Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 45 No. 1 (2024)

A Fuzzy Graph Theory Approach to Symmetric Key Cryptography

C. Ruby Sharmila[1], S. Meenakshi[2]

[1] Research Scholar, Department of Mathematics, Vels Institute of Science Technology and Advanced Studies, Chennai, Tamil Nadu.E-mail: sharmi.ruby2011@gmail.com

[2]Head &Associate Professor, Department of Mathematics, Vels Institute of Science Technology and Advanced Studies, Chennai, Tamil Nadu.

Abstract:

Today's network relies heavily on cryptography. While exchanging data over a network, data security is a major concern. A high level of confidentiality is required for the data. When transmitting data from one party to another, cryptography hides or manipulates the original data so that no third party can translate it. Today's society relies more on digital technology, which is utilized for a variety of purposes including banking, passwords, e-commerce, etc. Given that graphs are easily transformed into matrices, graph theory is used in the subject of cryptography. In this paper, a modified affine cipher and fuzzy graph are used to convert the message with the symmetric key.

Keywords: Encryption, Decryption, Symmetric key, Fuzzy Graph, Graph theory

1. Introduction:

Nowadays with the growth of the network, data are being exchanged widely over the networks. The basic need in every growing field is communication. Everyone wishes to protect his or her data to be safe and secure whether it is personal or professional. In order to conceal the contents of the original message and ensure that only the intended parties can read and handle the data, cryptography involves transforming the communication into an unintelligible form.The one who transmits the communication and the one who accepts it is, respectively, the sender and the receiver. A perpetrator makes an unlawful attempt to intercept the communication.The hidden or concealed communication is known as a stream cipher, whereas the original statement sent is known as the mainstream. Encryption is the procedure of transcribing plain text to ciphertext, as well as decryption is the converse.

There are various types of cryptographic methods exists which is used to encrypt or hide the secret message. The cryptographic method uses a ‘key’, which is used to hide the secret message. Key acts as a password to lock and unlock the secret message. A cryptographic algorithm is classified into two types based on the type of key it utilizes. Algorithms with symmetric and asymmetric keys. The sender and recipient share the same key in the symmetric key algorithm. Before transmitting the secret message, the sender uses the key to hide or encrypt it. Later, the recipient decrypts the secret message using the same key. In an asymmetric key algorithm sender and receiver shares different key. In this case, "secret key" and "shared key" passwords were both utilized. The stream cipher is encrypted by the sender using the "secret key" before being sent, and decrypted by the recipient using the "shared key".

[2] Discusses a novel method for using graph theory in cryptography. An Euler graph is created from the first message. The graph is then split up into several matrices and given to the recipient. With the help of a modified affine cipher, [4] converts each character in the mainstream into a numeric value that is then plotted into a graph and sent to the recipient. In [1,3], a new encryption technique that divides the mainstream into numerous blocks is presented. These blocks are then turned into graphs, and the recipient is given a copy of the graphs. In [5], a new algorithm for symmetric key cryptography is utilized to encrypt and decrypt the communication. This paper

proposes a new system in which, Fuzzy graph is used to represent the ciphertext, then for additional security, by multiplying these matrices with the symmetric secret key, the graphs are first transformed into a number of matrices and then transferred to the recipient.
The rest of the manuscript is composed of 2. Preliminaries 3. Proposed Methodology 4. Conclusion.

2. Preliminaries:

Affine Cipher:

This is a monoalphabetic substitution cipher with numeric values for each alphabet. It is useful for encryption and decryption. As a result, each letter will be replaced by another letter through the use of a substitution cipher. The alphabets corresponding numeric values are as follows:

Table 1: Encoding Table

A	B	C	D	E	F	G	H	I	J	K	L	M
0	1	2	3	4	5	6	7	8	9	10	11	12
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
13	14	15	16	17	18	19	20	21	22	23	24	25

Graph:

A set V of vertices (also defined as nodes) and a set E of edges constitute a graph, G = (V, E).

Complete Graph:

A complete graph consists of a graph where every vertex is connected with each other vertex. Kn stands for a complete graph of order n.

Adjacency Matrix:

A simple labeled graph's adjacency matrix, also known as the connection matrix, is a matrix with rows and columns labeled by graph vertices. If there is an edge between two vertices, then the value is considered to be 1, else it is considered to be 0. The adjacency matrix needs to contain 0s on the diagonal for a straightforward graph without self-loops. The adjacency matrix for an undirected graph is symmetric.

Fuzzy Graph:

The pair G : (σ, μ) denotes a fuzzy graph with V as the underlying set, where σ: V → [0, 1] is a fuzzy subset and μ: V × V → [0, 1] is a fuzzy relation on σ such that μ(x, y) ≤ σ(x) ∧ σ(y) for any x, y ∈ V, where ∧ denotes for the minimum.σ the *fuzzy vertex set* of *G* and μ the *fuzzy edge set* of *G*, respectively.

Complete Fuzzy Graph:

If μ(x, y) = σ(x)∧σ(y) for any x, y ∈ V, then a fuzzy graph G: (σ, μ) is complete.

3. PROPOSED WORK:

In this paper, we have developed new techniques for data security using cryptography concepts to provide better security, integrity, accuracy, privacy, and confidentiality of data when stored in public networks. The system aims to conceal the original message so that brute-force attacks cannot access it without keys.

Encryption Process:

Step 1: Using the encoding table, transform each standard character into a numeric value x.

Step 2: Establish the values of a and b that satisfy the criteria $GCD\ (a,m) = 1$ and $0 \leq a,b \leq m$,

where m=26.

Step 3: Obtain the letter *E(y)* corresponding to the value y, where $y = (ax + b) \bmod m$.

Step 4: Get the digit Z, wherein Z is the ASCII representation of the letter E(y).

Table 2: ASCII value

A	B	C	D	E	F	G	H	I	J	K	L	M
65	66	67	68	69	70	71	72	73	74	75	76	77
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
78	79	80	81	82	83	84	85	86	87	88	89	90

Step 5: Obtain the letter E (*r*) corresponding to each value $(Z- d) \bmod m$using the encoding table, where d is the variation between the encoding table's highest and lowest indices.

Step 6: For the letters E (*r*), obtain the corresponding membership value by

M(*r*) = encoding value of E (*r*) / 26.

Step 7:Make a number of blocks of size n-1 out of these M (r). If block size < n-1 then impose padding membership value to fulfill the data size.

Step 8:Make each block appear as fuzzy graphs, by using each block's membership value as the vertex set.

Step 9: Ensure that the fuzzy graph is complete.

Step 10: Choose a unique character for every block. The unique character is the letter that represents the summation of the elements in K if it is an initial block. The unique character is the final character of the preceding block it is not the starting block. This unique character should be added to the start of the previous block. Represent these unique characters by their corresponding membership value.

Step 11: Construct the corresponding adjacency matrix M.

Step 12: Deliver CT to the receiver after calculating CT = M x SK, where SK is the secret key matrix which is symmetric.

Example:

Let's see the encryption process of the message "DO THE BEST YOU CAN"

Let a = 11, b = 3, $SK = \begin{bmatrix} 1 & 6 & 14 & 20 & 24 \\ 10 & 2 & 7 & 15 & 21 \\ 17 & 11 & 3 & 8 & 16 \\ 22 & 18 & 12 & 4 & 9 \\ 25 & 23 & 19 & 12 & 5 \end{bmatrix}$ and $|SK| = 415625 \neq 0$, hence $(SK)^{-1}$ exists and n = 5.

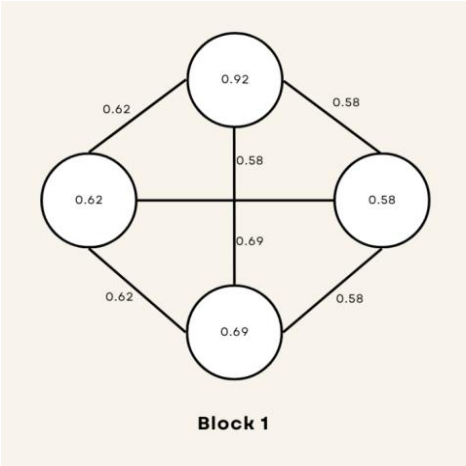
Table 3: Encryption Calculation

Mainstream	D		O	T	H	E	B	E	S	T	Y	O	U	C	A	N

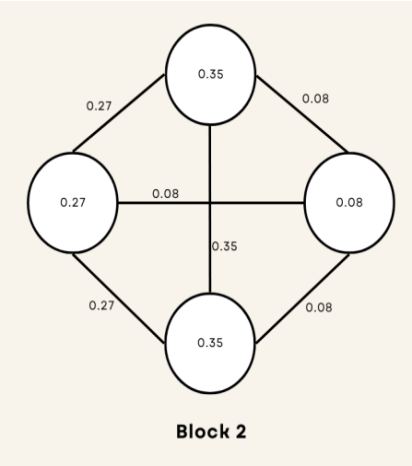
x	3		14	19	7	4	1	4	18	19	24	14	20	2	0	13
y	10		1	4	2	21	14	21	19	4	7	1	15	25	3	16
E (*y*)	K		B	E	C	V	O	V	T	E	H	B	P	Z	D	Q
Z	75		66	69	67	86	79	86	84	69	72	66	80	90	68	81
(Z-d) mod26	24		15	18	16	9	2	9	7	18	21	15	3	13	17	4
E (*r*)	Y		P	S	Q	J	C	J	H	S	V	P	D	N	R	E
M (*r*)	0.92		0.58	0.69	0.62	0.35	0.08	0.35	0.27	0.69	0.81	0.58	0.12	0.5	0.65	0.15

Block size = 4 and Number of blocks = $\frac{Plainstream}{Block\ size} = \frac{15}{4} = 3.75 \approx 4$

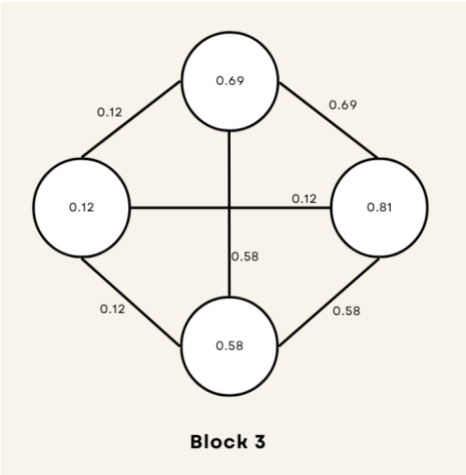
Using M(*r*) complete fuzzy graphs are obtained for each block by taking the following vertex sets V1 = {0.92, 0.58, 0.69, 0.62}, V2 = {0.35, 0.08, 0.35, 0.27}, V3 = {0.69, 0.81, 0.58, 0.12} and V4 = {0.5, 0.65, 0.15, 0}.



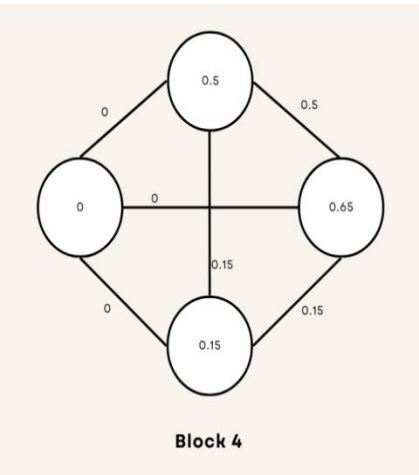
Block 1



Block 2



Block 3



Block 4

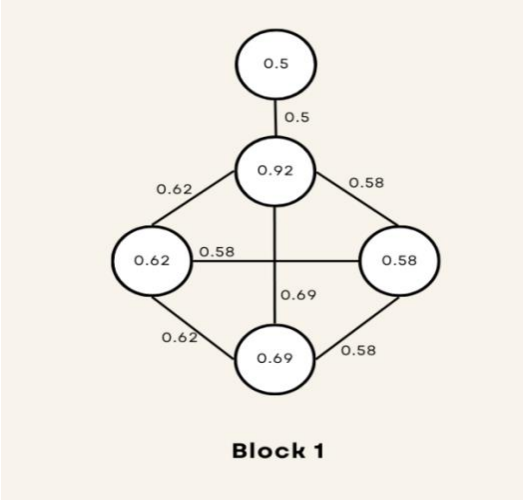
Pick out each block's unique character.

The initial block's (Block 1) unique character ≡ char (($\sum SK$) mod m)≡ char (325 mod26)= 13=N,

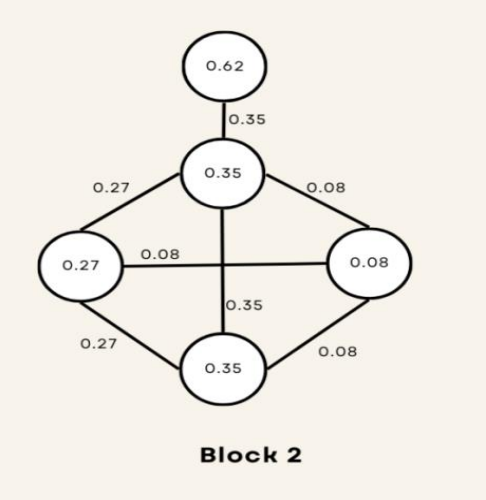
Membership value of N = 0.5.

The final character of the preceding block must be the unique characters of the other blocks and represent the unique character by its membership value.

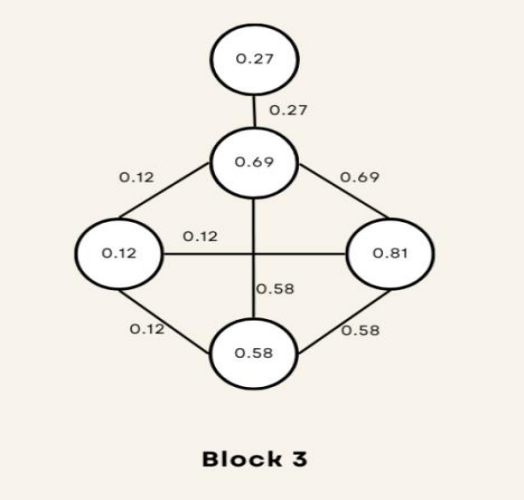
Adding the membership value to the complete fuzzy graph.



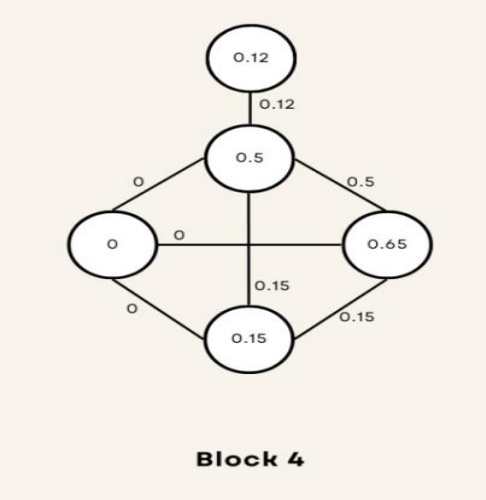
Block 1



Block 2



Block 3



Block 4

Multiply the resulting fuzzy graphs adjacency matrices M_i with the secret key matrix SK.

$CT_i = M_i \times SK$

$$CT_1 = \begin{bmatrix} 0 & 0.5 & 0 & 0 & 0 \\ 0.5 & 0 & 0.58 & 0.69 & 0.62 \\ 0 & 0.58 & 0 & 0.58 & 0.62 \\ 0 & 0.69 & 0.58 & 0 & 0.62 \\ 0 & 0.62 & 0.58 & 0.62 & 0 \end{bmatrix} \begin{bmatrix} 1 & 6 & 14 & 20 & 24 \\ 10 & 2 & 7 & 15 & 21 \\ 17 & 11 & 3 & 8 & 16 \\ 22 & 18 & 12 & 4 & 9 \\ 25 & 23 & 19 & 13 & 5 \end{bmatrix} = \begin{bmatrix} 5 & 1 & 3.5 & 7.5 & 10.5 \\ 41.04 & 36.06 & 28.8 & 25.46 & 30.59 \\ 34.06 & 25.86 & 22.8 & 19.08 & 20.5 \\ 32.26 & 22.02 & 18.35 & 23.05 & 26.87 \\ 29.7 & 18.78 & 13.52 & 16.42 & 27.88 \end{bmatrix}$$

$$CT_2 = \begin{bmatrix} 0 & 0.35 & 0 & 0 & 0 \\ 0.35 & 0 & 0.08 & 0.35 & 0.27 \\ 0 & 0.08 & 0 & 0.08 & 0.27 \\ 0 & 0.35 & 0.08 & 0 & 0.27 \\ 0 & 0.27 & 0.08 & 0.27 & 0 \end{bmatrix} \begin{bmatrix} 1 & 6 & 14 & 20 & 24 \\ 10 & 2 & 7 & 15 & 21 \\ 17 & 11 & 3 & 8 & 16 \\ 22 & 18 & 12 & 4 & 9 \\ 25 & 23 & 19 & 13 & 5 \end{bmatrix} = \begin{bmatrix} 3.5 & 0.7 & 2.45 & 5.25 & 7.35 \\ 16.16 & 15.49 & 14.47 & 12.55 & 14.18 \\ 9.31 & 7.81 & 6.65 & 5.03 & 3.75 \\ 11.61 & 7.79 & 7.82 & 9.4 & 9.98 \\ 10 & 6.28 & 5.37 & 5.77 & 9.38 \end{bmatrix}$$

$$CT_3 = \begin{bmatrix} 0 & 0.27 & 0 & 0 & 0 \\ 0.27 & 0 & 0.69 & 0.58 & 0.12 \\ 0 & 0.69 & 0 & 0.58 & 0.12 \\ 0 & 0 & 0.58 & 0 & 0.12 \\ 0 & 0.12 & 0.12 & 0.12 & 0 \end{bmatrix} \begin{bmatrix} 1 & 6 & 14 & 20 & 24 \\ 10 & 2 & 7 & 15 & 21 \\ 17 & 11 & 3 & 8 & 16 \\ 22 & 18 & 12 & 4 & 9 \\ 25 & 23 & 19 & 13 & 5 \end{bmatrix} = \begin{bmatrix} 2.7 & 0.54 & 1.89 & 4.05 & 5.67 \\ 27.76 & 22.41 & 15.09 & 14.8 & 23.34 \\ 22.66 & 14.58 & 14.07 & 14.23 & 20.31 \\ 18.66 & 10.3 & 8.08 & 14.9 & 22.06 \\ 5.88 & 3.72 & 2.64 & 3.24 & 5.52 \end{bmatrix}$$

$$CT_4 = \begin{bmatrix} 0 & 0.12 & 0 & 0 & 0 \\ 0.12 & 0 & 0.5 & 0.15 & 0 \\ 0 & 0.5 & 0 & 0.15 & 0 \\ 0 & 0.15 & 0.15 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 6 & 14 & 20 & 24 \\ 10 & 2 & 7 & 15 & 21 \\ 17 & 11 & 3 & 8 & 16 \\ 22 & 18 & 12 & 4 & 9 \\ 25 & 23 & 19 & 13 & 5 \end{bmatrix} = \begin{bmatrix} 1.2 & 0.24 & 0.84 & 1.8 & 2.52 \\ 11.92 & 8.92 & 4.98 & 7 & 12.23 \\ 8.3 & 3.7 & 5.3 & 8.1 & 11.85 \\ 4.05 & 1.95 & 1.5 & 3.45 & 5.55 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

CT_1, CT_2, CT_3, and CT_4 will be sent to the receiver.

Decryption Process:

Step 1: Receive several matrices as cipher streamCT_1, CT_2, CT_3, and CT_4.

Step 2: The matrix M is determined by multiplying CT with $(SK)^{-1}$.

Step 3: Construct a fuzzy graph with a matrix of adjacencies M.

Step 4: To get the complete fuzzy graph and the vertex set, omit the unique character membership value.

Step 5: From each vertex set, convert the membership value to equivalent characters by multiplying the membership value by 26.

Step 6: Using the encoding table, determine the encoding value "s" of the relevant characters.

Step 7: Consider p = s+ (m x q) and where $q = \begin{cases} 1, \ if \ \ s \geq 14 \\ 2, \ if \ \ s < 14 \end{cases}$.

Step 8: The letters E(y) with the ASCII value of p + d are calculated.

Step 9: Using the encoding table, determine the matching E(y) numerical value.

Step 10: Get the letters that relate to the given criterion, $a^{-1}(y-b)$ *mod m*; exclude the padding characters.

Example:

Decrypting the received message.

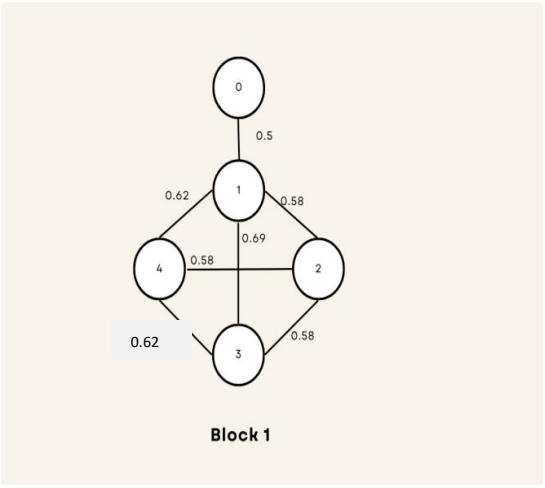
$$\text{Let S } K^{-1} = \begin{bmatrix} -0.06993 & 0.09278 & -0.02813 & 0.012174 & 0.01409 \\ 0.06842 & -0.15534 & 0.119242 & -0.03895 & 0.01259 \\ 0.00568 & 0.06613 & -0.15402 & 0.11936 & -0.02698 \\ -0.00611 & 0.00171 & 0.06577 & -0.15558 & 0.09175 \\ 0.0295 & -0.00503 & 0.0064 & 0.06926 & -0.06446 \end{bmatrix}$$

$M_i = CT_i \times (SK)^{-1}$

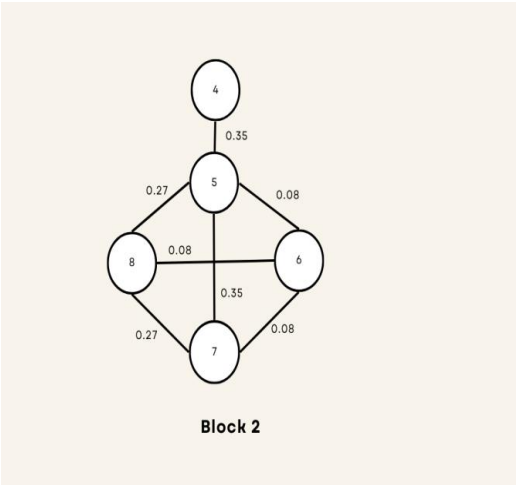
$$M_1 = \begin{bmatrix} 0 & 0.5 & 0 & 0 & 0 \\ 0.5 & 0 & 0.58 & 0.69 & 0.62 \\ 0 & 0.58 & 0 & 0.58 & 0.62 \\ 0 & 0.69 & 0.58 & 0 & 0.62 \\ 0 & 0.62 & 0.58 & 0.62 & 0 \end{bmatrix}, \quad M_2 = \begin{bmatrix} 0 & 0.35 & 0 & 0 & 0 \\ 0.35 & 0 & 0.08 & 0.35 & 0.27 \\ 0 & 0.08 & 0 & 0.08 & 0.27 \\ 0 & 0.35 & 0.08 & 0 & 0.27 \\ 0 & 0.27 & 0.08 & 0.27 & 0 \end{bmatrix}$$

$$M_3 = \begin{bmatrix} 0 & 0.27 & 0 & 0 & 0 \\ 0.28 & 0 & 0.69 & 0.58 & 0.12 \\ 0 & 0.69 & 0 & 0.58 & 0.12 \\ 0 & 0.58 & 0.58 & 0 & 0.12 \\ 0 & 0.12 & 0.12 & 0.12 & 0 \end{bmatrix}, \quad M_4 = \begin{bmatrix} 0 & 0.12 & 0 & 0 & 0 \\ 0.12 & 0 & 0.5 & 0.15 & 0 \\ 0 & 0.5 & 0 & 0.15 & 0 \\ 0 & 0.15 & 0.15 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

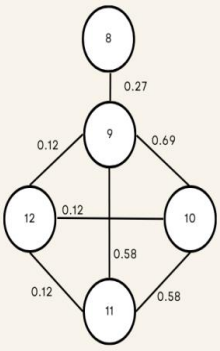
Drawing the fuzzy graphs with the adjacency matrices M_1, M_2, M_3, and M_4



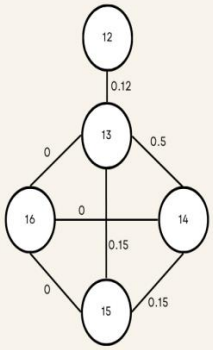
Block 1



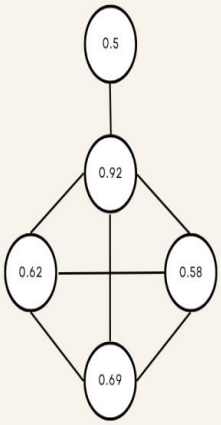
Block 2



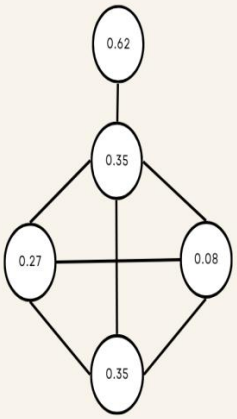
Block 3



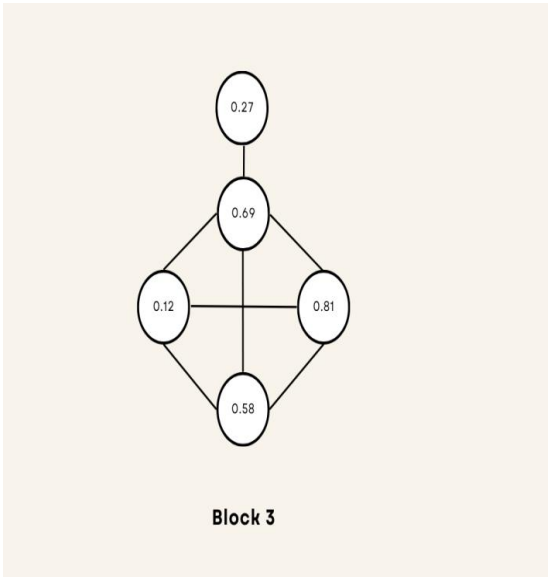
Block 4

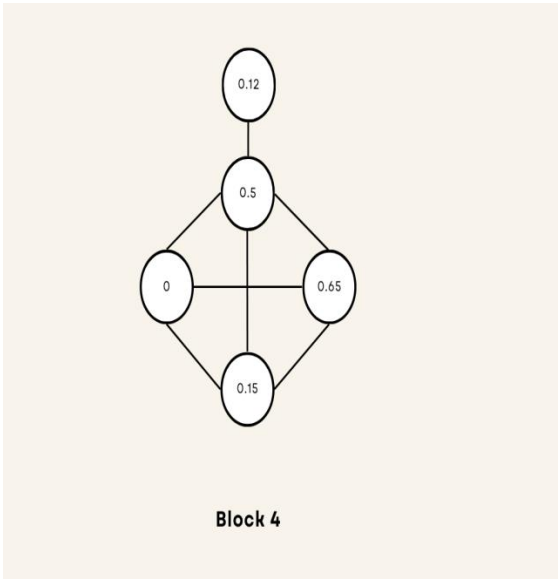


Block 1

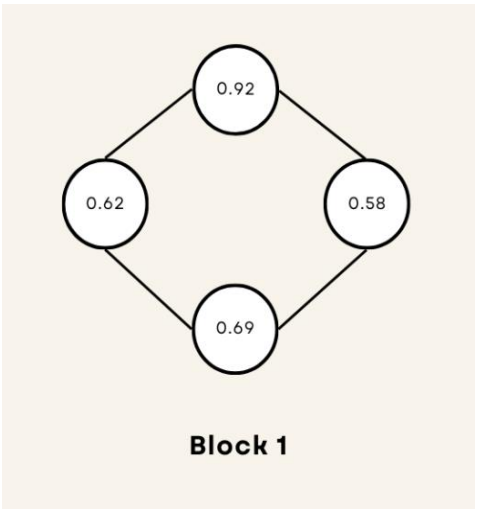


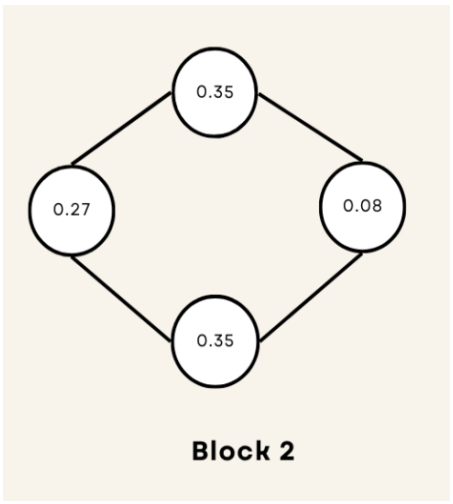
Block 2

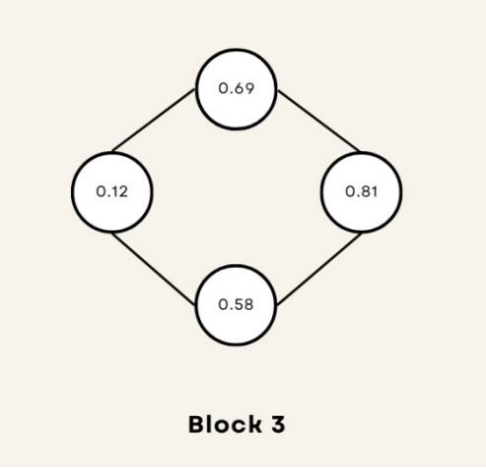




omitting the membership value ofthe unique characters.







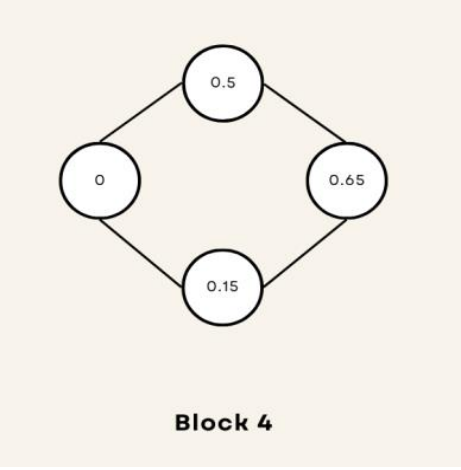


Table 4: Decryption Calculation

M(*r*)	0.92	0.58	0.69	0.62	0.35	0.08	0.35	0.27	0.69	0.81	0.58	0.12	0.5	0.65	0.15	0
E (*r*)	Y	P	S	Q	J	C	J	H	S	V	P	D	N	R	E	A
S	24	15	18	16	9	2	9	7	18	21	15	3	13	17	4	0
P	50	41	44	42	61	54	61	59	44	47	41	55	65	43	56	52
p+25	75	66	69	67	86	79	86	84	69	72	66	80	90	68	81	77
E (*y*)	K	B	E	C	V	O	V	T	E	H	B	P	Z	D	Q	M
Y	10	1	4	2	21	14	21	19	4	7	1	15	25	3	16	12
X	3	14	19	7	4	1	4	18	19	24	14	20	2	0	13	15
Plain Text	D	O	T	H	E	B	E	S	T	Y	O	U	C	A	N	P

4. Result:

To obtain the mainstream above calculations are to be done in Table 4. Ignore the last character since it is a padding character. Hence finally we get the message “DO THE BEST YOU CAN”. By applying this method, a large ciphertext is generated in comparison to the mainstream. This approach uses an extremely difficult-to-guess secret key, a matrix of order (nxn).

5. Conclusion:

In this research, a brand-new cryptographic technique is suggested for enhancing the security and confidentiality of information. Data is encrypted faster with symmetric key cryptography than with asymmetric cryptography when large amounts of data need to be encrypted. The most crucial aspect of this exchange is shared keys sent across a secured route. In this study, we provide a novel approach to tackle this issue utilizing fuzzy graphs. We calculate the membership value for the original message after converting it into encoded text through some

calculations. Using these membership values several fuzzy graphs were obtained. Further, all these fuzzy graphs are transformed into matrices. These matrices are multiplied with the secret key matrix to produce the stream cipher. An example is also used to demonstrate the encryption and decryption processes.

References:

[1] Wael Mahmoud Al Etaiwi, Encryption algorithm using graph theory, Journal of Scientific Research & Reports, 3(19):2519-2527, 2014.

[2] P. Amudha, A.C. Charles Sagayaraj, and A. C. Santha Sheela, An Application of graph theory in cryptography, International Journal of Pure and Applied Mathematics, 119(13):375 – 383.

[3] SafaaHraiz and Wael Etaiwi, Symmetric encryption algorithm using graph representation. International Conference on Information Technology (ICIT), 501-506, IEE 2017.

[4] Manisha Kumari and V.B. Kirubanad, Data encryption and decryption using graph plotting, International Journal of Civil Engineering and Technology (IJCIET) Volume, 9:36-46,2018.

[5] P.A.S.D. Peera and G.S. Wijesiri, Encryption and Decryption in Symmetric Key Cryptography Using Graph Theory, Psychology and Education 58(1):3420-3427,2021.

[6] Talal AL-Hawary, Complete Fuzzy Graphs, International J. Math. Combin. Vol.4, 26-34, 2011.

[7] Christof Paar and Jan Pe[z], Understanding cryptography: A textbook for students and practitioners, Springer Science & Business Media, 2009.

[8] Michael G. Voskoglou, TarasankarPramanik, Chapter 19, “Fuzzy Graphs and Fuzzy Hypergraphs”, IGI Global, 2020 Publication.