Bilecik Şeyh Edebali Üniversitesi Fen Bilimleri Dergisi
Bilecik Seyh Edebali University Journal of Science
11(2), 236-243, 2024
DOİ: 10.35193/bseufbd.1323521
e-ISSN: 2458-7575 (https://dergipark.org.tr/tr/pub/bseufbd)

Araştırma Makalesi - Research Article

İkame Kutularının Lineer Olmama Değerini Optimize Etme

Optimizing Nonlinearity Value of Substitution Boxes

Fırat Artuğer[1*]

Geliş / Received: 06/07/2023 ***Revize / Revised: 14/09/2023*** ***Kabul / Accepted: 16/10/2023***

ÖZ

Şifreleme algoritmalarında en önemli gereksinimlerden bir tanesi karıştırma olarak adlandırılmaktadır. Bu nedenle şifrelenecek verinin etkin bir şekilde karıştırılması gerekmektedir. İkame kutusu (s-box), bu gereksinimi sağlayan en önemli yapılardan bir tanesidir. Bu yapının en önemli özelliklerinden biri olan lineer olmama değeri ne kadar yüksek olursa karıştırmayı o kadar sağlıklı bir şekilde yerine getirecektir. İkame kutularının elde edilmesinde birçok teknik kullanılmaktadır. Bu tekniklerden en çok kullanılan, optimizasyon tekniğidir. Bu teknikte, başlangıçta genellikle kaos yardımıyla bir s-box elde edilir. Daha sonra bir optimizasyon tekniği kullanılarak elemanların konumları değiştirilir. Uygunluk değeri olarak lineer olmama kriteri kullanılır. Yeni konumlandırmalardan sonra lineer olmama değeri arttığında s-box yapısı güncellenmektedir. Bu çalışmada öncelikle s-box yapılarında lineer olmama değerinin nasıl optimize edildiği açıklanmıştır. Daha sonra sinüs kosinüs algoritması kullanılarak bir s-box optimize edilmiştir. Elde edilen s-box yapısının, 500 iterasyon sonunda lineer olmama değeri 108 olarak gözlemlenmiştir. Ayrıca bir s-box yapısının diğer performans kriterleri de açıklanmıştır.

Anahtar Kelimeler- İkame kutusu (s-box), Kaos, Lineer olmama, Optimizasyon

ABSTRACT

One of the most important requirements in encryption algorithms is called confusion. For this reason, the data to be encrypted must be effectively mixed. Substitution box (s-box) is one of the most important structures that meet this requirement. The higher the nonlinearity value, which is one of the most important features of this structure, the healthier the mixing will be. Many techniques are used to obtain substitution boxes. It is the most used optimization technique among these techniques. In this technique, an s-box is obtained initially, usually with the help of chaos. The positions of the elements are then changed using an optimization technique. The nonlinearity criterion is used as the fitness value. When the nonlinearity value increases after new positioning, the s-box structure is updated. In this study, first, explains how the nonlinearity value is optimized in s-box structures. Then an s-box is optimized using the sine cosine algorithm. The nonlinearity value of the obtained s-box structure was observed to be 108 at the end of 500 iterations. Other performance criteria of an s-box structure are also described.

Keywords- Substitution box (s-box), Chaos, Nonlinearity, Optimization

[1]*Sorumlu yazar iletişim: firatartuger@munzur.edu.tr (https://orcid.org/0000-0002-4096-0458
Bilgisayar Mühendisliği Bölümü, Munzur Üniversitesi, Mühendislik Fakültesi, Tunceli, Türkiye

I. GİRİŞ

Hayatımızın neredeyse tamamen dijitalleştiği günümüzde veri en değerli varlıktır. Verilerin hem depolanması hem de iletilmesi anında en önemli sorunlardan bir tanesi güvenliktir. Günümüzde verilerin güvenliğini sağlayan en temel teknik simetrik şifreleme algoritmalarıdır. Simetrik şifreleme algoritmaları, blok ve akış olmak iki üzere iki temel tekniğe dayanmaktadır. Akış şifrelemede bitler tek tek şifrelenir. Bu algoritmalar hızlı ve güvenlidir. Ancak verinin boyutu arttıkça uygulaması neredeyse imkânsız hale gelmektedir. Bir diğer teknik ise blok şifrelemedir. Günümüz blok şifreleme standardı AES algoritmasıdır ve 8 bit güçlü bir s-box yapısı kullanılmaktadır [1]. Blok şifrelemede veri bloklara bölünür ve her blok kendi içinde şifrelenir. Daha sonra şifrelenmiş bloklar birleştirilerek şifreli veri elde edilir. Blok şifreleme algoritmalarının güvenliği sağlamasındaki en temel yapı taşlarından bir tanesi s-box yapılarıdır. Çünkü s-box, blok şifreleme algoritmasının genellikle lineer olmayan tek birimidir. Lineer olmama değeri ne kadar yüksek olursa s-box o kadar güçlü olacaktır. 8 bit bir s-box 256 değer içermektedir. Buda güçlü bir s-box aramak için arama uzayının 256! olduğu anlamına gelmektedir. Bu geniş arama uzayında güçlü bir s-box aramak oldukça zor bir problemdir. Bu tarz problemlerin çözümü için optimizasyon teknikleri sıklıkla kullanılmaktadır.

Bir s-box matematiksel olarak denklem 1 ‘de verildiği gibi ifade edilebilir. Yani elemanların başka elemanlarla yer değiştirdiği lineer olmayan bir haritalama işlemidir [2]. Güçlü s-box yapıları geliştirmek için literatürde birçok yaklaşım bulunmaktadır. Bunlardan ilki kaos tabanlı yaklaşımlardır. Kaos tabanlı yaklaşımlar genellikle kaotik haritalara dayanmaktadır [3-8]. Kaos tabanlı yöntemler rastgele oluşturuldukları için güçlüdür. Ancak bu yöntemler ile geliştirilen s-box yapılarının genellikle lineer olmama değeri düşüktür. Bu değerin düşük olması şifreleme algoritmasını diferansiyel saldırılara karşı dirençsiz hale getirmektedir. Buda bu yöntemlerin en önemli dezavantajıdır. Ancak son yıllarda bu dezavantajı gidermek için çeşitli yaklaşımlar önerilmiştir [9, 10]. Bu yaklaşımlar sayesinde kaotik s-box yapılarını performansı iyileştirilmiştir. Bir diğer yaklaşım ise matematiksel dönüşümlerdir [11-13]. Matematiksel dönüşümler ile geliştirilen s-box yapıları genellikle istatistiksel olarak oldukça güçlüdürler ve lineer olmama değerleri yüksektir. Ancak, pratik ve cebirsel olarak yapılabilecek şifre çözümleme saldırılarına karşı çeşitli zayıflıkları mevcuttur. Bunlardan farklı olarak hücresel otomata modelleri [14,15], DNA-RNA zincir yapıları [16-18] kullanılarak s-box yapıları oluşturulmuştur. Son olarak s-box yapılarında lineer olmama değerini arttırmak için optimizasyon yöntemleri sıklıkla kullanılmıştır. Bu teknikler ile s-box yapılarının lineer olmama değeri arttırılabilmektedir. Ancak bu algoritmaların hesaplama karmaşıklığı yüksektir. Buda bu yöntemlerin dezavantajı olarak görülebilir.

Şu ana kadar literatürde yapılan optimizasyon tabanlı s-box geliştirme yöntemlerine kısaca değinilmiştir. Farah ve arkadaşlarının yaptığı çalışmada öğretme öğrenme tabanlı optimizasyon algoritması kullanılarak yeni bir s-box üreteci önerilmiştir. Bu yöntem, toplam sekiz turdan meydana gelmektedir. Her tur iki dönüşümden oluşur. Bunlardan birincisi satır sola kaydırma ve diğeri ise sütun yönünde döndürme işlemidir. Bu operasyonlar sayesinde lineer olmama değeri 106.5 olan bir s-box geliştirilmiştir [19]. Ahmed ve arkadaşları tarafından ateş böceği optimizasyon algoritmasını kullanan yeni bir yöntem önerilmiştir. Bu yöntemde de başlangıç s-box ‘ı için kaotik harita kullanılmıştır. Bu yaklaşımda, uygunluk fonksiyonu yardımıyla rehberli arama yapılarak lineer olmama değeri 107.5 olan bir s-box üretilmiştir [20]. Zamli, güvenlik uygulamalarında kullanılabilecek etkili bir s-box için uyarlanabilir ajan kahramanlar ve korkaklar algoritması olarak tanımlanan yeni bir metasezgisel algoritma geliştirmiştir. Bu algoritma ajan kahramanlar ve korkaklar algoritmasına dayanmaktadır. Başlangıç popülasyonu için ise literatürde sıklıkla kullanılan kaotik haritalardan biri olan çadır haritası kullanılmıştır. Bu yeni metasezgisel algoritma sayesinde lineer olmama değeri 109.75 olan güçlü bir s-box elde edilmiştir [21]. Alhadawi ve arkadaşları tarafından guguk kuşu arama algoritmasına dayanan yeni bir s-box üreteci önerilmiştir. Başlangıç popülasyonu ayrık zamanlı kaotik haritalara dayanmaktadır. Bu yöntemin temel avantajı, Genetik algoritma ve Parçacık sürü optimizasyonu algoritmalarına kıyasla verimli rastgelelik ve daha düşük ayarlanabilir parametreler yardımıyla gösterilmesidir. Bu yaklaşım sayesinde lineer olmama değeri 108.5 olan bir s-box elde edilmiştir [22]. Wang ve arkadaşları tarafından etkili bir s-box geliştirmek için genetik algoritmaya dayanan yeni bir algoritma geliştirilmiştir. Bu yöntemde boole fonksiyonu s-box ‘ın kromozomu olarak alınır. Başlangıç popülasyonunu için yine kaotik yaklaşımlar temel alınmıştır. Genetik algoritmanın etkili çaprazlama ve mutasyon birimleri sayesinde lineer olmama değeri 110.25 ‘e kadar arttırılmıştır [23]. Artuğer ve Özkaynak tarafından yapılan çalışmada genetik algoritmanın çaprazlama ve mutasyon birimleri farklı bir şekilde kullanılarak lineer olmama değerini 111.75 ‘e kadar çıkarmıştır [24]. Alhadawi ve arkadaşları tarafından ateş böceği optimizasyon algoritması bir kez daha s-box üretiminde kullanılmıştır. Başlangıç popülasyonu için kaotik haritalardan yararlanılmıştır. Bu yöntemde optimizasyon, sabit bir uygunluk fonksiyonunu azaltarak iyi çözümleri hızlı bir şekilde aramaktadır. Bu algoritma sayesinde lineer olmama değeri 107 olan bir s-box üretilmiştir [25]. Ahmad ve Al-Solami ‘nin yaptığı çalışmada kesirli sıralı zaman gecikmeli hopfield sinir ağına dayanan yeni bir algoritma önerilmiştir. Algoritmanın evrimi lineer olmama değerini arttırmak üzerine kurulmuştur. Bu yöntemde lineer olmama değeri 111.25 olarak hesaplanmıştır [26]. Wang ve arkadaşları tarafından, s-box elde etme problemi bir gezgin satıcı problemine uyarlanmıştır. Optimizasyon için genetik algoritma, başlangıç popülasyonu için ise kaos kullanılmıştır. Bu

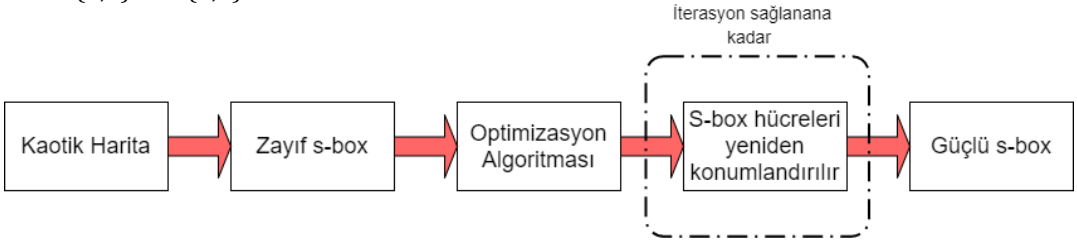
çalışmada genetik algoritmanın evrim süreci sayesinde lineer olmama değeri 108 olan bir s-box üretilmiştir [27]. Ahmad ve arkadaşları tarafından en popüler metasezgisel algoritmalardan biri olan karınca kolonisi optimizasyon algoritmasına dayanan yeni bir yöntem önerilmiştir. Başlangıç s-box 'ı için yine kaos kullanılmıştır. Bu yöntemde, s-box elde etme problemi yine bir gezgin satıcı problemine dönüştürülerek arama gerçekleştirilmiştir. Bu sayede lineer olmama değeri 107 olan bir s-box elde edilmiştir [28]. Chen tarafından tavlama metodu kullanılarak yeni bir s-box üreteci geliştirilmiştir. Bu yöntemde yine kaotik haritalara dayanmaktadır. Bu sayede lineer olmama değeri 104 olan bir s-box üretilmiştir [29]. Khan ve arkadaşları tarafından güvenli bir şifreleme algoritması geliştirmek için güçlü s-box 'lar kullanmayı benimseyen yeni bir yöntem önerilmiştir. Bu yöntemde s-box üretiminde parçacık sürü optimizasyonu kullanılmaktadır. Başlangıç popülasyonu rastgele üretilmektedir. Buradaki temel felsefe parçacık konumlarının s-box üretiminde kullanılmasıdır. Bu yöntem ile 112 lineer olmama değerine ulaşılmıştır [30]. Hematpour ve Ahadpour tarafından parçacık sürü optimizasyonunu kullanılarak yeni bir s-box üreteci geliştirilmiştir. Lineer olmama değeri 106.5 olan bir s-box elde edilmiştir [31]. Zamli ve arkadaşları tarafından, futboldan esinlenilerek geliştirilmiş tiki-taka optimizasyon algoritmasına dayanan yeni bir algoritma geliştirilmiştir. Bu yöntemde başlangıç popülasyonu için 5 farklı kaotik haritadan bir tanesi seçilebilmektedir. Buda algoritmayı çeşitlendirmektedir. Bu algoritma ile 109.25 lineer olmama değerine sahip olan bir s-box üretilmiştir [32]. Tian ve Lu tarafından, bir diğer önemli optimizasyon algoritmalarından biri olan bakteriyel yiyecek arama optimizasyonuna dayanan yeni bir s-box üreteci önerilmiştir. Başlangıç popülasyonu için iç içe geçmiş lojistik harita temel alınmıştır. Bu algoritma optimizasyon aşamasında uygunluk fonksiyonları olarak doğrusal olmayan ve diferansiyel tekdüzeliği kullanması sayesinde lineer olmama değeri 107.5 olan bir s-box elde etmiştir [33]. Alzaidi ve arkadaşları tarafından, son yıllarda geliştirilen güçlü optimizasyon algoritmalarından biri olan sinüs kosinüs algoritmasına dayalı yeni bir yöntem önerilmiştir. Bu yöntemde başlangıç popülasyonu için yeni bir kaotik harita kullanılmıştır. Bu algoritmanın güçlü kaotik dinamikleri sayesinde lineer olmama değeri 109.5 olan bir s-box üretilmiştir [34]. Ahmad ve arkadaşları tarafından güçlü s-box yapıları üretmek için parçacık sürüsü optimizasyonu kullanılmıştır. Bu yöntemde başlangıç popülasyonu için zengin dinamiklere sahip olan Renyi haritası kullanılmıştır. Bu yöntemde lineer olmama değeri 111.5 'e kadar arttırılmıştır [35]. Kang ve Wang 'ın yaptığı çalışmada genetik operasyonlar kullanılarak lineer olmama değeri 108 olarak elde edilmiştir [36]. Zamli ve arkadaşlarının yaptığı çalışmada çıplak köstebek faresi algoritması kullanılarak 109.75 lineer olmama değerine ulaşılmıştır [37].

Çalışmanın ikinci bölümünde ikame kutularının nasıl optimize edildiği açıklanmış olup, sinüs kosinüs algoritması kullanılarak yeni bir s-box üreteci oluşturulmuştur. Üçüncü bölümde elde edilen s-box yapısı analiz edilmiş olup, literatürdeki diğer çalışmalarla performans karşılaştırması yapılmıştır. Dördüncü bölümde ise sonuçlar tartışılmış olup önerilerde bulunulmuştur.

II. İKAME KUTULARININ OPTİMİZASYONU

İkame kutularında en önemli kriter lineer olmama değeridir. Bu yüzden optimizasyon teknikleri uygulandığında genellikle uygunluk değeri olarak lineer olmama kullanılır. Bu yöntemlerin uygulanmasında öncelikle kaotik bir harita veya farklı bir teknikle bir s-box elde edilir. Bu s-box yapısının elde edilmesi oldukça kolaydır. Kaotik haritanın başlangıç parametreleri belirlenir ve bu harita ile bir değer üretilir. Daha sonra bu değer bir tam sayıya dönüştürülür ve [0-256] aralığında olabilmesi için 256 değerine göre mod işlemi uygulanır. Son olarak elde edilen bu değer ikame kutusunda yoksa eklenir. Varsa yeni bir değer üretilerek süreç devam eder. Bu şekilde 256 hücre dolana kadar bu işleme devam edilir. Bu şekilde bir s-box yapısı elde edilmiş olunur. Ancak bu yapı kriptografik olarak zayıf özellikler gösterecektir. Özellikle lineer olmama değeri 107 'nin altında kalacaktır. AES algoritmasında bu değer 112 'dir. 112 değerine ulaşılamasa bile buna yakın değerler elde edilmesi istenmektedir. Optimizasyon burada devreye girmektedir. Optimizasyon algoritmasının burada kullanımı çok çeşitli olsa bile, genellikle bu algoritmalardan bir tanesiyle iki veya daha fazla hücre belirlenir ve konumları değiştirilir ya da örneğin genetik algoritmada çaprazlama sürecinde satırlar veya sütunlar yer değiştirilip lineer olmama değeri arttırılabilmektedir. Bu şekilde lineer olmama değeri arttığında s-box güncellenir. Belli bir iterasyona kadar bu süreç uygulandığında lineer olmama değerinin oldukça arttığı görülmektedir. İkame kutularının optimizasyon süreci için temel sistem modeli şekil 1 'de verilmiştir.

$$S: \{0,1\}^m \rightarrow \{0,1\}^n \quad (1)$$



Şekil 1. S-box yapılarının optimizasyonu için sistem modeli

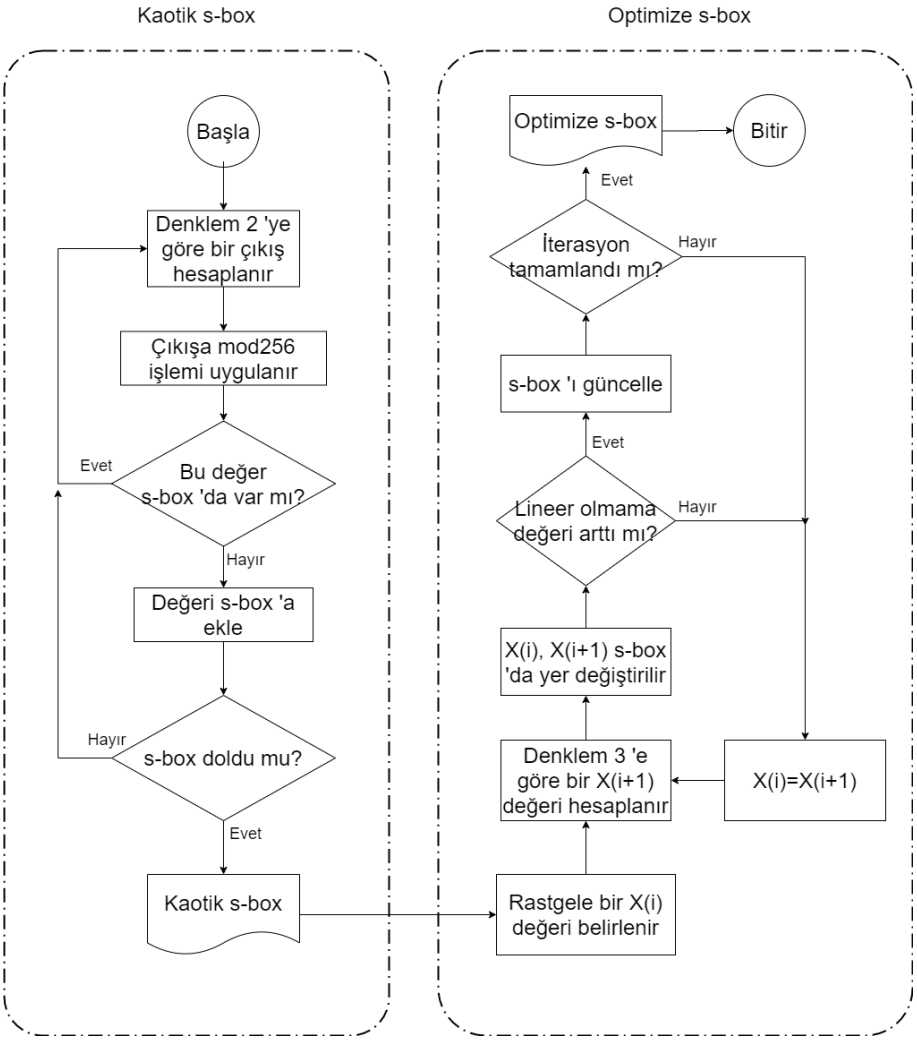
Bu çalışmada sinüs kosinüs algoritması kullanılarak bir s-box yapısı optimize edilmiştir. Başlangıç için kaotik çadır harita kullanılmıştır. Çadır haritanın matematiksel modeli denklem 2 'de verilmiştir.

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

Sinüs kosinüs algoritması, optimizasyon problemlerini çözmek için matematiksel sinüs ve kosinüs fonksiyonlarının döngüsel modellerini kullanan popülasyon tabanlı etkili bir optimizasyon algoritmasıdır. Yakın zamanda Seyedali Mirjalili tarafından önerilmiş ve çoğu optimizasyon algoritmasından çok daha hızlı bir yakınsamaya sahip olduğu kanıtlanmıştır [38]. X_i^t 'nin konumunu r_4 parametresine göre güncellemek için denklem 3 kullanır. Denklem 3 'de verilen X_i^{t+1} değeri s-box yapısında X_i^t ile konum değiştirecek olan değer olacaktır.

$$X_i^{t+1} = \begin{cases} X_i^t + r_1 \, x \, sin(r_2) \, x \, |r_3 \, x \, P_i^t - X_i^t|, & r_4 < 0.5 \,) \\ X_i^t + r_1 \, x \, cos(r_2) \, x \, |r_3 \, x \, P_i^t - X_i^t|, & r_4 \geq 0.5 \,) \end{cases} \tag{3}$$

$Burada\ X_i^t$ i. boyutunun t. İterasyonundaki güncel çözümüdür. r_1, r_2, r_3, r_4 rastgele sayılardır. Özellikle r_4, 0 ile 1 aralığında rastgele bir sayıdır. P_i i. boyuttaki hedef noktanın pozisyonudur. Yani başlangıçta rastgele bir X_i^t değeri seçilir. Daha sonra bu değere göre bir X_i^{t+1} değeri elde edilerek bunların konumları değiştirilir. Bu işlemden sonra lineer olmama değeri arttığında s-box güncellenir. X_i^t değeri bir sonraki adımda X_i^{t+1} değeri yapılarak belli bir iterasyona kadar algoritma devam ettirilir. Kaotik bir s-box yapısını üretip lineer olmama değerini optimize etmek için önerilen algoritmanın akış diyagramı şekil 2 'de verilmiştir. Önerilen algoritma belirlenen iterasyon sayısına göre tek döngüde çalıştığı için O(n) karmaşıklığına sahiptir.



Şekil 2. Önerilen algoritmanın akış diyagramı

III. ANALİZ SONUÇLARI

Bir s-box yapısının performansını analiz etmek için literatürde var olan farklı yaklaşımlar mevcuttur. Bunlar; Bijektiflik, Katı Çığ Kriterleri (SAC), Lineer olmama, Bit Bağımsızlık Kriteri (BIC) ve giriş/çıkış XOR dağılımı kriterleridir. Bijektiflik, birebir ve örten bir fonksiyon anlamına gelmektedir. Yani 256 değer içeren s-box yapısında, 0 ile 256 arasındaki değerlerinden her birinin sadece bir kere kullanılması gerekmektedir. Bu çalışmada önerilen s-box yapısı ve literatürdeki diğer çoğu çalışma bu kriteri sağlamaktadır. Daha etkin bir karıştırma işlemi için s-box yapılarının bijektif olması önerilmektedir. SAC kriteri, girdi verisinde bir bit değiştiğinde, çıktı verisinde meydana gelecek olan değişikliği ölçmektedir [39]. Bu değişikliğin 0.5 veya buna yakın bir değer olması istenir. Yani girdide bir bit değiştiğinde çıktıda bitlerin yarısına yakınının değişmesi beklenmektedir. Bu sayede saldırganların bitleri değiştirerek yapabilecekleri herhangi bir saldırının önüne geçilmiş olunur. Lineer olmama kriteri, affine dönüşümünde boole işlevinin bit sayısındaki değişikliğini ifade eder. Bu kriter bir s-box yapısındaki en önemli değer olarak görülmektedir. Bu değerin olabildiğince yüksek olması istenir. Günümüz standardı olan AES algoritmasında bu değer 112 ‘dir. BIC kriteri, Webster ve Tavares tarafından önerilmiş olup, i ve j olarak tanımlanan iki bitin bağımsız bir şekilde değişmesi gerektiği ve herhangi birinin diğerinden çıkarılmasının mümkün olmadığı bir olayın meydana gelişinin bağımsızlığına dayanmaktadır [39]. BIC kriterinde hem lineer olmama değerinin olabildiğince yüksek olması, hem de SAC değerinin 05 ‘e yakın olması beklenmektedir. XOR dağılım kriteri Biham ve Shamir tarafından sunulmuştur [40]. Girişlerdeki XOR değerleri ile çıkışta meydana gelen XOR değerleri aynı olasılığa sahip olmalıdır. Yani, bir s-box ‘ın giriş ve çıkış olasılık dağılımına olabildiğince izin verilmediğinde, bu yapı diferansiyel kriptanalize karşı dirençli hale gelebilmektedir.

Bu çalışmada öncelikle kaotik çadır harita kullanılarak tablo 1 ‘de verilen s-box yapısı elde edilmiştir. Daha sonra sinüs kosinüs algoritması ile 500 iterasyonda optimize edilmiş s-box yapısı tablo 2 ‘de verilmiştir. Lineer olmama değeri 102.75 ‘den 108 ‘e çıkmıştır. Bu değer literatürdeki birçok optimizasyon tabanlı s-box yapısını geride bırakmaktadır. Ayrıca literatürdeki diğer optimizasyon tabanlı s-box geliştirme yöntemlerinin performans değerleri de incelenmiştir. Bu değerler tablo 3 ‘de verilmektedir. Bu tabloda görüldüğü gibi birçok optimizasyon tekniği kullanılmıştır. Bu çalışmanın temel amacı lineer olmama değerini optimize etmek olduğu için çalışmalar bu değere göre kıyaslanmıştır.

Tablo 1. Çadır harita ile elde edilen s-box

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

Tablo 2. Tablo 1 ‘de verilen s-box yapısının optimize edilmiş hali

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**A**	**B**	**C**	**D**	**E**	**F**
0	18	175	83	81	247	196	130	111	65	137	103	191	52	132	105	152
1	50	45	209	51	142	117	226	96	78	92	55	13	46	84	159	80
2	236	188	228	16	144	14	118	141	59	114	32	178	97	169	66	244
3	181	148	48	172	102	100	121	164	47	217	154	71	86	158	95	136
4	238	205	239	180	156	12	87	157	218	250	230	7	219	143	220	231
5	6	170	21	147	58	120	210	153	69	39	22	116	24	109	168	34

6	101	166	249	211	193	93	75	146	225	0	44	1	177	150	253	165
7	68	140	187	113	245	190	233	108	195	135	199	125	56	173	110	27
8	36	162	73	10	115	184	31	122	185	9	208	251	123	19	227	207
9	53	29	4	119	194	134	106	139	242	204	223	126	11	192	163	213
A	174	99	43	72	167	3	202	63	176	252	203	128	38	240	186	85
B	40	2	212	112	35	5	235	255	155	221	28	94	216	133	189	88
C	241	232	61	161	182	179	160	41	23	171	145	201	82	37	224	206
D	67	229	8	183	127	104	57	60	49	138	42	91	64	151	234	77
E	248	254	237	131	215	76	17	70	243	20	30	62	214	149	26	98
F	129	222	200	54	246	107	198	33	15	197	74	90	79	89	25	124

Tablo 3. Optimizasyon tabanlı s-box üretme yöntemlerinin performans karşılaştırması

	Lineer olmama			BIC		SAC	Max.
Optimizasyon Tekniği	**Ort**	**Min**	**Max**	**Lineer O.**	**SAC**	**Ort**	**XOR**
Öğretme-öğrenme algoritması [19]	**106.5**	104	110	105.2	0.4984	0.5120	10
Ateşböceği algoritması [20]	**107.5**	106	108	104.3	0.5001	0,4944	10
Uyarlanabilir ajan kahramanlar ve korkaklar algoritması [21]	**109.75**	108	112	104.35	0.5009	0.5068	10
Guguk kuşu arama algoritması [22]	**108.5**	106	110	103.85	0.5011	0.4995	10
Genetik algoritma [23]	**110.25**	110	112	104.07	0.5021	0.4953	10
Genetik algoritma [24]	**111.75**	110	112	104	0.5033	0.4968	12
Ateşböceği algoritması [25]	**107**	106	108	104.6	0.4974	0.496	10
Kesirli sıralı zaman gecikmeli hopfield sinir ağı algoritması [26]	**111.25**	110	112	102.57	0.5034	0.5007	10
Genetik algoritma [27]	**108**	108	108	90	0.4950	0.5068	10
Karınca kolonisi algoritması [28]	**107**	106	110	105.5	0.5010	0.5015	10
Tavlama algoritması [29]	**104**	102	106	103.2	0.4971	0.4980	10
Parçacık sürü optimizasyonu [30]	**112**	112	112	110	0.5134	0.5431	-
Parçacık sürü optimizasyonu [31]	**106.5**	104	108	105.85	0.4995	0.5036	10
Tiki-Taka algoritması [32]	**109.25**	106	110	104.07	0.5005	0.5017	10
Bakteriyel yiyecek arama optimizasyonu [33]	**107.5**	106	110	103.7	0.5025	0.5093	10
Sinüs kosinüs algoritması [34]	**109.5**	108	110	104.07	0.5020	0.4985	10
Parçacık sürü optimizasyonu [35]	**111.5**	108	112	110.28	-	0.5022	6
Genetik algoritma [36]	**108**	-	-	-	-	-	-
Çıplak köstebek faresi algoritması [37]	**109.75**	-	-	104.14	0.5041	0.4998	10
Kaotik çadır harita (Tablo 1)	**102.75**	100	106	103.71	0.4966	0.4978	12
Sinüs kosinüs algoritması (Tablo 2)	**108**	106	110	103.57	0.4976	0.4939	10

IV. SONUÇ VE ÖNERİLER

Şifreleme algoritmalarında en önemli gereksinimlerden bir tanesi karıştırma işlemidir. Bu işlem blok şifreleme algoritmalarında genellikle s-box yapıları ile gerçekleştirilir. Çünkü s-box lineer olmayan bir yapıdır ve veriyi etkin bir şekilde karıştırır. Güçlü s-box yapıları elde etmek için birçok teknik kullanılmaktadır. Bunlardan bir tanesi optimizasyon tekniğidir. Bu çalışmada öncelikle s-box yapılarının nasıl optimize edildiği açıklanmıştır. Daha sonra çadır harita yardımıyla elde edilen güçsüz bir s-box, sinüs kosinüs algoritması ile optimize edilmiştir. Başlangıçtaki güçsüz s-box yapısının lineer olmama değeri 500 iterasyon sonunda 102.75 'den 108 'e çıktığı gözlemlenmiştir. Ayrıca literatürde kullanılan diğer değerlendirme kriterleri de açıklanmış olup önerilen s-box yapısı analiz edilmiştir. Yapılan optimizasyon işlemi sonunda lineer olmama ve XOR dağılımı değerlerinde artış olmuştur. Diğer kriterler için çoğu çalışmada benzer değerler elde edilmektedir. Bunlardan bir tanesi SAC değeridir. SAC değeri bu çalışmada 0.5 değerine oldukça yakın bir değerdir.

Ayrıca literatürdeki diğer optimizasyon tabanlı s-box geliştirme yönteminin performans değerleri incelenmiştir. Tablo 3 'e bakıldığında bu yöntemler arasında sadece bir çalışmanın 112 değerine ulaştığı görülmektedir. Bu sonuç aslında optimum çözümün henüz elde edilemediğini gösteriyor. Gelecek çalışmalarda araştırmacılar için sıcak bir konu olmaya devam edecektir. Özellikle yeni geliştirilen metasezgisel algoritmalar farklı şekillerde kullanılarak daha etkin sonuçların elde edilebileceği düşünülmektedir. Bu algoritmalar hem zaman bakımından hem de kriptografik özellikler olarak daha iyi sonuçlar verebilir.

KAYNAKLAR

[1] J. Daemen and V. Rijmen, ''AES proposal: Rijndael,'' in Proc. 1st Adv. Encryption Conf., CA, USA, 1998, pp. 1–45.
[2] Artuğer, F., & Özkaynak, F. (2021). An effective method to improve nonlinearity value of substitution boxes based on random selection. *Information Sciences*, 576, 577-588.
[3] Liu, G., Yang, W., Liu, W., & Dai, Y. (2015). Designing S-boxes based on 3-D four-wing autonomous chaotic system. *Nonlinear dynamics*, 82(4), 1867-1877.
[4] Liu, L., Zhang, Y., & Wang, X. (2018). A novel method for constructing the S-box based on spatiotemporal chaotic dynamics. *Applied sciences*, 8(12), 2650.
[5] Özkaynak, F., Çelik, V., & Özer, A. B. (2017). A new S-box construction method based on the fractional-order chaotic Chen system. *Signal, Image and Video Processing*, 11(4), 659-664.
[6] Khan, M., & Shah, T. (2015). An efficient construction of substitution box with fractional chaotic system. *Signal, Image and Video Processing*, 9(6), 1335-1338.
[7] Özkaynak, F., & Yavuz, S. (2013). Designing chaotic S-boxes based on time-delay chaotic system. *Nonlinear Dynamics*, 74(3), 551-557.
[8] 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 dynamics*, 87(2), 1081-1094.
[9] Artuğer, F., & Özkaynak, F. (2020). A novel method for performance improvement of chaos-based substitution boxes. *Symmetry*, 12(4), 571.
[10] Artuğer, F., & Özkaynak, F. (2022). A method for generation of substitution box based on random selection. *Egyptian Informatics Journal*, 23(1), 127-135.
[11] Anees, A., & Chen, Y. P. P. (2020). Designing secure substitution boxes based on permutation of symmetric group. *Neural Computing and Applications*, 32(11), 7045-7056.
[12] Javeed, A., Shah, T., & Ullah, A. (2020). Construction of non-linear component of block cipher by means of chaotic dynamical system and symmetric group. *Wireless Personal Communications*, 112(1), 467-480.
[13] Siddiqui, N., Khalid, H., Murtaza, F., Ehatisham-Ul-Haq, M., & Azam, M. A. (2020). A novel algebraic technique for design of computational substitution-boxes using action of matrices on Galois field. *IEEE Access*, 8, 197630-197643.
[14] Alexan, W., ElBeltagy, M., & Aboshousha, A. (2022). Rgb image encryption through cellular automata, s-box and the lorenz system. *Symmetry*, 14(3), 443.
[15] Haque, A., Abdulhussein, T. A., Ahmad, M., Falah, M. W., & Abd El-Latif, A. A. (2022). A Strong Hybrid S-Box Scheme Based on Chaos, 2D Cellular Automata and Algebraic Structure. *IEEE Access*, 10, 116167-116181.
[16] Farhan, A. K., Ali, R. S., Yassein, H. R., Al-Saidi, N. M. G., & Abdul-Majeed, G. H. (2020). A new approach to generate multi S-boxes based on RNA computing. *Int. J. Innov. Comput. Inf. Control*, 16(1), 331-348.
[17] Mohamed, A. G., Korany, N. O., & El-Khamy, S. E. (2021). New DNA coded fuzzy based (DNAFZ) S-boxes: Application to robust image encryption using hyper chaotic maps. *IEEE Access*, 9, 14284-14305.
[18] Basha, H. A. M. A., Mohra, A. S. S., Diab, T. O. M., & El Sobky, W. I. (2022). Efficient image encryption based on new substitution box using DNA coding and bent function. *IEEE Access*, 10, 66409-66429.
[19] Farah, T., Rhouma, R., & Belghith, S. (2017). A novel method for designing S-box based on chaotic map and teaching–learning-based optimization. *Nonlinear dynamics*, 88(2), 1059-1074.
[20] Ahmed, H. A., Zolkipli, M. F., & Ahmad, M. (2019). A novel efficient substitution-box design based on firefly algorithm and discrete chaotic map. *Neural Computing and Applications*, 31(11), 7201-7210.
[21] Zamli, K. Z. (2021). Optimizing S-box Generation based on the Adaptive Agent Heroes and Cowards Algorithm. *Expert Systems with Applications*, 115305.
[22] Alhadawi, H. S., Majid, M. A., Lambić, D., & Ahmad, M. (2021). A novel method of S-box design based on discrete chaotic maps and cuckoo search algorithm. *Multimedia Tools and Applications*, 80(5), 7333-7350.
[23] Wang, Y., Zhang, Z., Zhang, L. Y., Feng, J., Gao, J., & Lei, P. (2020). A genetic algorithm for constructing bijective substitution boxes with high nonlinearity. *Information Sciences*, 523, 152-166.
[24] Artuğer, F., & Özkaynak, F. (2022). SBOX-CGA: substitution box generator based on chaos and genetic algorithm. *Neural Computing and Applications*, 34(22), 20203-20211.
[25] Alhadawi, H. S., Lambić, D., Zolkipli, M. F., & Ahmad, M. (2020). Globalized firefly algorithm and chaos for designing substitution box. *Journal of Information Security and Applications*, 55, 102671.

[26] Ahmad, M., & Al-Solami, E. (2020). Evolving dynamic S-boxes using fractional-order hopfield neural network based scheme. *Entropy*, 22(7), 717.
[27] Wang, Y., Wong, K. W., Li, C., & Li, Y. (2012). A novel method to design S-box based on chaotic map and genetic algorithm. *Physics Letters A*, 376(6-7), 827-833.
[28] Ahmad, M., Bhatia, D., & Hassan, Y. (2015). A novel ant colony optimization based scheme for substitution box design. *Procedia Computer Science*, 57, 572-580.
[29] Chen, G. (2008). A novel heuristic method for obtaining S-boxes. *Chaos, Solitons & Fractals*, 36(4), 1028-1036.
[30] Khan, L. S., Hazzazi, M. M., Khan, M., & Jamal, S. S. (2021). A novel image encryption based on rossler map diffusion and particle swarm optimization generated highly non-linear substitution boxes. *Chinese Journal of Physics.*
[31] Hematpour, N., & Ahadpour, S. (2021). Execution examination of chaotic S-box dependent on improved PSO algorithm. *Neural Computing and Applications*, 33(10), 5111-5133.
[32] Zamli, K. Z., Kader, A., Din, F., & Alhadawi, H. S. (2021). Selective chaotic maps Tiki-Taka algorithm for the S-box generation and optimization. *Neural Computing and Applications*, 1-18.
[33] Tian, Y., & Lu, Z. (2017). Chaotic S-box: Intertwining logistic map and bacterial foraging optimization. *Mathematical Problems in Engineering*, 2017.
[34] Alzaidi, A. A., Ahmad, M., Ahmed, H. S., & Solami, E. A. (2018). Sine-cosine optimization-based bijective substitution-boxes construction using enhanced dynamics of chaotic map. *Complexity*, 2018.
[35] Ahmad, M., Khaja, I. A., Baz, A., Alhakami, H., & Alhakami, W. (2020). Particle swarm optimization based highly nonlinear substitution-boxes generation for security applications. *IEEE Access*, 8, 116132-116147.
[36] Kang, M., & Wang, M. (2022). New Genetic Operators for Developing S-Boxes With Low Boomerang Uniformity. *IEEE Access*, 10, 10898-10906.
[37] Zamli, K. Z., Din, F., & Alhadawi, H. S. (2023). Exploring a Q-learning-based chaotic naked mole rat algorithm for S-box construction and optimization. *Neural Computing and Applications*, 1-23.
[38] Mirjalili, S. (2016). SCA: a sine cosine algorithm for solving optimization problems. *Knowledge-based systems*, 96, 120-133.
[39] Webster, A. F., & Tavares, S. E. (1985, August). On the design of S-boxes. In Conference on the theory and application of cryptographic techniques (pp. 523-534). Springer, Berlin, Heidelberg.
[40] Biham, E., & Shamir, A. (1991). Differential cryptanalysis of DES-like cryptosystems. *Journal of CRYPTOLOGY*, 4(1), 3-72.