New Dynamical Key Dependent S-Box based on chaotic maps

F. J. Luma1, H. S. Hilal[1] and A. Ekhlas[2]

[1], Department of Computer Science, University of Technology,
P.O Box 35092, Baghdad, www.uotechnology.edu.iq
[2] Department of Computer Science, Mustansiriyah University, baghdad, Iraq,

Abstract: *The strength and security of cryptographic algorithms is determined by substitution non-linear S-boxes, so the construction of cryptographically strong S-boxes is important in the design of secure cryptosystems. In In this paper, an efficient method for designing dynamical key dependent S-boxes based on 2D logistic map and 2D cross map. S-box is a nonlinear transformation where each byte of the State is replaced by another byte using the substitution table. Each individual byte of State is mapped into a new byte by using 2D cat map. One S-box is used for each message. The aim of the proposed approach is to generate more secure S-boxes. The generated S-boxes will increase the complexity and has better results in security analysis. The new S-box is analyzed and tested for the following criteria: avalanche effect, strict avalanche effect, key sensitivity, differential and linear cryptanalysis. All the results have shown that the proposed method is a good candidate for designing dynamical S-boxes that can be widely used in block cipher.*

Keyword: *Dynamical S-Box, Block Cipher, Key-Dependence, 2D logistic map, 2D cross map, 2D cat map.*

I. Introduction

The S-box is the core component of most block ciphers, such as the Data Encryption Standard (DES), the Advanced Encryption Standard (AES) and so on. A strong block cipher should be resistant to various attacks, such as linear and differential cryptanalysis. This is generally achieved if the S-boxes used satisfy a number of criteria, such as the bijection, avalanche effect, the strict avalanche criteria (SAC) and so on. S-boxes are fixed and they used in SPN cipher systems as the important nonlinear component. The S-box used in encryption process could be chosen under the control of key, instead of being fixed.

According to the chaotic systems properties such as the ergodic, mixing and random-like behavior, it seems to be convenient and simple to obtain "good"
S-Boxes by modifying slightly the initial conditions or system parameters. Many approaches for obtaining S-Boxes based on chaos have been
presented:-

- Jakimoski and Kocarev, 2001 were presented the first chaotic S-boxes. They have proposed a four-step method to create Sboxes by using chaotic maps, which includes choosing a chaotic map, discretizing the chaotic map, key Two well-known chaotic maps (exponential and logistic) are used to generate the S-boxes, and suggested key-dependent chaotic S-boxes to enhance the security, in which the parameters and the number of iterations of the discretized exponential map were considered as the keys [1]. Their methodology includes N-th iteration of chaotic for each logistic map, where they choose N=1000.
- In 2005, Tang et al., proposed a method based on baker map. In this method, by iterating a chaotic logistic map, a 8-bit sequence of binary random variables is generated from a real value trajectory obtained and turn it to a decimal integer on the range of 0-2, then an integer table can be obtained. Then a key-dependent permuting is used to shuffle the table nonlinear by a Baker map [2].
- In [3], another method proposed by Tang and Liao that is based on discretized chaotic map. This method consists of: First, an integer sequence that can be regarded as secret key K, K= X0= {1, 2, ..., 2 }, is obtained in an arbitrary way. Second, for a given M = 2n and A, the chaotic map iterates more than k times with the initial value X_0, one can obtain a permuted integer sequence {X}. Finally, by translating the {X} to a 2n/2×2n/2 table, the S-box is obtained.
- In 2007, Chen et al., proposed another method for designing S-boxes based on three-dimensional baker map which has more intensive chaotic characters than the two-dimensional one [4].
- Muhammad Asim et al, 2007 propose method based on the mixing property of the piecewise linear chaotic map (PLCM). In this method, the output range [0.1, 0.9] is divided into 256 intervals of equal length. Label each region sequentially from 0 to m, where m is equal to 255. Iterate the PLCM using the selected initial condition.
- Whenever the PLCM visits a particular region, store that number in an array S [5].
- In 2009, R.Yin, propose a new key-dependent S-box based on the iteration of continuous logistic chaotic maps. Since S-boxes operate in the discrete state space, they making a mapping between these discrete

states and the continuous states of chaotic systems [6].

- J. Peng, S. Jin, L. Lei, R. Jia, 2012 propose a method for generating dynamical key-dependent S-boxes based on hyperchaotic Chen system . The secret key first is mapped to the initial condition and control parameter for the hyperchaotic system. Then, iterate the Chen system to generate a hyperchaotic sequence which is subsequently used to construct the S-box [7].
- Mona Dara and Kooroush Manochehri, 2013 propose a new method based on chaotic logistic map that uses cipher key to generate initial value. They use cipher key to generate initial value of logistic map (X_0) and outputs of logistic map as the values dynamic S-box [8].
- Cristian-Iulian Rîncu, Vasile-Gabriel IANA, 2014 combine simple chaotic maps, which are frequently used in chaos based cryptography.

They use three of the chaotic maps Logistic map, Tent map and Piece-Wise Linear Chaotic Map. These maps will be iterated starting from the initial condition x0, the output becoming the next initial condition. a number of iterations of the three chaotic maps are performed, equal to the value of the S -box dimension, in order to ensure the beginning of a normal chaotic operating mode. Then the iteration of the three chaotic dynamical systems is continuing and at each iteration the real value is kept of each system. The three real values are transformed into integer values [9]. In this paper, we propose new method based on the mixing property of the chaotic maps (2D logistic map, 2D cross map) to design dynamical dependent key S-box. One S-box is used to encrypt each message. Each byte is permuted using 2D cat map before substituted with the byte in the proposed S-box.

The remaining part of the paper is organized as follows: the chaotic functions analysis is given in Section 2. The description of the algorithm is given in Section 3. Section 4 presents the criteria for a ‘‘good n × n bit a cryptographically S-box. Section 5 presents the experiment results of the propose S-box.

II. Basic Theory.

In this paper we used three chaotic maps: 2D cross map, 2D logistic map and 2D cat map to construct the new S-box.

2.1 2D cross map.

$$\chi_{i+1} = 1 - \mu y^2$$
$$y_{i+1} = \cos(k.\cos^{-1}\chi_i), \chi, y \in [1.-1] \tag{1}$$

Where μ and k are the control parameters of the system, respectively. When μ=2 and k=6, this system exhibits a great variety of dynamics behavior 10][.

Manisha Raj1, Shelly Garg [11] proposed a new image encryption algorithm based on DNA sequence addition and five chaotic maps (Logistic Map, Cross Chaotic Map, Duffing map, Tinkerbell map, Gingerbreadman map). The chaotic maps are compared through the simulation results, histogram analysis and correlation analysis. They have found out that Cross Chaotic Map, showed best results than other chaotic maps. It is sensitive to the secret keys, it has larger key space, and it gives best encrypted image.

2.2 2D logistic map.

One of the most known and widely used chaotic systems is the 1 D Logistic map, which is defined as follows [11] :

$$f(x) = \mu\chi(1 - \chi) \qquad \chi \in (0,1) \tag{2}$$

where μ is the control parameter. The system is in chaos on condition that 3.569<μ<4.0.

The 2D logistic map is an extension of 1D logistic map. It increases the key space as well as the dependency on control parameters. In 2D logistic map, it is bit harder to guess the secret information. It also exhibits greater amount of chaotic behavior on the generation of sequence [12]. In overall, it increases the complexity of the algorithm.

$$f(x) = \begin{cases} \chi_{i+1} = \mu_1 \chi_i (1 - \chi_i) + \gamma_1 y_i^2 \\ y_{i+1} = \mu_2 y_i (1 - y_i) + \gamma_2 (\chi_i^2 + \chi_i y_i) \end{cases} \tag{3}$$

Where (xi,yi) is the pair-wise point at the ith iteration and μ1, μ2, μ2, γ1 and γ2 are the system parameters. When 2.75 <μ1<=3.4, 2.75 < μ2<=3.45, 0.15 <γ1<=0.21, 0.13 <γ2<=0. 15, the system is in chaotic state and can generate two chaotic sequences in the region (0, 1].

The 2D logistic map has a higher complexity compared to the conventional 1D logistic map. The complexities of the 1D and 2D logistic maps can be measured by using various means such Information Entropy and Lyapunov Exponent with respect to different pairs of initial values [13]. By this comparison they found that the 2D logistic map has a higher information entropy scores than 1D logistic map, which implies that its trajectory is more random-like. Meanwhile, the 2D logistic map also has a larger Lyapunov exponent than the 1D logistic map, which implies that the 2D logistic map is more dynamic.

2.3 2D Cat Map.

A 2D Cat map is first presented by V.I. Arnold in the research of ergodic theory. Let the coordinates of a positions P = {(x, y) | x, y = 1, 2, 3. . . N}, a 2D Cat map with two control parameters [14] is as follows:

$$\begin{cases} x_{i+1} = (x_i + a y_i) \bmod N \\ y_{i+1} = (b x_i + (ab + 1) y_i) \bmod N \end{cases} \tag{4}$$

Where, a, b are positive integers which are control parameters and (x’, y’) is the new position of the original pixel position (x, y) of N x N plain-image when cat map is applied once to the original. By replacing the position of the image pixel points with new coordinate, cat map permutes/shuffles the organization of pixels of plain-image. After several iterations, the correlation among the adjacent pixels is disturbed completely and the image appears distorted and meaningless.

Inverse transformation for deciphering is given as follows: -

$$\begin{cases} x_i = (x_{i+1}(ab + 1) - a y_{i+1}) \bmod N \\ \quad y_i = (y_{i+1} - b * x_{i+1}) \bmod N \end{cases} \tag{5}$$

III. Design Dynamical Key Dependent S-Box.

Substitution is a nonlinear transformation which performs confusion of bits. A nonlinear transformation is essential for every modern encryption algorithm and is proved to be a strong cryptographic primitive against linear and differential cryptanalysis. Nonlinear transformations are implemented as lookup tables (S-boxes). The central idea of the proposed S-box is based on the mixing property of chaotic nonlinear dynamical systems. The proposed S-box is a table of 16 × 16 integer values (256 bytes). The S-box is created by using 2d logistic map and cross map. The main idea of the proposed S-box consists of the following major steps:-

Step1:- the initial condition (x0 and y0) is input to the 2D cross map and 2D logistic map. These numbers are floating point numbers where the precision is 10^{-16} for each of x0 and y0, considered as the keys of the proposed S-box.

Step2:- Iterate the cross map and logistic map 100 times and ignore the results, in order to eliminate the transient effect of chaotic map.

Step3:- Iterate the cross map and logistic map one time. The two outputs of cross map(x_1 and y_1) are Xored with the two outputs of logistic map (x_2 and y_2) :-

$$\begin{cases} nx = x_1 \bigoplus x_2 \\ ny = y_1 \bigoplus y_2 \end{cases} \quad (6)$$

Step4:- the new outputs (nx and ny) are translated to integer numbers belong [0..255] by using the following equation:-

$$\begin{cases} n_1 = nx\ mod\ 256 \\ n_2 = \ ny\ mod\ 256 \end{cases} \quad (7)$$

Step5: the two integer numbers 1 2 are inserted to the S-box table while they are not found in the S-box table in order to avoid repeating the same numbers.
Step6:- repeat from step 2 until the S-box contains a permutation of all possible 256 byte values.

Table (1) shows an example of S-box created from the initial conditions (x0 = 0.2376589876543123 and y0= 0.9765432786554125).

S-box is a nonlinear transformation where each byte of the State is replaced by another byte using the substitution table. Each individual byte of State is mapped into a new byte in the following way (as shown in figure (1-a)): The leftmost 4 bits of the byte are used as a row value and the rightmost 4 bits are used as a column value. These row and column values are permuted to anther values for row and column by using the cat chaotic map eq(4).
The byte in the new row and column is then served as indexes into the S-box to select a unique byte output value.

For example, if S-box in figure 1 is used to substitute the byte (190) which
is 10111110 in binary, the leftmost 4 bits (R=1011 = 11=b) is the row value and the rightmost 4 bits(C=1110=14=e) is the column value.
This row and column values are diffuses using cat map eq(4) (with the
control parameters a =41and b=68) as follows;

$$\begin{cases} R' = (11 + 14 \times 41) mod\ 16 = 9 \\ C' = (11 \times 68 + 14 \times (41 \times 68 + 1) mod\ 16 = 2 \end{cases} \quad (8)$$

Finally, use the new row (9) and column (2) as index to find the substituted byte in the S-box (in table 1) which is the byte (102).

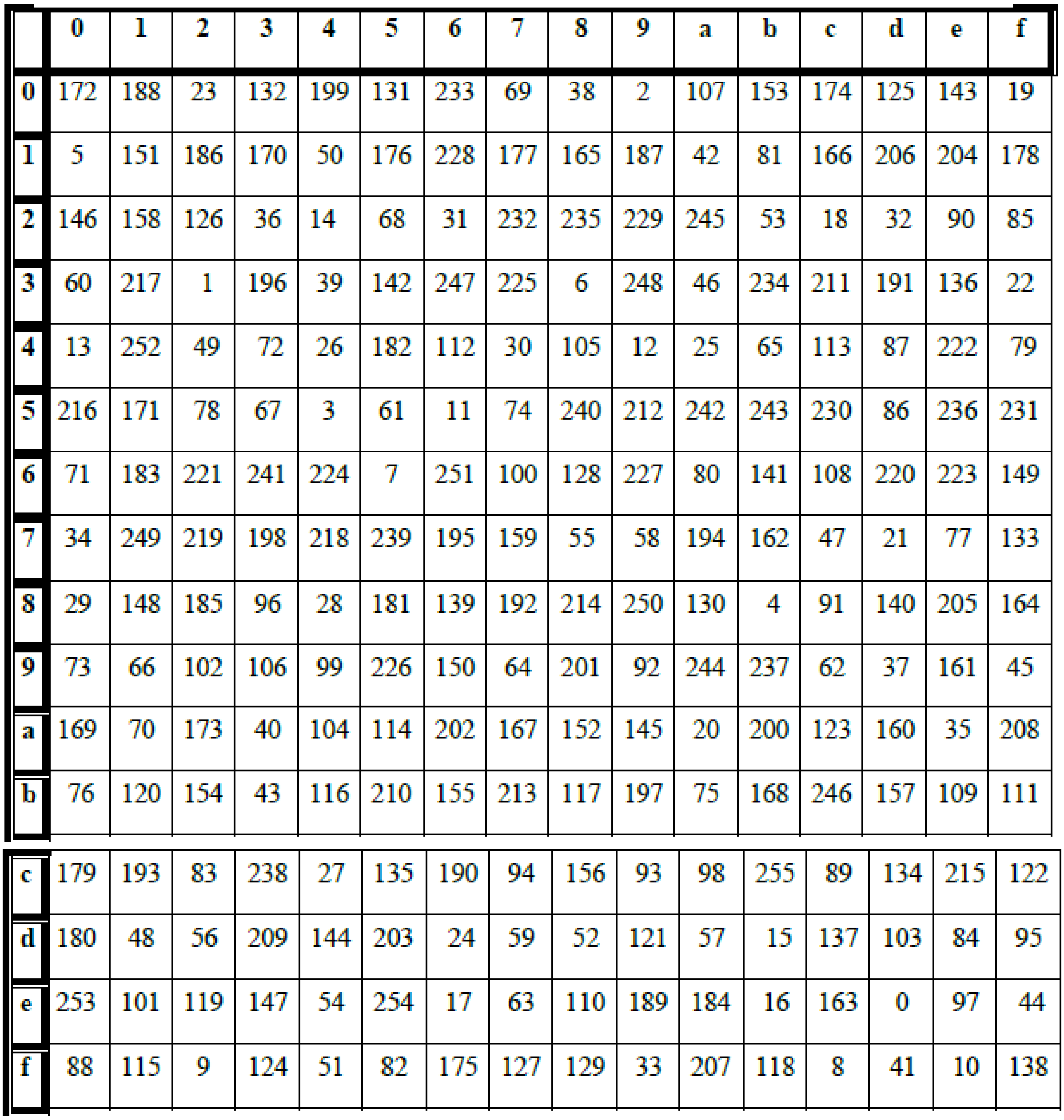
The inverse substitute byte transformation is performed in the following way (as shown in figure (1-b)): the byte is search in the S-box table and take its
row and column. These row and column values are return to their real value by using the inverse of cat map. Then the new row and column is
concatenate to make the original byte.

For example, the substituted byte (102) from previous example can be return to its original byte value in the following way: search the S-box table to find the byte (102) and take its row (9) and column (2). This row and column values input to inverse cat map to produce the original row and column as follows:

$$\begin{cases} R = (9 \times (41 \times 68 + 1) - 41 \times 2) mod\ 16 = 11 \\ C = (2 - 68 \times 9) mod\ 16 = 14 \end{cases} \quad (9)$$

So, the row (11) which is in binary (1011) and column (14) which is in binary (1110) are concatenating to produce the original byte
(10111110=190).

Table(1): S-box created from the 2d logistic map and cross map using the proposed procedure.



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

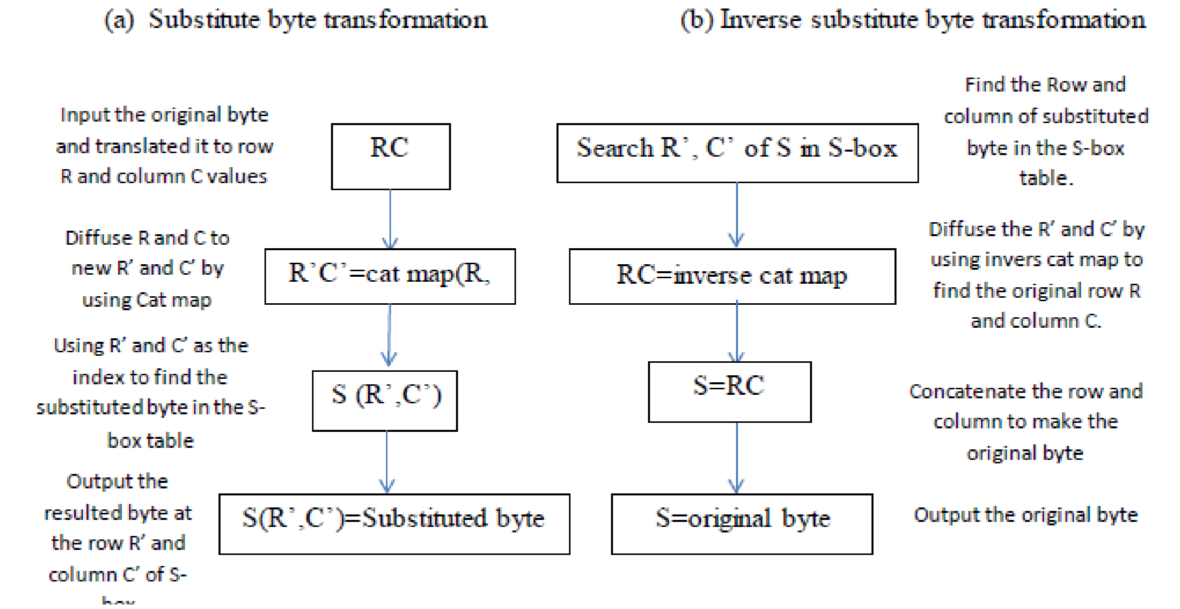

(a) Substitute byte transformation
(b) Inverse substitute byte transformation
Input the original byte and translated it to row R and column C values
RC
Diffuse R and C to new R' and C' by using Cat map
R'C'=cat map(R,
Using R' and C' as the index to find the substituted byte in the S-box table
S (R',C')
Output the resulted byte at the row R' and column C' of S-box
S(R',C')=Substituted byte
Search R', C' of S in S-box
Find the Row and column of substituted byte in the S-box table.
RC=inverse cat map
Diffuse the R' and C' by using invers cat map to find the original row R and column C.
S=RC
Concatenate the row and column to make the original byte
S=original byte
Output the original byte

Figure (1-a) : the substitute transformation of the byte value in S-box table and (1-b) the inverse substitute transformation of the byte in the S-box.

IV. Criteria for a ‘‘good n × n bit a cryptographically S-box.

The only nonlinear components in cryptosystems are the S -boxes. The differential cryptanalysis was introduced by Biham and Shamir for DES-like

cryptosystems [15]. After that Dawson and Tavares [16] expanded S-boxes' design criteria based on information theory and revealed how S-boxes

provided immunity to the differential attack. In general, some cryptographic properties are widely accepted as the essential properties for “good” S-boxes

and can be used to evaluate the S-box [18,17,8, 9, 6, 7,4] .

There are several properties which are cryptographically desirable in an s-box. They are:-

- The Bijective Property.
- The avalanche effect.
- The strict avalanche criteria (SAC).
- Differential attacks.
- Nonlinearity.
- Key sensitivity.

4.1 The Bijective Property.

An n x n bit s-box need to be a bijection that is, that every possible input

vector x maps to a unique output vector y. The bijective property is checked by using the method introduced in the literature [18, 19]. If the Boolean function f(x)=(f1,f2,….,fn) of an S-box such that: -

$$wt\left(\sum_{i=1}^{n} a_i f_i\right) = 2^{n-1} \qquad (8)$$

,where ai ϵ {0,1} , (a1,a2,…,an)=(0,0,…,0) and wt (·) is the Hamming weight (the Hamming weight is the number of "1" bits in the binary sequence, the S-box is bijective.

4.2 The avalanche effect.

Any secure component in a cryptographic algorithm has one of the key characteristic representatives which is the avalanche effect [9]. By presenting this characteristic, we can sure that half of the S-box output bits will be modified when complementing a single bit in the input vector. These characteristics are tested according to the avalanche vectors by using [9]:

$$Av_i = S(x) \oplus S(x_i) \quad (9)$$

Where the difference between x and xi lies only in i bit. When the complementing of any unique input bit determined the modification of N/2 output bits on average, the avalanche effect is fulfilled.

4.3 The strict avalanche criteria (SAC).

The strict avalanche criterion (SAC) was introduced by A.F. Webster and Tavares in 1985.SAC is satisfied when single input bit in an S-box is changed, each output bit should change with a probability of one half. The SAC of an S-box is tested by constructing the dependence matrix. If each element and the mean value of the matrix are both close to the ideal value 0.5, the S-box is considered as nearly fulfills the SAC. The dependence matrix should be calculated According to [20] by following those steps: -

1. Let x a random vector of an n-bits and its m-bits ciphertext y, y=s(x).
2. A set of n-vectors (x_l, x_2, ... ,x_n), such that x and xj differ only in one bit j, for all j 1≤ j ≤ n . Another set of vectors (y_1, y_2, ... ,y_n) is created such that $y_j=s(x_j)$.
3. The avalanche vectors (v ,v ,..,v) is calculated by:

$$\mathbf{v_j} = \mathbf{y} \oplus \mathbf{y_j} \qquad (10)$$

4. The bit i of v_j is added to element a_{ij} of m x n dependence matrix A.
5. Repeat these steps for k random vector x.
6. Each element of the matrix A is divided by k.

The value of each element in A (aij) varies from 0 to 1 which gives an idea about plaintext bit j and ciphertext bit i. If aij is close to 1 then indicate that if bit j is complemented then bit i should change its value. If all element of the matrix A have a value close to 0.5, then the S-box satisfies the SAC criterion.

4.4 Differential attacks.

Differential cryptanalysis was introduced the first time by Biham and Shamir in [15]. The S-box should have the differential uniformity to resist the differential cryptanalysis. This mean the nonlinear transformation S-box should ideally have differential uniformity. The differential approximation probability of a given S-box, DPs, is a measure for differential uniformity and is defined as [17]: -

$$DPs = \max_{\Delta x \neq 0, \Delta y} \left(\frac{\#\{x \in \frac{X}{s(x)} \oplus s(x \oplus \Delta x) = \Delta y\}}{2^n} \right) \qquad (11)$$

Where where X is the set of all possible input values, elements, Δx is the input differential and Δy is the intruder first calculates differential pairs (Δx, Δy) forΔy input differential Δx and the output differential 2n is the number of its output differential. The each$_{=s(\ x}$S$_{)}$-$_{\oplus}$box$_{s(x}$where$_{\oplus\Delta}$ the$_{x),}$then find the result of DPs.

This equation means that the DPs is the maximum probability of the output differential Δy when the nonzero input differential is Δx. Ideally, a nonlinear S-Box should have a differential uniformity: an input differential Δx is mapped to a unique output differential Δy.

4.5 Nonlinearity.

Linear cryptanalysis studies linear probability approximations of the cryptosystem. The S-box must resistant to this type of attack. The goal is to construct linear equations between input plaintext and output cipher text and enumerate all linear approximations of the S-Box in a linear approximation table. Otherwise, the nonlinearity is measured by calculating probability approximation (LPs) [17]: -

$$\mathrm{LPs} = \max_{a,b\neq 0}\left(\frac{\#\{x\in X \,|\, x\cdot a = s(x)\cdot b\} - 2^{n-1}}{2^{n-1}}\right)^2 \qquad (12)$$

Where a, b ϵ {I, 2, .., 2n-1}, x.a is the parity of the binary product of x and a. To increase linear attack complexity, the value of LPs must be decreased.

4.6 Key sensitivity.

Sensitivity analysis is the study of how the uncertainty in the output of a model can be apportioned to different sources of uncertainty in the model input [21]. An essential factor for the S-Box is the sensitivity on the key. In other words, a small changing in the keys should cause a large change in the S-Box. This means that a small difference on key values, the created S-boxes should be completely uncorrelated.

To ensure the sensitivity of the key, the analysis is done using Pearson's correlation coefficients. Consider a pair of S-boxes given by: S1 = [x1, . . . , xN] and S2 = [y1, . . . , yN]. Therefore, the corresponding correlation coefficient is:

$$C_{s1,s2} = \frac{\sum_{i=0}^{N-1}(x_i-\bar{x}).(y_i-\bar{y})}{[\sum_{i=0}^{N-1}(x_i-\bar{x})^{\,2}]^{\frac{1}{2}}.[\sum_{i=0}^{N-1}(y_i-\bar{y})^{\,2}]^{\frac{1}{2}}}$$

Where the mean values of S_1 and S_2 are:

$$\bar{x} = \sum_{i=0}^{N-1} x_i/N$$

And

$$\bar{y} = \sum_{i=0}^{N-1} y_i/N$$

To ensure the sensitivity of the key, the analysis is done using Pearson's
value of CS1, S2 is to ±1, the stronger the correlation between the two S-boxes. In the case of two independent S-boxes, the value of CS1,S2 is equal to0.

5. Experiment results.

In this section, we construct 100 dynamical key dependent S-Box from nearby or successive keys and the Criteria for good S-box results are as
follows: -

1. **The Bijective Property: -**All the S-boxes are bijective. The value of all generated S-box is 128, which is the same as the ideal value. This mean that in each S-box every possible input vector x maps to a unique output vector y.

2. **The avalanche effect: -** we test the avalanche effect for the S-box in table (1). Figure (2) shows the frequency of the number of the modified bits belonging to the output values obtained after changing each of the input N positions for all of the S-box elements in table(1). Because the complementing of any unique input bit determined the modification of N/2 (8/2=4) output bits on average, the avalanche effect is considered fulfilled. We found that all the 100 created S-boxes are fulfilled the avalanche effect.

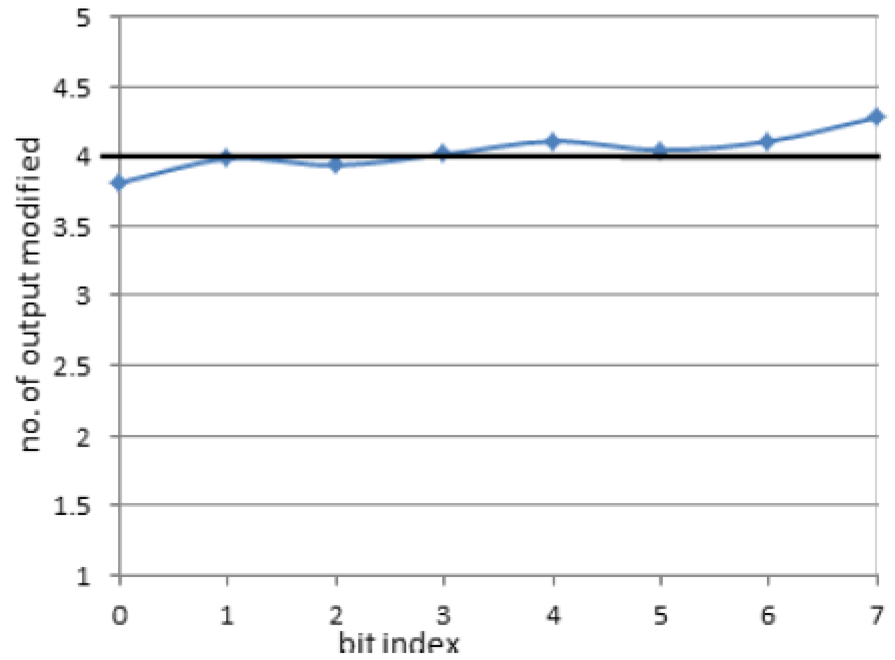


Figure (2): The results of the avalanche effect test.

Figure (2): The results of the avalanche effect test.

3. The strict avalanche criteria (SAC):- the dependence matrix for the S-box in table (1), with 8 input bits, it is not difficult to compute a complete dependence matrix containing the elements for all the 2N input vectors that can appear, is calculated and shown in table (2). The mean value is 0.51342 which is close to ideal value (0.5). All the mean values of the dependence matrixes of 100 S-boxes are located within [0.46, 0.53], which are also close to the ideal value 0.5, indicating that all the S-boxes have excellent SAC property.

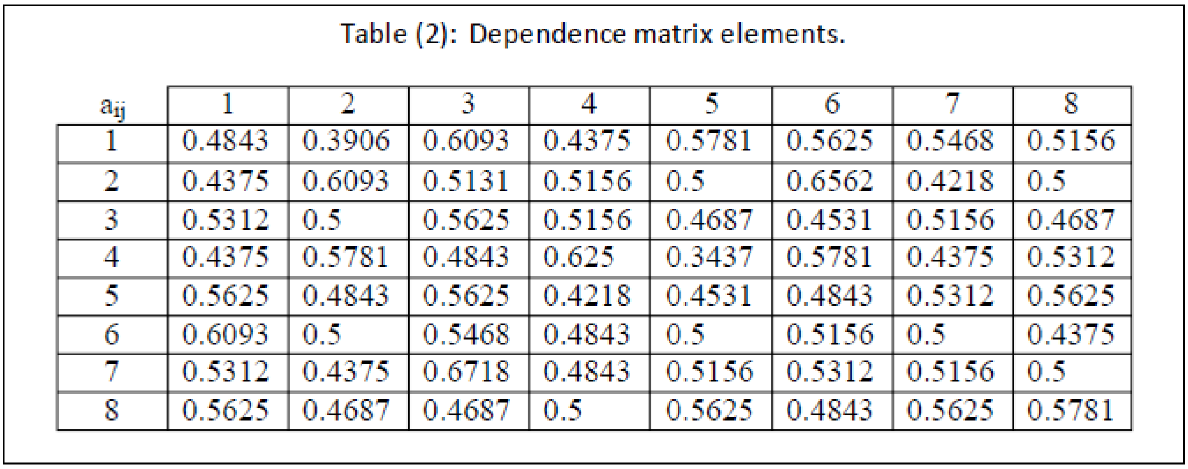


Table (2): Dependence matrix elements.

a_{ij}	1	2	3	4	5	6	7	8
1	0.4843	0.3906	0.6093	0.4375	0.5781	0.5625	0.5468	0.5156
2	0.4375	0.6093	0.5131	0.5156	0.5	0.6562	0.4218	0.5
3	0.5312	0.5	0.5625	0.5156	0.4687	0.4531	0.5156	0.4687
4	0.4375	0.5781	0.4843	0.625	0.3437	0.5781	0.4375	0.5312
5	0.5625	0.4843	0.5625	0.4218	0.4531	0.4843	0.5312	0.5625
6	0.6093	0.5	0.5468	0.4843	0.5	0.5156	0.5	0.4375
7	0.5312	0.4375	0.6718	0.4843	0.5156	0.5312	0.5156	0.5
8	0.5625	0.4687	0.4687	0.5	0.5625	0.4843	0.5625	0.5781

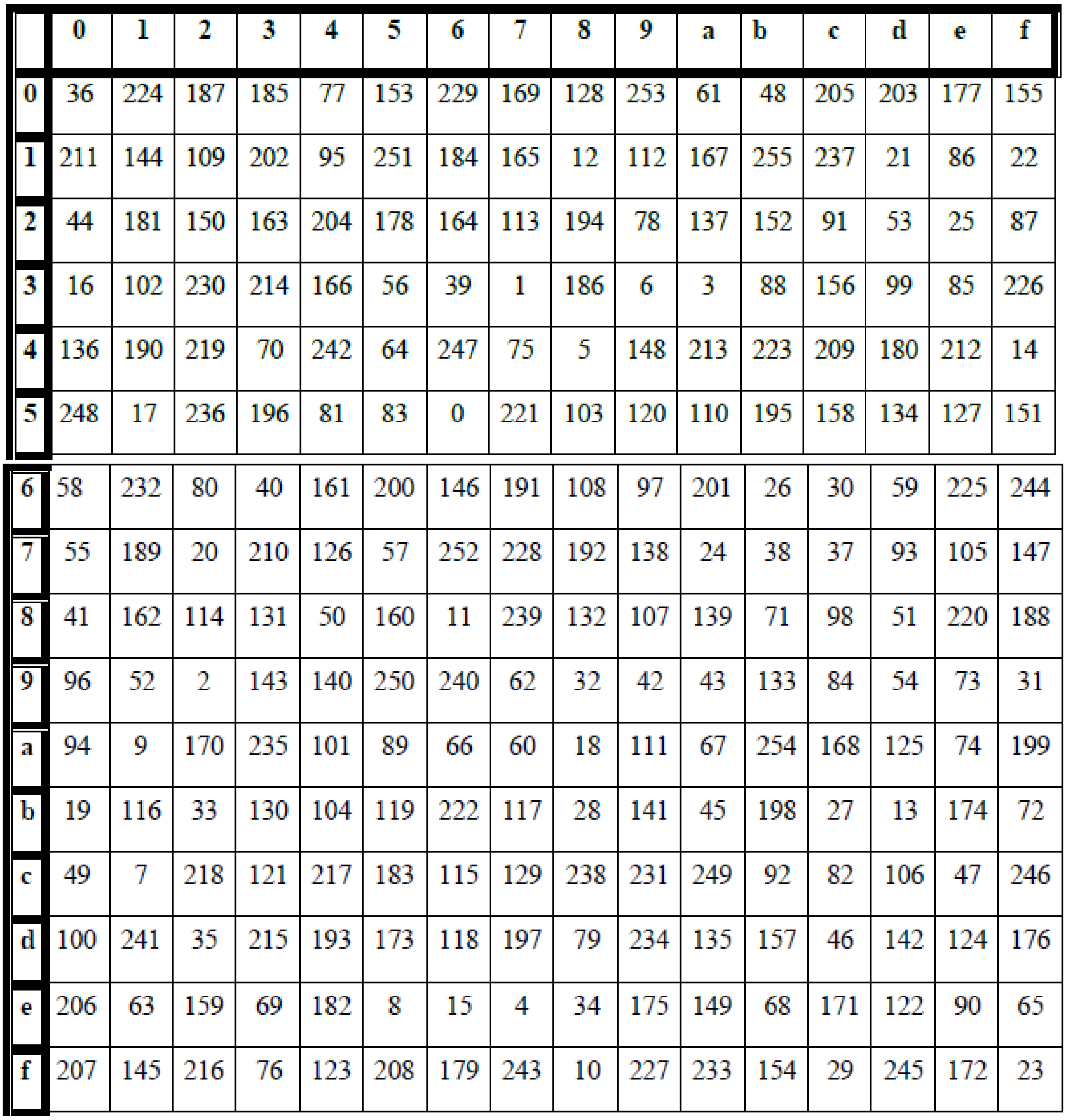
4. Differential attacks: - The DPs of the S-Box in table (1) is calculated and the result is 12/256 which is close to the ideal value (2^{-6}≤ DPAES-SBox≤ 2^{-4}). All the DPs of 100 S-boxes are located within [10/256, 12/256], which are also close to the ideal value, indicating that all the S-boxes have good immunity against differential attacks.

5. Nonlinearity: - The linear approximation probability of the S-Box in table (1) is evaluated to LPs=0.07910. Linear approximation probability of statistic AES S-Box varies from 2^{-2} to 2^{-3}, so decreasing LPs, leads to increasing linear attack complexity. All the LPs of 100 S-Box are calculated and the average of LPs is 0.0760089. This means that all the S-boxes can resistant to this type of attack.

6. Key sensitivity: - In order to investigate the S-box's sensitivity to the secret keys, the Pearson's correlation coefficients between the S-boxes in table (1) and table (3) with the key1 and key2 are calculated. In here, key1 is set to
x_0 = 0.2376589876543123 and y_0= 0.9765432786554125 key2 is set to
x_0 = 0.237658987654312**4** and y_0= 0.976543278655412**6**

Where the key1 is slightly changed to key2. The S-boxes generated by key1 and key2 are shown in table (1) and table (2), respectively, and the correlation coefficient between them -0.0199495, indicating that the S-boxes are very sensitive to the keys.

Table (2): S-box created from the 2d logistic map and cross map using the proposed procedure.



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

V. Conclusions

The S-box is the core component of most block ciphers. In this paper, new
method for creating dynamical key-dependent S-boxes based on 2D logistic map and 2D cross map is presented. The results show that generated S- Boxes have very low linear and differential probabilities and satisfy also the avalanche criterion and strict avalanche criterion (SAC). In addition, the S-box's sensitivity to the keys is investigated by the correlation coefficient. The algorithm used to generate S-boxes and the selected chaotic maps ensure a simple and quick implementation using simple digital computing systems. Because the proposed S-box is satisfied all the criteria of good S-box, we can used for designing block cipher with dynamical S-boxes.

References

[1]. Jakimoski G, Kocarev L. "Chaos and cryptography: block encryption ciphers". IEEE Trans Circ Syst—I 2001;48(2):163–9 boxes based on chaotic maps, Chaos Solitons Fractals, 23(2): 413-419.

[2]. Tang, G., X. Liao and Y. Chen, 2005. "A novel method cryptography for designing S-boxes based on chaotic maps", Chaos 40(1): 505-519. Solitons Fractals, 23(2): 413-419.

[3]. Tang, G. and X. Liao, 2005. "A method for dynamical S-boxes based on discretized chaotic map", Chaos Solitons Fractals, 23(5): 1901-1909.

[4]. Chen, G., Y. Chen and X. Liao, 2007. "An extended method for obtaining S-boxes based on three- dimensional chaotic baker maps", Chaos Solitons Fractals, 31(3): 571-579.

[5]. Muhammad Asim et al. "Efficient and Simple Method for Designing Chaotic S-Boxes", ETRI Journal, Volume 30, Number 1, February 2008.

[6]. Ruming Yin, Jian Yuan, Jian Wang, Xiuming Shan, Xiqin Wang, "Designing key-dependent chaotic S-box with larger key space", Chaos, Solitons and Fractals 42 (2009) 2582–2589.

[7]. J. Peng, S. Jin, L. Lei, R. Jia, "A Novel Method for Designing Dynamical Key-Dependent S-Boxes based on Hyperchaotic System", International Journal of Advancements in Computing Technology(IJACT)

[8]. Volume4, Number18,October. 2012, doi:10.4156/ijact.vol4.issue18.33.

[9]. Mona Dara and 2Kooroush Manochehri, "A Novel Method for Designing S-Boxes Based on Chaotic Logistic Maps Using Cipher Key", World Applied Sciences Journal 28 (12): 2003-2009, 2013 ISSN 1818-4952.

[10]. Cristian-Iulian Rîncu, Vasile-Gabriel IANA, "S-Box Design Based on Chaotic Maps Combination", 978-1-4799-2385-4/14/$31.00 ©2014 IEEE.

[11]. Ling Wang, Quen Ye Yaoqiang, Yongxing zou , Bo Zang, "An Image Encryption Scheme based on cross chaotic map" 2008 IEEE.

[12]. Manisha Raj , Shelly Garg , " An Innovative Approach: Image Encryption with Chaotic Maps using DNA Addition Operation", International Association of Scientific Innovation and Research (IASIR), IJSWS 14-337; © 2014, IJSWS All Rights Reserved.

[13]. Musheer Ahmad and M. Shamsher Alam, "A New Algorithm of Encryption and Decryption of Images Using Chaotic Mapping", Musheer

[14]. Ahmad et al /International Journal on Computer Science and Engineering, Vol.2(1), 2009, 46-50.

[15]. Yue Wu, Gelan Yang, Huixia Jin and Joseph P. Noonan, "Image Encryption using the Two-dimensional Logistic Chaotic Map",

[16]. Li S, Mou X and Cai Y, "Pseudo random bit generator based on couple chaotic systems and its application in stream cipher cryptography", Lecture notes in computer science, 2247, 2001, 316-329, Springer-Verlag, Berlin.

[17]. Biham E, Shamir A. "Differential cryptanalysis of DES-like cryptosystems". J Cryptol 1991; 4(1): 3 – 72.

[18]. Dawson M, Tavares SE. "An expanded set of S-box design criteria based on information theory and its relation to differential- like attacks". In: Advances in cryptology: Proc of Eurocrypt 91. New York: Springer- Verlag; 1991. pp. 352–67.

[19]. Ghada Zaibi, Abdennaceur Kachouri, Fabrice Peyrard and Daniele Foumier-Prunaret. "On Dynamic chaotic S-BOX", 978-1-4244-4624-7/09/$25.00 ©2009 IEEE.

[20]. Carlisle Adams and Stafford Tavares and Stafford. "The Structured Design of Cryptographically Good S-Boxes". Journal of Cryptology (1990) 3:27-41 International Association for Cryptologic Research.

[21]. Adamas C, Tavares S. "Good S-boxes are easy to find". In: Advances in cryptology: Proc. of CRYPTO 89. In: Lecture notes in computer science; 1989. pp. 612–5.

[22]. A. F. Webster and S. E. Tavares, "On the Design of S-Boxes", in Advances in Cryptology: Proc. of CRYPTO '85, Springer-Verlag , Berlin,1986, pp. 523-534.

[23]. Ascough II, J.C., T.R. Green, L. Ma, and L.R. Ahjua. "Key Criteria and Selection of Sensitivity Analysis Methods Applied to Natural Resource Models". 1USDA-ARS, Great Plains Systems Research Unit, Fort Collins, CO 80526.