INTERNATIONAL JOURNAL OF
Telecommunications
AIR DEFENSE COLLEGE

Nonlinearity Improvement for Ascon Substitution Box

Citation: Bakr, Mohamed.; Korany, Noha. *Inter. Jour. of Telecommunications, IJT'2024, Vol. 04, Issue 01, pp. 01-13, 2024.*

Editor-in-Chief: Youssef Fayed.

Received: 12/02/2024.

Accepted: 15/03/2023.
Published: 20/03/2024.

Publisher's Note: The International Journal of Telecommunications, IJT, stays neutral regard-ing jurisdictional claims in published maps and institutional affiliations.

Mohamed Bakr [1] *, and Noha Korany [2]

[1] Air Defense College, Alexandria University, Egypt; Mohamedbakr.8010.adc@alexu.edu.eg
[2] Faculty of Engineering, Alexandria University, Egypt; Noha.korany@alexu.edu.eg

Abstract: The technological scene is developing towards employing tiny-sized devices. For a variety of functions that encompass sensing, identification, and decision-making. These devices are restricted in their resources to be secure. The National Institute of Standards and Technology (NIST) has published the Ascon, a standard lightweight cryptography (LWC) algorithm for data gathered by restricted devices in 2023. The fundamental function of Ascon is the permutation function. The main core of the permutation function is the substitution box (S-box). This paper proposes an S-box based on chaotic systems using coupled map lattices (CML), which agrees with LWC algorithms due to its low complexity. The proposed S-box suggests high performance and security for restricted devices. The security robustness is tested using various cryptographic criteria, such as nonlinearity, strict avalanche criteria, and differential approximation probabilities. The proposed S-box approaches significant resistance to both linear and differential cryptanalysis. So, it could be replacing the substitution and linear diffusion layers of Ascon permutation to improve the nonlinearity and randomness.

Keywords: Ascon, IoT, Lightweight Cryptography, Nonlinearity, Information Security.

1. Introduction

The Internet of Things (IoT) depends on a wide selection of technologies to integrate digital and physical areas, as physical objects are combined with sensors and actuators in one integrated system [1]. The IoT network is structured to be made up of three layers: the perception layer, which is responsible for sensing and acquiring data from surroundings; the transmission layer, which is responsible for distributing sensitive information onto safe storage on cloud platforms; and the application layer, which enables decision-making [2]. The constrained IoT devices present a big challenge to security as they have tiny dimensions with limited computing power and limited memory size to process and store the data, which is not appropriate with the existing security schemes [3, 4]. Traditional security algorithms are constructed by using complex cryptographic functions, which demand a lot of resources. So, it is not suitable for IoT devices with constrained resources [5]. Thus, the LWC appeared to satisfy the constrained IoT hardware requirements with high performance and appropriate security. The LWC includes a lot of variant primitives, such as message authentication codes, encryption, and hashing functions. Since it is easy for an adversary to modify the encrypted data without being observed, such primitives cannot be employed in secure communication [6]. The use of authenticated encryption with additional data (AEAD) techniques may overcome this issue. While the additional data is not encrypted but just authenticated, which can be any auxiliary data to the main data, the AEAD method ensures the data's authenticity and integrity, in addition to confidentiality [7]. The NIST and Bernstein launched the competition for authenticated encryption: security, applicability, and robustness (CAESAR) initiative in 2013. In 2018, the CAESAR portfolio ended, with Ascon being the best choice [8]. The NIST issued a request to standardize lightweight AEAD in the same year. Finally, NIST announced Ascon as the LWC standard to secure data moving

through resource-constrained devices in 2023 [9]. Ascon was advanced in 2014 by a group of investigators. The concept behind it relies on sponge structure in conjunction with monkey duplex [10]. The construction makes it simple to take advantage of Ascon in various configurations, and its permutation performs a round transformation continuously 12 times. Integrating round constants, nonlinear substitutes applying S-boxes, and a linear diffusion layer results in a round transformation [11]. In literature, substitution and permutation are primitive cryptographic operations introduced by Claude Shannon in 1949 [12]. The core of any substitution permutation network (SPN) is the substitution process. It is achieved through the S-box. For LWC algorithms, Prathiba proposed a 4-bit S-box with high perfor-mance, but the security is powerless compared to high-end bit S-boxes such as 8-bit S-boxes [13]. Chaos theory and the mathematics following it are used to develop techniques for cryptography. Research has proposed many S-boxes depending on chaos systems. Matthews launched the initial attempt to integrate chaos theory into cryptography in 1989 [14]. For constructing S-boxes, chaotic maps are employed to develop S-boxes that have the advantage of simple mathematical calculation with rapid processing time, but they have a low nonlinearity (NL) [15–22]. Yuanyuan recommended a two dimensional exponential quadratic chaotic map with ergodicity and improved unpredictability within a broad control parameter range [23]. Sine maps can create random integer sequences with good NL values [24–29]. Thakor's proposed S-box design employs chaotic theory by enhancing the sine map to provide random behavior for the S-box [30]. CML, as a sort of spatiotemporal chaos system (SCS), has received attention from researchers in recent years. The CML illustrates a dynamic development in both time and space. It illustrates more complicated nonlinear events than one dimensional chaos systems while having less numerical difficulty than high-dimensional chaos systems [31–38].

This research is driven by the fact that Ascon is the standard algorithm for securing data transfer between sensors, embedded systems, and IoT networks, which outperformed all competitors in the CAESAR and NIST competitions. So, this paper's aimed at improving randomness and the NL of the Ascon S-box, then reducing the code size and the processing time of Ascon. Applying an S-box depending on the CML system to supplant the substitution and diffusion layers in Ascon. This paper is to construct a chaos based S-box that is efficient and suitable for use in the LWC. The proposed S-box consists of 5-bit input and output depending on the modified sine map using the CML to have a chaotic system. The rest of the research is organized in the following manner: Section 2 provides a detailed review of the Ascon algorithm, the chaotic system, and the Spatiotemporal chaos system. Section 3 shows the suggested S-box construction method, followed by the flowchart and the generator function code in Python. Section 4 describes the S-box security criteria. Section 5 demonstrates the performance analysis of the S-box and then finalizes the conclusion of the research in the last section.

2. Preliminary

2.1 Ascon

Christoph, Maria, Florian, and Martin presented Ascon [39], which is based on duplex sponge construction but uses a stronger initialization and finalization keyed function for AEAD in both versions (Ascon-128 and Ascon-128a) [40], and on sponge construction with both versions (Ascon-Hash and Ascon-Xof) for the hashing algorithms [41]. Ascon has been built for strong security. Ascon-128 affords 128-bit security level, limiting the number of handled P and A blocks to 2^{64} blocks/key [42]. Resilience against forgery attacks may be achieved with complexity $2^{c}/_{2}$. Key recovery attacks, which may be found with a complexity lower than min (2^{l}, $2^{c}/_{2}$) equals 2^{128} [43]. AEAD Ascon is composed of four steps: initialization of state (S), processing of additional data

(A), processing of plaintext (P) or ciphertext (C), and finalization steps for encryption and decryption algorithms.

Ascon-128 parameters are the secret key (K), whose key length (l) is 128-bit, the block size of data is 64-bit, and the numbers of rounds are a and b as 12 and 6, respectively. The initialization and finalization permutations (p^a) and the intermediate permutation (p^b). Ascon-128 operates on a 128-bit state that is updated using the permutations p^a and p^b which are the initial components. The permutations use an iterative round change based on a SPN that consists of a constant addition layer, a substitution layer, and a diffusion layer. The 320-bit of S is concatenated ($\|$) from two parts as exterior part (S_r) of r-bit and interior part (S_c) of c-bit as $S = S_r \| S_c$, where the rate (r) and capacity (c), at $c = 320 - r$, and then S subdivided into five register words of 64-bit as $w_0, \dots, w_4$. In the step of constant addition, a round constant is added to the register word w_2 of S in each round. The 5-bit S-box is used 64 times simultaneously at 320-bits by the substitution layer to update S. The diffusion layer diffuses each $w_0, \dots, w_4$. The inputs for the AEAD are K and a nonce (N) with 128-bit length, P, and A of arbitrary length. Ascon output that consists of the C equal in length with P along with an authentication tag (T) of size 128-bit that authenticates both the A and the C (see Figure. 1-a), the encryption process ($E_{k,r,a,b}$) of Ascon [39]. The verified decryption inputs K, N, A, C, and T produce an output. The output will be P if the verification of T is effective, or an error ($\perp$) if the verification of T fails (see Figure. 1-b), the decryption process ($D_{k,r,a,b}$) of Ascon [39].

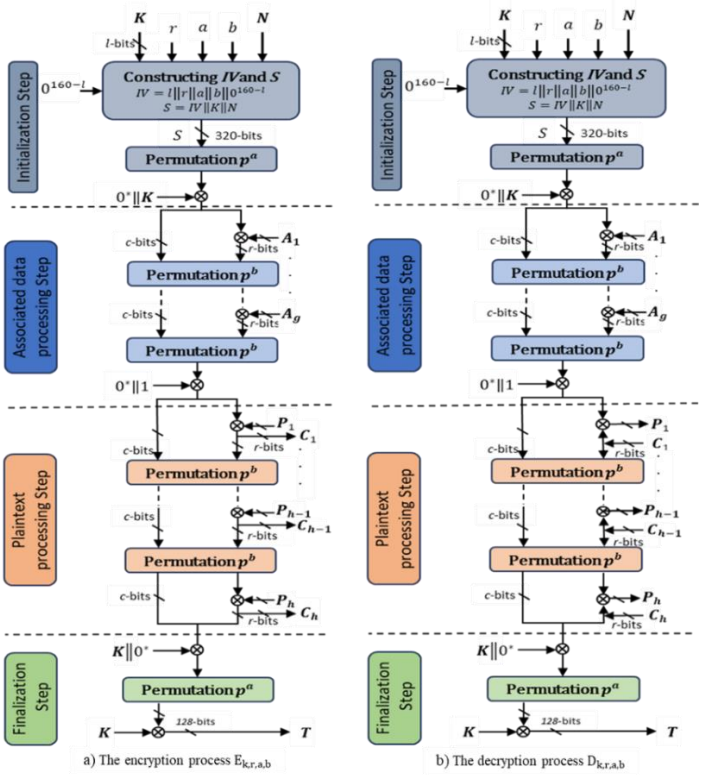


Figure 1. Ascon-128 encryption and decryption process.

The Initialization. Constructing the initialization vector (IV) then concatenated with S, $IV \leftarrow l\|r\|a\|b\|0^{160-l}$, $IV = 80400c0600000000$, where every digit denotes a hexadecimal of 4-bit so IV consist of 64-bit as (4×16) bit. $S \leftarrow IV\|K\|N$ of 320-bit as (64+128+128) bit, the length of K and N are 128-bit. The output of initialization step $S \leftarrow p^a(S) \oplus (0^{320-l}\|K)$[39].

Processing on A. It attaches a single 1 and the smallest number of 0s to A to acquire a multiple of r-bits and split A into g-blocks (g is the numeral of blocks of A) of r-bits $A_1\| \dots \|A_g = A\|1\|0^{r-1-(|A| \bmod r)}$, where |A| is the

length of the bitstring A in bits. If A is undefined (∅) at $|A| = 0$, so no buffering is applied and $g = 0$. For each block A_I at $(1 \le i \le g)$, the $S \leftarrow p^b((S_r) \oplus A_i)\|S_c)$, the $S \leftarrow S \oplus (0^{319}\|1)$ [39].

Processing on P or C. It attaches a single 1 and the smallest number of 0s to P to acquire a multiple of r-bits and split P into h-blocks (h is the numeral of blocks to P) of r-bits $P_1\| \ldots \|P_t = P\|1\|0^{r-1-(|P| \bmod r)}$, |P| is the length of the bitstring P in bits. During each encryption iteration $C_i \leftarrow S_r \oplus P_I$ to extract of one C_I respectively for all block excepting the last C_h. $S \leftarrow p^b(C_i\|S_c)$, where $(1 \le i < h)$. $S \leftarrow C_i\|S_c$. The last C_h block is $\lfloor C_h \rfloor_{|P| \bmod r}$ (truncated to the first r-bits) to the unpadded last P_h block. During each decryption Iteration $P_i \leftarrow S_r \oplus C_I$ to calculated P_I the respectively for all block excepting the last P_h. $S \leftarrow p^b(C_i\|S_c)$. The last P_h block is $\lfloor S_h \rfloor_l \oplus C_h$, where $0 \le l \le r - 1$ of the unpadded last C_h block. The $S \leftarrow (S_r \oplus (P_h\|1\|0^{r-1-l}))\|S_c$ [39].

The Finalization. $S \leftarrow p^a(S \oplus (0^r\|K\|0^{-l}))$. T (consist of the last 128-bit of S) as $T \leftarrow \lceil S \rceil^{128} \oplus \lceil K \rceil^{128}$ (truncated to the last 128-bits) [39].

2.2 Chaotic Systems

One (1D) and two (2D) dimensional maps are the dual main groups into which chaotic systems are applied for cryptography applications and produce promising results. Chaos has been applied to developing S-boxes with low-dimensional systems. Chaotic behaviors appear from nonlinear dynamical formulas called logistic and sine maps, as defined in eq. (1) and eq. (2), respectively [14].

$$x_{n+1} = \mu \, . \, x_n(1 - x_n) \tag{1}$$

$$x_{n+1} = \alpha . sin(\pi x_n) \tag{2}$$

If x_n is a variable that represents the ratio of the existing population to the highest population and μ is the control behavior of the logistic map, $(0 \le \mu \le 4)$, although α is the control behavior of the sine map, $(0 \le \alpha \le 1)$ [23, 27]. A bifurcation diagram shows how a system's behavior varies when its parameters change. It is used to illustrate how a system changes from an arranged to a chaotic pattern. The Lyapunov exponent diagram illustrates a system's sensitivity to slight initial conditions. The positive value of the Lyapunov exponent reflects a chaotic pattern, while a negative value denotes a periodic pattern. Diagrams of bifurcation and Lyapunov exponent were studied for logistic map and sine map (see Figure. 2). Both maps' diagrams contain a long cutoff period followed by a short chaotic phase if $(3.57 \le \mu \le 4)$, (see Figure. 2-a and 2-d), and if $(0.87 \le \alpha \le 1)$, (see Figure. 2-b and 2-e). Cryptographers regularly use multi-dimensional chaotic systems for high-resource devices due to their computational complexity, while logistic and sine maps are more suitable for constrained resource devices with low computational and difficulty processes [20]. Thakor proposed an enhanced sine map, which is defined in eq. (3) [30], by extending the control behavior coefficient α to be 4 instants of 1 to extend the chaotic behavior interval as $(1.5 \le \mu \le 3.29) \cup (3.5 \le \mu \le 4)$ and decrease the cutoff interval (see Figure. 2-c and 2-f).

$$x_{n+1} = sin(\pi . \mu . x_n(1 - x_n)) \tag{3}$$

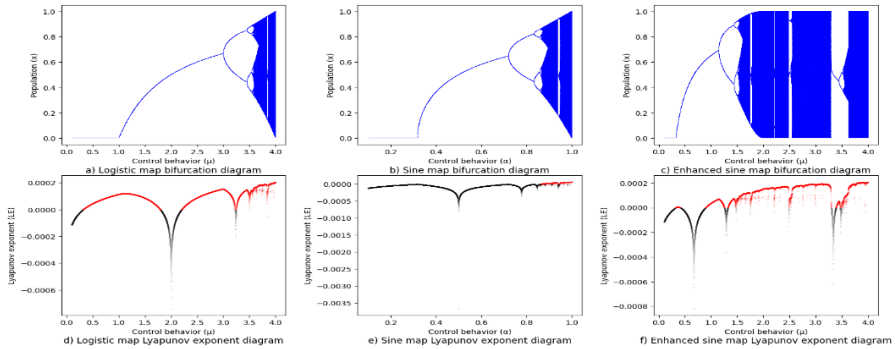


Figure 2. Bifurcation and Lyapunov exponent diagrams for logistic, sine, and enhanced sine maps

2.3 Spatiotemporal Chaos System (SCS)

A typical SCS is CML, which can be utilized in cryptography. It has been shown to have high security and high efficiency. CML outperforms low- and high-dimensional chaotic schemes because they obtain a wider scale of parameters, which improves the security of the cipher and makes CML systems more suitable for cryptography. The general form of CML is defined in eq. (4) as a 2D system in space (lattice) and time, so the CML can be represented as 1D system by defining the t = 0. So, the 1D CML system is defined in eq. (5).

$$x_L^{t+1} = (1-\varepsilon)f[x_L^t] + \frac{\varepsilon}{2}(f[x_{L-1}^t] + f[x_{L+1}^t]) \tag{4}$$

$$x_L^1 = (1-\varepsilon)f[x_L^0] + \frac{\varepsilon}{2}(f[x_{L-1}^0] + f[x_{L+1}^0]) \tag{5}$$

, where x_L^t is the state of the L-th element of the lattice at time t, ε is the coupling variable $0 \le \varepsilon \le 1$, and f is a local map function. The local map can be a logistic or sine map. Diagrams of bifurcation and Lyapunov exponent for the logistic CML (see Fig. 3-a and 3-d), which has a small chaos period of $3.57 \le \mu \le 4$. The sine CML (see Figure. 3-b and 3-e) has a more chaotic phase of $0.85 \le \alpha \le 4$ but still has a cutoff period. But the enhanced sine CML (see Figure. 3-c and 3-f) has more cutoff periods.

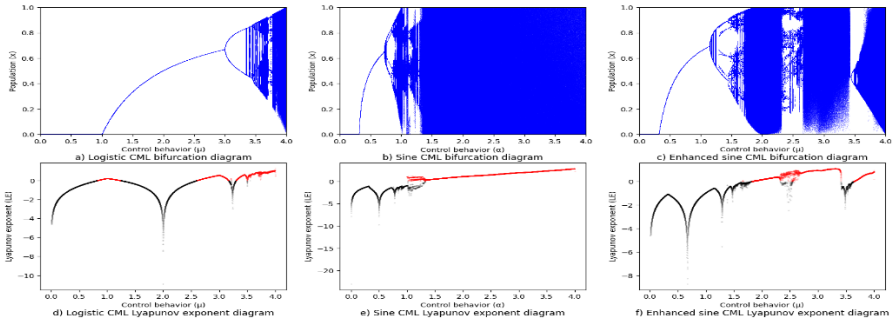
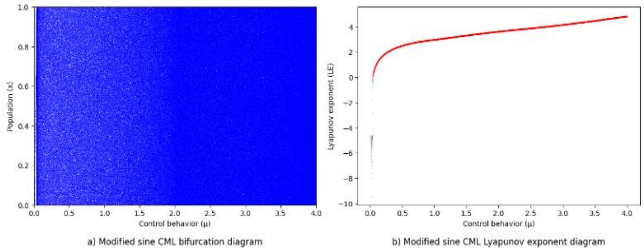


Figure 3. Diagrams of bifurcation and Lyapunov exponent for CML system with logistic, sine, and enhanced sine maps

3. Construction the Proposed Substitution Box

This section provides an overview of the design criteria of the proposed S-box for use with Ascon in place of the substitution and linear diffusion layers. The S-box transforms the input values to unique output values of 5-bit, $\mathbb{F}_2^5 \rightarrow \mathbb{F}_2^5 : X \rightarrow S(X)$, where the 5-bit input digits create 2^5 possible input values. The S(X) values change based on the chaos generator, which is based on the 1D CML system defined in eq. (6), which consists of set lattices. The local map (f) for the CML is defined as a modified sine map, which depends on the advantages of the enhanced sine map [30], which are expressed in eq. (3) and multiplied by the number of lattices (L) as shown in eq. (7). The change improves the chaos performance of the CML, which has a broad control parame-

ter without cutoff. Diagrams of bifurcation and Lyapunov exponent demonstrate the improvement in chaos performance along with all control behavior parameters ($0.01 \leq \mu \leq 4$) (see Figure. 4-a and 4-b). The enhancement CML system is utilized to produce the proposed S-box named Modified CML (MCML S-box).



a) Modified sine CML bifurcation diagram b) Modified sine CML Lyapunov exponent diagram

Figure 4. Diagrams of bifurcation and Lyapunov exponent for modified sine CML

$$x^1(i) = (1-\varepsilon)f[x^0(i)] + \frac{\varepsilon}{2}(f[x^0(i-1)] + f[x^0(i+1)]), \quad 0 \leq i < L \tag{6}$$

$$f(x_i) = L.\sin\left(\pi.\mu.x_i(1-x_i)\right), \quad 0 \leq x \leq 1 \tag{7}$$

The flowchart of the construction of the MCML S-box (see Figure. 5) demonstrates the steps as follows:

1. Set the value of $x_0 = 0.5$, $\mu = 3.19$ and $\varepsilon = 0.3$, where $0 \leq x_i \leq 1$, $0 \leq \mu \leq 4$ and $0 \leq \varepsilon \leq 1$.
2. Compute the $f(x^0{}_i)$ sequence using the modified sine map as in eq. (7).
3. Compute the $x^1(i)$ sequence using MCML as in eq. (6).
4. Repeat step 3: if $L = 32$, where $0 \leq i < L$. So, repeat it L times to generate a chaotic sequence.
5. Construct the 5-bit MCML S-box with the chaos output of the MCML.

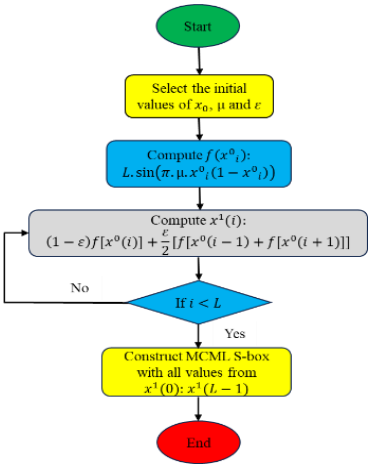


Figure 5. Flowchart of MCML S-box construction

From running the Python function named ***Generate_S_box ()***, it generates a 5-bit MCML S-box as illustrated in Table 1, the 5-bit input value X as a decimal number, and chaos it to have an output of a nonreported value as S(X).

Table 1. The MCML S-box with values of $x_0 = 0.5$, $\mu = 3.19$ and $\varepsilon = 0.3$

X	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
S(X)	31	22	16	6	20	29	19	1	23	17	26	21	7	30	5	0
X	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
S(X)	10	15	27	4	25	12	28	8	13	14	3	18	9	11	24	2

4. Security Criteria

This section explains the security strength of the 5-bit MCML S-box [44]. The MCML S-box shown in Table 1 is used while judging the listed security criteria.

4.1 Bijective Characteristic

A function is bijective if it is both injective and surjective, meaning each input value changes to a unique output and every output has a corresponding input value. For an output vector $Y = y_1, \ldots, y_n$, the Hamming weight $H_{wt}(.)$ is defined as follows [45]:

$$H_{wt}(Y) = \sum_{\Gamma=1}^{2^n} (y_1^\Gamma \oplus y_2^\Gamma \ldots \oplus y_n^\Gamma) = 2^{n-1} \tag{8}$$

where $y_j \in \{0,1\}$ and for 5-bit MCML S-box $(y_1, y_2, y_3, y_4, y_5) \neq (0,0,0,0,0)$. The $H_{wt}(Y)$ for all values of S(X) is illustrated in Table 2. The summation of $H_{wt}(Y)$ is equal to $16 = 2^{n-1}$, which means the MCML S-box has a balancing in the numbers of 0^s and 1^s, and includes all different values from 0 to 31. Therefore, the MCML S-box achieves the bijective characteristic.

4.2 Nonlinearity (NL)

The S-box with low NL scores is regarded as cryptographically vulnerable and may jeopardize the cryptosystem. The NL may be quantified using the Hamming distance. It is the distance between any related input-output pairs $(x_i,\ y_j)$ [45].

Table 2. Hamming weight H_{wt} and distance (H_d) for the MCML S-box

Γ	X	$S(X) = Y$	H_{wt}	H_d	Γ	X	$S(X) = Y$	H_{wt}	H_d
	$(x_1, \ldots, x_5)$	$(y_1, \ldots, y_5)$				$(x_1, \ldots, x_5)$	$(y_1, \ldots, y_5)$		
1	(00000)	(11111)	1	5	17	(10000)	(01010)	0	3
2	(00001)	(10110)	1	4	18	(10001)	(01111)	0	4
3	(00010)	(10000)	1	2	19	(10010)	(11011)	0	2
4	(00011)	(00110)	0	2	20	(10011)	(00100)	1	4
5	(00100)	(10100)	0	1	21	(10100)	(11001)	1	3
6	(00101)	(11101)	0	2	22	(10101)	(01100)	0	3
7	(00110)	(10011)	1	3	23	(10110)	(11100)	1	2
8	(00111)	(00001)	1	2	24	(10111)	(01000)	1	5
9	(01000)	(10111)	0	5	25	(11000)	(01101)	1	3
10	(01001)	(10001)	0	2	26	(11001)	(01110)	1	4
11	(01010)	(11010)	1	1	27	(11010)	(00011)	0	3
12	(01011)	(10101)	1	4	28	(11011)	(10010)	0	2
13	(01100)	(00111)	1	3	29	(11100)	(01001)	0	3
14	(01101)	(11110)	0	3	30	(11101)	(01011)	1	3
15	(01110)	(00101)	0	3	31	(11110)	(11000)	0	2
16	(01111)	(00000)	0	4	32	(11111)	(00010)	1	4

$$NL = H_d(x_i, y_j) = \frac{1}{2^n} \sum_{\Gamma=1}^{2^n} \sum_{i=j=1}^{n} \left(x_i^\Gamma \oplus y_j^\Gamma\right) \tag{9}$$

As illustrated in Table 2, the minimum value of H_d for the 5-bit MCML S-box is 1, while the maximum is 5. Consequently, the average H_d is 3 for the MCML S-box.

4.3 Strict Avalanche Criteria (SAC)

SAC works on two concepts: completeness (Ω) and the avalanche effect (Ψ). The Ω states that each of its output bits depends on each of its input bits. The Ψ states that any insignificant change in input bits produces significant variations in output bits. The Ω and Ψ are defined in eq. (10) and eq. (11), respectively [46]. The input vector $X = (x_1, \ldots, x_n)$, $x_i \in \{0,1\}$, there is a corresponding output vector S(X) = Y. Altering i-th bit in X results to Y_j such that $Y_j = S(X_i)$, where $i = 1, \ldots, n$ and $j = 1, \ldots, m$.

$$\Omega = 1 - \frac{1}{n \times m} \sum (i,j) \left| e_{i,j} = 0 \right. \tag{10}$$

$$\Psi = 1 - \frac{1}{\#\Gamma \times n \times m} \sum_{i=1}^{n} \sum_{j=1}^{m} \left| e_{i,j} - \frac{1}{2} \#\Gamma \right. , \qquad e_{I,j} = \left\{ X \in \mathbb{F}_2^n \left| (Y)_i \neq \left(Y_j\right)_i \right. \right\} \tag{11}$$

where, $\#\Gamma$ is the number of inputs. The Ω and Ψ value close to 1 indicates that S-box has strong NL and good diffusion. The MCML S-box Ω and Ψ values are 1 and 0.82 respectively. SAC applies when a variation in one input bit leads to a variation in at least 50% of the output bits. The avalanche vector $V_{i,j} = Y \oplus Y_j$ is obtained [47]. Repeating for all i-th of vector X, the (5 × 5) matrix $V_{i,j}^{\Gamma}$ is obtained. Then, the process is repeated 2^n times for all input vectors X. Then the SAC matrix is calculated according to eq. (12). Table 3 demonstrates that the SAC matrix with the average SAC value of the MCML S-box is 0.5.

$$SAC = \frac{1}{2^n} \sum_{\Gamma=1}^{2^n} V_{i,j}^{\Gamma}, \quad At\ i = j \tag{12}$$

Table 3. The SAC analysis of the MCML S-box

0.625	0.500	0.750	0.500	0.750
0.500	0.375	0.625	0.625	0.500
0.500	0.625	0.500	0.375	0.500
0.375	0.500	0.375	0.250	0.625
0.500	0.375	0.375	0.375	0.500

4.4 Bit Independence Criterion (BIC)

BIC applies when a variation in i-th bit leads to an individual variation in every output bit of j and q, where i, j, q ∈ (1, …, m) and j ≠ q. Accordingly, the two output bits of an S-box, y_j and y_q, if $y_j \oplus y_q$ shows high NL and fulfill the SAC, then the S-box has a proper BIC. BIC-SAC calculated by stating the vectors Y_1, …, Y_5 for each vector X. An avalanche vector $V_{i,j,q} = N_{i,j} \oplus M_{i,j}$is obtained, where $N_{i,j} = y_i \oplus y_j$ of Y and $M_{i,j} = y_i \oplus y_j$ of Y_q [47]. Repeating for all n-bit ($n = 5$) of vector X, the (5 × 5) matrix $V_{i,j,q}^{\Gamma}$ is obtained. Then, the process is repeated 2^n times for all input vector X. Then BIC-SAC matrix is calculated according to eq. (13). S-box must have a BIC-SAC value near 0.5 to improve the resistance property. Table 4 displays that the BIC-SAC matrix with the average BIC-SAC value of the MCML S-box is 0.5.

$$BIC - SAC = \frac{1}{2^n}\sum_{\Gamma=1}^{2^n} V_{i,j,q}^{\Gamma}, \quad At\ j \neq q \tag{13}$$

Table 4. The BIC-SAC analysis of the MCML S-box

0	0.500	0.375	0.625	0.625
0.500	0	0.375	0.500	0.375
0.375	0.375	0	0.625	0.500
0.625	0.500	0.625	0	0.500
0.625	0.375	0.500	0.500	0

4.5 Differential Approximation Probability (DAP)

The output/input variation Δx, Δy is arranged in a table known as the difference distribution table (DDT), which is used to examine the behavior and calculate the DAP value for an S-box. A higher DAP value signifies that the S-box has a lower resistance to differential attacks. The DAP is the highest probability of Δx, Δy then divides by 2^n [48]. For each value x, $(\Delta x, \Delta y) \in [0,\ 2^{n-1}]$ as in eq. (14). Table 5 demonstrates that the differential probability values while the highest value of the MCML S-box is 8, i.e., DAP value is 0.25.

$$\text{DAP} = \max_{\Delta x \neq 0, \Delta y} (\#\{x \in X | S(x) \oplus S(x \otimes \Delta x) = \Delta y\}) \times \frac{1}{2^n} \tag{14}$$

Table 5. The DDT of the MCML S-box

Δx \ Δy																															
	32	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	4	0	0	0	0	0	2	4	0	0	0	0	4	0	2	2	2	4	2	0	0	0	0	0	2	0	0	0	2	2	0
	0	2	0	4	0	2	0	2	0	6	2	0	2	0	0	0	2	0	0	0	0	0	0	0	2	0	4	2	0	0	4	2
	0	2	4	0	2	0	0	0	4	0	0	2	2	2	2	0	0	0	0	0	0	2	0	2	2	0	0	2	0	0	4	0
	0	2	2	0	2	2	4	0	0	2	2	2	0	0	0	2	0	4	0	0	2	0	0	2	0	0	0	2	0	2	0	0
	0	0	2	0	0	2	2	2	2	0	2	0	2	0	2	0	2	0	0	0	4	0	2	0	4	0	0	0	4	0	0	0
	0	0	0	2	0	0	2	0	2	0	2	4	2	0	2	0	2	2	0	0	2	0	2	0	0	2	0	0	2	2	2	0
	0	0	0	2	2	0	4	2	0	0	0	0	0	0	2	4	0	4	0	2	0	0	2	2	0	2	2	0	0	0	0	2
	0	0	4	0	0	0	2	0	0	2	0	0	0	4	4	0	0	0	2	0	0	2	0	2	2	0	0	2	2	0	4	0
	0	0	0	2	2	0	2	0	0	0	0	0	2	0	2	2	4	0	0	2	4	2	2	0	0	0	0	0	2	0	0	4
	0	2	0	0	0	2	0	4	0	0	2	0	0	0	4	2	0	2	2	0	0	0	0	0	6	2	0	0	0	2	2	0
	0	0	0	0	0	0	2	0	0	0	2	2	0	2	0	0	2	0	0	4	0	4	4	0	0	2	0	0	2	0	4	2
	0	0	2	0	2	0	0	0	2	4	0	0	4	0	0	2	0	0	2	4	0	0	2	0	4	2	0	0	2	0	0	0
	0	2	0	0	2	0	0	2	4	0	2	0	0	0	2	2	2	0	0	0	0	2	2	0	0	2	0	0	0	2	6	0
	0	0	0	2	0	4	2	2	2	2	2	2	2	0	0	0	2	2	2	2	2	0	0	0	0	0	2	0	0	0	0	0
	0	2	2	0	4	0	0	0	0	0	0	0	0	4	0	0	2	0	0	0	2	0	0	4	0	2	4	0	2	2	0	2
	0	0	0	2	0	2	0	2	0	2	2	4	0	0	2	0	2	0	0	4	2	2	0	0	0	0	0	4	0	2	0	0
	0	2	2	0	2	2	2	2	0	2	2	2	0	0	0	2	0	0	2	0	0	2	2	2	0	2	0	2	0	0	0	0
	0	0	2	2	2	0	0	0	4	0	0	2	0	0	2	2	0	0	2	2	0	0	0	2	0	2	8	0	0	0	0	0
	0	0	2	2	0	4	0	2	2	2	0	2	0	0	0	0	4	0	4	0	0	2	0	0	0	0	0	2	0	2	0	2

0	2	0	2	2	2	0	0	0	0	2	2	0	0	2	2	0	2	2	2	0	0	2	0	0	2	0	0	0	0	0	4
0	2	0	2	0	0	0	0	2	2	0	4	4	2	2	0	0	2	0	0	0	0	0	2	2	0	0	0	2	2	0	2
0	2	2	0	2	0	0	2	0	2	0	0	0	2	0	0	2	0	0	4	0	4	2	0	2	0	2	0	0	0	0	4
0	2	4	2	2	0	2	0	0	2	2	0	0	2	0	2	0	0	2	0	6	0	0	0	0	0	0	2	2	0	0	0
0	2	2	2	2	0	2	0	0	0	0	2	0	0	0	0	2	0	0	0	0	2	4	2	0	4	0	2	0	4	0	0
0	0	0	2	2	4	0	2	0	2	0	0	2	0	0	2	2	2	0	0	2	2	0	2	2	0	0	2	0	0	2	0
0	2	0	0	0	2	2	0	0	0	0	0	2	4	0	0	0	2	0	2	4	0	0	2	0	0	4	2	4	0	0	0
0	0	0	0	0	2	0	0	2	0	2	0	2	0	4	0	0	0	0	0	0	0	0	2	2	2	0	4	2	4	2	2
0	0	2	0	0	0	0	4	2	0	2	0	0	2	0	0	0	4	2	0	0	0	0	0	2	2	0	0	2	2	0	6
0	0	0	4	0	2	2	0	0	0	2	0	2	4	0	0	0	0	2	2	2	2	0	4	0	0	0	2	0	0	2	0
0	2	0	0	0	0	2	0	0	0	0	2	0	0	0	2	0	2	2	0	0	2	4	2	2	0	4	2	2	2	0	0
0	0	0	0	2	0	0	2	0	2	2	0	4	0	0	4	0	2	2	0	0	2	2	0	0	2	2	0	0	2	2	0

4.6 Linear Approximation Probability (LAP)

The greatest value of unbalance in the input-output components is found using LAP. Assume that the input differential Δx and the output differential Δy, and X is a set of all possible inputs with cardinality 2^n. The LAP is defined as:

$$LAP = \max_{\Delta x, \Delta y \neq 0} \left| \frac{\#\{x = X | x.\Delta x = S(X).\Delta y\}}{2^n} - \frac{1}{2} \right| \tag{15}$$

The maximum duplicate outcome of x. Δx = S(X). Δy for all x ∈ X found in the suggested MCML S-box is 8, i.e., LAP evaluate is 0.25. The smaller value of LAP close to 0 proposes better security [49].

5. Performance Analysis

The performance analysis of the suggested MCML S-box in terms of security is compared with the 5-bits S-boxes of Ascon, Thakor, and PRIMATE. Table 6 demonstrates the security criteria analysis. The proposed S-box achieved the highest NL value and has the ideal value for SAC and BIC-SAC criteria.

Table 6. Comparison of S-boxes security criteria analysis

S-Box	Bijective	NL	SAC	BIC-SAC	LAP	DAP
Proposed	True	3	0.50	0.50	0.25	0.25
Ascon [50]	True	2.5	0.51	0.58	0.25	0.25
Thakor [30]	True	2.65	0.57	0.53	0.25	0.25
PRIMATE [51]	True	2.5	0.52	0.54	0.375	0.0625

Table 7 demonstrates the criteria analysis for a set of proposed S-boxes with various values x_0, μ and ε. The security analysis results are very promising for having an appropriate security performance.

Table 7. The security analysis of Proposed S-boxes with x_0, μ and ε are variables

x_0	μ	ε	Bijective	NL	Ω	Ψ	SAC	BIC-SAC	DAP	LAP
0.1	3.431	0.3	True	3	1	0.84	0.500	0.512	0.19	0.25
0.1	3.235	0.5	True	3	1	0.88	0.500	0.512	0.25	0.25
0.2	3.484	0.7	True	3	1	0.88	0.490	0.500	0.25	0.25

0.3	1.159	0.8	True	3	1	0.85	0.515	0.487	0.25	0.25
0.4	3.306	0.4	True	3	1	0.88	0.490	0.500	0.19	0.25
0.4	2.121	0.8	True	3	1	0.83	0.495	0.487	0.25	0.25
0.5	2.696	0.1	True	3	1	0.84	0.510	0.500	0.25	0.25
0.5	3.519	0.9	True	3	1	0.78	0.500	0.525	0.19	0.25
0.6	1.444	0.2	True	3	1	0.80	0.470	0.525	0.25	0.25
0.6	3.341	0.5	True	3	1	0.89	0.545	0.487	0.25	0.25
0.7	3.265	0.3	True	3	1	0.81	0.535	0.525	0.25	0.25
0.8	1.322	0.1	True	3	1	0.84	0.520	0.500	0.25	0.25
0.8	3.938	0.7	True	3	1	0.81	0.515	0.487	0.25	0.25
0.9	0.991	0.6	True	3	1	0.81	0.515	0.525	0.19	0.25
0.9	2.750	0.8	True	3	1	0.85	0.495	0.500	0.25	0.25

The Ascon-128 algorithm was reorganized by Python, followed by replacing the substitution and diffusion layers with a the proposed MCML S-box to have a modified Ascon algorithm called Ascon-MCML. The both techniques were run on the same platform with Windows 11, Intel (R) Core i7-1165G7, 2.80 GHz, and Python 3.11 are used. Both techniques were used to encrypt and decrypt a plaintext "Hello" with an associated data "Welcome", while the key and nonce are random data. The input and output parameters for both algorithms are displayed in Table 8. Table 8 demonstrates that the Ascon-MCML has a faster processing time and a smaller code size than the Ascon-128 as both algorithms run in the same operating system for the same plaintext and associated data.

Table 8. The input and output parameters for Ascon-128 and Ascon-MCML

Parameters	Ascon-128	Ascon-MCML
Key	0x9f33f9d471da5f21aa596c9fad752f80	0x1cd0fb3d7fda909b722d97a22bde83c
Nones	0xbb267406435996777c68c60ab73264c	0x04fc3e1344e5f6c393ee40dd03100bd
Plaintext	0x48656c6c6f	0x48656c6c6f
Associated data	0x57656c636f6d65	0x57656c636f6d65
Ciphertext	0xe6c28814cb	0x3dbe3ff16e
Tag	0xbbd17d6039e7496150b58bd0eb1e1e4	0x5d3f71123cc177057defc9533b35e860
Processing Time	1.9173622131347656	1.7447471618652344
Code Size	7.8056640625	7.349609375

6. Conclusion

This paper is concerned with developing a novel S-box creation approach based on the CML system that is compatible with the LWC algorithms to achieve the security requirements of IoT devices. The creation method is based on the modified sine map as a function of CML, and uses the chaotic sequence produced by the MCML system to randomly mix the components of the resultant S-box, significantly enhancing its unpredictability. The bifurcation and Lyapunov exponent diagrams demonstrate that the MCML system has a very wide chaos period without any cutoff, so the MCML performance is more efficient than the original CML system. The findings of the performance experiment demonstrate the effectiveness and great suitability of the proposed S-box for LWC algorithms, which are based on 5-bit construction and have appropriate security with light performance. The investigational findings show that all S-box requirements were satisfied, including robust con-

frontation with differential and linear cryptanalysis in addition to randomness and the best NL value, which has a margin of 0.5 more than the original Ascon S-box, and an ideal value of completeness with fitting values of SAC, BIC-SAC, LAP, and DAP, which asserts improvement of diffusion and confusion for S-box. Accordingly, the MCML S-box superseded the substitution and linear diffusion layers of the Ascon set. From the application of the encryption and decryption processes, it shows that it has a smaller code size and faster processing time than Ascon-128 while maintaining the same security level of 128-bit as Ascon-128.

References

1. Atzori, L., Iera, A., Morabito, G.: The Internet of Things: A survey. Computer Networks. 54, 2787–2805 (2010). https://doi.org/10.1016/J.COMNET.2010.05.010
2. Ammar, M., Russello, G., Crispo, B.: Internet of Things: A survey on the security of IoT frameworks. Journal of Information Security and Applications. 38, 8–27 (2018). https://doi.org/10.1016/J.JISA.2017.11.002
3. Aziz, A., Singh, K.: Lightweight Security Scheme for Internet of Things. Wirel Pers Commun. 104, 577–593 (2019). https://doi.org/10.1007/S11277-018-6035-4/FIGURES/9
4. Panchami, V., Mathews, M.M.: A Substitution Box for Lightweight Ciphers to Secure Internet of Things. Journal of King Saud University - Computer and Information Sciences. 35, 75–89 (2023). https://doi.org/10.1016/J.JKSUCI.2023.03.004
5. Eisenbarth, T., Kumar, S., Paar, C., Poschmann, A., Uhsadel, L.: A survey of lightweight-cryptography implementations. IEEE Design and Test of Computers. 24, 522–533 (2007). https://doi.org/10.1109/MDT.2007.178
6. Dinu, D., Corre, Y. Le, Khovratovich, D., Perrin, L., Großschädl, J., Biryukov, A.: Triathlon of lightweight block ciphers for the Internet of things. J Cryptogr Eng. 9, 283–302 (2019). https://doi.org/10.1007/S13389-018-0193-X/TABLES/5
7. Rogaway, P.: Authenticated-encryption with associated-data. Proceedings of the ACM Conference on Computer and Communications Security. 98–107 (2002). https://doi.org/10.1145/586110.586125
8. Crypto competitions: CAESAR: Competition for Authenticated Encryption: Security, Applicability, and Robustness, https://competitions.cr.yp.to/caesar.html
9. NIST Selects 'Lightweight Cryptography' Algorithms to Protect Small Devices | NIST, https://www.nist.gov/news-events/news/2023/02/nist-selects-lightweight-cryptography-algorithms-protect-small-devices
10. Bertoni, G., Daemen, J., Peeters, M., Assche, G. Van: Duplexing the sponge: single pass authenticated encryption and other applications. Cryptology ePrint Archive. (2011)
11. Ascon – Authenticated Encryption and Hashing, https://ascon.iaik.tugraz.at/
12. Shannon, C.E.: Communication Theory of Secrecy Systems. Bell System Technical Journal. 28, 656–715 (1949). https://doi.org/10.1002/J.1538-7305.1949.TB00928.X
13. Prathiba, A., Bhaaskaran, V.S.K.: Lightweight S-Box Architecture for Secure Internet of Things. Information 2018, Vol. 9, Page 13. 9, 13 (2018). https://doi.org/10.3390/INFO9010013
14. Matthews, R.: On the derivation of a "chaotic" encryption algorithm. Cryptologia. 13, 29–42 (1989). https://doi.org/10.1080/0161-118991863745
15. Jakimoski, G., Kocarev, L.: Chaos and cryptography: Block encryption ciphers based on chaotic maps. IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications. 48, 163–169 (2001). https://doi.org/10.1109/81.904880
16. Tang, G., Liao, X., Chen, Y.: A novel method for designing S-boxes based on chaotic maps. Chaos Solitons Fractals. 23, 413–419 (2005). https://doi.org/10.1016/J.CHAOS.2004.04.023
17. Zhu, D., Tong, X., Wang, Z., Zhang, M.: A Novel Lightweight Block En-cryption Algorithm Based on Combined Chaotic System. Journal of In-formation Security and Applications. 69, 103289 (2022). https://doi.org/10.1016/J.JISA.2022.103289
18. Alshammari, B.M., Guesmi, R., Guesmi, T., Alsaif, H., Alzamil, A.: Implementing a Symmetric Lightweight Cryptosystem in Highly Constrained IoT Devices by Using a Chaotic S-Box. Symmetry 2021, Vol. 13, Page 129. 13, 129 (2021). https://doi.org/10.3390/SYM13010129
19. Mansouri, A., Wang, X.: A novel one-dimensional sine powered chaotic map and its application in a new image encryption scheme. Inf Sci (N Y). 520, 46–62 (2020). https://doi.org/10.1016/J.INS.2020.02.008
20. Alawida, M., Teh, J. Sen, Mehmood, A., Shoufan, A., Alshoura, W.H.: A chaos-based block cipher based on an enhanced logistic map and simultaneous confusion-diffusion operations. Journal of King Saud University - Computer and Information Sciences. 34, 8136–8151 (2022). https://doi.org/10.1016/J.JKSUCI.2022.07.025
21. Teh, J. Sen, Alawida, M., Sii, Y.C.: Implementation and practical problems of chaos-based cryptography revisited. Journal of Information Security and Applications. 50, 102421 (2020). https://doi.org/10.1016/J.JISA.2019.102421
22. Özkaynak, F.: Construction of robust substitution boxes based on chaotic systems. Neural Comput Appl. 31, 3317–3326 (2019). https://doi.org/10.1007/S00521-017-3287-Y/TABLES/5
23. Si, Y., Liu, H., Zhao, M.: Constructing keyed strong S-Box with higher nonlinearity based on 2D hyper chaotic map and algebraic operation. Integration. 88, 269–277 (2023). https://doi.org/10.1016/j.vlsi.2022.10.011
24. Zhu, H., Qi, W., Ge, J., Liu, Y.: Analyzing Devaney Chaos of a Sine–Cosine Compound Function System. https://doi.org/10.1142/S0218127418501766. 28, (2019). https://doi.org/10.1142/S0218127418501766

25. Hua, Z., Zhou, B., Zhou, Y.: Sine Chaotification Model for Enhancing Chaos and Its Hardware Implementation. IEEE Transactions on Industrial Electronics. 66, 1273–1284 (2019). https://doi.org/10.1109/TIE.2018.2833049
26. Belazi, A., El-Latif, A.A.A.: A simple yet efficient S-box method based on a chaotic sine map. Optik (Stuttg). 130, 1438–1444 (2017). https://doi.org/10.1016/J.IJLEO.2016.11.152
27. Dong, C., Rajagopal, K., He, S., Jafari, S., Sun, K.: Chaotification of Sine-series maps based on the internal perturbation model. Results Phys. 31, 105010 (2021). https://doi.org/10.1016/J.RINP.2021.105010
28. Demir, F.B., Tuncer, T., Kocamaz, A.F.: A chaotic optimization method based on logistic-sine map for numerical function optimization. Neural Comput Appl. 32, 14227–14239 (2020). https://doi.org/10.1007/S00521-020-04815-9/TABLES/4
29. Hua, Z., Jin, F., Xu, B., Huang, H.: 2D Logistic-Sine-coupling map for image encryption. Signal Processing. 149, 148–161 (2018). https://doi.org/10.1016/J.SIGPRO.2018.03.010
30. Thakor, V.A., Razzaque, M.A., Darji, A.D., Patel, A.R.: A novel 5-bit S-box design for lightweight cryptography algorithms. Journal of Information Security and Applications. 73, (2023). https://doi.org/10.1016/J.JISA.2023.103444
31. Kaneko, K.: Pattern dynamics in spatiotemporal chaos: Pattern selection, diffusion of defect and pattern competition intermettency. Physica D. 34, 1–41 (1989). https://doi.org/10.1016/0167-2789(89)90227-3
32. Kaneko, K.: Theory and Applications of Coupled Map Lattices. (1993)
33. Li, P., Li, Z., Halang, W.A., Chen, G.: A stream cipher based on a spatiotemporal chaotic system. Chaos Solitons Fractals. 32, 1867–1876 (2007). https://doi.org/10.1016/J.CHAOS.2005.12.021
34. Zhang, Y., Wang, X., Liu, L., Liu, J.: Fractional Order Spatiotemporal Chaos with Delay in Spatial Nonlinear Coupling. https://doi.org/10.1142/S0218127418500207. 28, (2018). https://doi.org/10.1142/S0218127418500207
35. Wang, X., Guan, N., Zhao, H., Wang, S., Zhang, Y.: A new image encryption scheme based on coupling map lattices with mixed multi-chaos. Scientific Reports 2020 10:1. 10, 1–15 (2020). https://doi.org/10.1038/s41598-020-66486-9
36. Xian, Y., Wang, X., Teng, L., Yan, X., Li, Q., Wang, X.: Cryptographic system based on double parameters fractal sorting vector and new spatiotemporal chaotic system. Inf Sci (N Y). 596, 304–320 (2022). https://doi.org/10.1016/J.INS.2022.03.025
37. Zia, U., McCartney, M., Scotney, B., Martinez, J., AbuTair, M., Memon, J., Sajjad, A.: Survey on image encryption techniques using chaotic maps in spatial, transform and spatiotemporal domains. Int J Inf Secur. 21, 917–935 (2022). https://doi.org/10.1007/S10207-022-00588-5/TABLES/5
38. Azam, N.A., Murtaza, G., Hayat, U.: A novel image encryption scheme based on elliptic curves and coupled map lattices. Optik (Stuttg). 274, (2023). https://doi.org/10.1016/j.ijleo.2023.170517
39. Dobraunig, C., Eichlseder, M., Mendel, F., Schläffer, M.: Ascon v1.2: Lightweight Authenticated Encryption and Hashing. Journal of Cryptology. 34, 1–42 (2021). https://doi.org/10.1007/S00145-021-09398-9/TABLES/21
40. Andreeva, E., Daemen, J., Mennink, B., Assche, G. Van: Security of Keyed Sponge Constructions Using a Modular Proof Approach. https://doi.org/10.1007/978-3-662-48116-5
41. Bertoni, G., Daemen, J., Peeters, M., Assche, G. Van: Sponge-Based Pseu-do-Random Number Generators.
42. Sönmez Turan, M., McKay, K., Chang, D., Çalık, Ç., Bassham, L., Kang, J., Kelsey, J.: Status Report on the Second Round of the NIST Lightweight Cryptography Standardization Process. (2021). https://doi.org/10.6028/NIST.IR.8369
43. Jovanovic, P., Luykx, A., Mennink, B.: Beyond 2^{c/2} Security in Sponge-Based Authenticated Encryption Modes. Cryptology ePrint Archive. (2014)
44. Adams, C., Tavares, S.: Good s-boxes are easy to find. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). 435 LNCS, 612–615 (1990). https://doi.org/10.1007/0-387-34805-0_56
45. Cusick, T.W., Stanica, P.: Cryptographic Boolean Functions and Applications. Cryptographic Boolean Functions and Applications. 1–232 (2009). https://doi.org/10.1016/B978-0-12-374890-4.X0001-8
46. Jia, Y.M., Li, C.J., Zhang, Y., Chen, J.: The high-order completeness analysis of the scaled boundary finite element method. Comput Methods Appl Mech Eng. 362, 112867 (2020). https://doi.org/10.1016/J.CMA.2020.112867
47. Williams, H.C. ed: Advances in Cryptology — CRYPTO '85 Proceedings. 218, (1986). https://doi.org/10.1007/3-540-39799-X
48. Matsui, M.: Linear cryptanalysis method for DES cipher. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). 765 LNCS, 386–397 (1994). https://doi.org/10.1007/3-540-48285-7_33/COVER
49. Biham, E., Shamir, A.: Differential cryptanalysis of DES-like cryptosystems. Journal of Cryptology. 4, 3–72 (1991). https://doi.org/10.1007/BF00630563
50. Dobraunig, C., Eichlseder, M., Mendel, F., Schläffer, M.: Cryptanalysis of Ascon. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). 9048, 371–387 (2015). https://doi.org/10.1007/978-3-319-16715-2_20/COVER
51. Andreeva, E., Bilgin, B., Bogdanov, A., Luykx, A., Mendel, F., Mennink, B., Mouha, N., Wang, Q., Yasuda, K.: PRIMATEs v1. competi-tions.cr.yp.to. (2014)