitution-box structure can be used in many information security applications in the future studies.

5.1. General Properties of Some Chaotic Maps

The successful role of chaotic systems in shaping the source of entropy has brought new research questions to the agenda of researchers. One of these areas of study has been how different chaotic systems can improve the randomness characteristics of the entropy source, or in other words, the design metrics of cryptological components. However, it has been observed in the results obtained in the recently published results that this area still needs to be investigated in more detail.

For example, in Figure 5.1 the graphs of Lyapunov exponents of discrete-time chaotic systems are shown. Lyapunov exponents are a chaos analysis tool. The positive Lyapunov exponent indicates that the system has an unpredictable behavior. In other words, one of the randomness requirements is satisfied.

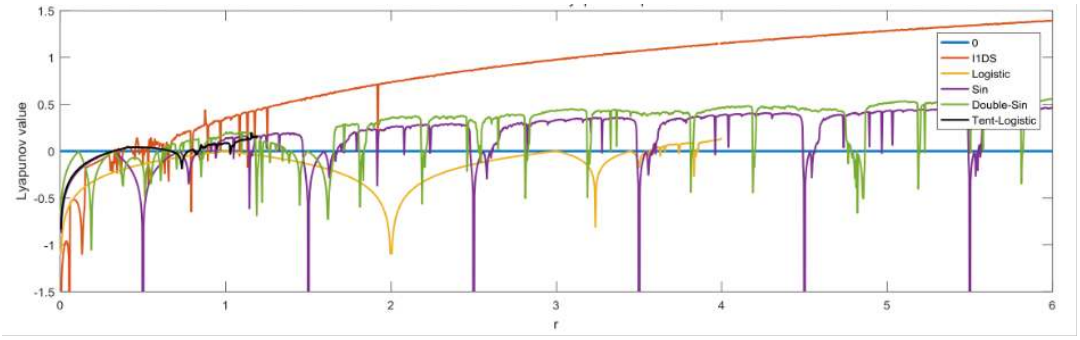

Figure 5.1. Lyapunov exponents of different discrete-time chaotic systems

The graph in Figure 5.1 shows that the outputs obtained have larger positive Lyapunov exponents, especially when more than one chaotic system is applied in a hybrid form. This indicates

that the role of alternative chaotic systems in cryptological component design should be investigated. Because the discrete-time chaotic system has an advantage over other types of chaotic systems thanks to its simple mathematical models. Mathematical models of several discrete time chaotic systems are given in Eq. (5.1)-(5.6), respectively [32].

Logistic Map:

$$x_{n+1} = a x_n (1 - x_n), \quad x_n \in [0,1], \quad a \in [3.5, 4] \tag{5.1}$$

Tent Map:

$$x_{n+1} = \begin{cases} a x_n & x_i < 0.5 \\ a(1 - x_n) & x_i \geq 0.5 \end{cases}, \quad x_n \in [0,1], \quad a \in [1,2] \tag{5.2}$$

Sine Map:

$$x_{n+1} = a \sin(\pi x_n), \quad x_n \in [0,1], \quad a \in [0.85,1] \tag{5.3}$$

Circle Map:

$$x_{n+1} = x_n + a - \frac{b}{2\pi} \sin(2\pi x_n) mod 1, \quad x_n \in [0,1], \quad a \in [0,1], b \in [0,4\pi] \tag{5.4}$$

May Map:

$$x_{n+1} = x_n \exp[a(1 - x_n)], \quad a \in [0,5] \tag{5.5}$$

Gaussian Map

$$x_{n+1} = \exp(-a {x_n}^2) + c, \quad a \in [4.7,17], c \in [-1,1] \tag{5.6}$$

5.2. S-box Generation Approach

The flowchart of the approach used to generate the s-box in this section is given in Figure 5.2. The proposed approach has a similar structure to the approach proposed in chapter 4. While in the previous method, the chaotic system values are directly converted to s-box structures, in this approach, the chaotic state variables are converted into a binary bit array via a quantization function. Then the bit values are divided into 8-bit-length pieces and converted to values between 0-255. Finally, these values are used directly in the creation of s-box structures. For the analysis results, the program described in the previous sections was used. At least 100 different s-box structures are tested for each chaotic system. Of these, only 10 with nonlinearity criteria of 106 and above are listed between Table 5.1, Table 5.2, Table 5.3, Table 5.4, Table 5.5 and Table 5.6, respectively. The rest of the dataset is made publicly available to researchers on the website.

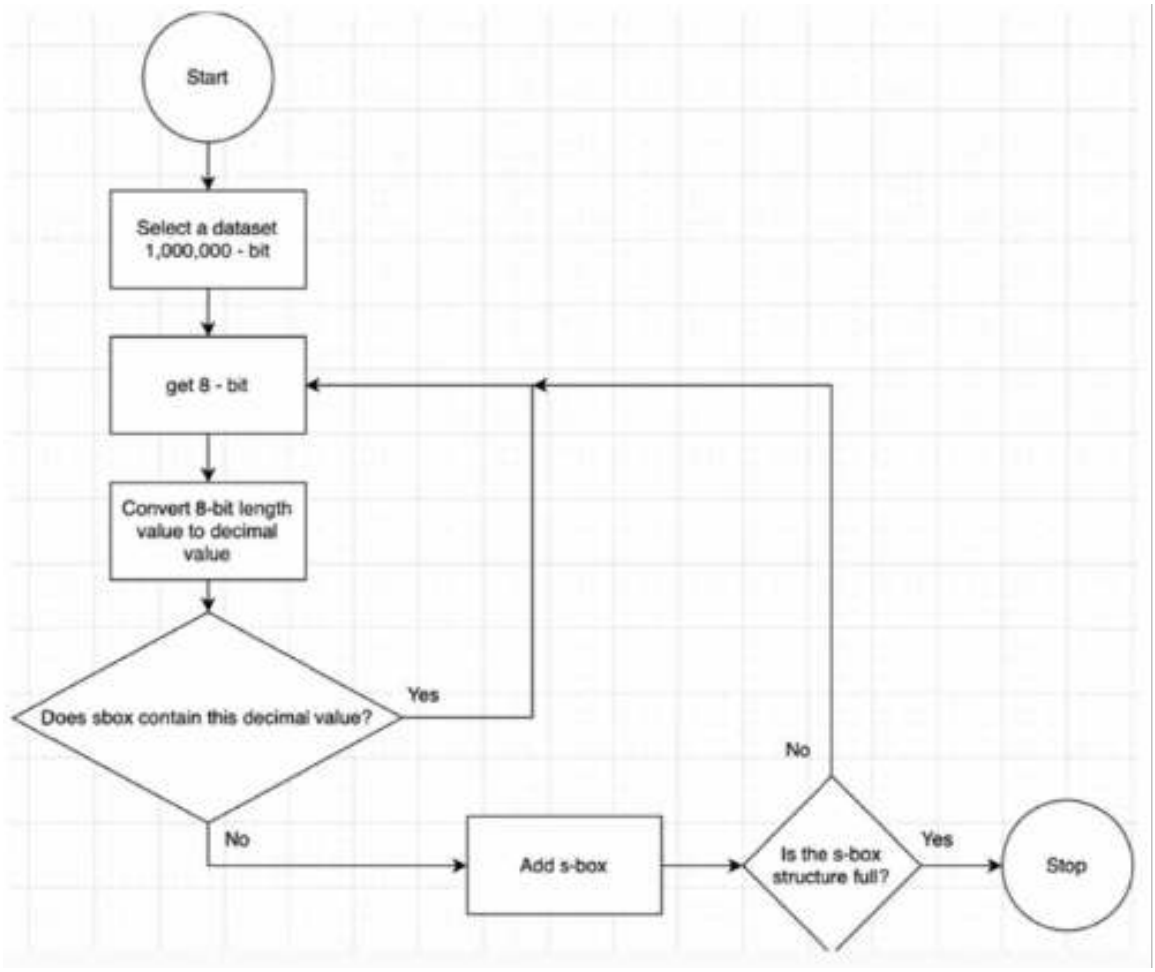

Figure 5.2. Flowchart of proposed s-box generator approach

Table 5.1. Analysis results for s-boxes based on Circle map

S-box Name	Nonlinearity			Bit Independence Criterion		Strict Avalanche Criterion			Maximum I/O XOR
	min	max	avg	SAC	NL	min	max	avg	
circle 1	102	110	106	0,4967	103,86	0,4062	0,5781	0,5056	10
circle 2	102	108	106	0,4998	102,5	0,4219	0,5625	0,5056	10
circle 3	104	108	106	0,4913	102,71	0,4219	0,5781	0,4927	10
circle 4	104	108	106,25	0,4985	103	0,4062	0,5625	0,4998	10
circle 5	104	108	106	0,4999	103,79	0,4219	0,5938	0,5034	10
circle 6	102	110	106	0,4967	103,86	0,4062	0,5781	0,5056	10
circle 7	104	108	106	0,4965	104,29	0,3906	0,6406	0,501	10
circle 8	104	110	106	0,4923	103,86	0,375	0,5781	0,4988	10
circle 9	102	110	106,25	0,5015	103,93	0,3281	0,5781	0,5015	10
circle 10	106	108	106,25	0,4985	103,71	0,4062	0,5781	0,5012	10

Table 5.2. Analysis results for s-boxes based on Gaussian map

S-box Name	Nonlinearity			Bit Independence Criterion		Strict Avalanche Criterion			Maximum I/O XOR
	min	max	avg	SAC	NL	min	max	avg	
Gaus map 1	104	108	106,25	0,5013	103,79	0,3906	0,6094	0,4998	10
Gaus map 2	104	108	106	0,4991	103,93	0,4219	0,6406	0,509	10
Gaus map 3	104	110	106,5	0,498	103,79	0,3906	0,5938	0,5076	10
Gaus map 4	102	108	106	0,4974	102,64	0,4061	0,6094	0,5081	10
Gaus map 5	102	110	106,5	0,4961	103,57	0,4219	0,625	0,5073	10
Gaus map 6	100	108	106	0,5019	103,64	0,3906	0,6094	0,4983	10
Gaus map 7	104	108	106,25	0,5031	103,71	0,3906	0,625	0,4941	10
Gaus map 8	100	110	106	0,5034	103,29	0,4219	0,5625	0,4961	10
Gaus map 9	104	108	106	0,4966	103,86	0,4062	0,5781	0,4983	10
Gausmap 10	102	108	106	0,4963	102,93	0,3906	0,5938	0,5037	10

Table 5.3. Analysis results for s-boxes based on Logistic map

S-box Name	Nonlinearity			Bit Independence Criterion		Strict Avalanche Criterion			Maximum I/O XOR
	min	max	avg	SAC	NL	min	max	avg	
Logistics 1	104	108	106,25	0,4956	103,93	0,4062	0,5938	0,4983	10
Logistics 2	104	108	16	0,4995	103,14	0,4062	0,5938	0,4978	10
Logistics 3	100	108	106	0,5019	103,64	0,3906	0,6094	0,4983	10
Logistics 4	104	108	106,25	0,5031	103,71	0,3906	0,625	0,4941	10
Logistics 5	104	108	106	0,4932	104,36	0,3906	0,5781	0,4973	10
Logistics 6	104	108	106	0,4966	103,86	0,4062	0,5781	0,4983	10
Logistics 7	104	108	106	0,502	101,86	0,3906	0,5938	0,4995	10
Logistics 8	104	110	106	0,5004	103,29	0,4062	0,5938	0,5012	10
Logistics 9	104	108	106,25	0,4997	103,21	0,3906	0,6406	0,4963	10
Logistics10	102	108	106	0,4989	103,64	0,4219	0,5938	0,4995	10

Table 5.4. Analysis results for s-boxes based on May map

S-box Name	Nonlinearity			Bit Independence Criterion		Strict Avalanche Criterion			Maximum I/O XOR
	min	max	avg	SAC	NL	min	max	avg	
May map 1	106	108	106,75	0,5004	103,29	0,4062	0,6094	0,5051	10
May map 2	104	108	106	0,5056	103,71	0,4062	0,6094	0,5061	10
May map 3	104	108	106,25	0,5012	102,93	0,4219	0,625	0,5042	10
May map 4	104	110	106,5	0,4992	103,29	0,4062	0,625	0,5044	10
May map 5	104	108	106,25	0,5043	103,21	0,4219	0,5781	0,499	10
May map 6	104	108	106,5	0,4996	102	0,4219	0,6094	0,5059	10
May map 7	104	108	106,25	0,5038	102,57	0,4062	0,5938	0,5049	10
May map 8	104	108	106	0,4912	102,29	0,4219	0,6094	0,5059	10
May map 9	102	108	106	0,4969	103,07	0,4062	0,625	0,5005	10
May map10	102	108	106	0,4939	104,43	0,3594	0,6094	0,5005	10

Table 5.5. Analysis results for s-boxes based on Sin map

S-box Name	Nonlinearity			Bit Independence Criterion		Strict Avalanche Criterion			Maximum I/O XOR
	min	max	avg	SAC	NL	min	max	avg	
Sin 1	102	108	106,25	0,4955	104,43	0,4062	0,6094	0,5	10
Sin 2	104	108	106	0,4946	104,43	0,4062	0,5781	0,509	10
Sin 3	104	108	106	0,5002	102,93	0,4062	0,6094	0,4917	10
Sin 4	104	108	106,5	0,5034	103,64	0,4375	0,5781	0,5034	10
Sin 5	102	108	106	0,5001	103,79	0,3906	0,5938	0,5061	10
Sin 6	104	108	106	0,5034	104,64	0,3906	0,5938	0,5034	10
Sin 7	102	108	106	0,5017	103,86	0,4062	0,5938	0,5054	10
Sin 8	104	108	106,5	0,4985	102,86	0,4219	0,5938	0,5129	10
Sin 9	104	108	106	0,5003	103,86	0,4062	0,6094	0,4985	10
Sin 10	104	108	106,25	0,4935	103,43	0,4062	0,5625	0,4954	10

Table 5.6. Analysis results for s-boxes based on Tent map

S-box Name	Nonlinearity			Bit Independence Criterion		Strict Avalanche Criterion			Maximum I/O XOR
	min	max	avg	SAC	NL	min	max	avg	
Tent 1	104	110	106,5	0,5022	103,79	0,4219	0,625	0,4951	10
Tent 2	102	110	106	0,5025	104,43	0,4062	0,6094	0,4988	10
Tent 3	104	108	106,25	0,4997	104,21	0,3906	0,5938	0,5029	10
Tent 4	104	108	106,25	0,4995	102,93	0,375	0,6094	0,5056	10
Tent 5	102	108	106	0,4995	103	0,3594	0,5938	0,4961	10
Tent 6	104	110	106,25	0,4994	104,07	0,4062	0,5625	0,4951	10
Tent 7	104	108	106	0,5013	103,93	0,3594	0,5938	0,4985	10
Tent 8	102	110	106,25	0,501	104,36	0,4219	0,5938	0,5056	10
Tent 9	104	108	106,25	0,4978	102,86	0,5219	0,6094	0,5073	10
Tent 10	102	108	106	0,4918	103,29	0,3906	0,5938	0,4922	10

All of the sixty different s-box structures generated by ten different s-box structures produced from each of the six different chaotic systems in this section have better s-box design metrics than the s-box structures generated in the previous three sections and most designs in the literature. In addition, the s-box structures presented here constitute only a part of the publicly shared dataset. It is thought that these outputs will provide a great advantage for researchers in many practical applications.

6. CONCLUSIONS

The most remarkable of the application areas where chaotic systems are successful is undoubtedly the design of cryptographic components. Thousands of examples of chaos-based design are the most important quantitative indicator of this claim. A prominent chaos-based cryptographic component design among these studies has been the substitution boxes (s-boxes). The fact that the design metrics of these components can be measured more easily has helped researchers to evaluate and interpret the outputs more easily. In this context, improving the nonlinearity criterion of s-box structures based on random selection has been a hot research topic. Recently, especially the demonstration that chaos-based s-box structures can be an alternative to prevent application-oriented attacks has increased this hot research topic many times over.

In this thesis, it is aimed to investigate random selection based s-box approaches in detail, to bring new designs to the literature and to present a projection for future studies. Within the scope of the studies, many s-box structures have been obtained through four different approaches. The evaluation of all these results has been tried to be emphasized in the following items.

- The classical rand() function of programming languages is used as an entropy source. The advantages and disadvantages of these structures compared to chaos-based structures are discussed. It has been observed that classical pseudo-random number generators (PRNG) have a nonlinearity value close to chaos-based approaches.
- Among the s-box structures produced using the classical rand() function, the most successful twenty s-box structures are presented as a dataset. It is thought that this dataset may be important in terms of creating a ready-made initial dataset that can be used in studies to improve nonlinearity measurement by applying post-processing techniques in the future.
- In order to evaluate the success of pseudo-random number generators from a different perspective, the role of an entropy source in the s-box design process, based on true random number generators (TRNG), which constitute the other leg of random number generators, is analyzed. In the thesis, an entropy source has been created using a TRNG hardware, and then its success in the s-box design was comprehensively analyzed. More than five hundred s-box structures were used in the analysis. Again, among these, eighty different successful s-box structures were shared as a public dataset.
- As a third approach, s-box structures were generated by using the entropy source created by a noise source. 13 successful s-box structures based on this entropy source were also made available to researchers.
- Finally, 176 different s-box structures with nonlinearity values of 106 and above were generated for six different chaotic systems.

All these results still indicate that more chaotic systems are a more powerful resource for shaping s-box structures than both DRNG and TRNG based designs. Hundreds of different chaotic systems waiting to be explored are a robust resource from which new data sets can be created. It indicates that these outputs can be used successfully in many practical designs in the future.

REFERENCES

[1] M. Harbawi and A. Varol, "An improved digital evidence acquisition model for the Internet of Things forensic I: A theoretical framework," 2017 5th International Symposium on Digital Forensic and Security (ISDFS), 2017, pp. 1-6, doi: 10.1109/ISDFS.2017.7916508.

[2] B. Arslan, S. Gunduz and S. Sagiroglu, "A review on mobile threats and machine learning based detection approaches," 2016 4th International Symposium on Digital Forensic and Security (ISDFS), 2016, pp. 7-13, doi: 10.1109/ISDFS.2016.7473509.

[3] J. Daemen and V. Rijmen, "AES proposal: Rijndael," in Proc. 1st Adv. Encryption Conf., CA, USA, 1998, pp. 1–45.

[4] M. Suárez-Albela, T. M. Fernández-Caramés, P. Fraga-Lamas and L. Castedo, "A Practical Performance Comparison of ECC and RSA for Resource-Constrained IoT Devices," 2018 Global Internet of Things Summit (GIoTS), 2018, pp. 1-6, doi: 10.1109/GIOTS.2018.8534575.

[5] B. Chevallier-Mames, M. Ciet and M. Joye, "Low-cost solutions for preventing simple side-channel analysis: side-channel atomicity," in IEEE Transactions on Computers, vol. 53, no. 6, pp. 760-768, June 2004, doi: 10.1109/TC.2004.13.

[6] https://www.crowdsupply.com/13-37/infinite-noise-trng

[7] A. M., Garipcan, E. Erdem, Implementation and Performance Analysis of True Random Number Generator on FPGA Environment by Using Non-periodic Chaotic Signals Obtained from Chaotic Maps. Arab J Sci Eng 44, 9427–9441 (2019). https://doi.org/10.1007/s13369-019-04027-x

[8] A. M., Garipcan, E. Erdem, A TRNG using chaotic entropy pool as a post-processing technique: analysis, design and FPGA implementation. Analog Integr Circ Sig Process 103, 391–410 (2020). https://doi.org/10.1007/s10470-020-01605-0

[9] FIPS PUB 202, FEDERAL INFORMATION PROCESSING STANDARDS PUBLICATION, SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions, August 2015, http://dx.doi.org/10.6028/NIST.FIPS.202

[10] A. S. Muhammad, F. Özkaynak, "SIEA: Secure Image Encryption Algorithm Based on Chaotic Systems Optimization Algorithms and PUFs", Symmetry 2021

[11] T. Cusick and P. Stanica, Cryptographic Boolean Functions and Applica- tions. Amsterdam, The Netherlands: Elsevier, 2009.

[12] C. Wu and D. Feng, Boolean Functions and Their Applications in Cryp- tography. Berlin, Germany: Springer, 2016.

[13] 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, 2015.

[14] I. Hussain, H. Shah, A. Mahmood, and M. 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, 2013.

[15] M. Khan and T. Shah, "A novel image encryption technique based on Hénon chaotic map and S8 symmetric group," Neural Comput. Appl., vol. 25, nos. 7–8, pp. 1717–1722, 2014.

[16] 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.

[17] A. Belazi, M. Khan, A. Latif, and S. Belghith, "Efficient cryptosys- tem approaches: S-boxes and permutation–substitution-based encryption," Nonlinear Dyn., vol. 87, no. 1, pp. 337–361, 2017.

[18] I. Hussain, T. Shah, M. Gondal, and H. Mahmood, "A novel method for designing nonlinear component for block cipher based on TD-ERCS chaotic sequence," Nonlinear Dyn., vol. 73, no. 1, pp. 633–637, 2013.

[19] K. Khalid Butt, G. Li, F. Masood, S. Khan, A Digital Image Confidentiality Scheme Based on Pseudo-Quantum Chaos and Lucas Sequence, Entropy 2020, 22(11), 1276; https://doi.org/10.3390/e22111276

[20] F. Özkaynak, “On the effect of chaotic system in performance character- istics of chaos based S-box designs,” Phys. A, Stat. Mech. Appl., vol. 550, Jul. 2020, Art. no. 124072, doi: 10.1016/j.physa.2019.124072.

[21] M. Ş. Açikkapi and F. Özkaynak, "A Method to Determine the Most Suitable Initial Conditions of Chaotic Map in Statistical Randomness Applications," in IEEE Access, vol. 9, pp. 1482-1494, 2021, doi: 10.1109/ACCESS.2020.3046470.

[22] F. Özkaynak, “From biometric data to cryptographic primitives: A new method for generation of substitution boxes,” in Proc. ACM Int. Conf. Biomed. Eng. Bioinformat., Bangkok, Thailand, Sep. 2017, pp. 27–33. doi: 10.1145/3143344.3143355.

[23] F. Artuğer and F. Özkaynak, “A novel method for performance improvement of chaos-based substitution boxes,” Symmetry, vol. 12, no. 4, p. 571, Apr. 2020.

[24] I. Hussain, T. Shah, H. Mahmood, and M. Gondal, Construction of S8 Liu J S-boxes and their applications,” Comput. Math. Appl., vol. 64, no. 8, pp. 2450–2458, 2012.

[25] I. Hussain, T. Shah, M. Gondal, W. Khan, and H. Mahmood, “A group theoretic approach to construct cryptographically strong substitution boxes,” Neural Comput. Appl., vol. 23, no. 1, pp. 97–104, 2013.

[26] I. Hussain, T. Shah, and M. Gondal, “A novel approach for designing substitution-boxes based on nonlinear chaotic algorithm,” Nonlinear Dyn., vol. 70, no. 3, pp. 1791–1794, 2012.

[27] M. Khanand, and T. Shah, “An efficient construction of substitution box with fractional chaotic system,” Signal, Image Video Process., vol. 9, no. 6, pp. 1335–1338, 2015.

[28] F. Özkaynak, V. Çelik, and A. B. Özer, “A news -box construction method based on the fractional-order chaotic Chen system,” Signal, Image Video Process., vol. 11, no. 4, pp. 659–664, 2017.

[29] 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., pp. 1–10, 2019. doi: 10.1007/s40998-019-00230-6.

[30] 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, 2018. doi: 10.3390/app8122650.

[31] G. Chen, “A novel heuristic method for obtaining S-boxes,” Chaos, Soli- tons Fractals, vol. 36, no. 4, pp. 1028–1036, 2008.

[32] X. Wang, J. Yang, “A novel image encryption scheme of dynamic S-boxes and random blocks based on spatiotemporal chaotic system”, Optik - International Journal for Light and Electron Optics 217 (2020) 164884

[33] N.A., Khan, M. Altaf, and F. A. Khan, “Selective encryption of JPEG images with chaotic based novel S-box.” Multimed Tools Appl (2020). https://doi.org/10.1007/s11042-020-10110-5

[34] M. Khan, T. Shah, and M. Gondal, “An efficient technique for the construction of substitution box with chaotic partial differential equation,” Nonlinear Dyn., vol. 73, no. 3, pp. 1795–1801, 2013.

[35] 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, 2014.

[36] 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.

[37] M. Khan, T. Shah, and S. Batool, “A new implementation of chaotic S-boxes in CAPTCHA,” Signal, Image Video Process., vol. 10, no. 2, pp. 293–300, 2016.

[38] 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, 2005.
[39] 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.
[40] 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, 2005.
[41] N. Hematpour and S. Ahadpour, “Execution examination of chaotic S- box dependent on improved PSO algorithm,” Neural Comput. Appl., Aug. 2020, doi: 10.1007/s00521-020-05304-9.
[42] 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, Feb. 2011.
[43] 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, 2010.
[44] M. Khan, T. Shah, H. Mahmood, M. 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, 2012.
[45] 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, 2007.
[46] M. Khan, T. Shah, H. Mahmood, and M. Gondal, “An efficient method for the construction of block cipher with multi-chaotic systems,” Nonlinear Dyn., vol. 71, no. 3, pp. 489–492, 2013.
[47] M. Khan and Z. Asghar, “A novel construction of substitution box for image encryption applications with Gingerbreadman chaotic map and S8 permutation,” Neural Comput. Appl., vol. 29, no. 4, pp. 993–999, Feb. 2018. doi: 10.1007/s00521-016-2511-5.
[48] S. Jamal, M. Khan, and T. Shah, “A watermarking technique with chaotic fractional S-box transformation,” Wireless Pers. Commun., vol. 90, no. 4, pp. 2033–2049, 2016.
[49] B. Cassal-Quiroga and E. Campos-Canton, “Generation of Dynamical S-Boxes for Block Ciphers via Extended Logistic Map”, Mathematical Problems in Engineering Volume 2020, https://doi.org/10.1155/2020/2702653
[50] A. Freyre-Echevarria, I. Martinez-Diaz, C. M. L. Perez, G. Sosa-Gomez, and O. Rojas, “Evolving nonlinear S-boxes with improved theoretical resilience to power attacks,” IEEE Access, vol. 8, pp. 202728–202737, 2020, doi: 10.1109/ACCESS.2020.3035163.
[51] M. Khan, “A novel image encryption scheme based on multiple chaotic S-boxes,” Nonlinear Dyn., vol. 82, no. 1, pp. 527–533, 2015.
[52] M. Khan, T. Shah, and S. Batool, “Construction of S-box based on chaotic Boolean functions and its application in image encryption,” Neural Com- put. Appl., vol. 27, no. 3, pp. 677–685, 2016
[53] F. Artuğer and F. Özkaynak, “An Effective Method to Improve Nonlinearity Value of Substitution Boxes based on Random Selection”, Infirmation Sciences, 2021

CURRICULUM VITAE

Habib IBRAHIM

[redacted]

ACADEMIC ACTIVITIES

Proceedings:

1. Habib Ibrahim, Fatih Özkaynak, A Substitution-Box Structure Based on Crowd Supply Infinite Noise TRNG, IEEE The 9th International Symposium on Digital Forensics and Security (ISDFS 2021), Elazığ, TURKEY, June 28&29, 2021
2. Habib Ibrahim, Fatih Özkaynak, A Substitution Box Structure Based on Chaotic May Map, 14th Chaotic Modeling and Simulation International Conference, 8 - 11 June, 2021, Athens, Greece
3. Habib Ibrahim, Fatih Özkaynak, A Random Selection Based Substitution-box Structure Dataset for Cryptology Applications, IEEE EUROCON-2021 (2021 IEEE 19th International Conference on Smart Technologies (EUROCON), July 6 – 8, 2021 (in review).

Projects:

1. Researcher, Fırat University Scientific Research Project (TEKF 20.21), Generation of Substitution Boxes Based on Random Selection and Implementation of Its Practical Applications