Method for Designing Strong S-Boxes Based On Chaotic System

By

Urwa Aftab

MASTER OF PHILOSOPHY IN MATHEMATICS

DEPARTMENT OF MATHEMATICS
CAPITAL UNIVERSITY OF SCIENCE AND TECHNOLOGY
ISLAMABAD
2017

Method for Designing Strong S-Boxes Based On Chaotic System

By

Urwa Aftab

A research thesis submitted to the Department of Mathematics,
Capital University of Science and Technology, Islamabad
in partial fulfillment of the requirements for the degree of

MASTER OF PHILOSOPHY IN MATHEMATICS

DEPARTMENT OF MATHEMATICS
CAPITAL UNIVERSITY OF SCIENCE AND TECHNOLOGY
ISLAMABAD
2017

CAPITAL UNIVERSITY OF SCIENCE & TECHNOLOGY
ISLAMABAD

Islamabad Expressway, Kahuta Road, Zone-V, Islamabad
Phone: +92 51 111 555 666, Fax: 92 51 4486705
Email: info@cust.edu.pk, Website: http"//www.cust.edu.pk

CERTIFICATE OF APPROVAL

Method for Designing Strong S-Boxes Based On Chaotic System

by

Urwa Aftab

MMT153011

THESIS EXAMINING COMMITTEE

S No	Examiner	Name	Organization
(a)	External Examiner	Dr. Tayyab Kamran	Quaid-e-Azam University, Islamabad
(b)	Internal Examiner	Dr. Dur e Shawar	CUST, Islamabad
(c)	Supervisor	Dr. Rashid Ali	CUST, Islamabad

_____

Dr. Rashid Ali
Thesis Supervisor
Sep, 2017

_____

Dr. Muhammad Sagheer
Head
Department of Mathematics
Dated : Sep, 2017

_____

Dr. Muhammad Abdul Qadir
Dean
Faculty of Computing
Dated : Sep, 2017

Certificate

This is to certify that MS. Urwa Aftab has incorporated all observations, suggestions and comments made by the external evaluators as well as the internal examiners and thesis supervisor. The title of his Thesis is: Method for Designing Strong S-Boxes Based on Chaotic System

_____

Dr. Rashid Ali
(Thesis Supervisor)

A METHOD FOR DESIGNING STRONG S-BOXES BASED ON CHAOTIC SYSTEM

by Urwa Aftab

Submission date: 15-Sep-2017 03:11PM (UTC+0500)
Submission ID: 847357637
File name: Urwa_Aftab_1.doc (1.65M)
Word count: 11790
Character count: 44179

A METHOD FOR DESIGNING STRONG S-BOXES BASED ON CHAOTIC SYSTEM

ORIGINALITY REPORT

13%	7%	10%	6%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.cse.fau.edu Internet Source	2%
2	Submitted to University of South Africa Student Paper	1%
3	www.kiarash.net Internet Source	1%
4	Hussain, Iqtadar, and Tariq Shah. "Literature survey on nonlinear components and chaotic nonlinear components of block ciphers", Nonlinear Dynamics, 2013. Publication	1%
5	Belazi, Akram, Rhouma Rhouma, and Safya Belghith. "A novel approach to construct S-box based on Rossler system", 2015 International Wireless Communications and Mobile Computing Conference (IWCMC), 2015. Publication	1%
6	Chen, G.. "An extended method for obtaining S-boxes based on three-dimensional chaotic	<1%

"As far as the laws of mathematics refer to reality, they are not certain, as far as they are certain, they do not refer to reality".

Albert Einstein

Abstract

A substitution box (S-box) plays an essential role in symmetric cryptographic algorithms. The most important property of S-box is its non-linearity that strengthen the cryptographic security. The purpose of using S-boxes in cryptography is to increase the confusion ability of the cipher. A number of researchers proposed different methods for the construction of S-boxes based on Chaotic System. In this thesis, we have modified a method to construct S-boxes based on Chaotic Lorenz system. The modified method is also applied to another chaotic system to generate better chaotic S-boxes. After this, the proposed S-boxes are analyzed by using a software SET tool.

Acknowledgements

Starting with the name of **ALLAH** the most beneficent and the most merciful whose blessings are abundant and favors are unlimited.a

First of all, I am very thankful to my Allah Almighty for his blessing. Blessing of Allah Almighty brought me up in this stage. Now I would like to express my sincere gratitude to my supervisor Dr. Rashid Ali for his patience, motivation, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis. My sincere thanks also goes to Dr. Muhammad Sagheer and Dr. Shafqat Hussain for their appreciation and support.

I also feel honored to have such supporting friends. I would like to specially thank my friend Saba Majeed, Afsheen Nazar, Parsa Ghazanfar, Jahangir Aftab, Ammara Aftab, Aroush Aftab, Mahnoor, Asifa Ashraf, Laila Allah Ditta, Atiya illyas, for providing me the strength to get focused toward my main objectives.

Finally, I would like to thank my family for being always with me and bringing all the care and support for my career. Specially, I am grateful to my parents,brother and sisters who have given all the love and care and brought me up in this stage.

Urwa Aftab

Contents

List of Figures

List of Tables

Dedicated to

My Parents

without their effort and support, I would never have reached so far

Chapter 1

INTRODUCTION

1.1 Cryptography

From ancient time to present day, cryptography provide security during communication. **Cryptography** is the study of techniques for the securing communications and data in the presence of adversaries. It is the science of arcanum writing with the motive that remains secret from others [42]. It convers mathematical techniques that are related to characteristics of information security [34]. It performs encryption and decryption with the help of secret key. The key information is known to sender and receiver and security of data rely on key information. Now we explain Cryptosystem. **Cryptosystem** is a procedure that permit two parties to communicate each other securely. It consist of five elements:

1. Message Space M
2. Ciphertext Space C
3. Key Space $\mathcal{K}$
4. Encryption Algorithm E_k, $k \in \mathcal{K}$
5. Decryption Algorithm D_k, $k \in \mathcal{K}$

On the basis of keys, cryptography is divided into two main branches. Namely, the Symmetric key cryptography and the Asymmetric key cryptography.

In **Symmetric key cryptography**, same key is used for both data encryption and decryption. Sender and receiver share a common secret key for both data encryption and decryption. For example (DES) Data Encryption Standard [59] and (AES) Advanced Encryption standard[15].

In 1976 White field Diffie and Martin Hellman [19] have proposed the concept of public key cryptography or asymmetric key cryptography. In **Asymmetric key cryptography**, two keys are used one for encryption and the other for decryption. A sender designs two keys, one key is shared publicly and the other key is kept secret. Examples are RSA [47] El. Gamal cryptosystem [45] and Elliptic curve cryptosystem [5].

1.2 Where are S-Boxes in a Cryptography

Many symmetric key cryptosystems are used to encrypt and decrypt one block at a time such systems are referred to as block ciphers. These are designed on the basis of Shannon's theory of confusion and diffusion [10]. To make the encryption and decryption process efficient, the process substitution is done with the help of look-up tables. These look-up tables are known as substitution boxes (S-boxes). A typical **S-box** takes an n bits input to produce an m bits output. S-box shows corelation between key and ciphertext. Infact, S-box is the only component in symmetric block ciphers that provides nonlinearity in the encryption process. So several time it is very essential to choose S-box which resists hacking/Cryptanalysis. The desirable properties of an S-box are its design simplicity, fast encryption and decryption speed . There are many methods for making good S-Boxes such as the construction used in Blowfish [18] , DES [59] , AES [15], Serpent [6] , Chaotic Lorenz System, etc. Reasearchers have proposed many approaches and methods for the construction of a strong S-box. Later on, mathematical based approaches were used to generate the values for S-box. For instance, construction of S-box

for AES [15] block cipher is based on a transformation and a translation. For this S-box, $m = n = 8$ and all the operation are performed in the finite Galois field $GF(2^8)$. With this S-box a byte is replaced by another byte in various rounds of the AES encryption algorithms. The hexadecimal representation of AES S-box is a 16×16 table given below:

63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

TABLE 1.1: AES S-Box

All S-boxes are designed to have better resistance against linear and differential cryptanalysis attacks [27]. To protect a cryptosystem from these attacks, S-box must satisfy high non-linearity [4], low number of fixed and opposite fixed points, high algebraic degree [33] and low differential uniformity. High non-linearity increases the strength of an S-box.

1.3 Objective Of Thesis

As explained earlier, S-box has a key role in cryptography, so it is very important to design a cryptographically good S-box. Different methods were proposed to design S-boxes in conventional cryptography over the past decade [51]. Many researchers have proposed different methods to generate a strong S-box. There

are many criteria's for desinging S-box but most of those criteria's do not fulfil the requirements[50]. We will generate S-box with the help of Chaotic Lorenz system. Lorenz system is proposed by Edward Lorenz in 1963 [44]. Lorenz was trying to discover new model in which atmosphere is define by fluid equations. He developed mathematical model which consist of the following three(ODE's).

$$\begin{cases} \dfrac{dx}{dt} &= a(y-x) \\ \dfrac{dy}{dt} &= (bx-y-xz) \\ \dfrac{dz}{dt} &= (xy-cz) \end{cases} \tag{1.1}$$

Lorenz system has chaotic solutions for a certain values of parameters and initial conditions. If we choose $a = 10$, $b = 28$, $c = \frac{8}{3}$, $x = 0.1$, $y = 0$ and $z = 0$, or some nearby values, then the above system shows the chaotic behaviour. With the help of above Lorenz system, Özkaynak et al. in [41] proposed a method for generating a chaotic S-box. Their proposed S-box is given below:

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

TABLE 1.2: S-box

Note that in this S-box $S(74) = 74$. That is, 74 is a fixed point of S-box which is not a good sign from cryptanalysis point of view. The method in [41] was slightly modified for generating a chaotic S-box without any fixed point. We have solved the system (3.4) by using MATLAB built-in tool "ode45" and applied an affine mapping on the resulting S-box to remove the fixed point and to improve the strength of the S-box.

1.4 Software Tools For S-Boxes Analysis

Some tools are available that can be used to study of the properties of an S-box. A brief description of such tools is given below:

1. **Boolfun Package in *R***

   R is the free open source Mathematics software used for graphics and computing statistics. It works on various Windows, UNIX and Mac OS platforms, while the standard version of ***R*** does not support the evaluation of Boolean functions but it is possible to load a package named as **Boolfun** [12, 21] which provides functionality relation to the cryptographic analysis of Boolean functions.

2. **SageMath**

   SageMath library [53] is the free open source tool which contains a modules called Boolean functions and S-Box. It allows the algebraic treatment of S-Boxes and studied the cryptographic properties of boolean function. This tool can evaluate the cryptographic properties related to linear approximation matrix and difference distribution table for S-boxes and Boolean functions.

3. **VBF**

   VBF stands for Vector Boolean Function Library. Alverez-Cubero and Zuffiria [13] presented this tool for the analysis of vector boolean functions that are used to evaluate the cryptographic properties of S-boxes.

4. **SET**

   SET stands for S-box Evaluation Tool. It is a free open source Mathematics tool which is simple and easy to use. Stjepan Picek and team [55] presented this tool for the analysis of cryptographic properties of Boolean function and S-boxes.

5. **ApCoCoA**

   ApCoCoA stands for Applied Computations in Computer Algebra. ApCoCoA is developed by the ApCoCoA Team, which is chaired by Martin Kreuzer [3]. It is based on the Computer Algebra system CoCoA [24]. The information about ApCoCoA can be found on its official website (apcocoa.org).

The rest of the thesis is organized as follows

- **Chapter 1** provides the necessary background on Cryptography.
- **Chapter 2** presents the basic concepts and definition that are needed for Boolean functions, their general properties, and how these are used for the S-boxes.
- **Chapter 3** presents the basic concept of Chaotic Lorenz system and introduced an algorithm for the construction of S-boxes by using Lorenz equations and New Chaotic System. Further, checked the properties of S-boxes by using the S-box Evaluation Tool (SET)[55].
- **Chapter 4** In this chaphter we will present analysis of S-box properties and perform comparison with other Chaos-based S-boxes . In the end a brief conclusion is presented.

Chapter 2

PRELIMINARIES

This chapter presents and explain some basic terminologies which will later be used in the proceeding chapters.

2.1 Cryptography

It is the science of arcanum writing with the motive of keeping something secret from others [42]. Cryptography is a subject which convers mathematical techniques that are related to characteristics of information security [34] *i.e* confidentially, data probity authentication and ownership reliability. Crytography provides number of techniques which are helpful in security matters. Now we explain the Cryptosystem. **Cryptosystem** is a pair of algorithms that converts plaintext into ciphertext and vice versa with the help of key. Cryptosystem has the following components.

1. **Plaintext**: This is comprehensible message which is given to algorithm as input.

2. **Ciphertext**: This is muddle message denoted as output. It depends on comprehensible message and secret key.

3. **Encryption algorithm**: The encryption algorithm performs various swap and alteration on plaintext.

4. **Decryption algorithm**: It is important for the algorithm to run in reverse manner as well. It generate original plaintext with the help of secret key and ciphertext.

5. **Secret Key**: The key is independent of the plaintext and of the algorithm. The algorithm generate various outputs depending on the key.

Cryptography is further divided in the following two main categories:

1. **Symmetric key cryptosystem**

2. **Asymmetric key cryptosystem**

2.1.1 Symmetric key cryptosystem

It is the cryptosystem in which encryption and decryption are performed by same key. It is also known as secret key cryptosystem. Sender and receiver share a common secret key for both data encryption and decryption. For example (DES) Data Encryption Standard and (AES) Advanced Encryption standard [42]. Symmetric key cryptosystem are simple to use, easier to implement and have fast speed. On the other hand symmetric key cryptosystem have key management and security issues.

2.1.2 Asymmetric key cryptosystem

It is the cryptosystem in which encryption and decryption are accomplish using two distinct keys, a public key and a private key. It is known as public key encryption. Asymmetric encryption convert plaintext into ciphertext with the help of public key and an algorithm. Examples are RSA cryptosystem, El.Gamal cryptosystem, Elliptic curve cryptosystem [4]. The most genearally used public

key cryptosystem is RSA. Block and Stream Ciphers are the two main types of ciphers used in cryptography.

Definition 2.1.1. (**Block Cipher**)
A symmetric key cryptosystem which converts a block of plaintext p into a block of ciphertext c with the help of secret key k at a time is known as block cipher. Block cipher are proposed to provide plenty of confusion and diffusion [26]. A block cipher has two important parameters.

1. The block size
2. The key size

Data encryption Standard (DES)[17] is a symmetric block cipher which was published by IBM in 1970. DES uses 56 bit key to encrypt 64 bit data, having a block of 8 bytes. Due to its small key size, it was breakable through brute force within 24 hours, to overcome this drawback, $2DES$ and $3DES$ were designed. In $2DES$, 112 bit key is used for encryption. Similarly in $3DES$, 168 bit key is used for encryption. $3DES$ proved more secure as compared to others, but the drawback is that it has slow speed. In 2001 Vincent Regimen, John Daemon gave a more complicated algorithm called Rinjindael, which was named as Advanced encryption standard . It is a private key symmetric block cipher. It is six times faster than $3DES$. AES [17] use a key of 128 | 192 | 256 bits to encrypt 128 bit data, having a block of 16 bytes. Its main components is Substitution box (S-box). The purpose of such S-box is to produce non-linearity, confusion, and diffusion in the ciphertext. These are the only operations used for creating nonlinearity in the system. Since our works depends on the construction of S-boxes. Now we will present the brief introduction of S-box and boolean functions.

2.2 Finite Field $GF(2^8)$

Finite Field is a field that contains finite number of elements. It is a set on which the operations of addition, subtraction, muiltiplication and division are defined and

satisfy certain basic rules. It is denoted by $GF(p^n)$. The most common examples of finite fields are given by the integers mod p when p is a prime number. All the elements in finite field can be represented by polynomials of degree less than n, with coeffients from $GF(p) = \{0, 1, 2, \ldots, p-1\}$. From cryptographic point of view, we are mostly interested in finite fields with $p = 2$. That is, $GF(2)$ and $GF(2^n)$. The elements of $GF(2^n)$ can then equivalently represented by n-bits binary strings. For the finite field $GF(2^8)$, both the polynomial and binary representations are given below:

Decimal	Polynomial	Binary	Hexa
0	0	00000000	00
1	1	00000001	01
2	x	00000010	02
3	$x+1$	00000011	03
4	x^2	00000100	04
5	x^2+1	00000101	05
6	x^2+x	00000110	06
7	x^2+x+1	00000111	07
8	x^3	00001000	08
9	x^3+1	00001001	09
10	x^3+x	00001010	$0A$
.	.	.	.
.	.	.	.
.	.	.	.
255	$x^7+x^6+x^5+x^4+x^3+x^2+x+1$	11111111	FF

TABLE 2.1: Elements of Finite Field $GF(2^8)$

2.3 Substitution Box

S-boxes are very important component in a crypotosystem. In block ciphers, S-boxes are used to hide the correlation between the key and the ciphertext [22]. S-box are the only nonlinear component in cryptosystem. S-boxes are used in almost all conventional cryptographic algorithms, such as DES and AES. S-boxes are composed of highly nonlinear Boolean functions. Now we move towards the Boolean function and some basic definations.

Definition 2.3.1. Boolean Function

Boolean function is a function which is define as $f(x) : GF(2^n) \longrightarrow GF(2)$ where k is non-negative integer. Every value of n where $(n = 1, 2 \ldots, 8)$can be written as $x_1, x_2, x_3, x_4, \ldots, x_n$ [4]. A Boolean function explain how Boolean output values determine with the help of some logical calculations of Boolean input values. These functions are also helpful to design the circuits and chips of digital computers [11]. In cryptography, Boolean functions plays an important role for designing a substitution boxes.

Example 2.3.2. For $n = 3$, we have a mapping from $GF(2^n)$ to $GF(2)$.

$$f(x_1, x_2, x_3) = x_1 \oplus x_2 x_3$$

with input bits x_1, x_2 and x_3.

x_1	x_2	x_3	f
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

TABLE 2.2: Truth table of Boolean function

Definition 2.3.3. Vectorial Boolean Function

A Vectorial boolean function is a mapping $S : GF(2^n) \longrightarrow GF(2^m)$, where $GF(2)$ is the finite field having two elements. An (n, m) vectorial Boolean function(S-box) is a mapping:

$F = F(x_1, x_2, x_3, x_4, \ldots, x_n) = (f_1(x_1, x_2, \ldots, x_n), \ldots, f_m(x_1, x_2, \ldots, x_n))$

where each component function f_i, $i = 1, 2, \ldots, m$ is n-variable boolean function. Let $f_1, f_2, \ldots, f_m$ be m Boolean functions, where each f_i corresponds to vector F of length 2^n.

Example 2.3.4. For n=m=2, we have mapping from $GF(2^2)$ to $GF(2^2)$ and we get 4×4 S-box:

Inputs: [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15]

S-box : $[15 \quad 13 \quad 10 \quad 9 \quad 11 \quad 12 \quad 7 \quad 3 \quad 14 \quad 6 \quad 8 \quad 2 \quad 4 \quad 5 \quad 0 \quad 1]^t$

where each elements of S-box can be written as:

$$S = \begin{bmatrix} f_1 : 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ f_2 : 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ f_3 : 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 4 & 0 & 0 \\ f_4 : 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

That is, $S(0000)= 1111$, $S(0001)= 1011,\ldots,$ $S(1111)= 1000$. Since an S-box is used in both encryption and decryption, it should be a bijective mapping. This is to make sure that every S-box also has an inverse S-box.

Definition 2.3.5. (**Hamming weight**)
The number of non-zero digits in a binary sequence is called hamming weight. It is denoted by $\mathbf{wt}(x)$, where $x \in GF(2^n)$

Example 2.3.6. For $n = 8$. Let

$$x = 00110101 \text{ then } \mathbf{wt}(00110101) = 4$$

Definition 2.3.7. (**Hamming distance**) The hamming distance [14] between two Boolean functions $f(v), g(v) : GF(2^n) \longrightarrow GF(2)$ is defined as:

$$d(f, g) = \mathbf{wt}(f(v) \oplus g(v))$$

Here,

$$f(v) \oplus g(v) = f(v_0) \oplus g(v_0) \oplus f(v_1) \oplus g(v_1) \oplus \ldots \oplus f(v_{2^n-1}) \oplus g(v_{2^n-1})$$

Example 2.3.8. Consider the two Boolean functions, $f(v) = v_1v_2v_3$ and $g(v) = v_1 \oplus v_2v_3$ with input bits v_1, v_2, v_3. Hamming distance of these boolean functions is:

$$\begin{aligned} d(f, g) &= \mathbf{wt}(f(v) \oplus g(v)) \\ &= \mathbf{wt}(v_1v_2v_3 \oplus v_1 \oplus v_2v_3) \end{aligned}$$

i	$v_i = v_1v_2v_3$	$(f \oplus g)(v_i)$
0	0 0 0	0
1	0 0 1	0
2	0 1 0	0
3	0 1 1	0
4	1 0 0	1
5	1 0 1	1
6	1 1 0	1
7	1 1 1	1

TABLE 2.3: Truth table of $GF(2^3)$

Hence, the hamming distance of f and g is 4.

Definition 2.3.9. (**Balanced**)
A binary sequence of Boolean function f is called **balanced** if there are equal number of zeros and ones.

Example 2.3.10. To show binary sequence is balanced. Consider the example with boolean function,

$$f(v_1, v_2, v_3, v_4) = v_1 \oplus v_2v_3 \oplus v_4$$

is given below:

i	$\beta = v_1v_2v_3v_4$	$f(\beta_i)$
0	0 0 0 0	0
1	0 0 0 1	1
2	0 0 1 0	0
3	0 0 1 1	1
4	0 1 0 0	0
5	0 1 0 1	1
6	0 1 1 0	1
7	0 1 1 1	0
8	1 0 0 0	1
9	1 0 0 1	0
10	1 0 1 0	1
11	1 0 1 1	0
12	1 1 0 0	1
13	1 1 0 1	0
14	1 1 1 0	0
15	1 1 1 1	1

TABLE 2.4: Truth table of $GF(2^4)$

The last column contains 8 zeros and 8 ones. So the sequence of f is balanced.

Definition 2.3.11. (**Bijection**)
Bijection is a mapping in which each input bit produce a unique output. Let n be the possible input bits such as $\{0,1\}^n$ there exist a unique output bit. Every output vector should appear one time.

To check the bijective property of an $n \times n$ S-box a method is introduced in [56]. An $n \times n$ S-box is said to satisfy the bijective property if for $1 \leq i \leq n$ the boolean functions f_i of S are such that:

$$\mathbf{wt}(\Sigma_{i=1}^{n} a_i f_i) = 2^{n-1} \tag{2.1}$$

where $a_i \in (0,1)$, and $(a_1, a, \ldots, a_n) \neq (0,0,\ldots,0)$ and $\mathbf{wt}()$ is the hamming weight. The condition (2.1) in fact, guarantees that every boolean function f_i and all their combinations are 0/1 balanced, that is has equal number of zeros and ones. We illustrate the above definition by the following example.

Example 2.3.12. Let us consider 4×4 S-box and show that it is bijective.
Inputs: [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15]
S-box : [9 13 10 15 11 14 7 3 12 8 6 2 4 1 0 5]^t
where each elements of S-box can be written as:

$$S = \begin{bmatrix} f_1: & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ f_2: & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ f_3: & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ f_4: & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

That is, $S(0000)$= 1001, $S(0001)$= 1011,..., $S(1111)$= 0101.

Note that all the above boolean functions $f_1, \ldots, f_4$ are 0/1 balanced. That is, $\mathbf{wt}(f_i) = 8$ for $1 \leq i \leq 4$. One can show that hamming weight of all other combinations of f_i, that is, $f_1 + f_2$, $f_1 + f_3$, ..., $f_2 + f_3 + f_4$ is 8. Further, to check the property (2.1), we have implemented a code in ApCoCoA [3] (see Appendix A).

Definition 2.3.13. (**linearity**)
Linearity is the property of function which show that function can be represented graphically by a straight line. A **linearity** of boolean function $f(x) : GF(2^n) \rightarrow GF(2)$ is defined as:

$$f(x) = a \cdot x \qquad \text{where } a \text{ , } x \in GF(2^n)$$

Let us take some examples of some linear functions:

$$\begin{aligned} f_1(x_1, x_2) &= 0 \\ f_2(x_1, x_2) &= 1 \\ f_3(x_1, x_2) &= x_1 \\ f_4(x_1, x_2) &= x_2 \\ f_5(x_1, x_2) &= x_1 + 1 \\ f_6(x_1, x_2) &= x_2 + 1 \\ f_7(x_1, x_2) &= x_1 + x_2 \\ f_8(x_1, x_2) &= x_1 + x_2 + 1 \end{aligned}$$

The linear combination of two boolean function $f(x), g(x)$ is defined as

$$(f \oplus g)x = f(v) \oplus g(x)$$

Definition 2.3.14. (**Affine function**)
A Boolean function $f(x) : GF(2^n) \rightarrow GF(2)$, composed of linear function and a constant is called **Affine function** [17], which can be expressed as

$$f(x) = a \cdot x \oplus \varepsilon \quad \text{where } a, x \in GF(2^n) \text{ and } \varepsilon \in GF(2)$$

Definition 2.3.15. (**Permutation**)
Let S be a finite set. A permutation P on S is a bijection from S to itself (i.e,$P : S \longrightarrow S$).

Example 2.3.16. Let $S = (1, 2, 3, 4, 5)$. A permutation $P : S \longrightarrow S$ can be defined as follows:

$$P(1) = 3, P(2) = 5, P(3) = 4, P(4) = 2, P(5) = 1$$

Definition 2.3.17. (**Non-linearity**)
The *non-linearity* $NL(f)$, of a boolean function $f(v) : GF(2^n) \longrightarrow GF(2)$ is defined as the minimum hamming distance of f from any of its n variable affine functions [31].

$$NL(f) = \min_{g \in A} d(f, g)$$

For better understanding, we take an example:

Example 2.3.18. Let x_1 and x_2 are input bits and $f(x)$ is a boolean function: $f(x) = x_1 \oplus x_2$

x_1	x_2	$f(x)$	0	x_1	x_2	$x_1 \oplus x_2$
0	0	0	0	0	0	0
0	1	1	0	0	1	1
1	0	1	0	1	0	1
1	1	1	0	1	1	0

TABLE 2.5: Truth table

Where $0, x_1, x_2, x_1 \oplus x_2$ are the possible linear function of x_1 and x_2 and $d_1(f(x), 0) = 3$, $d_2(f(x), x_1) = 1$, $d_3(f(x), x_2) = 1$, $d_4(f(x), x_1 \oplus x_2) = 1$.

So,

$$N_f = \min(d_1, d_2, d_3, d_4) = 1$$

Definition 2.3.19. (**Walsh transform**) [61]
The measurement of correlation between the boolean function f and all of the linear combinations is known as **Walsh transform**. The Walsh transform of a boolean function f is defined by

$$WHT_f(\beta) = \sum (-1)^{f(v)+\beta.v} \qquad \text{where } \beta \in GF(2^n) \text{ for all } v \in GF(2^n).$$

The non-linearity of a Boolean function $f(x)$ can be given by Walsh transform by the following formula:

$$N_f = 2^{n-1}(1 - 2^{-n} \max_{\beta \in GF(2^n)} |WHT_f(\beta)|). \tag{2.2}$$

Further, we can check the non-linearity of S-box by using tool SET [55].

Definition 2.3.20. (**Correlation Immunity**)
A boolean function has a correlation immunity (CI) which denotes the independence size between the linear combination of input bits and output. Its functional order can be determined by a relationship between Walsh transform and hamming weight of its inputs. A boolean function is said to be correlation immunity if its $WHT_f(\beta) = 0$, whenever $1 \leqslant H(w) \leqslant p$. In other words there should not be statatistical pattern between output bits from the output vectors [57].

Definition 2.3.21. (**Absolute indicator and Sum of square indicator**) [37]
The absolute indicator of boolean function $h(v)$ is defined as the minimum absolute value of autocorrelation, which can be expressed as

$$\Delta_h = \max \mid \Delta_h(b) \mid \quad \text{where} \quad b \in GF(2^n)$$

The sum of square indicator of boolean function $h(v)$ also derived from autocorrelation function $\Delta_h(b)$ which can be expressed as

$$\sigma_h = \sum_{b \in GF(2^n)} \Delta_h^2(b)$$

Where, autocorrelation (AC) of boolean function $h(v)$ is defined by

$$\Delta_h(b) = \sum (-1)^{h(v)+h(v+b)} \quad \text{where} \quad v \in GF(2^n)$$

Example 2.3.22. Let us consider the example of Absolute indicator and Sum of square indicator. The boolean function

$$h(v_1, v_2, v_3) = v_1 + v_2 v_3 + v_3$$

to compute autocorrelation at $b = 110$.

$v = v_1v_2v_3$	$h(v)$	$h(v+b)$	$(-1)^{h(v)+h(v+b)}$	$dim2$	$dim1$	$dim0$
0 0 0	0	1	1	2	4	8
0 0 1	1	1	1	2	4	0
0 1 0	0	1	1	2	0	0
0 1 1	0	0	1	2	0	0
1 0 0	1	0	1	0	0	0
1 0 1	0	0	1	0	0	0
1 1 0	1	0	1	0	0	0
1 1 1	1	1	1	0	0	0

TABLE 2.6: Truth table of AC

Hence the Absolute indicator of $h(v)$ is 8 and Sum of square indicator of $h(v)$ is 64.

Before discussing the Strict Avalanche Criterion. We define the term Avalanche effect.

Definition 2.3.23 (**Avalanche effect**)**.**
If half of the output bits changed as the result of changing single input bit then this is called Avalanche effect. To understand Avalanche effect, choose a pair of n-bit plaintext vectors X and X_j which is dissimilar only in jth bit, and their corresponding output bits are $f(X)$ and $f(X_j)$ which are different at least in bit i. After this, taking XOR of output bits and we get:

$$V_j = f(X) \oplus f(X_j)$$

Each V_j contain n-bits, are called avalanche variables. If the above procedure is repeated for $1 \leq j \leq n$, for each j one half of the variables are equal to 1, then f having a good avalanche effect.

Now, we will explain the **strict avalanche property**. Although the above process can be used, we will apply an alternate method. First of all, n-bit random plaintext vector X is generated and find its corresponding ciphertext vector Y . Then, the n-vectors plaintext are:

$$(X_1, X_2, \ldots, X_n)$$

is formed such a way that X and X_i dissimilar only in jth bit. The corresponding ciphertexts vectors are:

$$(Y_1, Y_2, \ldots, Y_n)$$

Then, we have:

$$Y_i = f(X_i)$$

Thus, we obtained avalanche vectors:

$$(V_1, V_2, ...V_n)$$

such that

$$V_i = Y \oplus Y_i$$

Now value of V_i added in dependence matrix A. We repeat this process for large numbers of time. The degree of repeated procedure depends on the number of randomly generated plaintext vectors which is said to be r, and every element of matrix A divided by r. In matrix the value of 0 show that the ciphertext bits is totally independent of plaintext bits and 1 show that any change in plaintext will change the ciphertext. If the matrix has non zero entries, then transformation is complete. So it satisfies the strict avalanche criterion and each element having the value closer to one half(1/2)[56]. Further, to check the property , we can use SET Tool[55].

Definition 2.3.24. (**The Output bits independence criterion**)
The Output bits independence criterion (or BIC) is another desirable property for designing the S-Box. It become really hard to guess the design of system, by increasing independence between the bits. It shows that all avalanche variables should be independent pairwise for a specific set of avalanche vectors which are produced by single plaintext bit transpose. Consider two Boolean functions f_j and f_k which are these are output bits of S-box. For (BIC), the relation given below

should satisfied [58]

$$f_j \oplus f_k (j \neq k, 1 \leq j, k \leq n) \tag{2.3}$$

and should be highly non linear, also satisfies (SAC), than each pair of output bits having equall to zero if the input bit is change. correlation is close to zero when any input bit is transpose. There is another method to find (SAC) of equation (2.3) which does not require dependence matrix. The Dynamic Distance (DD) of function :

$$f : (0,1)^n \longrightarrow (0,1)$$

is defined as:

$$DD(f) = \max_{d \epsilon (0,1)^n wt(d)=1} \frac{1}{2} \left| 2^{n-1} - \sum_{x=0}^{2^n-1} f(x) \oplus f(x \oplus d) \right|$$

The function satisfies (SAC) if DD have small integer value and closer to zero[7]. Further, to check the output bits independence criterion , we can use SET Tool[55].

Definition 2.3.25. (**Algebric immunity**)
An Algebric Immunity of two boolean functions $f(v)$ and $g(v)$ is defined as the lowest degree of non-zero function g such that either

$$(f+1)g = 0 \quad \text{or} \quad f.g = 0$$

where a boolean function f is said to admit an annihilating function g if $f.g = 0$.

Example 2.3.26. Consider the two boolean functions

$$f(v) = v_1 + v_2 \quad \text{and} \quad g(v) = v_2$$

to compute the algebric immunity;

$v = v_1v_2$	$f(v)$	$f.g$	$(f+1)$	$(f+1)g$
0 0	0	0	1	0
0 1	1	1	0	0
1 0	1	0	0	0
1 1	0	0	1	0

TABLE 2.7: Truth table of AI

From the above table it shows that $(f+1)g = 0$.

Definition 2.3.27. (**Algebraic degree**)
An algebraic degree of S-box is related with the nonlinearity measures. An algebric degree of boolean function $h(v)$ is defined as the highest degree of a function h, which can be expressed as

$$deg(h) = n - 1$$

Higher algebraic degree is considered more better than the lower algebraic degree.

Definition 2.3.28. (**Transparency order**)
The number of power traces to identity the correct key is known as transparency order. The smaller value of transparency order, provides a high resistance against differential power analysis (DPA) attacks where, DPA (Differential Power Analysis) is a strong cryptanalytic technique which is used to remove secret data from cryptographic device. If the transparency order of a S-box is high then S-box cannot achieve its resistance against differential power analysis (DPA) attacks depends on the quality of the measurements an attacker can achieve.

Definition 2.3.29. (**Fixed** (F_p) **and Opposite fixed points**(OF_P))
The number of these (F_p) and (OF_P) should kept as low as possible to avoid attacks in any statistic cryptanalysis [36].

Chapter 3

Designed S-box by Chaotic Lorenz System

In this Chaphter, first we explain Chaos Theory. Secondly, we modify the algorithm of Lorenz System presented in [41] for designing an S-box. Further, we also introduce a new Chaotic System and its algorithm for constructing substitution boxes.

3.1 Chaos Theory

Chaos theory is the result of natural science discoveries in the field of non-linear dynamics. Nonlinear dynamics is the study of the temporal evolution of non-linear systems. Chaos theory is also used in non-linear system[40]. A brief examination of the mathematics and behaviour of chaotic system provides means for understanding the relevance of this theory to the complexity of social phenomena. Chaos theory is summarized by Edward Lorenz. It is the branch of mathematics which emphasize on the behaviour of dynamic system [8]. Dynamic system is a system in which function rely on time dependent point in a geometrical space, *i.e*, moving pendulum, water flow in pipe etc. Moreover, Chaotic System is highly sensitive about the initial conditions. Sensitivity to initial condition mean that each point

on the chaotic system are very close to the other points on the trajectories. So that the smaller change in initial conditions may lead significantly large change in the behaviour. Sensitivity to initial condition is known as the **Butterfly effect** [25]. In other words chaos theory is the science of unpredictable surprises.
Chaos theory has several applications in the following discipline[54].

1. Meteorology
2. Physics
3. Sociology
4. Environmental science
5. Computer science
6. Engineering
7. Economics
8. Biology
9. Ecology
10. Philosophy

3.1.1 Why Chaotic System is Used

"Chaos was the law of nature, Order was the dream of man"[60]. It is new and interesting field of abstract and complex mathematics. Until 1960 the world of science was relatively simple. Everthing could be explained with formulas and had a predictable behavoiur. As the story goes, one day Edward Lorenz was working on weather forecasting machine, he decided to examine the past day sequence with more detail. He types the number from the previous day computer record and went to get a coffee. When he returned, he couldn't believe in his eyes. The new weather was nothing like the original. It was completely different. Then he

realized that weather is a Chaotic system. Actually Chaos system is sensitive to initial condition. In the beginning, minor change leads to major change in future prediction. Chaos teaches us to believe on unexpected. It deals with the non linear things that are difficult to predict such as turbulence, weather, the stock market, brain states and so on [16]. These phenomenons are often captures by the fractal mathematics [43].

3.1.2 Lorenz System

Lorenz system is proposed by Edward Lorenz in 1963 [44]. Lorenz was trying to discover new model in which atmosphere is define by fluid equations. He select his model as a distinct set of ordinary differential equations(ODEs), called Lorenz equations. He has chosen this system because of two reasons. First, the atmospheric equation with thousands values of variables and parameters become more complicated even the mordern machines can not predict accurately. Secondly he wanted to show that, even simple equations also have unpredictable results. He developed mathematical model which consist of three(ODE's) [44]. These Lorenz equations are also helpfull to simplify models for Lasers, Electric circuit, DC Motors etc. Lorenz system are non-linear and non-periodic. Further Lorenz system has chaotic solutions for a certain values of parameters and initial conditions. The system of Lorenz equations are:

$$\begin{cases} \dfrac{dx}{dt} & = a(y-x) \\ \dfrac{dy}{dt} & = (bx-y-xz) \\ \dfrac{dz}{dt} & = (xy-cz) \end{cases} \tag{3.1}$$

where x, y, z, t are the variable and a, b, c are the chaotic parameter's. For the chaotic behaviour parameter values are $a = 10$, $b = 28$, $c = \frac{8}{3}$ and initial vaues

are $x = 0.1$,$y = 0$, $z = 0$. The solutions of Lorenz system are called Chaotic solutions. The set of Chaotic solutions of Lorenz system is called Lorenz attractor [28], which plots a butterfly image on plotting, as shown below.

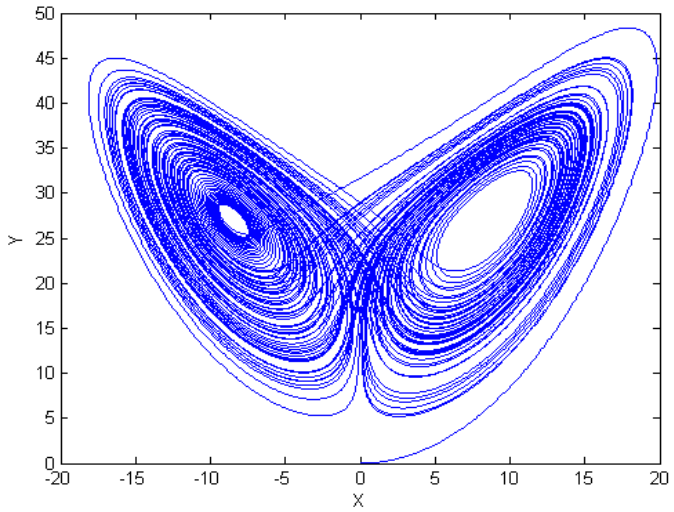

FIGURE 3.1: Lorenz System x-y Graph
[38]

3.1.3 Properties of Lorenz Equations

Now we discuss some important features of Lorenz system. The important properties of Lorenz equations are:

- These equations are autonomous(the right side of Lorenz equations are independent of time).
- They have only first order time derivative.
- These equations are non-linear, second and third equations involve the quadratic terms xy and xz.
- The solution of above equations is bounded.

For further details we refer to [46].

3.2 S-box and Lorenz System

In [41] Özkaynak et al. proposed algorithm to use Lorenz system for genearating an S-box. They solved the Lorenz system (3.4) with selected parameter values by applying 4-steps Runge-Kutta method. In this section, we describe the algorithm presented in [41] for generating an S-box. Instead of using $4th$ order Runge-Kutta method, we solved the system using MATLAB built-in tool "ODE45"[51]. The code is given in Appendix A. To design S-box following steps will be followed. These steps consist of two phases, diffusion is performed in the first phase and substitution operation is applied in the second phase. All these steps are described in the following algorithm.

Algorithm 3.2.1. Lorenz Chaotic S-Box

Input: The Lorenz Equations (3.4)

Output: Chaotic S-box

1. Get the system trajectories by solving the Lorenz system of equations (3.4) by using the MATLAB tool **ode45** with the following parameters:

$$a = 10, \quad b = 28, \quad c = \frac{8}{3}.$$

2. After ignoring some initial values, get a sample of 256 data points from all the points on these trajectories after a fixed step of size $h = \frac{\text{Total number of data points}}{256}$.

3. Assign a code from $0 - 255$ to the sampled 256 data points.

4. Sort the above codes in ascending order with respect the corresponding data points. In this way, initialize the proposed S-box by writing the sorted codes in the form of a 16×16 table.

5. For each row number i $(1 \leq i \leq 16)$, rotate the row by applying a left shift of $(i - 1)$ points to update the above S-box.

6. For each column number j $(1 \leq j \leq 16)$, rotate the column by applying a downward shift of $(16 - j)$ points to update the above S-box.

3.2.1 Implementation and Results

Using the MATLAB tool **ode45** [51] and ApCoCoA [3], we implemented the above algorithm 3.2.1 and generated an S-box as follows:

For the first step of Algorithm 3.2.1, we have computed 100,000 y values using MATLAB code given in Appendix A. The initial conditions for the system are taken as $x = 0.1$, $y = 0$ and $z = 0$. After ignoring starting 9984 (*i.e.* 39×256) values for the transitions to die out, the following data is obtained after applying Step 3 of the algorithm.

Data	Code	Data	Code	Data	Code	Data	Code	Data	Code	Data	Code	Data	Code	Data	Code
-6.779	0	23.745	32	2.635	64	-0.127	96	5.257	128	1.799	160	-1.967	192	17.012	224
-11.879	1	-6.091	33	19.198	65	0.448	97	13.830	129	12.208	161	-2.844	193	-3.503	225
-4.695	2	-11.880	34	-3.132	66	5.676	98	3.086	130	1.318	162	-16.835	194	-5.707	226
-10.933	3	-5.167	35	-5.270	67	12.506	99	6.959	131	-0.888	163	1.444	195	-15.539	227
-6.270	4	-6.890	36	-16.661	68	-6.311	100	12.496	132	-4.930	164	1.571	196	-0.862	228
-6.665	5	-12.277	37	0.124	69	-12.960	101	2.467	133	-17.892	165	11.545	197	-6.429	229
-12.337	6	-3.691	38	-3.058	70	-4.146	102	11.143	134	3.652	166	0.362	198	-13.636	230
-4.093	7	-11.617	39	-19.751	71	-8.970	103	4.691	135	10.989	167	-3.419	199	0.367	231
-11.142	8	-5.077	40	2.759	72	-8.895	104	2.970	136	4.665	168	-15.725	200	-1.498	232
-5.840	9	-5.522	41	4.763	73	-4.691	105	15.212	137	2.367	169	-0.130	201	-14.535	233
-5.970	10	-14.455	42	17.804	74	-13.599	106	0.563	138	13.912	170	-1.608	202	3.359	234
-13.539	11	-2.427	43	-0.822	75	-3.429	107	2.130	139	0.707	171	-12.667	203	5.978	235
-3.189	12	-9.093	44	0.505	76	-7.783	108	14.975	140	0.641	172	0.395	204	14.806	236
-10.179	13	-8.170	45	5.283	77	-10.894	109	-1.189	141	6.051	173	3.030	205	1.678	237
-6.747	14	-2.825	46	15.660	78	-3.266	110	-2.150	142	13.415	174	15.530	206	8.462	238
-4.275	15	-13.625	47	-5.086	79	-12.746	111	-13.723	143	-3.642	175	-0.301	207	8.926	239
-14.921	16	-1.947	48	-13.365	80	-3.459	112	0.586	144	-12.388	176	0.293	208	1.404	240
-2.063	17	-2.982	49	-3.355	81	-4.712	113	2.155	145	-3.020	177	3.322	209	9.619	241
-6.500	18	-16.555	50	-6.242	82	-16.081	114	13.142	146	-2.704	178	22.649	210	5.377	242
-13.546	19	0.582	51	-13.824	83	-1.068	115	-0.005	147	-15.423	179	-4.800	211	-0.387	243
-1.407	20	-0.474	52	-2.143	84	-6.071	116	-1.799	148	0.118	180	-10.724	212	-0.328	244
-8.631	21	-4.783	53	-9.854	85	-14.642	117	-11.105	149	-0.638	181	-5.872	213	-3.865	245
-8.103	22	-18.253	54	-6.498	86	-0.127	118	-1.719	150	-6.060	182	-4.041	214	-21.839	246
-0.565	23	4.446	55	-2.360	87	-3.965	119	2.238	151	-13.150	183	-15.381	215	6.136	247
-5.761	24	12.375	56	-13.293	88	-19.939	120	12.087	152	3.947	184	-1.539	216	12.327	248
-15.182	25	3.757	57	-1.459	89	2.511	121	1.355	153	12.990	185	-5.567	217	4.751	249
2.414	26	4.356	58	-1.262	90	5.399	122	-1.043	154	2.758	186	-15.635	218	7.540	250
7.983	27	16.223	59	-10.067	91	16.592	123	-5.939	155	3.525	187	-0.499	219	11.254	251
9.230	28	0.937	60	-3.001	92	-0.961	124	-14.096	156	16.740	188	-5.344	220	4.014	252
-0.425	29	5.326	61	2.578	93	-0.606	125	3.305	157	-0.002	189	-16.634	221	12.421	253
0.280	30	16.388	62	12.770	94	-6.039	126	11.404	158	2.375	190	1.256	222	4.339	254
3.305	31	-0.226	63	1.216	95	-12.049	127	3.773	159	16.636	191	2.007	223	6.295	255

Using the above table, Step 4 resulted is the following S-box:

246	120	71	54	165	194	68	221	50	114	200	218	227	179	215	25
16	117	233	42	156	83	143	230	47	106	19	11	80	88	183	101
111	203	176	6	37	127	34	01	39	8	149	3	109	212	13	91
85	44	103	104	21	45	22	108	36	0	14	5	18	86	229	100
4	82	33	116	182	126	10	155	213	9	24	226	217	41	220	67
35	79	40	164	211	53	113	2	105	15	102	7	214	119	245	38
175	225	112	107	199	81	110	12	66	70	177	92	49	193	46	178
43	87	142	84	17	192	48	148	150	202	216	232	89	20	90	141
115	154	124	163	228	75	181	125	23	219	52	29	243	244	207	63
201	96	118	147	189	180	69	30	208	198	231	204	97	76	138	51
144	172	171	60	95	222	162	153	240	195	196	237	160	223	139	145
151	169	190	26	133	121	93	64	186	72	136	205	130	157	31	209
234	187	166	57	159	184	252	254	58	55	168	135	249	73	128	77
61	242	122	98	235	173	247	255	131	250	27	238	239	28	241	167
134	251	158	197	152	161	248	56	253	132	99	94	185	146	174	129
170	236	140	137	206	78	59	62	123	191	188	224	74	65	210	32

TABLE 3.1: S-box after Step 4 of Algorithm 3.2.1

Applying Step 5, we get the updated S-box as follows:

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

TABLE 3.2: S-box after Step 5 of Algorithm 3.2.1

Finally Step 6 resulted in following S-box:

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

TABLE 3.3: S-box after Step 6 of Algorithm 3.2.1

The S-box given in Table 3.3 is obtained by the implementation of Algorithm 3.2.1. The properties of this S-box are then investigated through S-box Evaluation Tool SET [55]. The results are summerized as given below:

- S-Box is balanced.
- Non-Linearity (NL_f)is 104.
- Correlation Immunity (CI) is 0.
- Absolute Indicator (Δ_h) is 104.
- Sum of Square Indicator (σ_h) is 302464.
- Algebraic Degree (deg_h)is 7.
- Algebraic Immunity (AI) is 4.
- Transparency Order (T_G) is 7.793.
- Number of Fixed Points (F_p) is 2.
- Number of Opposite Fixed Points (OF_p) is 0.

Further, we applied an affine mapping on S-box to remove fixed points with the help of ApCoCoa [3].

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

TABLE 3.4: S-box after an affine mapping

The S-box given in Table 3.4 is obtained by the implementation of an affine mapping. The properties of this S-box are then investigated through S-box Evaluation Tool SET [55]. The results are summerized and given below:

- S-Box is balanced.
- Non-Linearity (NL_f)is 104.
- Correlation Immunity (CI) is 0.
- Absolute Indicator (Δ_h) is 104.
- Sum of Square Indicator (σ_h) is 302464.
- Algebraic Degree (deg_h)is 7.
- Algebraic Immunity (AI) is 4.
- Transparency Order (T_G) is 7.817.
- Number of Fixed Points (F_p) is 0.
- Number of Opposite Fixed Points (OF_p) is 1.

3.3 A New Method For Chaotic S-Box

In this section we will describe two systems which involve in new Chaotic System.

- Rossler System
- Lorenz System

3.3.1 Rossler System

Rossler system is quadratic non-linear system of equations introduced by Rossler in 1970. Rossler was influenced by three dimension geometry flows. So that Rossler introduced system of equations with parameters $g = 0.2, r = 0.2$ and $e = 5.7$. Rossler system is one of his well known system [48]. Some feature of Rossler system can be obtained by linear methods such as eigen vector but main property of Rossler system deduced by non-linear methods.
The set of solutions of Rossler system are called Rossler attractor [48]. Rossler attractor designed by Otto Rossler 1976. Rossler attractor also behave like Lorenz attractor.

The system of Rossler equations are:

$$\begin{cases} \dfrac{dx}{dt} &= -y - z \\ \dfrac{dy}{dt} &= (x + gy) \\ \dfrac{dz}{dt} &= r + z(x - e) \end{cases} \tag{3.2}$$

where $x = y = z = 0$ are the initial conditions, $g = r = 0.2, e = 5.7$ are the parameters. The Rossler attractors become more simple for its two equations when $z = 0$, then we will examine its behaviour in xy-plane [29].

$$\begin{cases} \dfrac{dx}{dt} &= -y - z \\ \dfrac{dy}{dt} &= (x + gy) \end{cases} \tag{3.3}$$

Now, write up the second system (3.1.2) which involve in new Chaotic System . The system of Lorenz equations are:

$$\begin{cases} \dfrac{dx}{dt} &= a(y-x) \\ \dfrac{dy}{dt} &= (bx - y - xz) \\ \dfrac{dz}{dt} &= (xy - cz) \end{cases} \tag{3.4}$$

where $a = 10, b = 28, c = \frac{8}{3}$ are the parameters and $x = 0.1, y = 0, z = 0$ are the initial conditions. By adding two chaotic systems (3.2), (3.4) we get a new Chaotic System

$$\begin{cases} \dfrac{dx}{dt} &= a(y-x) - y - z \\ \dfrac{dy}{dt} &= bx - y - 20xz + x + gy \\ \dfrac{dz}{dt} &= 5xy - cz + r + z(x-e) \end{cases} \tag{3.5}$$

where $a = 10, b = 28, c = \frac{8}{3}$, $g = r = 0.2$ and $e = 5.7$ [30] are the parameters By solving new chaotic system we get number of solutions, which plot a butterfly image as shown in the figure 3.2

3.4 S-box and New Chaotic System

The following steps will be processed to design S-box. This Algorithm also consist of two phases, Diffusion is perform in first phase and Substitution operation apply in second phase. The detail is given below:

Algorithm 3.4.1. New Chaotic S-Box

Input: New Chaotic System (3.5)

Output: Chaotic S-box

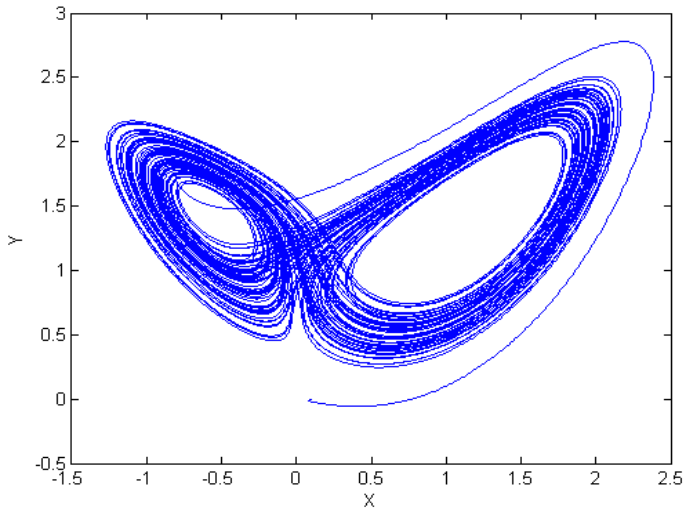

FIGURE 3.2: New Chaotic System
[30]

1. Obtain system trajectories by solving the New Chaotic System (3.5) by using the MATLAB tool **ode45** with the following parameters:

$$a = 10, \quad b = 28, \quad c = \frac{8}{3}, \quad g = 0.2, \quad r = 0.2, \quad e = 5.7.$$

2. After ignoring some initial values, get a sample of 256 data points from all the points on these trajectories after a fixed step of size $h = \frac{\text{Total number of data points}}{256}$.

3. Assign a code from $0 - 255$ to the sampled 256 data points.

4. Sort the above codes in ascending order with respect the corresponding data points. In this way, initialize the proposed S-box by writing the sorted codes in the form of a 16×16 table.

5. For each row number i $(1 \leq i \leq 16)$, rotate the row by applying a left shift of $(i - 1)$ points to update the above S-box.

6. For each column number j $(1 \leq j \leq 16)$, rotate the column by applying a downward shift of $(16 - j)$ points to update the above S-box.

3.4.1 Implementation and Results

Using the MATLAB tool **ode45** [51] and ApCoCoA [3], we implemented the above algorithm 3.4.1 and generated an S-box as follows:

For the first step of algorithm 3.4.1, we have computed 100,000 y values using MATLAB code given in Appendix A. The initial conditions for the system are taken as $x = 0.1$, $y = 0$ and $z = 0$. After ignoring starting 9984 (*i.e.* 39×256) values for the transitions to die out, the following data is obtained after applying Step 3 of the algorithm.

Data	Code	Data	Code	Data	Code	Data	Code	Data	Code	Data	Code	Data	Code	Data	Code
-1.495	0	-0.835	32	-0.418	64	-0.151	96	0.001	128	0.171	160	0.388	192	1.227	224
-1.427	1	-0.827	33	-0.406	65	-0.146	97	0.001	129	0.183	161	0.388	193	1.330	225
-1.349	2	-0.822	34	-0.405	66	-0.145	98	0.007	130	0.186	162	0.403	194	1.335	226
-1.316	3	-0.806	35	-0.404	67	-0.133	99	0.017	131	0.200	163	0.414	195	1.364	227
-1.238	4	-0.802	36	-0.404	68	-0.132	100	0.018	132	0.211	164	0.439	196	1.437	228
-1.203	5	-0.771	37	-0.397	69	-0.131	101	0.020	133	0.216	165	0.443	197	1.483	229
-1.188	6	-0.768	38	-0.362	70	-0.129	102	0.021	134	0.216	166	0.457	198	1.541	230
-1.173	7	-0.727	39	-0.362	71	-0.127	103	0.022	135	0.221	167	0.474	199	1.571	231
-1.147	8	-0.702	40	-0.358	72	-0.126	104	0.023	136	0.223	168	0.489	200	1.682	232
-1.115	9	-0.695	41	-0.337	73	-0.125	105	0.038	137	0.235	169	0.492	201	1.765	233
-1.105	10	-0.683	42	-0.322	74	-0.115	106	0.039	138	0.240	170	0.509	202	1.775	234
-1.092	11	-0.673	43	-0.308	75	-0.114	107	0.067	139	0.249	171	0.518	203	1.801	235
-1.075	12	-0.665	44	-0.308	76	-0.106	108	0.074	140	0.250	172	0.539	204	1.911	236
-1.060	13	-0.663	45	-0.304	77	-0.105	109	0.078	141	0.251	173	0.582	205	1.947	237
-1.007	14	-0.661	46	-0.298	78	-0.103	110	0.079	142	0.252	174	0.588	206	2.067	238
-1.003	15	-0.622	47	-0.290	79	-0.100	111	0.079	143	0.259	175	0.602	207	2.107	239
-0.993	16	-0.589	48	-0.284	80	-0.096	112	0.086	144	0.265	176	0.656	208	2.108	240
-0.962	17	-0.572	49	-0.276	81	-0.077	113	0.094	145	0.266	177	0.669	209	2.132	241
-0.959	18	-0.556	50	-0.269	82	-0.061	114	0.097	146	0.268	178	0.715	210	2.165	242
-0.936	19	-0.529	51	-0.261	83	-0.058	115	0.098	147	0.281	179	0.735	211	2.228	243
-0.931	20	-0.522	52	-0.253	84	-0.058	116	0.105	148	0.285	180	0.757	212	2.245	244
-0.926	21	-0.521	53	-0.245	85	-0.052	117	0.108	149	0.288	181	0.809	213	2.390	245
-0.920	22	-0.516	54	-0.243	86	-0.052	118	0.112	150	0.290	182	0.823	214	2.450	246
-0.919	23	-0.506	55	-0.238	87	-0.046	119	0.113	151	0.300	183	0.868	215	2.666	247
-0.912	24	-0.504	56	-0.219	88	-0.042	120	0.121	152	0.320	184	0.901	216	2.703	248
-0.911	25	-0.458	57	-0.205	89	-0.039	121	0.125	153	0.330	185	1.023	217	2.786	249
-0.911	26	-0.455	58	-0.192	90	-0.033	122	0.131	154	0.338	186	1.035	218	2.853	250
-0.893	27	-0.446	59	-0.189	91	-0.028	123	0.133	155	0.340	187	1.041	219	3.178	251
-0.885	28	-0.440	60	-0.180	92	-0.020	124	0.142	156	0.342	188	1.110	220	3.183	252
-0.883	29	-0.437	61	-0.177	93	-0.015	125	0.155	157	0.349	189	1.126	221	3.211	253
-0.882	30	-0.435	62	-0.175	94	-0.013	126	0.160	158	0.352	190	1.126	222	3.332	254
-0.837	31	-0.432	63	-0.164	95	-0.007	127	0.163	159	0.386	191	1.163	223	3.343	255

Using the above table, Step 4 resulted is the following S-box:

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

TABLE 3.5: S-box after Step 4 of Algorithm 3.4.1

Applying Step 5, we get the updated S-box as follows:

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

TABLE 3.6: S-box after Step 5 of Algorithm 3.4.1

Finally Step 6 resulted in following S-box:

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

TABLE 3.7: S-box after Step 6 of Algorithm 3.4.1

The S-box given in Table 3.7 is obtained by the implementation of algorithm 3.2.1. The properties of this S-box are then investigated through S-box Evaluation Tool SET [55]. The results are summerized as given below:

- S-Box is balanced.
- Non-Linearity (NL_f)is 98.
- Correlation Immunity (CI) is 0.
- Absolute Indicator (Δ_h) is 104.
- Sum of Square Indicator (σ_h) is 260224.
- Algebraic Degree (deg_h)is 7.
- Algebraic Immunity (AI) is 4.
- Transparency Order (T_G) is 7.790.
- Number of Fixed Points (F_p) is 1.
- Number of Opposite Fixed Points (OF_p) is 1.

Moreover, we applied an affine mapping on S-box to remove fixed points with the help of ApCoCoa [3].

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

TABLE 3.8: S-box after an affine mapping

The S-box given in Table 3.8 is obtained by the implementation of an affine mapping. The properties of this S-box are then investigated through S-box Evaluation Tool SET [55]. The results are summerized as given below:

- S-Box is balanced.
- Non-Linearity (NL_f)is 105.
- Correlation Immunity (CI) is 0.
- Absolute Indicator (Δ_h) is 104.
- Sum of Square Indicator (σ_h) is 282496.
- Algebraic Degree (deg_h)is 7.
- Algebraic Immunity (AI) is 4.
- Transparency Order (T_G) is 7.795.
- Number of Fixed Points (F_p) is 0.
- Number of Opposite Fixed Points (OF_p) is 1.

Chapter 4

Conclusion

An S-box has a major role in Symmetric Key Cryptography. Thus, it is very important to design a strong S-box. While the design of suitable S-box is a difficult job, but several criteria have been proposed which provide protection against attacks. In this chapter, to conclude our work, we will discuss the security analysis of our following proposed S-box.

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

TABLE 4.1: Proposed S-box

4.1 Criteria for a good S-box

The following properties of S-box are very essential for cryptographically strong S-box.[2]

1. Bijection.
2. Nonlinearity.
3. Strict avalanche criterion(SAC).
4. The Output bits independence criterion.

For the above S-box these properties are guarented as:

1. **Bijection:** For the bijective property the S-box must satisfy the equation (2.1) of Chapter 2. Using our code in ApCoCoA [3] (see Appendix A), we find that **wt** = 128
2. **Non-linearity:** As explained in Chapter 2, using SET tool, our S-Box has a reasonably good non-linearity of 95.
3. **Strict Avalanche Criteria:** As explained in Chapter 2, using SET tool, our S-Box has a value 0.499 which is closer to ideal value 0.5.
4. **The Output bits independence criteria:** For this property, the S-box must satisfy the (SAC) and should highly non-linear as explained in Chapter 2. So that our S-box also satisfied this property.

4.2 Performance analysis of S-box

Now we analyze the S-box 4.1 by comparing with the other S-boxes that are given in literature[7, 56, 58]. We will compare some properties of proposed S-box with Chen's, Tang's and Wang's S-boxes. These S-boxes are given below:

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

TABLE 4.2: Chen S-box

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

TABLE 4.3: Tang S-box

204	20	35	175	254	121	225	44	195	68	83	85	216	156	180	138
96	93	20t2	219	80	106	108	64	217	218	176	56	22	243	8	36
151	237	69	220	143	213	127	17	40	214	78	215	28	29	183	249
255	164	94	133	23	170	196	181	124	189	74	61	155	104	2	45
89	158	66	197	233	139	11	252	247	187	174	122	84	163	131	140
171	234	244	160	223	37	150	221	190	182	32	82	146	43	62	129
79	235	70	60	200	109	227	201	167	34	169	102	222	16	21	193
232	14	87	168	236	113	41	212	67	125	120	154	75	57	152	205
86	63	107	90	245	132	194	73	207	48	50	101	25	103	33	231
242	0	210	173	159	166	161	157	58	141	130	177	179	229	250	162
153	188	211	52	149	145	246	238	208	9	199	117	118	91	142	137
178	72	4	165	115	31	134	71	251	54	128	5	98	147	92	116
240	114	12	191	203	111	198	172	206	19	135	253	88	110	30	99
6	65	76	95	97	3	1	248	148	186	226	13	49	209	185	59
77	38	42	144	230	192	55	119	81	239	7	112	105	51	27	39
184	224	228	241	123	136	46	47	18	24	53	26	100	126	15	10

TABLE 4.4: Wang S-box

1. In order to see the bijection of above mentioned S-boxes , we have to calculate hamming weight by using the equation (2.1). The resultant values of all the S-boxes are 128 which is evident, that all the S-boxes 4.1,4.2,4.3 and 4.4 are bijective.

2. Additionally we will check the non-linearity [39] of S-boxes. As per calculations, we conclude that the non linearity of proposed S-box 4.1 is 105, which is similar to the non-linearity of Chen S-Box 4.2 and Tang S-Box 4.3. But another researcher, Wang's non-linearity of S-Box 4.4 is 103, which is less than in comparison to the non-linearity of our S-box 3.8

3. Moreover, we have examined the strict avalanch criterion, the value of present research's S-box is 0.499 which is closer to the ideal value 0.5 provided in previous studies that is compare to the value of Chen's S-Box 4.2,Tang's S-Box 4.3 and Wang's S-Box 4.4 *i.e* 0.499, 0.4995 and 0.4972 respectively[7]. Thus our S-box4.1 satisfies Strict avalanche criterion (SAC).

4. All of the above S-boxes also satisfy (BIC), as to fullfil this property, S-box should be non-linear and at the same time must satisfy (SAC) as welled. So in current study these two properties are satisfied by all the S-boxes.

It is concluded that, the S-box of this study meets the required properties, which proves that our proposed S-box is cryptographically strong and can resist against cryptanalysis attacks.

Appendix A

Lorenz System

A.1 Matlab code for Lorenz System

```
clear all;
close all
clc;
clf
format long

global aa bb cc
aa=10.0;
bb =28.0;
cc=8/3;
y = [0.1; 0.0; 0.0];        % initial conditions

T = 100;                        % end time point

%tspan = 0:40/30000:40;         % time mesh
t=linspace(0,T,100000);          % time mesh (alternate)

options = odeset('Reltol',1e-8,'Abstol',1e-8,'stats','on');
[tout yout] = ode45('loren',t,y,options);

sol=[tout yout];

```

```
figure(3)
plot(yout(:,1),yout(:,3));

for i=40:296;
%disp(yout(i*256,1))      % x values at 256th discrete time poits
disp(yout(i*256,2))     % y values at 256th discrete time poits
%disp(yout(i*256,3))      % z values at 256th discrete time poits
end
break
```

Listing A.1: Matlab Code for Lorenz System

Appendix B

Properties of S-Box

B.1 Code for Bijective property of S-box

```
Define Biject(Sb,N)
  If Len(Sb) <> 2^N Then Error("Wrong Inputs"); EndIf;
  BinSb:=[];
  Foreach Trm In Sb Do
    Append(BinSb,Dec2Bin(Trm,N));
  EndForeach;
  Ais:=[];
  For I:=1 To 2^N Do
    Append(Ais,Dec2Bin(I-1,N));
  EndFor;
  PrintLn("Inputs",Ais);
  PrintLn("Outputs",BinSb);
  V:=[];
  For I:=1 To 2^N Do
    Append(V,Mod(ScalarProduct(Ais[I],BinSb[I]) ,2));
  EndFor;
  PrintLn("Vector AiFi=",V);
  PrintLn("Its Weight is: ");
  Return Len(NonZero(V))$;
```

Listing B.1: Bijective Property Using ApCoCoA

Bibliography

[1] C. Adams, S. Tavares, "Good S-boxes are easy to find". In Conference on the Theory and Application of Cryptology (pp. 612-615). Springer New York (1989).

[2] C. Adams, S. Tavares, "The structured design of cryptographically good S-boxes". journal of Cryptology, 3(1), 27-41 (1990).

[3] I. S. Anja, Moldenhauer, Rosenberger, Gerhard, Rosenthal, Kristina, "On the Tits alternative for a class of finitely presented groups with a special focus on symbolic computations" (2016).

[4] B. Ann, "Cryptographic properties of Boolean functions and S-boxes". Diss. phd thesis-2006, (2006).

[5] E. Brow, "Elliptic Curve Cryptography", Math 189A; Algebraic Geometry, December 2010.

[6] S. Bruce, J. Kelsey, D. Whiting, D. Wagner, C. Hall, N. Ferguson, T. Kohno, and M. Stay, "The Twofish teams final comments on AES Selection". AES round 2 (2000).

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

[8] M. Christopher Danforth, "Chaos in an Atmosphere Hanging on a Wall". Mathematics of Planet Earth 2013. Retrieved 4 April (2013).

[9] C. Claude, "Boolean functions for cryptography and error correcting codes", Boolean models and methods in mathematics, computer science, and engineering 2 : 257 (2010).

[10] S. Claude, "Communication theory of secrecy systems". Bell Labs Technical Journal 28.4 (1949).

[11] C. Claude, "Vectorial Boolean functions for cryptography". Boolean models and methods in mathematics, computer science, and engineering 134 : 398-469 (2010).

[12] T. R. Core. "R: A language and environment for statistical computing". R Foundation for Statistical Computing, Vienna, Austria. 2013." (2014).

[13] A. Cubero, J. Antonio, and P. J. Zufiria, "A c++ class for analysing vector boolean functions from a cryptographic perspective". In: Proceedings of the 2010 International Conference on Security and Cryptography (SECRYPT), pp. 19 (2010).

[14] J. Daemen, V. Rijmen, "The design of Rijndael: AES-the advanced encryption standard". Springer Science & Business Media (2013).

[15] S. S. Deva, and C. P. Arya. "Algebraic Construction and Cryptographic Properties of Rijndael Substitution Box". Defence Science Journal 62.1 : 32-37 (2012).

[16] F. Diacu, P. Holmes, "Celestial Encounters: The Origins of Chaos and Stability". Princeton University Press (1996).

[17] A. J. Elbirt, "Understanding and applying cryptography and data security". CRC press (2009).

[18] F. Eric, and F. R. Ferard, F. Rodier "On the nonlinearity of Boolean functions". Proceedings of WCC2003, Workshop on coding and cryptography. (2003).

[19] B. Feng, R. H. Deng, W. Geiselmann, C. Schnorr, R. Steinwandt, and H. Wu. "Cryptanalysis of two sparse polynomial based public key cryptosystems". International Workshop on Public Key Cryptography. Springer Berlin Heidelberg, (2001).

[20] B. J. Fino, V. R. Algazi, "Unified Matrix Treatment of the Fast Walsh-Hadamard Transform". IEEE Transactions on Computers. 25 (11): 11421146. doi:10.1109/TC.1976.1674569 (1976).

[21] L. Frdric, "The boolfun Package: Cryptographic Properties of Boolean Functions." (2012).

[22] C. Guo, Y. Chen, and X. Liao, "An extended method for obtaining S-boxes based on three-dimensional chaotic Baker maps". Chaos, Solitons , Fractals 31.3 : 571-579 (2007).

[23] H. Haken, "Analogy between higher instabilities in fluids and lasers". Physics Letters A. 53 (1): 7778. Bibcode:1975PhLA...53...77H. doi:10.1016/0375-9601(75)90353-9 (1975).

[24] S. Kaspar, "Computing border bases without using a term ordering" (2013).

[25] L. D. Kiel, E. W. Elliott, "Chaos theory in the social sciences: Foundations and applications". University of Michigan Press (1996).

[26] L. R. Knudsen, & M. Robshaw, "The block cipher companion. Springer Science & Business Media" (2011).

[27] R. Lars , Knudsen, "Contemporary Block Ciphers". Lectures on Data Security 1998.

[28] G. A. Leonov, N. V. Kuznetsov, N. A. Korzhemanova, D. V. Kusakin, "Lyapunov dimension formula for the global attractor of the Lorenz system". Communications in Nonlinear Science and Numerical Simulation. 41: 84103. doi:10.1016/j.cnsns.2016.04.032 (2016).

[29] C. Letellier, P. Dutertre, B. Maheu, "Unstable periodic orbits and templates of the Rssler system: toward a systematic topological characterization". Chaos. 5 (1): 272281. doi:10.1063/1.166076 (1995).

[30] C. Letellier, V. Messager, "Influences on Otto E. Rssler's earliest paper on chaos". International Journal of Bifurcation & Chaos. 20 (11): 35853616. doi:10.1142/s0218127410027854 (2010).

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

[32] A. Mamadolimov, H. Isa, & M. S. Mohamad, "Practical bijective S-box design". arXiv preprint arXiv:1301.4723 (2013).

[33] S. Martin. "On cryptographic properties of random Boolean functions". Journal of Universal Computer Science 4.8 : 705-717 (1998).

[34] A. P. Menezes , P. Van Oorschot , & S. A. Vanstone, "Handbook of applied cryptography". CRC press (1996).

[35] S. Mister C. Adams. "Practical S-Box Design" (PostScript). Workshop on Selected Areas in Cryptography (SAC '96) Workshop Record. Queen's University. pp. 6176. Retrieved 2007-02-20 (1996).

[36] K. Mohamed, M. N. M. Pauzi, F. H. H. M. Ali, S. Ariffin, & N. H. N. Zulkipli, "Study of S-box properties in block cipher". In Computer, Communications, and Control Technology (I4CT), 2014 International Conference on (pp. 362-366). IEEE (2014).

[37] Y. Nawaz, "Design of stream ciphers and cryptographic properties of nonlinear functions" (2007).

[38] O. Nelles, "Nonlinear system identification: from classical approaches to neural networks and fuzzy models". Springer Science & Business Media (2013).

[39] K. Nomizu, S. Sasaki, "Affine Differential Geometry" (New ed.), Cambridge University Press, ISBN 978-0-521-44177-3 (1994).

[40] F. Omori, "On the aftershocks of earthquakes". Journal of the College of Science, Imperial University of Tokyo. 7: 111200 (1894).

[41] F. Özkaynak, & A. B. Özer, "A method for designing strong S-Boxes based on chaotic Lorenz system". Physics Letters A, 374(36), 3733-3738 (2010).

[42] C. Paar, & J. Pelzl, "Understanding cryptography: a textbook for students and practitioners". Springer Science & Business Media (2009).

[43] S. Paul, Addison, "Fractals and chaos: an illustrated course". CRC Press. pp. 4446. ISBN 978-0-7503-0400-9. Retrieved 2011-02-05 (1997).

[44] A. N. Pchelintsev, "Numerical and Physical Modeling of the Dynamics of the Lorenz System". Numerical Analysis and Applications. 7 (2): 159167 (2014).

[45] S. Rashmi, and S. Kumar, "Elgamals algorithm in cryptography". International Journal of Scientific , Engineering Research 3.12 : 1-4 (2012).

[46] C. Robert , Hilborn, "Chaos and Nonlinear Dynamics: An Introduction for Scientists and Engineers" (second ed.), Oxford University Press. ISBN 978-0-19-850723-9 (2000).

[47] R. Ronald L. A. Shamir, and L. Adleman. "A method for obtaining digital signatures and public-key cryptosystems". Communications of the ACM 21.2 : 120-126 (1978).

[48] O. E. Rssler, "An Equation for Continuous Chaos", Physics Letters, 57A (5): 397398, doi:10.1016/0375-9601(76)90101-8 (1976).

[49] O. E. Rssler, "An Equation for Hyperchaos", Physics Letters, 71A (2,3): 155157 (1979).

[50] B. Schneier, . "Applied Cryptography, Second Edition. John Wiley & Sons". pp. 296298, 349 ISBN (1996).

[51] N. A. F. Senan, " A brief introduction to using ode45 in MATLAB" (2012).

[52] C.Sparrow, "The Lorenz Equations: Bifurcations, Chaos, and Strange Attractors". Springer (1982).

[53] W. A. Stein, "Sage Mathematics Software" (Version 5.10). The Sage Development Team , http://www.sagemath.org (2013).

[54] H. Stephen , Kellert, "In the Wake of Chaos: Unpredictable Order in Dynamical Systems". University of Chicago Press. p. 32. ISBN 0-226-42976-8 (1993).

[55] P. Stjepan, L. Batina, D. Jakobovi, B. Ege, and M. Golub. "S-box, SET, match: a toolbox for S-box analysis." In IFIP International Workshop on Information Security Theory and Practice, pp. 140-149. Springer Berlin Heidelberg, (2014).

[56] G. Tang, X. Liao, & Y. Chen, "A novel method for designing S-boxes based on chaotic maps". Chaos, Solitons & Fractals, 23(2), 413-419 (2005).

[57] W. Thomas Cusick, & Stanica, Pantelimon, "Cryptographic Boolean functions and applications". Academic Press. ISBN 9780123748904 (2009).

[58] Y. Wang, K. W. Wong, C. Li, & Y. Li, "A novel method to design S-box based on chaotic map and genetic algorithm". Physics Letters A, 376(6), 827-833 (2012).

[59] S. William, "Cryptography and network security: principles and practices". Pearson Education India, (2006).

[60] G. Wills. "Henry Adams and The Making of America". Boston: Houghton Mifflin Co, (2005).

[61] C. K. Wu, & D. Feng, "Boolean functions and their applications in cryptography". Springer (2016).