



Strong s-box construction approach based on Josephus problem

Firat Artuğer¹

Accepted: 9 February 2024 / Published online: 10 July 2024
© The Author(s) 2024

Abstract

There are two basic requirements for symmetric encryption algorithms. The first of these is diffusion. The second and most important is confusion. In these algorithms, this requirement is usually met by s-box structures. Therefore, s-box structures must be strong. So, a cryptographically good s-box will make the encryption algorithm difficult to crack. However, obtaining a strong s-box is a rather difficult problem. In this study, Josephus circle logic is used to solve this problem. Initially, with a random s-box structure, the elements are replaced according to their Josephus positions, and the s-box is made stronger. In the proposed algorithm, according to the Josephus logic, the elements that kill each other are replaced and this process continues until one element remains. The last 30 surviving elements are replaced with all elements. In this way, three different s-boxes were obtained. In two of them, the nonlinearity value was 110.5, and in one of them, the nonlinearity value was 110.75. Fixed points in the proposed s-box structures were identified and eliminated. In addition, it has been proven because of the analysis that the obtained s-box structures also meet other cryptographic requirements. In addition to leaving behind most studies in the literature, the proposed method will also provide strong s-box structures for encryption algorithms to be built in the future.

Keywords S-box · High nonlinearity · Josephus problem · Chaos · Symmetric encryption

1 Introduction

Network technologies continue to evolve to meet today's increasing communication needs. One of the most important concepts during this communication is the security of the communication and the data transferred. Users are often worried about this. The topic of security is now a hot topic that attracts the attention of almost all people. However, with these developing technologies, attacks on these data are increasing day by day. At this point, the concept of cryptography emerges. The need for cryptographic structures has become inevitable for secure communication and the safe transfer of information. One of the most important pillars of cryptography science is the development of encryption algorithms to ensure confidentiality. Today's encryption standard AES (Daemen and Rijmen 1998) algorithm, frequently used DES algorithm (Standard 1999)

and many other important algorithms are called block cipher algorithms. These algorithms ensure that communication and data transfer are carried out securely. To fulfill today's requirements, the need for encryption algorithms in different sizes and speeds is increasing, especially according to the capacities of applications. Confusion and diffusion are two important requirements for block cipher algorithms (Artuğer and Özkaynak 2022a). The confusion requirement, which is the most important of these requirements, is usually provided by s-box structures. Because s-box is a nonlinear transformation and provides a mixing of data. Therefore, the s-box structure is one of the important factors that determine the strength of the block cipher algorithm.

The S-box can be seen as a simple nonlinear structure that transforms data from one form to another. However, considering the advantages it provides, its importance is an undeniable fact. Obtaining a strong s-box will turn into a very difficult problem, especially as its dimensions increase. For example, for AES and similar algorithms, an s-box structure consisting of 16 rows and 16 columns is used. That means 256 elements. Searching a table of 256 elements means 256! means situation. This means a

✉ Firat Artuğer
firatartuger@munzur.edu.tr

¹ Department of Computer Engineering, Munzur University, 62200 Tunceli, Turkey

problem that cannot be solved in a certain time. Therefore, researchers continue to work rapidly to bring different solutions to this problem. More than 150 studies have been done in the literature to obtain a strong s-box. This is one of the sides showing that the optimum solution is still not achieved. Because the advantages and disadvantages of the methods arise when obtaining an s-box. In other words, it says that the search for optimum solutions should continue. At the beginning of these existing approaches; chaos, optimization techniques, mathematical transformations, cellular automata, DNA and RNA chain structures. In addition to these, different techniques have been used. Chaos-based techniques are quite popular. They produce s-box structures that are easy to obtain but cryptographically weak. In mathematical transformations, cryptographically strong s-box structures are obtained. However, these structures may show weaknesses against some differential attacks (Örs et al. 2007). Optimization techniques are still frequently used. In these methods, cryptographically strong structures can be obtained. However, time complexity and computational costs can be high in these structures. Cellular automata structures and DNA-RNA chains are like optimization techniques. In this study, unlike these, an s-box generation architecture was created based on the Josephus problem. As a result of the analysis, it has been shown that this proposed method can be an alternative to the methods in the literature. Some of the above-mentioned studies, especially those developed in recent years, are mentioned below. In addition, a performance comparison was made with the proposed method.

General logic in optimization-based methods; It is the selection of the elements to be replaced in the s-box structure with the optimization algorithm. In these studies, two elements are selected and replaced by the optimization algorithm until a certain iteration. Then, if the fitness value has increased, the s-box structure is updated. As the fitness value, the most important parameter, the nonlinearity value, is used. In recent years, many optimization methods have been used to overcome the s-box generation problem. Some algorithms have been used many times in different ways and continue to be used. The time complexity of these algorithms is quite high. In other words, it may take a long time for the algorithm to obtain a strong s-box. Perhaps newly developed or to be developed metaheuristic algorithms can eliminate this problem. This could be a new research question for researchers. Looking at the studies so far; the studies referenced Wang et al. (2012, 2020), Kang and Wang (2022), Artuğer and Özkaynak (2022b), developed new s-box structures based on the genetic algorithm. Particle swarm optimization was used in the studies referenced Ahmad et al. (2020), Khan et al. (2021), Hematpour and Ahadpour (2021) and effective results were obtained. In the studies Alhadawi et al. (2020), Ahmed

et al. (2019), s-box generation algorithms were developed using the firefly algorithm. In the study Zamli et al. (2023), naked mole rat algorithm was used in the study. In the study Tian and Lu (2017), s-box structures were developed using the bacterial foraging optimization algorithm. In the study Farah et al. (2017), the learning-teaching optimization algorithm was used. In the study Zamli et al. (2021), effective s-box structures were developed using the tiki-taka algorithm. Hopfield neural network algorithm was used in the study Ahmad and Al-Solami (2020a). In the study Alhadawi et al. (2021), the cuckoo search algorithm was used. In the study Zamli (2021), the agent heroes and cowards optimization algorithms were used. In the study Chen (2008), the annealing algorithm was used. In addition, this study is one of the first optimization algorithms used to develop s-box. In the study Ahmad et al. (2015), ant colony optimization algorithm was used. In the study Kuznetsov et al. (2022), the s-box was developed using the hill climbing algorithm. In the study Alzaidi et al. (2018), strong s-box structures were tried to be obtained by using the sine cosine optimization algorithm. In the study Khan et al. (2023), strong s-box structures were obtained by using the gray wolf optimization algorithm. In the study Zamli et al. (2022), the s-box was created using the Elitist Barnacles Mating optimization algorithm. In the studies referenced Basha et al. (2022), Al-Wattar et al. (2015a, b), Kadhim et al. (2016), Mohamed et al. (2021), Farhan et al. (2020), algorithms for obtaining s-box were developed using DNA and RNA coding or chain structures. In these algorithms, results parallel to the optimization methods can be obtained in general. However, computational costs may increase with these methods. In studies Aboytes-González et al. (2018), Haque et al. (2022), Gangadari and Ahamed (2018), Szaban and Seredynski (2008), Mariot et al. (2019), Alexan et al. (2022), s-box was developed based on cellular automata structures. In these structures, it is like DNA-RNA coding and optimization techniques. Another approach is s-box structures obtained using chaotic maps. The advantage of these structures is that they are fast and less complex. However, nonlinearity values are generally low. Various methods have been proposed to improve the performance of such algorithms (Artuğer and Özkaynak 2020, 2021). Looking at the chaotic maps used based on studies, it is seen that chaotic Chen system in Özkaynak et al. (2017), logistic map in Özkaynak (2020a), Lorenz Duffing system in Ye and Zhimao (2018), Tent map in Tang and Liao (2005), Zhongtang system in Çavuşoğlu et al. (2017), Baker map in Tang et al. (2005), Lorenz systems in Özkaynak and Özer (2010), Khan et al. (2012), hyperchaotic system in Yang et al. (2023), a new discrete-time chaotic map in Lambić (2020), a one-dimensional discrete chaotic map in Lambić (2018), the labyrinth Rene Thomas system was used in Özkaynak (2020b), the partial

differential equation in Ahmed et al. (2019), the sinus map in Belazi and Abd El-Latif (2017), and the fractional chaotic system in Khan and Shah (2015). With the help of all these chaotic maps and many more chaotic maps, different s-box generation algorithms have been developed. However, since the nonlinearity value of these structures remains low, either the nonlinearity value of these structures should be increased, or alternative solutions should be developed. Finally, mathematical transformations are frequently used. The biggest advantage of these structures is that they are s-box structures that show very good cryptographic properties. Particularly when nonlinearity values are considered, they are one step ahead of other studies. However, such s-box structures have disadvantages against some application attacks (Örs et al. 2007). In mathematical transformation-based s-box structures; Boolean functions were used in the studies Hussain (2020), Bakunina and Dykyi (2022), Sokolov and Radush (2022), mobius group and finite field were used in the studies Arshad et al. (2022), Anees and Chen (2020). A group theory was used in the studies Razaq et al. (2022a, b). In the study Razaq et al. (2021), group theory and graph structure were used. In the study Isa et al. (2023), binomial power functions were used. In Khan and Shah (2014), a symmetrical group, in Ahmad and Al-Solami (2020b), algebraic structure and complex behavior, in Malik et al. (2020), rotation matrices, and in Ahmad et al. (2015), Siddiqui et al. (2020) a modular group action, it was desired to obtain a strong s-box.

Many different philosophies have been used to produce S-boxes, as mentioned above. It is thought that the Josephus problem approach used in this study can be among these philosophies. Thus, this study can be a source of inspiration for researchers to use the Josephus problem in the production of s-boxes or in solving different problems. This is one of the most important contributions of the study. Another contribution of the study is obtaining high nonlinearity values. Obtaining similar high nonlinearity values in 3 different s-box structures puts the study ahead of other studies in the literature. Finally, fixed points in the s-box structures obtained in this study were eliminated with an innovative approach. In this way, the recommended s-boxes have been made stronger. In most studies in the literature, fixed points are not controlled. This study shows the importance of fixed points in s-box structures and how they can be eliminated. In the remaining part of the study, the Josephus problem is explained in the second part. In the third chapter, the proposed algorithm is given together with the flow chart. In the fourth chapter, s-box performance metrics are explained, and 3 different s-box structures obtained with the proposed algorithm are analyzed. In the fifth section, the results are discussed.

2 Josephus problem

The Josephus Problem is named after the historian Flavius Josephus, who participated in the Jewish revolt against the Romans. As a general, Josephus managed to defend the fortress of Jotapata for 47 days. However, after the fall of the city, he was trapped in a nearby cave with 40 soldiers. Flavius and his friends, who said that it was more honorable to commit suicide instead of being killed by enemy soldiers, wanted to do this in a certain order. For this, they are first arranged to form a circle. Legend has it that Flavius Josephus will eventually survive this deadly game, thanks to his mathematical abilities. For this, he persuades the soldiers to kill each other. In this way, each soldier will kill the soldier next to him. Together with Josephus, a total of 41 people are lined up in a circle. Josephus immediately calculated the survivor and positioned himself in the appropriate place of the circle. So how did Josephus calculate this? In the problem, an example of which is given in Fig. 1 with 8 soldiers, the circle starts from the first soldier and moves to the left. In this way, the first soldier kills the

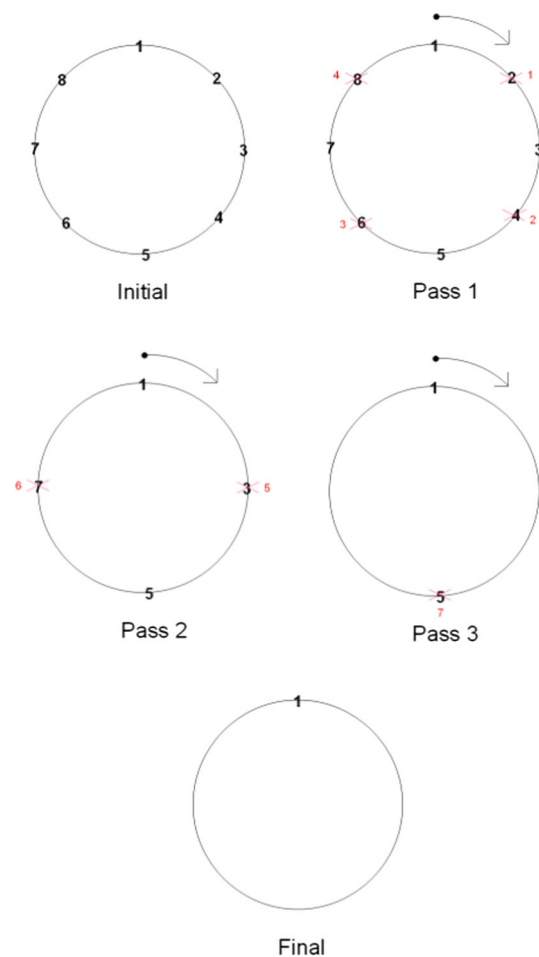


Fig. 1 Example of playing the Josephus problem with 8 players

second, the third soldier the fourth, the fifth soldier the sixth, and the seventh soldier the eighth. The four survivors after the first round continue in the same way. The first soldier kills the third, the fifth soldier kills the seventh. In the final round, the first soldier kills the fifth soldier and becomes the last survivor. Let's assume that $k(n)$ is the winning player. If there is an even number of soldiers, we can say that $k(2n) = 2k(n) - 1$. If there is an odd number of soldiers then $k(2n + 1) = 2k(n) + 1$. In addition, the surviving soldiers according to the number of some soldiers are given in Table 1.

When Table 1 is examined, one of the striking results is that the soldier in position 1 mostly survived. Namely, when the number of soldiers is one of the bases of 2, the first soldier will always survive. In other words, if there are soldiers in the numbers 1, 2, 4, 8, 16, 32, 64, 128, 256... we can say that the last soldier remaining is the first soldier.

3 Proposed algorithm

In the proposed method, firstly, a randomly generated s-box structure is taken. Then, considering the Josephus problem, there are 256 soldiers in this s-box structure, since there are 256 values. Looking at the structure in Fig. 1, the first soldier kills the second soldier. Accordingly, in the proposed method, the first value and the second value are replaced in the s-box. The nonlinearity value is then calculated. If this value increases or stays equal, the s-box is updated. In this way, the algorithm works one round for all elements. That is, up to 256 such elements as 3 and 4, 5 and 6, 7 and 8 are replaced, respectively, and checked according to the new s-box nonlinearity value. This way, the values killed at the end of a round will not be looked at again. The same process continues until only one element remains among the surviving values. Finally, the 30 surviving values are hovered by replacing all the elements in the s-box structure. In this way, a weak s-box structure can

become strong, especially with a high nonlinearity value. The general steps of the proposed algorithm are given below. In addition, the flow chart of the proposed method is given in Fig. 2.

Step 1 A random cryptographically weak s-box is taken.

Step 2 S-box values are located according to the Josephus problem.

Step 3 Elements that will kill each other are replaced in order.

Step 4 The nonlinearity value of the new s-box is calculated.

Step 5 If the new nonlinearity value is greater than the old nonlinearity value, the s-box is updated.

Step 6 Continue from Step 3 until one element remains.

Step 7 The last 30 elements are saved.

Step 8 These elements are replaced by all elements in turn.

Step 9 If the new nonlinearity value is greater than the old nonlinearity value, the s-box is updated.

Step 10 Continue from Step 8 until iteration is complete.

Step 11 The s-box with high nonlinearity value is obtained.

4 Analysis results

When we look at the literature, bijectivity, strict avalanche criterion (SAC), independence from output bits criterion (BIC), I/O XOR distribution, and nonlinearity values are generally used to analyze an s-box structure. Bijective means that each value should only be used once. In other words, since there are 256 values in the proposed s-box structures, each of the elements between 0 and 256 should be used only once. All the proposed s-box structures are bijective. Looking at the SAC value, a change made at the input for data to be encrypted should change half of the bits at the output (Webster and Tavares 1985). This value is taken as 0.5. These and similar values are considered valid for an s-box structure. In the proposed s-box structures, the SAC values were very close to 0.5. So, it can be said that this criterion is also met. Another and most important criterion is the nonlinearity value. This value is vital for an s-box structure. Because s-box structures are nonlinear structures and provide confusion. The linearity of these structures makes them easy to break. The nonlinearity values of these structures should be as high as possible so that no inference can be made by the attackers. In today's encryption standard AES algorithm, this value is 112, which is a perfect s-box structure. In s-box structures to be developed for alternative encryption algorithms, this value should be at least close to this value. In fact, some studies conducted today have shown that the nonlinearity value can exceed the AES algorithm. Nonlinearity values of 114.5 (Dimitrov 2020) and 116 (Dimitrov

Table 1 The last survivor by some soldier numbers

Number of soldiers	Surviving soldier
1	1
2	1
3	3
4	1
5	3
6	5
8	1
11	7
16	1
20	9

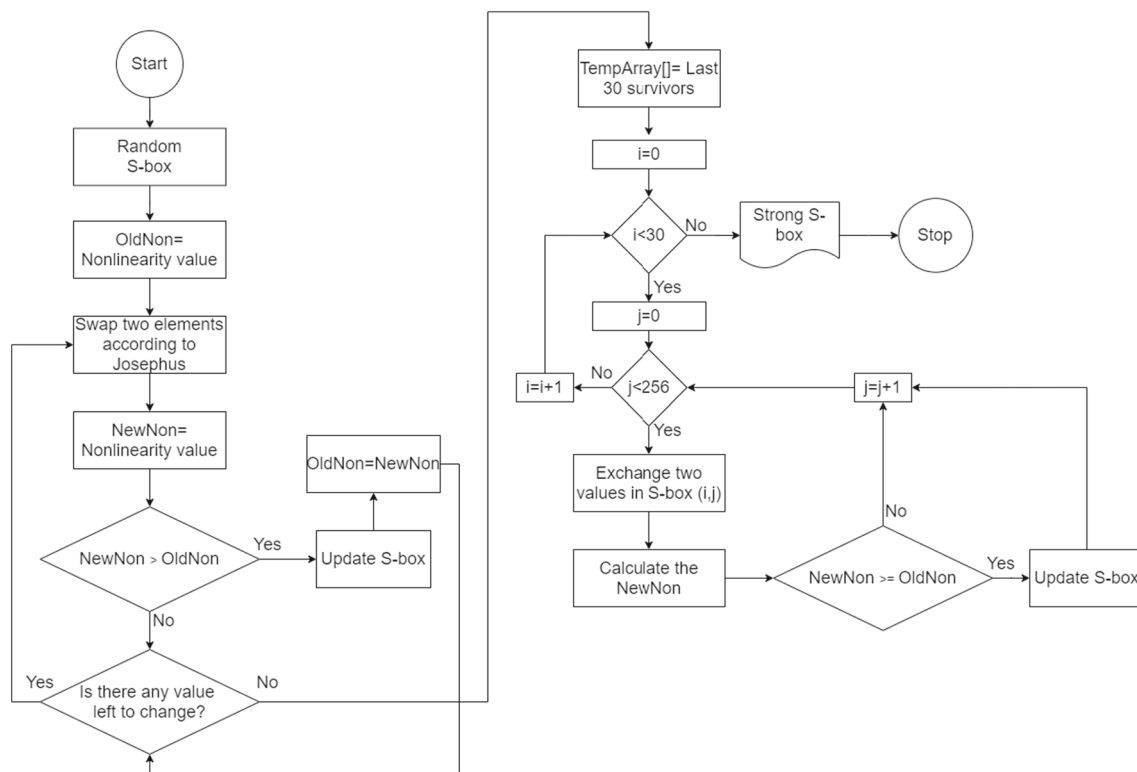


Fig. 2 Flow chart of the proposed method

2021; Zhao et al. 2023) are the highest scores obtained so far. Whether these values can be increased even higher may be a different source of motivation for researchers. However, most studies in the literature fall well below this value. In three different s-box structures obtained in this study, the nonlinearity values reached 110.5 in two and 110.75 in one. These values are acceptable. In other words, the proposed s-box structures also meet this criterion. In the input and output XOR values, which is another criterion, it is expected that the XOR values at the input and the XOR values at the output have the same probability (Biham and Shamir 1991). This makes the algorithm resistant to differential cryptanalysis. In the AES algorithm, the highest of this value is 4. This value may be slightly higher in the proposed algorithm and many algorithms in the literature. In the proposed s-box structures, this value was 10 in two and 12 in one. This is a criterion that needs improvement. However, the value of 10 has been an accepted value in the literature. Finally, the BIC value indicates that the bits should change independently of each other. In this value, both nonlinearity and SAC values are calculated. It is desired that the BIC-Nonlinearity value is as high as possible, and the BIC-SAC value is 0.5 or close to it. With the proposed algorithm, three different s-box structures that can be considered cryptographically strong were produced. These structures are given in Tables 2, 3, and 4, respectively.

When analyzing an s-box structure, the metrics given above are generally used (Si et al. 2023). However, fixed points and inverse fixed points are two very important concepts for an s-box. A fixed point refers to the situation where an input value matches its own value in an s-box. If expressed mathematically, it can be expressed as $S(a) = a$. On the other hand, inverse fixed point occurs when an input value matches the output value in an s-box. Mathematically, it can be expressed as $S(a) = b$, $S(b) = a$. Having fixed or inverted fixed points in an s-box may cause various vulnerabilities (Liu et al. 2020). Attackers can make many inferences here. For this purpose, after obtaining an s-box, it should be checked whether there is a fixed or inverse fixed point, and, if possible, this situation should be eliminated. The fixed points in the proposed s-box structures are as follows.

S-box1: 81, 82, 245.

S-box2: 24, 73.

S-box3: 66, 106.

There are some approaches in the literature to eliminate fixed and inverse fixed points (Liu et al. 2020, 2023; Si et al. 2021). However, in this study, these points were eliminated by using the philosophy in the proposed algorithm from a different perspective. Namely, after obtaining strong s-box structures with the proposed algorithm, fixed points are detected. Then, these fixed points are replaced by other elements, respectively. After the displacement

Table 2 Proposed sbox1

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

Table 3 Proposed sbox2

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

process, the nonlinearity value is checked. If this value remains equal, the s-box is updated. When this process is done sequentially for all fixed points, these points are quickly eliminated. In addition, this way the nonlinearity value will not decrease. With this philosophy, fixed points in the proposed s-box structures have been eliminated, and the final states of these s-box structures are given in Tables 5, 6, and 7 respectively.

In addition, the performance values of the proposed s-box structures in their initial state and after the fixed points are removed are given in Table 8. Thus, s-box structures are made stronger. Additionally, when Table 8 is

examined, it is seen that other criteria (BIC, SAC) also improved after the fixed points were eliminated.

The obtained s-box structures were compared with different s-box structures in the literature, especially according to the nonlinearity value. The performance comparison table is given in Table 9. In this table, there are various s-box generation approaches based on optimization, chaos, DNA–RNA coding–chain structures, mathematical transformations, and cellular automata. The proposed approach in Table 9 is compared with 48 studies. He left 31 of these studies behind and achieved the same value as one of them. This means that the proposed method can compete with current and future studies. Another contribution of the

Table 4 Proposed sbox3

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

Table 5 Proposed Final sbox1

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

study is the evaluation of the performance of different approaches. These different approaches have the potential to be a source of inspiration for researchers. In addition, to better understand the results given in Table 9, a performance graph was drawn according to nonlinearity values, and this graph is also given in Fig. 3.

5 Conclusions

Strong s-box structures are one of the main factors determining the strength of block cipher algorithms. Because they fulfill the need for confusion. These structures must be

nonlinear to provide strong confusion. In other words, they must have a high nonlinearity value. Obtaining an s-box with high nonlinearity is a difficult problem. In this study, a new algorithm is developed based on the Josephus problem to generate strong, high nonlinearity s-box structures. In the proposed algorithm, according to the logic of Josephus, each cell of the s-box is considered a soldier. Soldiers who were going to kill each other were thought of as cells that would swap places. In this way, when the nonlinearity value increases, the s-box structure is updated, and the process continues until the last cell survives. Three different s-boxes were obtained with the proposed algorithm. Two of them had a nonlinearity value of 110.5, and one had

Table 6 Proposed Final sbox2

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

Table 7 Proposed Final sbox3

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

Table 8 Performance of proposed s-box structures

S-box	XOR	BIC		SAC	Average nonlinearity
		Non	SAC		
S-box 1	10	102.86	0.5008	0.5107	110.5
S-box 2	10	103.14	0.5026	0.5037	110.5
S-box 3	12	104.57	0.4993	0.502	110.75
Final s-box 1	10	103	0.4998	0.51	110.5
Final s-box 2	10	103.07	0.5004	0.5029	110.5
Final s-box 3	12	104.21	0.5	0.5012	110.75

a value of 110.75. These values can be considered quite high. In addition, the proposed s-box structures also meet

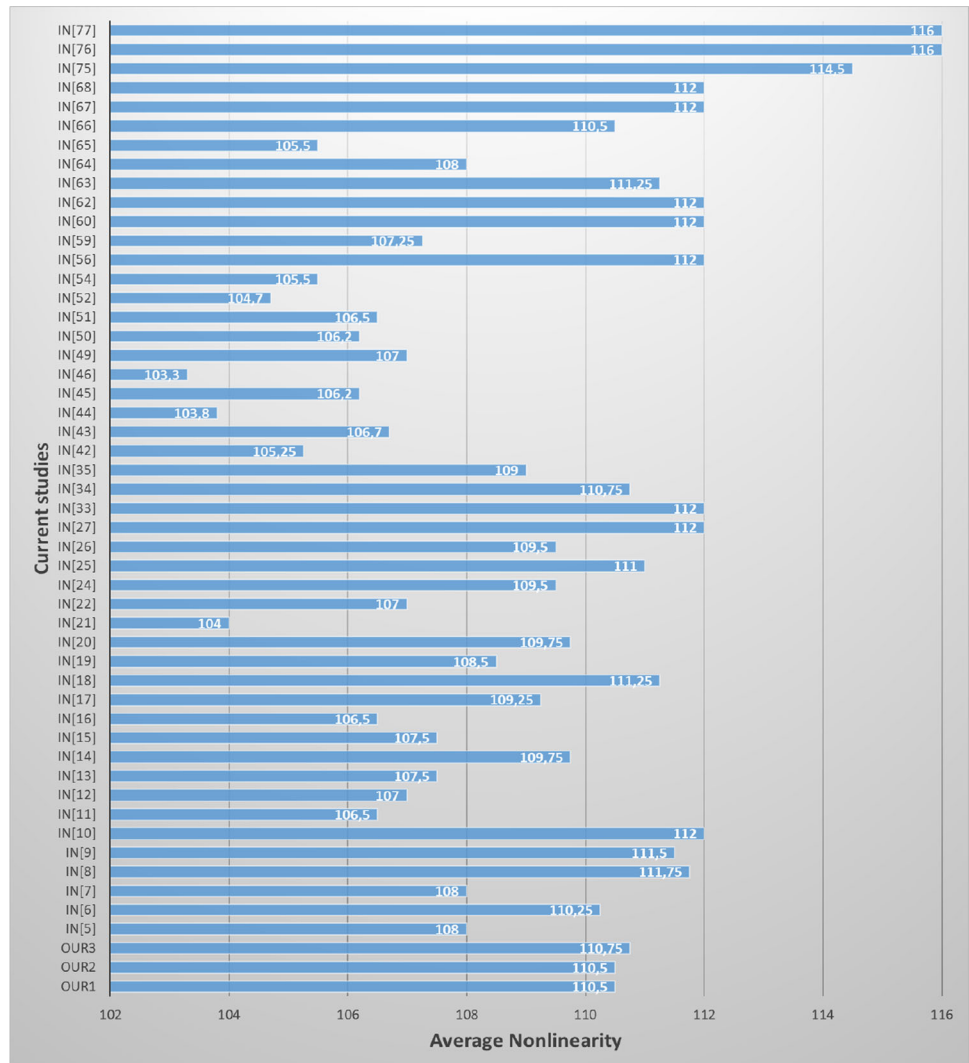
other cryptographic requirements. Obtaining similar successful results in three different s-box structures

Table 9 Performance values of different s-box generation approaches

S-box	XOR	BIC		SAC	Average nonlinearity
		Non	SAC		
Final S-box 1	10	103	0.4998	0.51	110.5
Final S-box 2	10	103.07	0.5004	0.5029	110.5
Final S-box 3	12	104.21	0.5	0.5012	110.75
In[5]	10	90	0.4950	0.5068	108
In[6]	10	104.07	0.5021	0.4953	110.25
In[7]	–	–	–	–	108
In[8]	12	104	0.5033	0.4968	111.75
In[9]	6	110.28	–	0.5022	111.5
In[10]	–	110	0.5134	0.5431	112
In[11]	10	105.85	0.4995	0.5036	106.5
In[12]	10	104.6	0.4974	0.496	107
In[13]	10	104.3	0.5001	0.4944	107.5
In[14]	10	104.14	0.5041	0.4998	109.75
In[15]	10	103.7	0.5025	0.5093	107.5
In[16]	10	105.2	0.4984	0.5120	106.5
In[17]	10	104.07	0.5005	0.5017	109.25
In[18]	10	102.57	0.5034	0.5007	111.25
In[19]	10	103.85	0.5011	0.4995	108.5
In[20]	10	104.35	0.5009	0.5068	109.75
In[21]	10	103.2	0.4971	0.4980	104
In[22]	10	105.5	0.5010	0.5015	107
In[24]	10	104.07	0.5020	0.4985	109.5
In[25]	–	113.35	0.4994	0.5	111
In[26]	10	104.21	0.5051	0.4980	109.5
In[27]	–	103.4	–	0.5012	112
In[33]	–	112	–	0.4998	112
In[34]	6	111.28	–	0.5012	110.75
In[35]	–	–	–	–	109
In[42]	10	102.6	0.4994	0.5037	105.25
In[43]	10	103.5	0.504	0.4976	106.7
In[44]	14	102.6	0.4958	0.5058	103.8
In[45]	10	102.3	0.5023	0.5039	106.2
In[46]	10	103.3	0.4995	0.4987	103.3
In[49]	10	104.5	0.5039	0.5007	107
In[50]	10	100	0.5288	0.501	106.2
In[51]	10	104.2	0.5003	0.4978	106.5
In[52]	10	103.3	0.4972	0.5034	104.7
In[53]	12	104.3	0.4988	0.5010	105.5
In[55]	4	112	0.504	0.4998	112
In[58]	6	107	–	0.501	107.25
In[59]	4	112	0.504	0.4998	112
In[61]	4	112	–	0.4985	112
In[62]	10	111.5	0.4950	0.5068	111.25
In[63]	4	–	–	–	108
In[64]	32	105.7	0.4994	0.4926	105.5
In[65]	8	–	–	0.5046	110.5
In[66]	4	112	0.4890	0.504	112

Table 9 (continued)

S-box	XOR	BIC		SAC	Average nonlinearity
		Non	SAC		
In[67]	4	112	—	0.4991	112
In[74]	—	—	—	—	114.5
In[75]	—	—	—	—	116
In[76]	—	104.86	0.5038	0.5020	116

Fig. 3 Performance comparison graph of different s-box generation approaches

demonstrates the effectiveness of the proposed method. In addition, fixed points were detected in the proposed s-box structures, and these were also eliminated. In this way, s-box structures have become even stronger. It also outperformed most studies when compared to other methods in the literature. This study can produce effective s-box structures for block cipher algorithms to be developed in the future.

Acknowledgements We thank the anonymous referees for their useful suggestions.

Author contributions FA was involved in the conceptualization, data curation, formal analysis, investigation, methodology and software, supervision visualization, original draft, writing—review and editing, funding acquisition, project administration and resources.

Funding Open access funding provided by the Scientific and Technological Research Council of Türkiye (TÜBİTAK).

Data availability Data sharing not applicable to this article as no datasets were generated or analysed during the current study.

Declarations

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

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Aboytes-González JA, Murguía JS, Mejía-Carlos M, González-Aguilar H, Ramírez-Torres MT (2018) Design of a strong S-box based on a matrix approach. *Nonlinear Dyn* 94(3):2003–2012
- Ahmad M, Al-Solami E (2020a) Evolving dynamic S-boxes using fractional-order hopfield neural network based scheme. *Entropy* 22(7):717
- Ahmad M, Al-Solami E (2020b) Improved 2D discrete hyperchaos mapping with complex behaviour and algebraic structure for strong S-boxes generation. *Complexity*. <https://doi.org/10.1155/2020/8868884>
- Ahmad M, Bhatia D, Hassan Y (2015) A novel ant colony optimization based scheme for substitution box design. *Procedia Comput Sci* 57:572–580
- Ahmad M, Khaja IA, Baz A, Alhakami H, Alhakami W (2020) Particle swarm optimization based highly nonlinear substitution-boxes generation for security applications. *IEEE Access* 8:116132–116147
- Ahmed HA, Zolkipli MF, Ahmad M (2019) A novel efficient substitution-box design based on firefly algorithm and discrete chaotic map. *Neural Comput Appl* 31(11):7201–7210
- Alexan W, ElBeltagy M, Aboshousha A (2022) Rgb image encryption through cellular automata, s-box and the lorenz system. *Symmetry* 14(3):443
- Alhadawi HS, Lambić D, Zolkipli MF, Ahmad M (2020) Globalized firefly algorithm and chaos for designing substitution box. *J Inf Secur Appl* 55:102671
- Alhadawi HS, Majid MA, Lambić D, Ahmad M (2021) A novel method of S-box design based on discrete chaotic maps and cuckoo search algorithm. *Multimed Tools Appl* 80(5):7333–7350
- Al-Wattar AS, Mahmood R, Zukarnain ZA, Udzir NI (2015a) Generating a new S-box inspired by biological DNA. *Int J Comput Sci Appl* 4(1):32–42
- Al-Wattar AH, Mahmood R, Zukarnain ZA, Udzir NI (2015b) A new DNA-based S-box. *Int J Eng Technol* 15(4):1–9
- Alzaidi AA, Ahmad M, Ahmed HS, Solami EA (2018) Sine-cosine optimization-based bijective substitution-boxes construction using enhanced dynamics of chaotic map. *Complexity* 2018:1–16
- Anees A, Chen YPP (2020) Designing secure substitution boxes based on permutation of symmetric group. *Neural Comput Appl* 32(11):7045–7056
- Arshad B, Siddiqui N, Hussain Z, Ehatisham-ul-Haq M (2022) A novel scheme for designing secure substitution boxes (S-boxes) based on Mobius Group and finite field. *Wirel Pers Commun* 124:1–22
- Artuğer F, Özkaynak F (2020) A novel method for performance improvement of chaos-based substitution boxes. *Symmetry* 12(4):571
- Artuğer F, Özkaynak F (2021) An effective method to improve nonlinearity value of substitution boxes based on random selection. *Inf Sci* 576:577–588
- Artuğer F, Özkaynak F (2022a) A method for generation of substitution box based on random selection. *Egypt Inform J* 23(1):127–135
- Artuğer F, Özkaynak F (2022b) SBOX-CGA: substitution box generator based on chaos and genetic algorithm. *Neural Comput Appl* 34(22):20203–20211
- Bakunina EV, Dykyi OV (2022) Synthesis method for S-boxes satisfying the criterion of correlation immunity of Boolean and 4-functions. *J Discrete Math Sci Cryptogr*. <https://doi.org/10.1080/09720529.2021.2018112>
- Basha HAMA, Mohra ASS, Diab TOM, El Sobky WI (2022) Efficient image encryption based on new substitution box using DNA coding and bent function. *IEEE Access* 10:66409–66429
- Belazi A, Abd El-Latif AA (2017) A simple yet efficient S-box method based on chaotic sine map. *Optik* 130:1438–1444
- Biham E, Shamir A (1991) Differential cryptanalysis of DES-like cryptosystems. *J Cryptol* 4(1):3–72
- Çavuşoğlu Ü, Zengin A, Pehlivan I, Kaçar S (2017) A novel approach for strong S-Box generation algorithm design based on chaotic scaled Zhongtang system. *Nonlinear Dyn* 87(2):1081–1094
- Chen G (2008) A novel heuristic method for obtaining S-boxes. *Chaos Solitons Fractals* 36(4):1028–1036
- Daemen J, Rijmen V (1998) AES proposal: Rijndael. In: *Proceedings of the 1st advanced encryption conference, CA, USA*, pp 1–45
- Dimitrov MM (2020) On the design of chaos-based S-boxes. *IEEE Access* 8:117173–117181
- Dimitrov MM (2021) A framework for fine-grained nonlinearity optimization of Boolean and vectorial Boolean functions. *IEEE Access* 9:124910–124920
- Farah T, Rhouma R, Belghith S (2017) A novel method for designing S-box based on chaotic map and teaching-learning-based optimization. *Nonlinear Dyn* 88(2):1059–1074
- Farhan AK, Ali RS, Yassein HR, Al-Saidi NMG, Abdul-Majeed GH (2020) A new approach to generate multi S-boxes based on RNA computing. *Int J Innov Comput Inf Control* 16(1):331–348
- Gangadari BR, Ahamed SR (2018) Programmable cellular automata-based low-power architecture to S-box: an application to WBAN. *Circuits Syst Signal Process* 37(3):1116–1133
- Haqae A, Abdulhussein TA, Ahmad M, Falah MW, Abd El-Latif AA (2022) A strong hybrid S-box scheme based on chaos, 2D cellular automata and algebraic structure. *IEEE Access* 10:116167–116181
- Hematpour N, Ahadpour S (2021) Execution examination of chaotic S-box dependent on improved PSO algorithm. *Neural Comput Appl* 33(10):5111–5133

- Hussain I (2020) True-chaotic substitution box based on Boolean functions. *Eur Phys J plus* 135(8):1–17
- Isa H, Syed Junid SAA, Z'aba MR, Endut R, Ammar SM, Ali N (2023) Enhancement of non-permutation binomial power functions to construct cryptographically strong S-boxes. *Mathematics* 11(2):446
- Kadhim FA, Majeed GHA, Ali RS (2016). Proposal new s-box depending on DNA computing and mathematical operations. In: 2016 Al-Sadeq international conference on multidisciplinary in IT and communication science and applications (AIC-MITCSA). IEEE, pp. 1–6
- Kang M, Wang M (2022) New genetic operators for developing S-boxes with low boomerang uniformity. *IEEE Access* 10:10898–10906
- Khan M, Shah T (2014) A novel image encryption technique based on Hénon chaotic map and S8 symmetric group. *Neural Comput Appl* 25(7):1717–1722
- Khan M, Shah T (2015) An efficient construction of substitution box with fractional chaotic system. *Signal Image Video Process* 9(6):1335–1338
- Khan M, Shah T, Mahmood H, Gondal MA, Hussain I (2012) A novel technique for the construction of strong S-boxes based on chaotic Lorenz systems. *Nonlinear Dyn* 70(3):2303–2311
- Khan LS, Hazzazi MM, Khan M, Jamal SS (2021) A novel image encryption based on Rossler map diffusion and particle swarm optimization generated highly non-linear substitution boxes. *Chin J Phys* 72:558–574
- Khan H, Hazzazi MM, Jamal SS, Hussain I, Khan M (2023) New color image encryption technique based on three-dimensional logistic map and Grey wolf optimization based generated substitution boxes. *Multimed Tools Appl* 82(5):6943–6964
- Kuznetsov A, Romeo L, Poluyanenko N, Kandy S, Kuznetsova K (2022) Optimizing Hill climbing algorithm parameters for generation of cryptographically strong S-boxes. *Electronics* 12(10):2338
- Lambić D (2018) S-box design method based on improved one-dimensional discrete chaotic map. *J Inf Telecommun* 2(2):181–191
- Lambić D (2020) A new discrete-space chaotic map based on the multiplication of integer numbers and its application in S-box design. *Nonlinear Dyn* 100(1):699–711
- Liu H, Kadir A, Xu C (2020) Cryptanalysis and constructing S-box based on chaotic map and backtracking. *Appl Math Comput* 376:125153
- Liu H, Liu J, Ma C (2023) Constructing dynamic strong S-box using 3D chaotic map and application to image encryption. *Multimed Tools Appl* 82(16):23899–23914
- Malik MSM, Ali MA, Khan MA, Ehatisham-Ul-Haq M, Shah SNM, Rehman M, Ahmad W (2020) Generation of highly nonlinear and dynamic AES substitution-boxes (S-boxes) using chaos-based rotational matrices. *IEEE Access* 8:35682–35695
- Mariot L, Picek S, Leporati A, Jakobovic D (2019) Cellular automata based S-boxes. *Cryptogr Commun* 11(1):41–62
- Mohamed AG, Korany NO, El-Khamy SE (2021) New DNA coded fuzzy based (DNAFZ) S-boxes: application to robust image encryption using hyper chaotic maps. *IEEE Access* 9:14284–14305
- Örs SB, Preneel B, Verbaauwhede I (2007) Side-channel analysis attacks on hardware implementations of cryptographic algorithms. In: *Wireless security and cryptography-specifications and implementations*, pp 213–247
- Özkaynak F (2020a) On the effect of chaotic system in performance characteristics of chaos based s-box designs. *Phys A Stat Mech Appl* 550:124072
- Özkaynak F (2020b) An analysis and generation toolbox for chaotic substitution boxes: a case study based on chaotic labyrinth Rene Thomas system. *Iran J Sci Technol Trans Electr Eng* 44(1):89–98
- Özkaynak F, Özer AB (2010) A method for designing strong S-Boxes based on chaotic Lorenz system. *Phys Lett A* 374(36):3733–3738
- Özkaynak F, Çelik V, Özer AB (2017) A new S-box construction method based on the fractional-order chaotic Chen system. *Signal Image Video Process* 11(4):659–664
- Razaq A, Ullah A, Alolaiyan H, Yousaf A (2021) A novel group theoretic and graphical approach for designing cryptographically strong nonlinear components of block ciphers. *Wirel Pers Commun* 116(4):3165–3190
- Razaq A, Akhter S, Yousaf A, Shuaib U, Ahmad M (2022a) A group theoretic construction of highly nonlinear substitution box and its applications in image encryption. *Multimed Tools Appl* 81(3):4163–4184
- Razaq A, Ahmad M, Yousaf A, Alawida M, Ullah A, Shuaib U (2022b) A group theoretic construction of large number of AES-like substitution-boxes. *Wirel Pers Commun* 122(3):2057–2080
- Si Y, Liu H, Chen Y (2021) Constructing keyed strong S-Box using an enhanced quadratic map. *Int J Bifurc Chaos* 31(10):2150146
- Si Y, Liu H, Zhao M (2023) Constructing keyed strong S-box with higher nonlinearity based on 2D hyper chaotic map and algebraic operation. *Integration* 88:269–277
- Siddiqui N, Yousaf F, Murtaza F, Ehatisham-ul-Haq M, Ashraf MU, Alghamdi AM, Alfakeeh AS (2020) A highly nonlinear substitution-box (S-box) design using action of modular group on a projective line over a finite field. *PLoS ONE* 15(11):e0241890
- Sokolov AV, Radush VV (2022) A method for synthesis of S-boxes with good avalanche characteristics of component Boolean and quaternary functions. *J Discrete Math Sci Cryptogr*. <https://doi.org/10.1080/09720529.2021.1964727>
- Standard DE (1999) Data encryption standard. In: *Federal Information Processing Standards Publication*, 112
- Szaban M, Seredynski F (2008) Application of cellular automata to create S-box functions. In: 2008 IEEE international symposium on parallel and distributed processing. IEEE, pp 1–7
- Tang G, Liao X (2005) A method for designing dynamical S-boxes based on discretized chaotic map. *Chaos Solitons Fractals* 23(5):1901–1909
- Tang G, Liao X, Chen Y (2005) A novel method for designing S-boxes based on chaotic maps. *Chaos Solitons Fractals* 23(2):413–419
- Tian Y, Lu Z (2017) Chaotic S-box: intertwining logistic map and bacterial foraging optimization. *Math Probl Eng* 2017:6969312
- Wang Y, Wong KW, Li C, Li Y (2012) A novel method to design S-box based on chaotic map and genetic algorithm. *Phys Lett A* 376(6–7):827–833
- Wang Y, Zhang Z, Zhang LY, Feng J, Gao J, Lei P (2020) A genetic algorithm for constructing bijective substitution boxes with high nonlinearity. *Inf Sci* 523:152–166
- Webster AF, Tavares SE (1985) On the design of S-boxes. In: *Conference on the theory and application of cryptographic techniques*. Springer, Berlin, pp 523–534
- Yang S, Tong X, Wang Z, Zhang M (2023) S-box generation algorithm based on hyperchaotic system and its application in image encryption. *Multimed Tools Appl* 82:1–25
- Ye T, Zhimao L (2018) Chaotic S-box: six-dimensional fractional Lorenz–Duffing chaotic system and O-shaped path scrambling. *Nonlinear Dyn* 94(3):2115–2126
- Zamli KZ (2021) Optimizing S-box generation based on the adaptive agent Heroes and Cowards algorithm. *Expert Syst Appl* 182:115305
- Zamli KZ, Kader A, Din F, Alhadawi HS (2021) Selective chaotic maps Tiki-Taka algorithm for the S-box generation and optimization. *Neural Comput Appl* 33:1–18

- Zamli KZ, Din F, Alhadawi HS, Khalid S, Alsolai H, Nour MK, Assam M (2022) Exploiting an Elitist Barnacles Mating Optimizer implementation for substitution box optimization. *ICT Express* 9(4):619
- Zamli KZ, Din F, Alhadawi HS (2023) Exploring a Q-learning-based chaotic naked mole rat algorithm for S-box construction and optimization. *Neural Comput Appl* 35:1–23
- Zhao M, Liu H, Niu Y (2023) Batch generating keyed strong S-Boxes with high nonlinearity using 2D hyper chaotic map. *Integration* 92:91–98

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