Journal of Theoretical and Applied Information Technology
15th December 2020. Vol.98. No 23

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

KEY DEPENDENT DYNAMIC S-BOXES BASED ON 3D CELLULAR AUTOMATA FOR BLOCK CIPHER

[1]SHARIFAH MD YASIN, [2]AYMAN MAJID HAMID, [3]NUR IZURA UDZIR, [4]SHERZOD TURAEV, [5]MUHAMMAD REZAL KAMEL ARIFFIN

[1]Senior Lecturer, Faculty Computer Science and Information Technology, Universiti Putra Malaysia, 43400 Serdang, Selangor, Malaysia

[1]Research Associate, Institute for Mathematical Research (INSPEM), Universiti Putra Malaysia, 43400 Serdang, Selangor, Malaysia

[2]Lecturer, Department Computer Science, Almustansiriah University, Baghdad, Iraq

[3]Associate Professor, Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, 43400 Serdang, Selangor, Malaysia

[4]Associate Professor, Department Computer Science and Software Engineering, United Arab Emirates University, United Arab Emirates

[5]Associate Professor, Faculty Science, Universiti Putra Malaysia, 43400 Serdang, Selangor, Malaysia

E-mail: [1]ifah@upm.edu.my, [2]aymanmajid@uomustansiriyah.edu.iq

ABSTRACT

Substitution boxes (S-Boxes) are critical components of numerous block ciphers deployed for nonlinear transformation in the cipher process where the nonlinearity provides important protection against linear and differential cryptanalysis. Classical S-Boxes are represented by predefine fixed table structures which are either use for Data Encryption Standard (DES) or Advanced Encryption Standard (AES). Based on cryptanalysis, it does not offer sufficient cipher protections. The S-boxes used in encryption process could be chosen to be key-dependent. For secure communication, we need a better design of S-boxes to be used for encryption and decryption. In this paper we proposed key dependent dynamic 3D cellular automata (CA) S-Boxes for block ciphers. Our work is based on the design of AES S-Boxes which are originally in 2D presentation. The conceptual framework of the 3D CA S-Boxes is to convert and apply the 3D CA rule to static AES S-Boxes. The methodology is to do conversion from the AES S-Boxes into 3D array of (8x8x4) S-boxes, and then applies the 3D CA Von Neumann rules to them. After a 3D array is obtained from the AES S-Box, the 3D CA is applied based on the round key. The 3D array S-Box are then converted back to the 2D array S-Box and finally it is improved to meet the requirements of good S-Boxes. The obtained S-Boxes is called key dependent dynamic 3D CA S-Boxes having interesting features with dynamic stretchy arrangement, which is functionally understood by CA. Our proposed 3D CA S-boxes are better in comparison with the AES S-Boxes with predefined fixed table structures. Experimental results shown that the proposed 3D CA S-Boxes have secure characteristics like nonlinearity, SAC, BIC and algebraic degree. The proposed S-Boxes can be implemented in any block cipher for secure communication.

Keywords: *Secure Communication, Block Cipher, Cellular Automata, Dynamic S-Boxes, Key Dependent*

1. INTRODUCTION

In the recent years, there has been a lot of effort to improve cryptographic techniques for securing communication and storing information. Cryptography in the past was used by military for protecting national security and diplomatic correspondents. Today modern communication requires cryptography to be deployed for privacy and security. It is a fundamental aspect of a secure communication transform readable information into unreadable form and vice versa using encryption and decryption techniques. Cryptographic techniques are widely used in secure

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

communication as it ensures and provides authentication, confidentiality, integrity and non-repudiation.

Advanced Encryption Standard (AES) is a block cipher which is an encryption system and is widely used for secure communication. AES uses fixed input block size of 128 bits, and a key size of 128 bits. The strength of block ciphers, like AES heavily depends on the construction of S-Boxes which must satisfy some essential properties to develop a secure cryptosystem which could resist algebraic attacks.

In this paper, we propose a key dependent (KD) dynamic 3D cellular automata (CA) S-Boxes based on Von Neumann rules for a block cipher. The essential nonlinear element of block ciphers (*DES*, Serpent, *GOST*, Blowfish, *AES* etc.) are the S-Boxes. S-Boxes are essentially lookup tables that substitute input bits with output bits [36]. Various studies have attempted to modify the AES S-Boxes to make it dynamic instead of static, in order to increase the security of the block cipher.

The outline of this paper as follows: Section II presents related works, and then followed by the methodology employed in Section III. The original S-Boxes, the functionalities of the S-boxes, and together with the properties of good S-Boxes is discussed in Section IV. In section V, the proposed design of new 3D CA S-Boxes are described. Section VI presents the illustration of the dynamic 3D CA S-Boxes and finally the last section concludes this paper.

2. RELATED WORK

Advanced Encryption Standards (AES) is a cryptographic technique being used to secure data. One of its components is the Substitution box (S-box) which is the only nonlinear part of AES and thus it is prone to attacks. An n × m S-box maps n input bits into m output bits, where n is not necessarily equal to m, nevertheless many ciphers deploys m = n where the number of input and output sizes are the same. When n = m, the S-Box is bijective, that is, each value in the output appears exactly once. However, a large dimension of n leads to larger lookup table which incurs cost in terms of time. Additionally, the design of block ciphers must adhere to Shannon's confusion and diffusion principles. Usually, S-boxes are the only nonlinear part of the cipher and they are used to provide confusion.

2.1 Substitution Boxes (S-Boxes)

In lightweight block ciphers such as PRESENT, smaller S-boxes with size 4×4 are mostly employed, and 8×8 S-boxes are found in the AES cipher [42]. The design of S-Boxes with good cryptographic properties is necessary for ciphers that utilize S-Boxes. There exists a clear trade-off between the choices of S-box sizes and properties. The larger S-box can be troublesome to implement in constrained environments. Consequently, a smaller and optimal S-boxes has low implementation costs and preferable.

Based on the NIST standard, good S-boxes should have criteria like bit independence criterion (BIC), nonlinearity (NL), strict avalanche criterion (SAC), linear probability (LP), and differential probability (DP). Generally, S-box with low DP value and high non-linearity is desirable in which the non-linearity of an S-box causes uncertainty in the output, which offers resistance against linear and differential cryptanalysis attacks [21]. An S-box satisfies the strict avalanche Criterion (SAC), if and only if for any single input bit of the S-boxes produces output bit with probability of one half.

Static S-Boxes in AES is the main attraction for the cryptanalyst to analyse. A weak S-box allows certain attacks onto the block cipher. The nonlinearity of the AES S-Boxes provides confusion properties for the AES algorithm. Currently, many algorithms have been proposed to modify the static nature of the AES S-Boxes. In the literature, it was found that arbitrary unknown, key-dependent substitution and permutation transformations is a good factor in enhancing the resistance of a block cipher against linear and differential cryptanalysis. Traditionally, static S-Boxes will be utilized in every individual round. Numerous analysts have experimented and have revealed that there is some shortcoming or weak points in the configuration and construction of the current static S-Box since it is static and known to all. Dynamic S-boxes have features where their values are not fixed and they give better result compare to a static nature.

Mariot [32] described cryptographic properties of S-boxes based on cryptanalytic attacks: Balancedness, Algebraic degree, Nonlinearity, Differential uniformity, and Resiliency. The linear cryptanalysis is a known plaintext attack invented by Matsui [34] which exploits the linear relationship between input and output of a cipher to discover cipher key bits. Differential cryptanalysis is a chosen plaintext attack invented by Biham and Shamir [9] in which an attacker analyse the plaintext and ciphertext pairs in order to discover the key bits.

The design of S-Boxes can roughly follow three directions, namely, algebraic constructions, random

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

search, and heuristics [40], and it is also possible to employ design strategies that represent various combinations of the aforementioned techniques. In [41], an S-Boxes of size 8×8 is generated with better resistance against side-channel attacks (SCA) as measured with the transparency order property. In [15], GA is used to generate S-Boxes with better SCA resilience for hardware and software implementation.

In [19], affine transformation is necessary in AES S-Boxes design to increase the complexity of the encryption process. AES S-box has cryptographic properties of 8×8 size with high algebraic degree of 7, high nonlinearity of 112, low autocorrelation of 32 and low differential uniformity of 4.

In [51] proposed the construction of dynamic S-Boxes by using chaotic maps, which enhances the security of the block ciphers based on the combination of two chaotic maps: one and three dimensional piecewise linear maps. The NIST test package is used to verify the randomness of those maps. The results from the security analysis displays that the dynamic S-Boxes based on two chaotic maps has resistance against linear and differential cryptanalysis. In [18] proposed dynamic S-Boxes based on 1D chaotic maps and can used in the AES algorithm.

In [23] proposed an algorithm which generates S-Boxes based on one-dimensional chaotic map (logistic and PWLCM). The generated S-Boxes are as good as AES S-Boxes with good nonlinearity and SAC properties. In [6] proposed a modified S-box which has gone through series of performance tests and passed the bijectivity, balance test, nonlinearity test, BIC and SAC.

The evaluation criteria on the security of an S-box design follows [28][46][6]: balance, bijective, nonlinearity, bit independence criterion (BIC) and strict avalanche criterion (SAC).

In [39] proposed an evolution of algebraic constructions that can be used to generate Boolean functions. It is proven that the approach is highly successful and is independent of the Boolean function size one needs to find. S-box Evaluation Tool namely SET can be used to evaluate the Boolean functions. A strong S-box is a critical part for a secured cryptosystem [49].

2.2 Key Dependent (KD) Dynamic S-Boxes

In the design of key dependent (KD) dynamic S-Boxes, the key is used in the creation of S-Boxes to provide dynamic nature. In [52] proposed KD S-Boxes and eight KD P-Boxes with efficient byte wise operation for fast speed implementation the internal structure of this algorithm resistant against linear and differential cryptanalysis. In [24] modified the KD S-Boxes generation algorithm and proposed a fast algorithm that generate KD S-Boxes. The NIST statistical test was applied to check the randomness of the generated S-Boxes. Results show that the KD S-Boxes provide algorithms that is resistant against algebraic attacks and the algebraic properties of the S-Boxes are as good as that of the AES S-Boxes.

In [16] used the RC4 algorithm with KD dynamic S-Boxes in which all the values of S-Box are dependent on input key and if any byte of input key is changed then different 256 values were generated and 256! S-Boxes can be generated using this method. In [25] proposed KD dynamic S-boxes that changes for every change of the secret key. The S-Boxes are resistant from both linear and differential cryptanalysis. They propose an approach of generating the random S-boxes by ensuring a dynamic change of the secret key. They have proven that the KD S-boxes are robust and unknown, therefore they are resistant from linear and differential cryptanalysis. In [13] used Chaotic Logistic Map to generate key dependent dynamic S-boxes for block ciphers.

In [27] proposed 128-bit AES-KDS block cipher with KD dynamic S-boxes which have 5 stages instead of 4 stages in a round of AES. On the encryption side the extra stage that is rotate S-Box added on the top of existing stages which rotates the elements of S-Box on the basis of round key. The algorithm used four cases to provide different level of security as shown in Table 1.

In [22] proposed a KD dynamic S-Boxes based on determinant matrix in which the dynamic S-Box changing in every round with reference to the expanded round key and the number of working rounds. The S-boxes provide better and robust security features for a block cipher. Their results shown that the proposed scheme consume less time and they are against linear and differential attacks.

In [29] proposed using LFSR (Linear Feedback Shift Register) to generate KD dynamic S-Boxes. This proposed algorithm leads to increase complexity and makes linear and differential cryptanalysis more difficult. They simulated the proposed scheme and the original AES with a fixed key using MATLAB and made comparison analysis. Their results show that the proposed scheme has better security than AES. In [12] proposed the use of a Chaotic Logistic Map to generate KD S-boxes for AES. The proposed S-boxes were analysed and tested for the avalanche effect, strict avalanche effect, bit independency criterion, nonlinearity, input/output XOR

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

distribution and key sensitivity. The S-boxes successfully passed all the criteria.

Table 1: Design Of Key Dependent Dynamic S-Boxes [27]

Case	Security Level	Method
1	moderate	S-Box rotation is based on only one byte of the round key
2	High	S-Box values are rotated on the bases of the whole round key
3	High	creating two subset of the round keys from key expansion algorithm in which one set of keys generated are used to find the value on which the values of the S-Box are rotated and the other set of keys are used to find the key for add round key operation
4	High	S-Box values are dependent on the whole key of the key generated from the set of keys of the set one.

In [1] proposed a KD dynamic S-Boxes by adding a secret value to the static index in order to shift the substitution to a secret location. They tested the proposed algorithm based on correlation coefficient (BIC) and strict avalanche criteria (SAC). The algorithm is against linear and differential attacks and it passes SAC and BIC. In [13] proposed a KD dynamic S-Box using RC4 algorithm. If any byte of the input key is changed, then 256 distinct values were generated. The 256 S-Boxes could be generated instantly.

The dynamic properties of the S-Boxes have advantages such as being unknown to the cryptanalyst. The properties support the block cipher to be resistance against attacks [2][3][4].

In [31] proposed an algorithm that generates S-Boxes using a pseudorandom number generator. The generated S-Boxes possess good linear and differential properties, better hamming weights with better avalanche effect and balanced output. In [7] proposed an algorithm that generates KD S-Boxes. The proposed S-Boxes are tested for nonlinearity, an XOR profile. The result shows that the S-Boxes are as good as the AES S-Box.

In [30] proposed KD dynamic S-Boxes using Chaos-based Rotational Matrices with strong cryptographic properties. They used different S-boxes in every round of AES, and therefore, it will be more difficult for an attacker to investigate each S-box, thus the cipher is resistant from linear and differential attacks. Table 2 shows the related review of KD dynamic S-Boxes.

Table 2: Review on Key Dependent (KD) Dynamic S-Boxes

	Reference	Proposed Method	Remarks
1.	[52]	-KD S-Boxes and eight key-dependent P-Boxes	resistance against linear and differential cryptanalysis
2.	[16]	-Use RC4 algorithm to generate KD dynamic S-Boxes	resistance against linear and differential cryptanalysis
3.	[22]	- KD dynamic S-boxes based on determinant matrix	resistance against linear and differential cryptanalysis
4.	[24]	-Modified KD S-Boxes generation algorithm -generate key dependant S-Boxes.	resistance against linear and differential cryptanalysis
5.	[29]	-Use LFSR (Linear Feedback Shift Register) to generate KD dynamic S-Boxes.	resistance against linear and differential cryptanalysis
6.	[12]	-Use Chaotic Logistic Map to generate KD S-boxes for AES.	Passed all criteria of a good S-Boxes
7.	[1]	-KD dynamic S-Boxes by adding a secret value to the static index in order to shift the substitution to a secret location	resistance against linear and differential cryptanalysis
8.	[13]	-KD dynamic S-Box with RC4 algorithm.	
9.	[31]	-Generates S-Boxes using the Pseudo-Random generator	S-Boxes possess good linear and differential properties.
10.	[7]	-KD S-Boxes	The S-Boxes are as good as the AES S-Boxes.
11.	[30]	-KD dynamic S-Boxes using Chaos-based Rotational Matrices	resistance against linear and differential cryptanalysis

From Table 2, we have analyzed thirteen papers based on key dependent S-Boxes. From the table, we can see that most researchers proposed key dependent S-boxes and their block ciphers are resistant from linear and differential cryptanalysis.

2.3 Cellular Automata (CA)

Cellular automata (CA) was introduced by Ulam [47] and Von Neumann [48]. Cellular Automata is a grid of cells that changes their state synchronously, according to a local update rule that specifies the new state of each cell based on the old states and its neighbors gives the global change of CA as shown in Figure 1.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

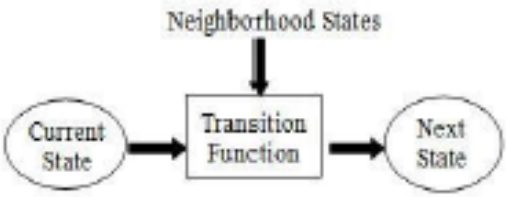

Figure 1: Current State, Transition Function, Next State and Neighbourhood States in CA (Kishore et al, 2011)

Figure 2 shows the grid cells for 1D CA, 2D CA and 3D CA. Cellular automata (CA) is a discrete model, which either arranged on a line or a grid of cells in which each cell exists in finite state i.e. either 0 or 1. Additionally, each cell in the CA applies a local rule over its neighbours in order to update its state and ensure the CA can exhibit very complex dynamical behaviours.

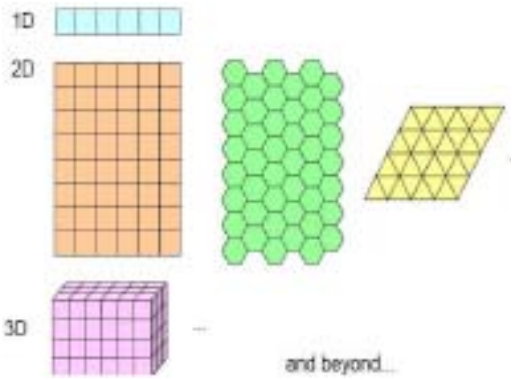

Figure 2: 1D CA, 2D CA and 3D CA (Kishore et al, 2011)

CA can be model as a quadruple set of {D, K, N. f} where D is the dimension of CA which can be 1D, 2D, 3D, etc; K is the set of all possible states of all cells in a CA; N is the neighbourhood states such as Von Neumann or Moore neighbourhoods; F is the transition function.

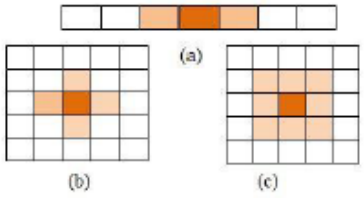

Figure 3: Neighbourhood in CA: (a) 3neighborhood, (b) Von Neumann Neighbourhood, and (c) Moore Neighbourhood (Kishore et al, 2011)

Figure 3 shows the type of neighbourhood in CA. In Figure 3(a), the 1D CA has two neighbours and then there is $2^{(2^{\wedge}3)}= 2^8 = 256$ rules can be generated.

Wolfram [50] firstly introduced 1D CA with Rule 30 for a pseudorandom number generator. Chowdhury and Subbaro [10] improved the 1D CA into an m x n 2D CA with restricted vertical neighbourhood. In [20] proposed the first block cipher based on CA which focused on the iterated behaviour of the CA. In [11] proposed CA with rule χ which is featured in a real-world cipher namely Keccak, which is also a part of NIST SHA-3 standard [8].

The 2D CA consists of 2D lattice of finite CA in which the global state of the CA evolves through local transitions. In [35] proposed a reconfigurable cryptosystem based on 2D CA with Von Neumann rule for image encryption using method of replacing the image pixel values by XORing them with a 2D CA key. They highlighted two good reasons for using the CA rule: Firstly, the number of CA evolution rules is very large and complex; Secondly, the recursive CA substitution requires integer arithmetic and/or logic operations which is easier to implement in hardware. Therefore, they used 2D CA to generate a good quality random number as key stream. Based on security analysis using statistical approach, the proposed scheme shows high quality encrypted images than the one produced from AES, which is due to the high quality of the key stream generated by the CA generator.

In [14] proposed a Programmable Cellular Automata (PCA) based on the elementary CA, and it is used to provide real time keys for a block cipher. Comparative analysis of the proposed scheme and AES are carried out for different key sizes ranging from 128, 192, and 256 bits. The results have shown that the proposed scheme has better execution time by 21%, 21%, and 19.6% respectively according to the key sizes.

In [45] proposed a combination of steganography and encryption scheme using 2D CA to secure message transmission through a network. Firstly, steganography is used for hiding the message to be sent behind an image. Secondly, the image obtained from previous step is encrypted with 2D CA using Moore neighbourhood model. Moore neighbourhood model has 9 cells at a time including the cell itself. The value of current cell (i.e. central cell) depends on its 8 neighbours. Table 3 shows the 2D CA with 9 cells. The rules used in the proposed method are Rule 2, Rule 8, Rule 32 and Rule 128. Their findings show that the transmitted message is much more secure when

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

compared to a simple encryption method. The proposed algorithm can be used for parallel processing of texts.

Table 3: 8 Neighbourhood CA rules

64	128	256
32	1	2
16	8	4

In [26] the PRNG based on CA produces high-rate random numbers and it shows better performance over the LFSR and other PRNGs.

In [44] proposed a CA based cryptosystem with high quality of randomness pattern for sensory data in wireless sensor network. The proposed scheme is resistant to brute force attacks and linear cryptanalysis attacks. It also uses less memory space in wireless sensor network.

In [38][40][41] used Genetic Programming to evolve CA rules that define S-boxes and their approach is able to generate large number of rules to produce S-boxes having optimal cryptographic properties, and with low implementation cost. Their technique can also be applied for larger S-boxes sizes. They highlighted that S-boxes with the largest possible nonlinearity are preferable for block ciphers in order to avoid linear attack. Also, the differential uniformity of an S-box needs to be as low as possible to avoid differential attacks. Another cryptographic criterion, the algebraic degree of an S-box should be as high as possible in order to thwart higher-order differential attacks.

In [33] proposed a 1D CA based S-Boxes which results in an extremely lightweight definition of the S-box with small implementation cost, and yields suboptimal cryptographic properties. Their design is an extension of [42] and satisfies the minimum set of criteria that includes bijectivity, high nonlinearity, and low differential uniformity. In [32] stated that CA based cryptosystem satisfies the principles of confusion and diffusion and it can be efficiently implemented in hardware with constrained computational resources. However, a general cryptographic analysis of CA is still missing in the literature.

In [5] proposed a 3D CA block cipher algorithms and they highlighted that the 3D CA rules offer better encryption technique than the 2D CA one, and the security of transmitted data is improved. The scheme has better confusion properties, the relation between key and cipher text is maximally complicated. In [43] proposed a secure 3D CA cryptosystem that has immunity to a brute-force attack. The cryptanalyst needs to search a huge search space to discover a possible set of rules, initial key, a discrete time-step, initial configuration, and boundary conditions. Thus, the 3D CA rules have a very useful feature due to their inherent computational complexity. The proposed scheme is also resistant from differential cryptanalysis.

The 3D CA are extensions of the 2D CA, and it requires checking neighbour cells in the X, Y, and Z directions. Neighbourhoods in CA refers to which cells around each cell influences its birth, survival and death. There are two type of neighbourhoods: 3D Moore and 3D Von Neumann. The first one extends to 26 possible neighbours (think of a Rubik's cube with the middle of the cube as the current cell), or consider a 3x3x3 3D grid of little cubes, in which the interior cube is the current cell, so the remaining 26 cubes around it are the neighbours of the centre cube. The latter one uses only neighbour cells sharing a face with current cell which gives the 6 cells in the +/- X, Y and Z axis direction from each cell, that is thinking of a 3D "plus sign" or cross shape.

In [26] proposed a PRNG based on 3D CA which is based on rule 43, 85, 170, and 201, with incremental boundary conditions. The 3D cell has three different dimensions with each dimension consists of eight bits. If all the neighbouring cells participate in the updating process then the complexity of the system exponentially increases. Koikara proposed the following three different updating mechanisms (X) which are chosen randomly whenever a cell needs to be updated:

$$X_{x,y}(t) = C_1 \oplus C_2 \oplus L_{x,y} \oplus R_{x,y} \oplus U_{x,y} \oplus D_{x,y}$$

$$X_{y,z}(t) = C_1 \oplus C_2 \oplus L_{y,z} \oplus R_{y,z} \oplus I_{y,z} \oplus O_{y,z}$$

$$X_{z,x}(t) = C_1 \oplus C_2 \oplus L_{z,x} \oplus R_{z,x} \oplus I_{z,x} \oplus O_{z,x}$$

where, C_1 and C_2 are the clock bits, and L, R, U, D, I, and O, represent the neighbouring cells in the left, right, up, down, in-page and out-page directions. In one cycle, each cell in the 3D structure passes through rule number 43, 85, 170, and 201 as shown in Table 4.

Table 4: 3D CA Rules for PRNG [26]

Rule#	C_1	C_2	L	R	U	D	I	O
43	0	0	1	0	1	0	1	1
85	0	1	0	1	0	1	0	1
170	1	0	1	0	1	0	1	0
201	1	1	0	0	1	0	0	1

In [26] proposed PRNG based on 3D CA using C++ language and compared with the PRNG based on 2D CA. She experimented the rule from Table 3 for 200 times and take the average results. The proposed 3D CA scheme is more secure and

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

random than the 2D CA, also, its complexity is almost analogous to the PRNG based on 2D CA, and suitable for real time applications.

In [24] proposed four algorithms for the generation of key-dependent S-boxes. They claimed the S-boxes are preferable for AES because they are more secure. In [17] proposed key-dependent S-boxes for AES.

3. METHODOLOGY

In this research, we propose a key dependent dynamic S-Box design and algorithms which have important features of 3D CA with Von Neumann rules. Our work is based on the AES S-Boxes. We convert the S-Boxes into a 3D design using 3D CA rules.

Figure 5 shows the methodology of our research which consists of four phases: Requirement analysis, Design of dynamic S-Boxes, Conversion of S-Box to 3D S-Box, and Testing of proposed S-Boxes. In Phase 1, the requirements of a secure S-Boxes are analysed, then in Phase 2, we design dynamic S-Boxes, in Phase 3, the S-boxes are converted into 3D CA dynamic S-Boxes and finally, in Phase 4 we test the proposed S-Boxes.

Phase 1: Requirements Analysis

This phase consists of analysis of the static AES S-Box. We study the properties of a good cryptographic S-Boxes and identify the requirements to ensure good S-Boxes are designed.

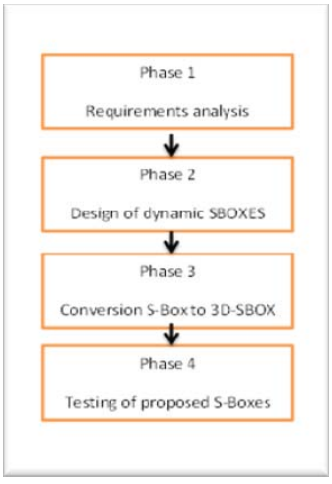

Figure 4 Methodology of 3D CA Key Dependent Dynamic S-Boxes

A. *Design of S-Boxes*

S-Boxes provides nonlinear transformation which supports the transformation of bits. It is essential for any encryption algorithm, to ensure that the S-Boxes are resistant against cryptanalytic attacks. The S-Boxes are invertible and implemented as lookup tables. The AES S-Boxes are presented as a 16x16 table of values as specified in Figure 5. It comprises of an arrangement of all possible 256 8-bit values. Each distinct byte of state array is mapped to a new byte value. For example, the hexadecimal value {95} in Figure 5 refers to the horizontal line number 6 and vertical column F of the S-Box, which holds the value {2A}, {72} to {40}.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
4	09	83	2C	1A	1B	6E	5A	A0	3B	52	D6	B3	29	E3	2F	84
5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

Figure 5: Static AES S-Box

Figure 6 shows the static AES inverse S-Box presented as a 16x16 lookup table. The properties of good S-Boxes have been widely used as a base for establishing new encryption strategies. These include nonlinearity, differential uniformity, and strict avalanche criterion.

Journal of Theoretical and Applied Information Technology
15th December 2020. Vol.98. No 23

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	52	09	6A	D5	30	36	A5	38	BF	40	A3	9E	81	F3	D7	FB
1	7C	E3	39	82	9B	2F	FF	87	34	8E	43	44	C4	DE	E9	CB
2	54	7B	94	32	A6	C2	23	3D	EE	4C	95	0B	42	FA	C3	4E
3	08	2E	A1	66	28	D9	24	B2	76	5B	A2	49	6D	8B	D1	25
4	72	F8	F6	64	86	68	98	16	D4	A4	5C	CC	5D	65	B6	92
5	6C	70	48	50	FD	ED	B9	DA	5E	15	46	57	A7	8D	9D	84
6	90	D8	AB	00	8C	BC	D3	0A	F7	E4	58	05	B8	B3	45	06
7	D0	2C	1E	8F	CA	3F	0F	02	C1	AF	BD	03	01	13	8A	6B
8	3A	91	11	41	4F	67	DC	EA	97	F2	CF	CE	F0	B4	E6	73
9	96	AC	74	22	E7	AD	35	85	E2	F9	37	E8	1C	75	DF	6E
A	47	F1	1A	71	1D	29	C5	89	6F	B7	62	0E	AA	18	BE	1B
B	FC	56	3E	4B	C6	D2	79	20	9A	DB	C0	FE	78	CD	5A	F4
C	1F	DD	A8	33	88	07	C7	31	B1	12	10	59	27	80	EC	5F
D	60	51	7F	A9	19	B5	4A	0D	2D	E5	7A	9F	93	C9	9C	EF
E	A0	E0	3B	4D	AE	2A	F5	B0	C8	EB	BB	3C	83	53	99	61
F	17	2B	04	7E	BA	77	D6	26	E1	69	14	63	55	21	0C	7D

Figure 6: Inverse of Static AES S-Box

1- Properties of good S-Boxes

The properties of good S-Boxes have been widely used as a base for establishing of new encryption strategies. These include nonlinearity, differential uniformity, and strict avalanche criterion [53].

Given (x, y)-S-box is a map, and

$S : \{0,1\}^x \rightarrow \{0,1\}^y$.

It comprises of n-variable component of Boolean functions:

$(f_1(x_1, ..., x_n), f_2(x_1, ..., x_n), ..., f_n(x_1, ..., x_n))$

each of which need to satisfy the S-Box properties.

The following are the list of several properties in S-box:

i) Robustness

Let $F = (f_1, f_2, \ldots, f_n)$ be an n×n S-box, where f_i is a component function of S-Box mapping $f_i : \{0,1\}n \rightarrow \{0,1\}$

$$r = (1 - \frac{N}{2^n})(1 - \frac{L}{2^n})$$

F must be Robust to against differential cryptanalysis [37].

ii) Balancing

$S: \{0,1\}^n \rightarrow \{0,1\}^m$ is balanced,

if $HW(f) = 2^{n-1}$.

The significance of the balance property is evaluated based on the higher the magnitude of function imbalance where a high probability linear approximation is obtained.

iii) Strict Avalanche Criterion (SAC)

SAC is defined as a change in one bit of input bits of S-Box should produce a change in half of the output bits of S-Box. It is difficult for the attacker to perform an analysis of cipher text [53].

iv) Nonlinearity

$S : \{0,1\}^x \rightarrow \{0,1\}^y$ is defined as the least value of nonlinearity of all nonzero linear combinations of x boolean functions.

$f_i : \{0,1\} \rightarrow \{0,1\}, i = x - 1, \ldots, 1, 0$.

The nonlinearity of an S-Box must be high to resist any linear cryptanalysis.

v) Differential Uniformity

The smaller the value of Differential Uniformity, the S-Box's resistant against differential cryptanalysis is much better.

vi) Linear Approximation

The lower the Linear Approximation value, the better is the S-Box's resistance against any linear cryptanalysis.

vii) Algebraic Complexity

The Algebraic Complexity is important to resist any interpolation attack and other attacks that are algebraic in nature.

viii) Fixed (Fp) and Opposite Fixed Points (OFp)

The number of Fp and OFp should be kept as low as possible to avoid any leakages in statistic cryptanalysis.

ix) Bit Independence Criterion

The bit independence is as highly desirable property. With increasing independence between bits, it becomes more difficult to understand and predict the design of the system.

Phase 2: Design of Dynamic S-Boxes

The proposed S-Boxes are designed using 3D CA Von Neumann rules. The rules are applied into the static AES S-Box. In this research, we proposed eight S-Boxes and the look-up table to construct eight different S-Boxes are based on 3 bits (i.e. $b_5b_6b_7$) which are selected from the round key of each round key.

Table 5 depicts that the names of rules in 3D CA Von Neumann (Left, Right, Up, Down, Back and Infrontof rules).

Table 5: Select the 3D S-Boxes Rules

b_7	b_6	b_5	Name of 3D rules
0	0	0	3D-Left-Up-Infrontof
0	0	1	3D-Left-Up-Back
0	1	0	3D-Left-Down-Infrontof
0	1	1	3D-Left-Down-Back
1	0	0	3D-Right-Up-Infrontof
1	0	1	3D-Right-Up-Back
1	1	0	3D-Right-Down-Infrontof
1	1	1	3D-Right-Down-Back

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Figure 7 shows the available 3D CA Von Neumann rules applied to 3D S-Box.

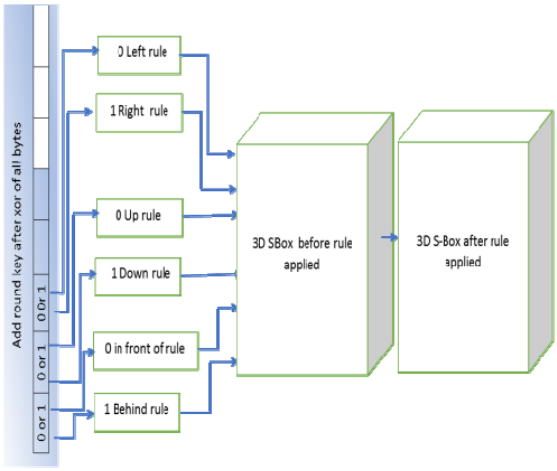

Figure 7 Selection 3D CA of One Round

The design of the proposed S-Boxes starts with conversion of the static AES S-Boxes into 3D array S-Boxes. Then, the 3D CA rules are applied onto the 3D array to construct new S-Boxes. The essence of applying Von Neumann rules is because it is easy to implement and it is reversible. Figure 8 illustrates the Von Neumann rules apply to 3D S-Box.

Table 5 shows the rules applicable to the 3D array S-Box based on the round key. The rules in Table 5 are applied to the 3D S-Boxes based on the 3-bit values obtained from the round key. If the value of b_7 is 0 then 3D-Left rule is applied and if its value is 1 then 3D-Right rule is applied. Simultaneously, for the value of b_6 is 0 then the 3D-UP rule is applied, if its value is 1 then the rule 3D-Down applied with the result stored as 3D array S-Boxes. With the value of b_5 is 0 then the 3D Infrontof rules is applied and the 3D-Back rule is applied when this bit value is 1.

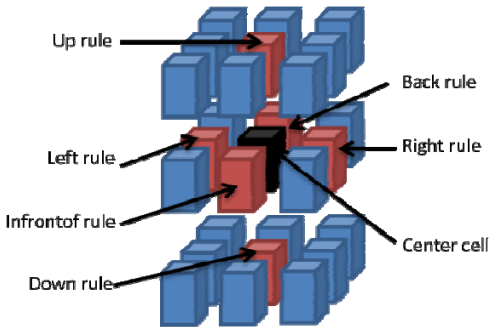

Figure 8: Von Neumann 3D Neighbourhoods

Figure 7 shows selection of the rules. We can see that we can construct different eight 3D S-Boxes depending on the values of 3 bits taken from the round key for each round. Based on Figure 8, the 3D Von Neumann rules are as follows:

1- 3D-Left rule: All the cells in the 3D S-Box shifts one cell to the left, while the last cell shift to the first cell.
2- 3D-Right rule: The reversal of the 3D-Left rule. In this rule all bytes in the 3D S-Box array shifts one cell to the right and the first cell shift circularly to the end of the row.
3- 3D-Up rule: All the bytes in the 3D S-Box shift one cell up and the cells on top shifts to the last cell on that column.
4- 3D-Down rule: this rule is the reversal of the 3D-Up rule and it works by shifting all the bytes of the 3D S-Box one cell down, and the cell down shifts to the top of the column.
5- 3D-Infrontof rule: All the cells in the 3D S-Box shifts one cell to the infront of layer-1 and the cells in the first layer shifts to the last layer.
6- 3D-Back rule: The reversal of the 3D-Infrontof rule and it is applied to all the cells in the 3D S-Box and shifts one cell to the back and the cells on the last layer (k=3) shifts to the beginning level where (k=0).

Phase 3: Conversion S-box to 3D S-Boxes

Construction of the 3D CA S-Boxes follow the steps below:

1) Take the original Rijndeal 2D S-Box and convert this array of bytes into 3D array of S-Boxes.
2) Apply 3D Von Neumann rules obtained from the round key bytes to the 3D array.
3) Convert the result of 3D-CA S-Boxes after applying the 3D rules into 2D-array of dynamic S-Box.
4) Enhance the new dynamic array to satisfy the properties of good S-Boxes.
5) Construct the inverse 3D-CA S-Box for each of the newly created dynamic S-Boxes
6) Store the results of the dynamic 3D S-Box into the arrays and these arrays will become ready for use in place of the static S-Boxes.

Journal of Theoretical and Applied Information Technology
15th December 2020. Vol.98. No 23

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A. *Conversion from 2D AES S-Boxes into 3D- array S-Boxes*

The 2D Rijndael S-Box contain array of bytes with 16 columns and 16 rows each cell of this array contains unique byte from 00 to FF in hexadecimal format. To convert this array into 3D array, we declare a new array called 3D-Array of 8 rows and 8 columns with 4 layers 3D-Array [8,8,4]. This is as illustrated in Figure 9.

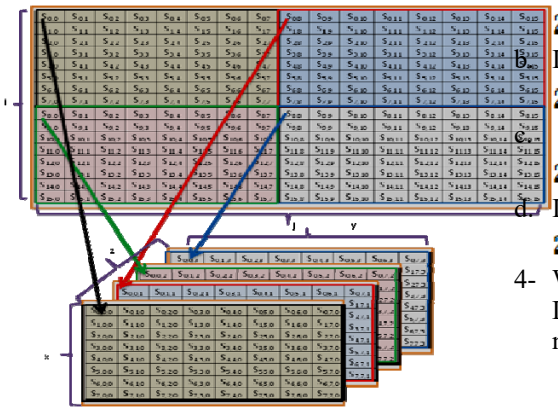

Figure 9: Conversion from 2D to 3D Array

To achieve a 3D array, the standards S-Box is divided into 4 equal parts. Each part contains 2D array of 8 rows and 8 columns. Let the index of 2D array be denoted as i for x-axis and j for y-axis.

1- If the value of i and j is less than 8 then we put this value in the 3D array of index $i, j, 0$.

$$3DSBOX[i, j, 0] = 2DSBOX[i, j] \quad (1)$$

2- Else if the value of i is greater than 8 and value of j is less than 8 then we put this value of S-Box in the 3D array of [i-8, j, 1].

$$3DSBOX[i-8, j, 1] = 2DSBOX[i, j] \quad (2)$$

3- Else if the value of i is less than 8 and value of j is more than 8 then we put the value in the third layer (k=2).

$$3DSBOX[i, j-8, 2] = 2DSBOX[i, j] \quad (3)$$

4- Finally, if both of i and j are more than 8 then we put the value of S-Box value in the last layer where (k=3).

$$3DSBOX[i-8, j-8, k] = 2DSBOX[i, j] \quad (4)$$

B. *Conversion from 3D S-Boxes arrays into dynamic 2D S-Boxes arrays.*

After applying 3D-rules to the 3D S-Boxes, the result obtained is stored and return back to 2D S-Boxes to make it look like standard look-up-table. Figure 10 illustrates the processes of convertor 3D S-Box into 2D S-Box. The following steps were taken:

1- For each dynamic 3D S-Box for instance 3D-Left-Up-Infrontof rule for the value of b5-b6 is 000.

2- Let consider a dimension with 8 for rows, 8 columns and 4 depths.

3- We read every cell in the 3D S-Boxes and identify the depth value.

a. If the value is 0 then we have

$$2DSBOXLUI[i, j] = 3DSBOXLUI[i, j, 0]$$

b. If the value is 1 then we have

$$2DSBOXLUI[i+8, j] = 3DSBOXLUI[i, j, 1]$$

c. If the value is 2 then we have

$$2DSBOXLUI[i, j+8] = 3DSBOXLUI[i, j, 2]$$

d. If the value is 3 then we have

$$2DSBOXLUI[i+8, j+8] = 3DSBOXLUI[i, j, 3]$$

4- We repeat this process for all the other 8 Dynamic S-boxes and store the result with the name of the rule used.

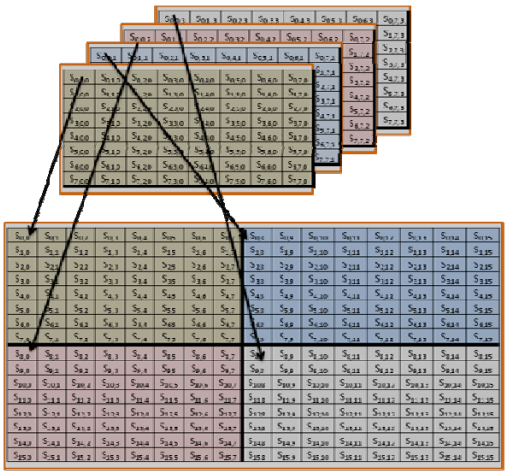

Figure 10: Conversion from 3D-Array to 2D Array of S-Box

C. *Enhanced the dynamic 3D S-Boxes*

To achieve good results for S-Boxes we try all possible rules and we found that our good results are similar to the standard S-Boxes provided if we take double rules for top and bottom and double rules for left and right. In this case, the values of i and j will be multipled by 2 to get good result. In this case, we can generate 63 new S-Boxes where all of them have the same property and as the standard AES S-Box and in our work, we choose only 8 dynamic 3D-CA S-Boxes. Apply the same steps of this dynamic 3D S-Box to all the eight S-Boxes generated and store the result into arrays.

Journal of Theoretical and Applied Information Technology
15th December 2020. Vol.98. No 23

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

D. Inverse operation of Dynamic S-boxes

The inverse operation for the dynamic 3D S-Boxes is the same as the static S-Boxes. For all generated dynamic 3D S-Boxes the inverse of the Dynamic S-Boxes is calculated as follows:

1- First, take one value form dynamic 3D S-Box.
2- Second, convert this value into hexadecimal format and convert the index value to hexadecimal value by concatenating the index of the row and the index of column.
3- From step 2 there is left hexadecimal value and right hexadecimal value.
4- In the new array, Inv-3D S-Boxes go to the row indexed of the left hexadecimal number while for the column the right hexadecimal number holds the value of the original index of the dynamic 3D S-Box.
5- The operation is repeated to all the cells in the dynamic 3D S-Box to construct the Inv-dynamic 3D S-Box.
6- Step 1- through step 5 is repeated to the other 8 dynamic 3D S-Boxes and the results is stored in a table of Inv-dynamic 3D S-Boxes with the name of the rule used.

E. Use dynamic 3D S-Boxes instead of original S-Box

After generating all the dynamic 3D S-Boxes tables with their Inv- tables, the following operation is performed:

- For each round of the AES, the key of 128-bits (16 bytes) is XORed with all of each byte to generate only one Byte.
- Then, the last 3 bits from the generated Byte is considered as the rules using for this round. For example, if the values of $b_5b_6b_7$ are 000 then the 3D S-Box LUI table is applied instead of the static S-Box.
- For the decryption of the same round $b_5b_6b_7$ is 000 the dynamic inv-3D S-Box LUI is used. In appendix A there are eight different dynamic 3D S-Box named with their rules and tables for their Inverse.

Phase 4: Testing of Proposed S-Boxes

The proposed S-Boxes are tested based on the arguments stated in the requirements for good S-Box. The dynamic S-Box is implemented using SET tool with program in C with computer Windows 7, Intel core i7, 8GB of the RAM and 2TB of the memory. After deploying this program, the result obtained for the proposed 3D S-Boxes is similar as the original AES S-Box.

4. RESULT AND ANALYSIS

The purpose of the S-Box is to map 8 input bits to 8 output bits using predefined table known as Look up table LUT μ : GF(Bn) → GF(Bm) (Panda et. al., 2011). Mathematically, the LUT based S-Box and 3D CA based S-Box are derived using Boolean functions in order to study the level of security using cryptographic properties.

In cryptography, the Boolean function used to encrypt the plain data must be diverse and mapping should be a one to one mapping from input to output, so as to provide enough security and proper decryption.

Finally, the 2^8 output bits are transformed into a single output bit using Boolean function $f_i : B^n \rightarrow B$. In the S-Box, if μ: $B^n \rightarrow B^m$ and hence there exists m number of function $\mu = \{f_1, f_2, \ldots, f_m\}$, where $i \in [1, m]$. The truth table in polarity form is written as follows:

$$f_k(x) = (-1)^{f(x)}$$
$$f_\beta(x) = (a_1 f_1(x) \oplus a_2 f_2(x) \oplus a_3 f_3(x) .. \oplus a_m f_m(x)) \quad (6)$$

f_β is a Boolean function of the linear combination of m functions $f_i(x)$, $i \leq m$, where $\alpha_i \in B^m$ are coefficient of the linear function.

Table 6: Analysis of 3D CA S-Boxes

Name of rule	LUI	LUB	LDI	LDB	RUI	RUB	RDI	RDB
Time execution (ms)	2759	2762	2772	2765	2746	2760	2763	2755
Nonlinearity	112	112	112	112	112	112	112	112
Corelation immunity	0	0	0	0	0	0	0	0
Algebraic degree	7	7	7	7	7	7	7	7
SAC	√	√	√	√	√	√	√	√
No. fixed points	0	0	0	0	0	0	0	0
Composite algebraic immunity	0	0	0	0	0	0	0	0
Delta uniformity	4	4	4	4	4	4	4	4

Table 6 illustrates the results obtained for the proposed 3D CA S-Boxes. The result shows eight different dynamic 3D CA S-Boxes namely LUI, LUB, LDI, LDB, RUI, RUB, RDI and RDB in which each of them have meaning as Left-Up-Infrontof, Left-Up-Back, Left_down-Infrontof, Left-Down-Back, Right-Up-Intfrontof, Right-Up-Back, Right-Down-Infrontof and Right-Down-Back rules respectively. From the result, we observed that each dynamic 3D CA S-Boxes is the same but with different execution times ranging from 2746 ms to 2772 ms. Also, we observed that the best

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

recorded time used to generate an S-Box is for the Right-Up-Infrontof and the time taken is 2746.00 ms. While the value to generate the 3D CA S-Box via the 3D S-Box rule Left-Down-infrontof took 2772.00 ms. Whereas, the original S-Box took about 2760.00. As such we conclude that the result is almost the same. Additionally, the nonlinearity value for standard S-Box is 112 similar as the new dynamic 3D CA-S-Boxes.

The proposed dynamic 3D CA S-Boxes satisfies the SAC property and it is near to 0.5 and is similar with AES S-Box. We also found that correlation immunity and non-fixed points and Composite algebraic immunity are all zero and those are similar as the standard S-Box where those values are zero.

The last observation from our result is the Delta uniformity and its value is 4 for all the proposed 3D CA S-Boxes and which is similar to AES. Also, the Algebraic degree and its value is 7 similar as the standard AES S-Box.

D4	A2	AF	9C	A4	72	C0	AD	81	4F	DC	22	2A	90	88	60
A5	E5	F1	71	D8	31	15	34	32	3A	0A	49	06	24	5C	E0
12	80	E2	EB	27	B2	75	07	C8	37	6D	8D	D5	4E	A9	[illegible]
3B	D6	B3	29	E3	2F	84	52	78	25	2E	1C	A6	B4	C6	[illegible]
CB	BE	39	4A	4C	58	CF	6A	3E	B5	66	48	03	F6	0E	70
F9	02	7F	50	3C	9F	A8	45	F8	98	11	69	D9	8E	94	E1
B6	DA	21	10	FF	F3	D2	76	A1	89	0D	BF	E6	42	68	8C
01	67	2B	FE	D7	AB	BC	30	0C	13	EC	5F	97	44	17	C4
EE	B8	14	DE	5E	0B	DB	46	82	C9	7D	FA	59	47	F0	CA
D3	AC	62	91	95	E4	79	C2	FD	93	26	36	3F	F7	CC	B7
56	F4	EA	65	7A	AE	08	6C	C7	23	C3	18	96	05	9A	04
DD	74	1F	4B	BD	8B	8A	E8	83	2C	1A	1B	6E	5A	A0	[illegible]
35	57	B9	86	C1	1D	9E	61	D1	00	ED	20	FC	B1	5B	53
1E	87	E9	CE	55	28	DF	9B	EF	AA	FB	43	4D	33	85	D0
99	2D	0F	B0	54	BB	16	41	A3	40	8F	92	9D	38	F5	51
A7	7E	3D	64	5D	19	73	C4	7C	77	7B	F2	6B	6F	C5	[illegible]

Figure 11 New 3D CA S-Box for Left-Up-Infrontof Rule

C9	70	51	4C	AF	AD	1C	27	A6	BF	1A	85	78	6A	4F	E2
63	5A	20	79	82	16	E6	7E	AB	F5	BA	BB	3B	C5	D0	B2
CB	62	0B	A9	1D	39	9A	24	D5	33	0C	72	B9	E1	8A	95
77	15	18	DD	17	C0	9B	29	ED	42	19	30	54	F2	4E	96
E9	E7	6D	DB	7D	57	87	8D	4B	1B	43	B3	44	DC	2D	09
53	EF	37	CF	E4	D4	A0	C1	45	8C	BD	CE	1E	F4	84	7B
0F	C7	92	FF	F3	A3	4A	71	6E	5B	47	FC	A7	2A	8C	FD
4F	13	05	F6	B1	26	67	F9	38	96	A4	FA	F8	8A	C1	52
21	08	88	B8	36	DE	C3	D1	0E	69	B6	B5	6F	2B	5D	EA
0D	93	EB	99	5E	94	AC	7C	59	E0	AE	D7	03	EC	C6	55
BE	68	01	E8	04	10	3C	F0	56	2E	D9	75	91	07	A5	02
E3	CD	25	32	3D	49	60	9F	81	C2	3F	E5	76	B4	41	6B
06	C4	97	AA	F7	FE	3E	A8	28	89	8F	40	9E	7F	[illegible]	[illegible]
DF	C8	66	90	00	2C	31	74	14	5C	61	86	0A	B0	83	D6
1F	5F	22	34	95	11	6C	2F	B7	D2	A2	23	7A	CA	80	D8
8E	12	FB	65	A1	EE	4D	9D	58	50	8B	DA	CC	98	73	64

Figure 12: New Inverse 3D CA S-Box for Left-Up-Infrontof Rule

Figure 11 and Figure 12 depict Left-up-infrontof S-Box and its inverse respectively. We concluded that the result for dynamic 3D CA S-Boxes is better than standard AES S-Box. We propose the use of the dynamic 3D S-Box instead of the standard AES S-Box in the future.

For cryptanalysis, the static AES S-Boxes is strong against Linear and Differential attack. Since our idea is based on the original S-Box and we used different S-Boxes in different layers depending on the value taken from the round key to make our algorithm more complicated. Therefore, our proposed 3DCA S-Boxes are stronger from attack by the known linear and differential attack.

5. DISCUSSION

The results of the 3D CA S-Boxes tests are good for balance criteria, sufficient differential uniformity, excellent nonlinearity, adequate algebraic degree, and SNR. Additionally, the results are comparable with the S-Boxes of the AES block cipher. From the results, it is proven that the 3D CA S-Boxes are not susceptible to linear cryptanalysis attack and also it is verified that all the 3D CA S-Boxes have nonlinearity properties. The results also verified that 3D CA S-Boxes have a good differential uniformity property since DU = 4/256. Therefore, these 3D CA-S-Boxes are not susceptible to differential cryptanalysis attack. Thus, we conclude that the dynamic feature provides the 3D CA S-Boxes to be resistance from linear and differential cryptanalysis. The 3D CA S-Boxes do not only provide a high non-linearity and complexity, the dynamic features of the 3D CA S-Boxes make any attempt of performing linear and differential attack exhausting or impossible. In comparison with the AES S-Boxes, the 3D CA S-Boxes have advantages of having dynamic characteristics, whereas the AES S-Boxes have static characteristics. The 3D CA S-Boxes have good cryptographic properties and satisfy the requirement of a secure S-Box.

Another contribution is the proposed new key expansion algorithm dependent on the dynamic 3D CA S-Box instead of static S-Box algorithm. Any change in the key expansion function makes the block cipher algorithm more secure as all rounds of the algorithm are affected by the key expansion. Hence, it becomes difficult to guess the results of the rounds if there is a minor change in a single bit of the key. This is due to the different 3D CA-S-Boxes used that made the whole key expansion changed. When the proposed algorithms are used in AES block cipher, it can be seen from the results of the security analysis that apart from the non-linear relationship provided by the key expansion

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

functions between the plaintext and the ciphertext, a good confusion and diffusion property is also achieved by the block cipher algorithm.

6. CONCLUSION AND FUTUREWORK

This research proposed key dependent dynamic S-Boxes based on 3D CA Von Neumann rules. Classical S-Boxes based on static tables usually have fixed design constructions. The new S-Boxes are evaluated by the same criteria for evaluation of the classical S-Boxes.

Our experimental results show that the proposed S-boxes have characteristics of high non-linearity and balanced as indicated by Hamming Weight values. Our experiment also shows low autocorrelation and distance which fulfil the Strict Avalanche Criterion. The proposed KD dynamic 3D CA S-Boxes are proven to have good cryptographic properties. They are more secure than classical AES S-boxes, and resistant against linear and differential cryptanalysis. The dynamic version of 3D CA S-Boxes is easy to use for any block cipher and supply high space of possible S-boxes. Moreover, they support all cryptographic properties, to provide ciphering which is more secure. Additionally, the proposed S-Boxes can be used in any block cipher and they are more compact with unexpected behaviour and these leads to an increase of security for the algorithm.

The proposed S-Boxes are using cellular automata rules which is possible to be implemented in hardware. For future research, the proposed S-Boxes could be improved to be lightweight and suitable for IoT implementation.

ACKNOWLEDGMENT

This work was supported by Ministry of Higher Education under FRGS 08-01-15-1717FR Grant no. 5524822, also supported by Ministry of Higher Education and scientific research in Iraq.

REFRENCES:

[1] Adi Narayana Reddy, K., "Secure Linear Transformation Based Cryptosystem using Dynamic Byte Substitution", No. 1, 2014, pp. 24–32.

[2] Al-Wattar, A. H., Mahmod, R., and Zukarnain, Z. A., "A New DNA Based Approach of Generating Key-Dependent MixColumns Transformation", International Journal of Computer Networks & Communications (IJCNC), 7(2), 2015, pp. 93–102.

[3] Al-Wattar, A. H., Mahmod, R., Zukarnain, Z. A., and Udzir, I., "A New DNA Based Approach of Generating Key-Dependent ShiftRows Transformation", International Journal of Network Security & Its Applications, 7(9), 2015.

[4] Al-Wattar, A. S., Mahmod, R., Zukarnain, Z. A., and Udzir, N. I. "Generating A New S-Box Inspired by Biological DNA", International Journal of Computer Science and Application, 4 November 2015.
https://doi.org/10.12783/ijcsa.2015.0401.04

[5] Amirthalingama, S. and Latha, K., "A study on encryption using three-dimensional cellular automata", ScienceAsia 42S, 2016, pp. 42–48.

[6] Antonio, R., Sison, A., and Medina, R., "Performance Analysis of the Modified Generated S-Box for Advanced Encryption Standards", In Proceedings of 2nd International Conference on Data Science and Information Technology (DSIT '19), July 19–21, Seoul, Republic of Korea, ACM, 2019.

[7] Ao, T., Rao, J., Dai, K., Zou, X., "Construction of high quality key-dependent S-boxes", IAENG Int. J. Comput. Sci., Vol. 44, No. 3, 2017, pp. 337–344.

[8] Bertoni, G., Daemen, J., Peeters, M., Assche, G. 2011. The Keccak reference, [Online] .Available: http://keccak.noekeon.org

[9] Biham E. and Shamir A., "Differential cryptanalysis of DES-like cryptosystems", Advances in Cryptology CRYPTO 1990, Springer-Verlag, 1990.

[10] Chowdhury D Roy, Subbaro P., "Chracterization of Two-Diminsional Cellular Automata Using Matrix Algebra", Journal Information Sciences, Vol. 71, 1993, pp. 289-314.

[11] Daemen, J., Govaerts, R., Vandewalle, J., "Invertible shift-invariant transformations on binary arrays", Applied Mathematics and Computation, Vol. 62, No. 2, 1994, pp. 259 – 277.

[12] Dara, M. and Manochehri, K., "A novel method for designing S-Boxes based on chaotic logistic maps using cipher key", World Applied Science Journal, Vol. 28, No. 12, 2013, pp. 2003–2009.

[13] Dara, M. and Manochehri, K., "Using RC4 and AES Key Schedule to Generate Dynamic S-Box in AES," Information Security Journal, Vol. 23, No. 1–2, 2014, pp. 1–9.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[14] Das, D. and Misra, R., "Programmable Cellular Automata Based Efficient Parallel AES Encryption Algorithm", International Journal of Network Security and Its Applications (IJNSA), Vol. 3, No.6, November 2011.

[15] Ege, B., Papagiannopoulos, K., Batina, L., and Picek, S. (2015). Improving DPA resistance of s-boxes: How far can we go? In 2015 IEEE International Symposium on Circuits and Systems, pp. 2013–2016.

[16] ElGhafar A., Rohiem A., Diaa A., Mohammed F. 2009. Generation of AES Key Dependent S-Boxes using RC4 Algorithm. 13th International Conference on Aero-space Sciences & Aviation Technology, ASAT- 13, May 26-28, 2009.

[17] El-Sheikh, M., El-Mohsen, O., Zekry, A. 2012. A new approach for designing key-dependent S-box defined over GF in AES. Int. J. of Computer Theory and Engineering, Vol. 4, pp.158-164.

[18] Ghada Z., Abdennaceur K., Fabrice P., Daniele F., "On Dynamic chaotic S-BOX", IEEE, 2009.

[19] Gangadari, B. R. and Ahamed, S. R., "Analysis and algebraic construction of s-box for AES algorithm using irreducible polynomials", In Contemporary Computing (IC3), 8th International Conference on, IEEE, 2015, pp. 526–530.

[20] Gutowitz, H., "Cryptography with dynamical systems", In Cellular Automata and Cooperative Systems. Springer, 1993, pp. 237–274.

[21] Heys, H., "A tutorial on linear and differential cryptanalysis", Cryptologia, Vol. 26, No. 3, 2002, pp. 189–221.

[22] Juremi, J., Yasin, S., Sulaiman, S., Saad, N., Ramli, J., "A survey on various dynamic S-Box implementation in block cipher encryption algorithm", Journal of Applied Technology and Innovation, Vol. 2, No. 1, 2018, pp.

[23] Katiyar, S. and Jeyanthi, N., "Pure Dynamic S-box Construction," Int. J. Comput., Vol. 1, No. 1, 2016, pp. 42–46.

[24] Kazlauskas, K., Smaliukas, R., Vaicekauskas, G. "A Novel Method to Design S-Boxes Based on Key-Dependent Permutation Schemes and its Quality Analysis", (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 7, No. 4, 2016, pp. 93-99.

[25] Kazlauskas, K. and Kazlauskas, J., "Key-dependent S-box generation in AES block cipher system", Informatica, Vol. 20, No. 1, 2009, pp. 23–34.

[26] Koikara, R., "A 3D-Cellular Automata based Pseudo-random Number Generator: Student Research Abstract", Proceedings of the 31st Annual ACM Symposium on Applied Computing SAC'16, 2016, April 04-08, Pisa, Italy, 2016, pp. 2111-2112.

[27] Krishnamurthy, G., and Ramaswamy V., "Making AES Stronger: AES with Key Dependent S-Box", International Journal of Computer Science and Network Security, 9(8), 2008.

[28] Lambić, D., "A novel method of S-box design based on chaotic map and composition method", Chaos, Solitons and Fractals, Vol. 58, 2014, pp. 16–21.

[29] Mahmoud, M., El Hafez, A., Elgarf, T., Zekry, A., "Dynamic AES-128 with key-dependent S-box", Int. J. of Engineering Research and Applications, Vol.3, 2013, pp.1662-1670.

[30] Malik, M., Ali, M., Khan, M., Ehatisham-Ul-Haq, M., Shah, S., Rehman, M., Ahmad, W., "Generation of Highly Nonlinear and Dynamic AES Substitution-Boxes (S-Boxes) Using Chaos-Based Rotational Matrices", IEEE Open Access, Vol. 8, 2020, pp. 35682-35695.

[31] Maram, B. and Gnanasekar, J., "Evaluation of Key Dependent S-Box Based Data Security Algorithm using Hamming Distance and Balanced Output", Journal, Vol. 5, No. 1, 2016, pp. 67–75.

[32] Mariot, L., "Cellular automata, boolean functions and combinatorial designs", PhD Thesis. Université Côte d'Azur, France, 2018.

[33] Mariot, L., Picek, S., Leporati, A., Jakobovic, D., "Cellular Automata based S-Boxes", 2018, pp. 1-26, https://eprint.iacr.org/2017/1055

[34] Matsui M., "Linear cryptanalysis method for DES cipher", Eurocrypt, Springer LNCS, Vol. 765, 1993, pp. 386-397.

[35] Mohsen, M., Median, Z., Zied, G., Rached, T., "Design of reconfigurable image encryption processor using 2D cellular automata generator", International Journal of Computer Science and Applications, Vol. 6, No, 4, 2009, pp 43 – 62.

[36] Paar, C., & Pelzl, J., "Understanding Cryptography", 2010, https://doi.org/10.1007/978-3-642-04101-3

[37] Panda, S., Sahu, M., Rout, U., and Nanda, S., "Encryption and Decryption algorithm using two dimensional cellular automata rules in Cryptography", Int. J. Commun. Network Security, Vol. 1, No. 1, 2011, pp. 18–23.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[38] Picek, S., Batina, L. and Jakobovi, D., "A Toolbox for S-box Analysis", 2014, pp. 140–149.

[39] Picek, S. and Jakobovic, D., "Evolving Algebraic Constructions for Designing Bent Boolean Functions", GECCO'16, July 20-24, Denver, CO, USA, 2016.

[40] Picek, S., Marchiori, E., Batina, L., and Jakobovic, D., "Combining evolutionary computation and algebraic constructions to find cryptography-relevant Boolean functions", In Proceedings of Parallel Problem Solving from Nature, 2014, pp. 822–831.

[41] Picek, S., Ege, B., Batina, L., Jakobovic, D., Chmielewski, L., and Golub, M., "On using genetic algorithms for intrinsic side-channel resistance: The case of AES S-Box", In Proceedings of the First Workshop on Cryptography and Security in Computing Systems, 2014, pp. 13–18.

[42] Picek, S., Mariot, L., Yang, B., Jakobovic, D., Mentens, N., "Design of S-boxes Defined with Cellular Automata Rules", In Proceedings of CF'17, Siena, Italy, 15-17 May 2017, 2017, pp. 409-414.

[43] Obaid Alezzani, A. and Yousif Al-Bayatti, H., "Cryptography Using Three-Dimensional Cellular Automata (3-D CA)", International Journal of Computer Science and Network (IJCSN), Vol. 5, Issue 1, February 2016, 2016, pp. 197-204.

[44] Roy, S., Karjee, J., Rawat, U., Praktik, D., Dey, N., "Symmetric key encryption technique: A cellular automata-based approach in wireless sensor network", International Conference of Security & Privacy (ICISP2015), Procedia Computer Science, 78, 2016, pp. 408-414.

[45] Sharma, P., Lal, N., Diwakar, M., "Text Security using 2D Cellular Automata Rules", Conference on Advances in Communication and Control Systems CAC2S'2013, 2013, pp. 363-368.

[46] Singh, A., Agarwal, P., and Chand, M., "Analysis of Development of Dynamic S-Box Generation," Comput. Sci. Inf. Technol., Vol. 59, No. 5, 2017, pp. 154–163.

[47] Ulam, S., "Random processes and transformations", In Proceedings of the International Congress on Mathematics, Vol. 2, 1952, pp. 264–275.

[48] Von Neumann, J., "Theory of self-reproducing automata", University of Illinois Press, 1966, pp. 1-11.

[49] Wang, Y., Lei, P. and Wong, K., "A Method for Constructing Bijective S-Box with High Nonlinearity Based on Chaos and Optimization", International Journal Bifurc. Chaos, Vol. 25, No. 10, 2015, pp. 1550127.

[50] Wolfram, S., "Cryptography with Cellular Automata", Proceedings os Crypto'85, 1985, pp. 429-432.

[51] Zaïbi, G., Kachouri, A., Peyrard, F., Foumier-Prunaret, D., "On dynamic chaotic S-BOX", Glob. Inf. Infrastruct. Symp. GIIS '09, July, 2009.

[52] Zhang, R., & Chen, L., "A Block Cipher Using Key-Dependent S-box and P-box", IEEE International Symposium on Industrial Electronics, 2008, pp. 1463–1468.

[53] Zhang, Y. and Xiao, D., "Cryptanalysis of S-box-only chaotic image ciphers against chosen plaintext attack", Nonlinear Dyn., Vol. 72(4), 2013, pp. 751-756.