An encryption technique using the adjacency matrices of certain graphs with a self-invertible key matrix

P. Mohan[1*], *K.* Rajendran[1], and *A.* Rajesh[1]

[1]Department of Mathematics, Vels Institute of Science, Technology & Advanced Studies (VISTAS), India

Abstract. The most important approaches for safeguarding our communications and data nowadays are message encryption techniques. The use of the internet and network communications has increased the pace of message encryption technology development. Sharing sensitive, private messages through unsecured networks makes it possible for an attack, theft, or hacking of the messages. In order to reduce this term, cryptographic methods have been found essential. There are several symmetric enciphering methods; a few examples are the Caesar Cipher, Hill Cipher, and others. In order to generate a complex cipher text, the enciphering method described in this article encrypts and decrypts the messages given to it using a self-invertible key matrix and an adjacency matrix of certain graphs like A graph, Centipede C2, Domino graph. Since we are using the self-invertible matrix as a key matrix, whose inverse always exists, thus we can decode the ciphertext without computing the inverse of the key matrix. The lessening in computational complexity facilitates our ability to determine the inverse of a key matrix.

1 Introduction

The mathematical technique of cryptography is employed to increase the security of data transfers and to safeguard communications, data, and images from hackers. Plain text and cipher text are both written in the framework of alphabets; however, they are not even the same alphabets. Sometimes letters or messages are written using certain characters, such as punctuation, numbers, blanks, or other unusual characters. The message units in this work are encoded using the following encoded table.

Table 1. Encoded Table.

A	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**J**	**K**	**L**	**M**	**N**
1	2	3	4	5	6	7	8	9	10	11	12	13	14
O	P	1	R	T	U	V	W	X	Y	Z		**.**	**?**
15	16	17	18	20	21	22	23	24	25	26	27	28	29

* Corresponding author: mohan.phd@velsuniv.ac.in

The notions of graph theory have many applications in mathematics and the subject of mathematics, and they are a major source for cryptography [11]. Numerous applications of graph theory are made in the field of cryptography [11]. In [14], the idea of an encryption method utilising a complete graph and a Hamiltonian stroll was used with the aid of a lower triangular matrix serving as the key matrix. A novel message encoding and decoding approach employing graph labelling was detailed in [2], and the upper triangular matrix was utilised as a key matrix in both cases. The symmetric encryption strategy using cycle graph, complete graph, and minimum spanning tree was explained in [12,14]. In order to demonstrate how graph theory and cryptography are related, [8] utilised the upper triangular matrix as a key matrix. The same key, often lower and upper triangular, was used for both sender and receiver in all of the symmetric encryption techniques previously discussed. These keys are shared by both users over any type of medians. Once the intermediaries are aware of the strategy, it is simple to crack. Sharing the key matrix through an insecure channel is similarly challenging. We have suggested the new strategy to lessen this, strengthen the key, and generate more security.

The self-invertible key matrix [1,6,7] is used as the key matrix in the method that has been suggested in this study. Because of this, we can conduct the decryption procedure without having to compute the inverse of the key matrix. With the help of the provided message units as the graph vertices, the sender should first find the adjacency matrix of an A-graph [3] or Centipede [13] or Domino graph [13]. This adjacency matrix can then be multiplied with the generated self-invertible key matrix. The output was then sent to the receiver over an unsecured channel, and the receiver should use the reverse process of this he can be able to read the original message. The remaining portion of this study is defined as follows: generation of self-inverting key was detailed in Section 2. A few specific graphs are shown in Section 3, the new recommended technique is explained in Section 4, and an implementation example is shown in Section 5. Section 6 provides the conclusion and a few recommendations for further study.

2 Self-Invertible Key Matrix Generation

If $R = R^{-1}$, or $R \cdot R^{-1} = R^{-1} \cdot R = I$ under modulo p then a matrix R is said to be self-invertible matrix, construct a self-invertible matrix by doing the actions listed below. Take any arbitrary $\frac{n}{2} \times \frac{n}{2}$ matrix R_{22}. With the help of R_{22} we may compute the other $\frac{n}{2} \times \frac{n}{2}$ matrices by the following properties:

$$R_{11} + R_{22} = 0, \quad R_{12} = I - R_{11}, \qquad R_{21} = I + R_{11}$$

Following the computation of $R_{11,}R_{12}, R_{21}$, and R_{22}, the self-invertible matrix R was produced by,

$$R = \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & \cdots & \cdots & r_{1n} \\ r_{21} & r_{22} & \cdots & \cdots & r_{2n} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ r_{n1} & r_{n2} & \cdots & \cdots & r_{nn} \end{bmatrix}$$

3 Specific Graphs

Graph: An ordered pair (V, E), where V is a graph's vertices and E is its edges, is known as a graph and is nothing more than a collection of vertices and edges.

Bull Graph: The name "bull graph" refers to a basic graph with five nodes and five edges that resembles a schematic drawing of a bull or ram (whose face is represented by the triangle and horns by the graph's two bridges) [3].

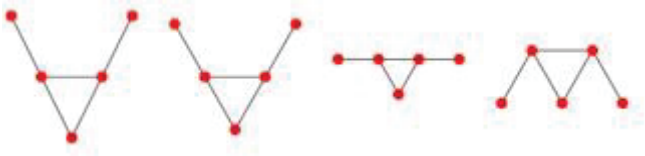

Fig. 1. The graph with six vertices shown below is the A graph. The 5-node bull graph is referred to as "A-graph" by one source [3].

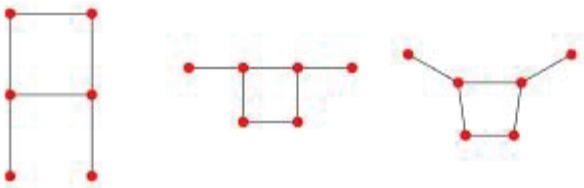

Fig. 2. The six-vertex graph shown below is an example of a domino graph. It is identical to both the (2,3)-grid graph and the 3-ladder graph.

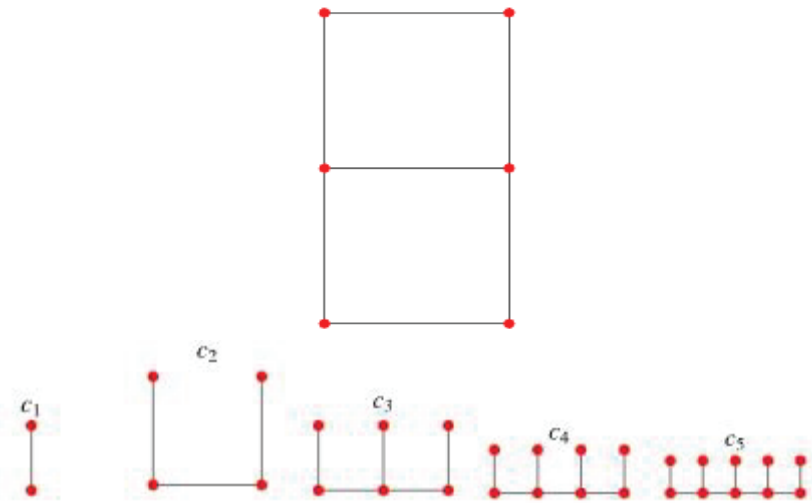

Fig. 3. The tree on 2n nodes that is created by combining the bottoms of n copies of the route graph P2 that are arranged in a row with edges is known as a "n-centipede graph," "n-centipede tree," or simply "n-centipede." As a result, it is isomorphic to the graph of (n-2) firecrackers.

4 The proposed cryptosystem

The suggested approach was described in depth in this part, and it involves employing the self-invertible key matrix as the key matrix together with the adjacency matrices of the A-graph or Centipede(C2) or Domino graphs.

4.1 Algorithm for proposed encryption technique

The steps listed below are used to perform encryption:

Step 1: In order to identify the starting letter of the given message unit, begins with the special character A. by connecting the sequential letters in the given plain text message units we create the required A-graph or Centipede (C2) or Domino graph.

Step 2: The message units are converted into their numerical equivalents using an encoded table (Table1).

Step 3: We find the numerical difference between the two adjacent vertices to compute the weights of each edge in a graph.

Step 4: The adjacency matrix for this graph was produced after taking addition modulo *p.*

Step 5: Using the commonly shared data and the procedures outlined in Section 2, the self-invertible matrix was produced.

Step 6: By multiplying the adjacency matrix with the newly produced self-invertible key matrix, the encrypted data for the original plaintext message was acquired. Finally, these encrypted matrices, the order of an adjacency matrix, and the matrix that was used to create the self-invertible matrix can be communicated with another user through an unsecured channel in either the form of row-wise or column-wise matrices.

4.2 Algorithm for proposed decryption technique

Step 1: The receiver can determine the order of the matrix, the encrypted matrix, and the matrix that aids in the generation of the self-invertible key matrix by backtracking through the information they have received.

Step 2: The receiver should create the self-invertible key matrix using the information from Section 2.

Step 3: Multiply the encrypted matrix by the self-invertible matrix that generated in step 2.

Step 4: Finally, the receiver should obtain the adjacency matrix of the required graph by taking addition modulo p for the resultant matrix of Step 3.

Step 5: After retracing the graph, the receiver can create the required graph with nodes and some specified weights.

Step 6: The message is calculated by adding the weights with their corresponding vertices. We are aware that vertex v1 is represented by the letter A and has the value 1, that v2 is equal to v1 plus the weight e1, etc.

5 Implementation example

5.1 Using A- graph

Suppose that User A(sender) wants to send the message “MANGO” to another user (User B(receiver)) using the technique which is explained in the above section, A-graph and its adjacency matrix, the key matrix that has been generated using Section 2.

Encryption- User A (The sender): Encryption is done by the following,

Initially, we should add a special character A as the beginning letter of the given plaintext message units and then convert the given message “A MAGNO” as the vertices of an A-graph. The vertices are joined from bottom to top by connecting sequential letters in the given message units.

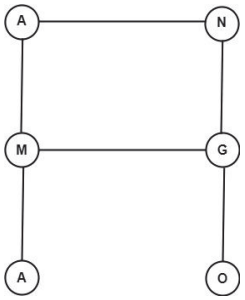

Fig. 4. A- Graph for given plaintext.

Using the encoded table (Table 1) we get, $A \to 1$, $M \to 13$, $A \to 1$, $N \to 14$, $G \to 7, O \to 15$.

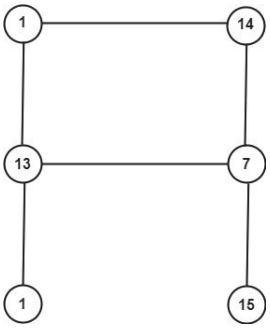

Fig. 5. Encoded A- Graph.

Weights of the edges of this graph are assigned by finding the numerical distance between the consecutive two connected vertices and then take addition modulo 29 as we are using 29 characters in the given encoded table (e1= Code M – Code A, e2 = Code A – Code M, …)

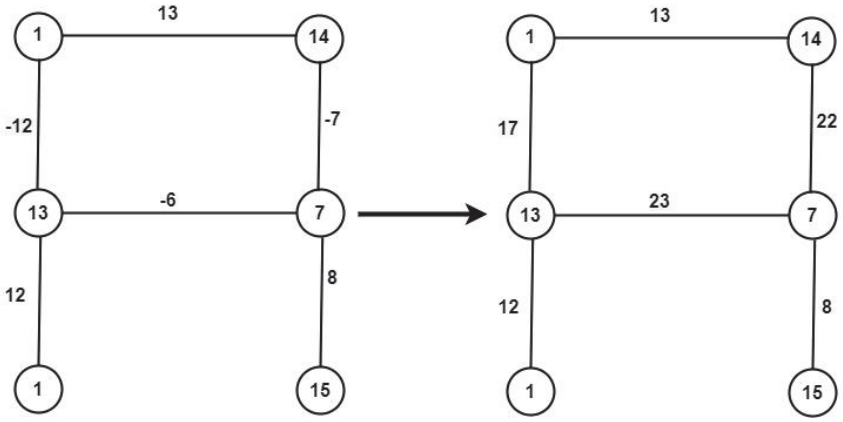

Fig. 6. Encoded A- graph with edge weights

The corresponding adjacency matrix of the above graph was computed, name it as ‘A’

$$A = \begin{bmatrix} 0 & 12 & 0 & 0 & 0 & 0 \\ 12 & 0 & 17 & 0 & 23 & 0 \\ 0 & 17 & 0 & 13 & 0 & 0 \\ 0 & 0 & 13 & 0 & 22 & 0 \\ 0 & 23 & 0 & 22 & 0 & 8 \\ 0 & 0 & 0 & 0 & 8 & 0 \end{bmatrix}$$

Now that the key matrix needs to be calculated, we use the $\frac{n}{2} \times \frac{n}{2}$ matrix R_{22} to construct the self-invertible key matrix R.

$$\text{Let } R_{22} = \begin{bmatrix} 1 & 5 & 1 \\ 4 & 9 & 6 \\ 2 & 4 & 1 \end{bmatrix} \text{ then } R_{11} = \begin{bmatrix} 28 & 24 & 28 \\ 25 & 20 & 23 \\ 27 & 25 & 28 \end{bmatrix},$$

$$R_{12} = I - R_{11} = \begin{bmatrix} 2 & 5 & 1 \\ 4 & 10 & 6 \\ 2 & 4 & 2 \end{bmatrix}, \text{ and } R_{21} = I + R_{11} = \begin{bmatrix} 0 & 24 & 28 \\ 25 & 21 & 23 \\ 27 & 25 & 0 \end{bmatrix}$$

$$\therefore R = \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix} = \begin{bmatrix} 28 & 24 & 28 & 2 & 5 & 1 \\ 25 & 20 & 23 & 4 & 10 & 6 \\ 27 & 25 & 28 & 2 & 4 & 2 \\ 0 & 24 & 28 & 1 & 5 & 1 \\ 25 & 21 & 23 & 4 & 9 & 6 \\ 27 & 25 & 0 & 2 & 4 & 1 \end{bmatrix}$$

Finally, the encrypted matrix was computed by multiplying A and R

$$C = A \cdot R = \begin{bmatrix} 0 & 12 & 0 & 0 & 0 & 0 \\ 12 & 0 & 17 & 0 & 23 & 0 \\ 0 & 17 & 0 & 13 & 0 & 0 \\ 0 & 0 & 13 & 0 & 22 & 0 \\ 0 & 23 & 0 & 22 & 0 & 8 \\ 0 & 0 & 0 & 0 & 8 & 0 \end{bmatrix} \cdot \begin{bmatrix} 28 & 24 & 28 & 2 & 5 & 1 \\ 25 & 20 & 23 & 4 & 10 & 6 \\ 27 & 25 & 28 & 2 & 4 & 2 \\ 0 & 24 & 28 & 1 & 5 & 1 \\ 25 & 21 & 23 & 4 & 9 & 6 \\ 27 & 25 & 0 & 2 & 4 & 1 \end{bmatrix}$$

$$C = \begin{bmatrix} 300 & 240 & 276 & 48 & 120 & 72 \\ 1370 & 1196 & 1341 & 150 & 335 & 184 \\ 425 & 652 & 755 & 81 & 235 & 115 \\ 901 & 787 & 870 & 114 & 250 & 158 \\ 791 & 1188 & 1145 & 130 & 372 & 168 \\ 200 & 168 & 184 & 32 & 72 & 48 \end{bmatrix}$$

The encrypted matrix can be transformed into a row or column matrix and delivered to another user over any type of median with specifying the order of the matrix, the matrix which aids in computing the self-invertible matrix.

[6, 300, 240, 276, 48, 120, 72, 1370, 1196, 1341, 150, 335, 184,425, 652, 755, 81, 235, 115, 901, 787, 870, 114, 250, 158,791, 1188, 1145, 130, 372, 168, 200, 168, 184, 32, 72, 48; 1, 5, 1, 4, 9, 6, 2, 4,1].

Decryption- User B (The receiver): Decryption is done by using following steps

With the received information, the receiver is able to identify the order of the matrix, encrypted matrix, the matrix which helps to generates the key matrix.

$$C = \begin{bmatrix} 300 & 240 & 276 & 48 & 120 & 72 \\ 1370 & 1196 & 1341 & 150 & 335 & 184 \\ 425 & 652 & 755 & 81 & 235 & 115 \\ 901 & 787 & 870 & 114 & 250 & 158 \\ 791 & 1188 & 1145 & 130 & 372 & 168 \\ 200 & 168 & 184 & 32 & 72 & 48 \end{bmatrix}$$

The receiver is also generating the self-invertible matrix as the procedure explained in Section 2.

$$\therefore R = \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix} = \begin{bmatrix} 28 & 24 & 28 & 2 & 5 & 1 \\ 25 & 20 & 23 & 4 & 10 & 6 \\ 27 & 25 & 28 & 2 & 4 & 2 \\ 0 & 24 & 28 & 1 & 5 & 1 \\ 25 & 21 & 23 & 4 & 9 & 6 \\ 27 & 25 & 0 & 2 & 4 & 1 \end{bmatrix}$$

$$C \cdot R = \begin{bmatrix} 300 & 240 & 276 & 48 & 120 & 72 \\ 1370 & 1196 & 1341 & 150 & 335 & 184 \\ 425 & 652 & 755 & 81 & 235 & 115 \\ 901 & 787 & 870 & 114 & 250 & 158 \\ 791 & 1188 & 1145 & 130 & 372 & 168 \\ 200 & 168 & 184 & 32 & 72 & 48 \end{bmatrix} \cdot \begin{bmatrix} 28 & 24 & 28 & 2 & 5 & 1 \\ 25 & 20 & 23 & 4 & 10 & 6 \\ 27 & 25 & 28 & 2 & 4 & 2 \\ 0 & 24 & 28 & 1 & 5 & 1 \\ 25 & 21 & 23 & 4 & 9 & 6 \\ 27 & 25 & 0 & 2 & 4 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 26796 & 24372 & 25752 & 2784 & 6612 & 3132 \\ 117810 & 105560 & 115321 & 12064 & 28675 & 13572 \\ 57565 & 51869 & 55709 & 6219 & 14645 & 7453 \\ 78909 & 71050 & 76631 & 8120 & 19307 & 9135 \\ 96599 & 86501 & 93728 & 10578 & 25085 & 12739 \\ 17864 & 16240 & 17168 & 1856 & 4416 & 2088 \end{bmatrix}$$

Taking addition modulo 29, we get, $26796 (\text{mod } 29) = 0$, $24372 (\text{mod } 29) = 12$, $25752 (\text{mod } 29) = 0$, …, $2088 \text{ (mod } 29) = 0$.

$$\therefore C \cdot R = \begin{bmatrix} 0 & 12 & 0 & 0 & 0 & 0 \\ 12 & 0 & 17 & 0 & 23 & 0 \\ 0 & 17 & 0 & 13 & 0 & 0 \\ 0 & 0 & 13 & 0 & 22 & 0 \\ 0 & 23 & 0 & 22 & 0 & 8 \\ 0 & 0 & 0 & 0 & 8 & 0 \end{bmatrix} = A$$

The Corresponding A-graph for the above adjacency matrix was formed

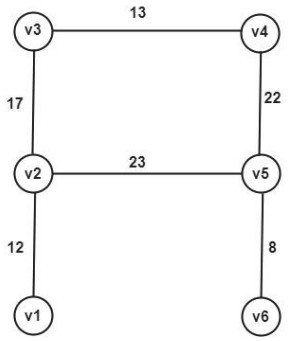

Fig. 7. A graph of decrypted adjacency matrix.

The vertices(nodes) of the above graph were constructed by adding numerical equivalent value of vertex with corresponding edge, since we are adding a special character A in the beginning so we know that the first vertex must be 1 so the remaining vertices are finding by let v1=1, so v2= 1 + 12 = 13, v3 = 13 + 17 = 30 = 1, v4 = 1 + 13 = 14, v5 = 14 + 22 = 36 = 7, v6 = 7 + 8 = 15.

$$\therefore \text{The vertices are } 1, 13, 1, 14, 7, 15.$$

$$\therefore \text{The message is } 1 \to A\,,\ 13 \to M, 1 \to A,\ 14 \to N,\ 7 \to G, 15 \to O \text{ i.e., A MANGO.}$$

5.2 Using Centipede(C2):

Suppose that the sender wants to send the message "BOY" to another user using the technique which is explained in Section 4, using Centipede C2 and its adjacency matrix, the self-invertible key matrix that has been explained in Section 2.

Encryption- User A (The sender): Encryption is done by the following,

Initially, we should add a special character A to the beginning letter of the given plaintext message units and then convert the given message "A BOY" as the vertices of Centipede C2. The vertices are joined from vertex1 to vertex 4 by putting sequential letters in the given message units as the vertices.

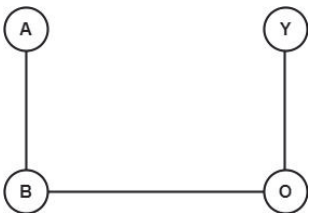

Fig. 8. Centipede (C2) of original message.

Using the encoded table (Table 1) we get, $A \to 1,\ B \to 2,\ O \to 15,\ Y \to 25.$

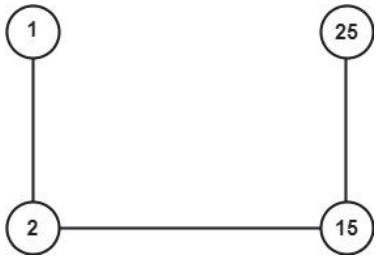

Fig. 9. Encoded Centipede (C2) graph.

Weights of the edges of this graph are assigned by finding the numerical distance between the consecutive two connected vertices and then take addition modulo 29 as we are utilising 29 character in the given encoded table (Table 1) (e1= Code B – Code A, e2 = Code O – Code, …)

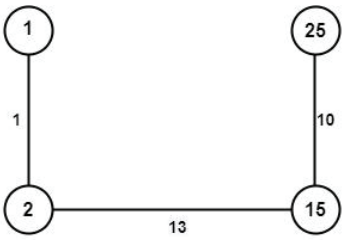

Fig. 10. Centipede (C2) with weights.

The corresponding adjacency matrix of the above graph was computed, name it as 'A'

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 13 & 0 \\ 0 & 13 & 0 & 10 \\ 0 & 0 & 10 & 0 \end{bmatrix}$$

To compute the key matrix, we construct the self-invertible key matrix 'R'. Let us consider the commonly shared $\frac{n}{2} \times \frac{n}{2}$ matrix $R_{22} = \begin{bmatrix} 1 & 5 \\ 1 & 4 \end{bmatrix}$. Then the remaining matrices are (under modulo 29)

$$R_{11} = \begin{bmatrix} 28 & 24 \\ 28 & 25 \end{bmatrix}, R_{12} = I - R_{11} = \begin{bmatrix} 2 & 5 \\ 1 & 5 \end{bmatrix}, R_{21} = I + R_{11} = \begin{bmatrix} 0 & 24 \\ 28 & 26 \end{bmatrix}$$

$$\therefore R = \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix} = \begin{bmatrix} 28 & 24 & 2 & 5 \\ 28 & 25 & 1 & 5 \\ 0 & 24 & 1 & 5 \\ 28 & 26 & 1 & 4 \end{bmatrix}$$

Finally, the encrypted matrix was computed by multiplying A and M

$$C = A \cdot R = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 13 & 0 \\ 0 & 13 & 0 & 10 \\ 0 & 0 & 10 & 0 \end{bmatrix} \cdot \begin{bmatrix} 28 & 24 & 2 & 5 \\ 28 & 25 & 1 & 5 \\ 0 & 24 & 1 & 5 \\ 28 & 26 & 1 & 4 \end{bmatrix}$$

$$C = \begin{bmatrix} 28 & 25 & 1 & 5 \\ 28 & 336 & 15 & 70 \\ 644 & 585 & 23 & 105 \\ 0 & 240 & 10 & 50 \end{bmatrix}$$

The encrypted matrix can be transformed into a row or column matrix and delivered to another user over any type of median with specifying the order of the matrix, the matrix which aids in computing the self-invertible matrix.

$$[\,4,28,25,1,5,28,336,15,70,644,585,23,105,0,240,10,50;1,5,1,4].$$

Decryption- User B (The receiver): Decryption is done by the following steps

With the received information, the receiver is able to identify the order of the matrix, encrypted matrix, the matrix which helps to generates the key matrix then the receiver separates the following matrix

$$C = \begin{bmatrix} 28 & 25 & 1 & 5 \\ 28 & 336 & 15 & 70 \\ 644 & 585 & 23 & 105 \\ 0 & 240 & 10 & 50 \end{bmatrix}$$

The receiver is also generating the self-invertible matrix as the procedure explained in Section 2.

$$\therefore R = \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix} = \begin{bmatrix} 28 & 24 & 2 & 5 \\ 28 & 25 & 1 & 5 \\ 0 & 24 & 1 & 5 \\ 28 & 26 & 1 & 4 \end{bmatrix}$$

$$C \cdot R = \begin{bmatrix} 28 & 25 & 1 & 5 \\ 28 & 336 & 15 & 70 \\ 644 & 585 & 23 & 105 \\ 0 & 240 & 10 & 50 \end{bmatrix} \cdot \begin{bmatrix} 28 & 24 & 2 & 5 \\ 28 & 25 & 1 & 5 \\ 0 & 24 & 1 & 5 \\ 28 & 26 & 1 & 4 \end{bmatrix}$$

$$= \begin{bmatrix} 1624 & 1451 & 87 & 290 \\ 12152 & 11252 & 477 & 2175 \\ 37352 & 33363 & 2001 & 6680 \\ 8120 & 7540 & 300 & 1450 \end{bmatrix}$$

Taking addition modulo 29 we get, $1624 \pmod{29} = 0$, $1451 \pmod{29} = 1$, $87 \pmod{29} = 0$, …, $1450 \pmod{29} = 0$.

$$\therefore C \cdot R = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 13 & 0 \\ 0 & 13 & 0 & 10 \\ 0 & 0 & 10 & 0 \end{bmatrix} = A$$

The Corresponding Centipede C2 for the above adjacency matrix was formed

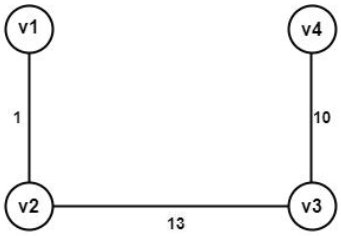

Fig. 11. Centipede (C2) of decrypted adjacency matrix.

The vertices(nodes) of the above graph were constructed by adding numerical equivalent value of vertex with corresponding edge, since we are adding a special character A in the beginning so we know that the first vertex must be 1 so the remaining vertices are finding by let v1=1, so v2= 1 + 1 = 2, v3 = 2 + 13 = 15, v4 = 15 + 10 = 25.

$$\therefore \text{The vertices are } 1, 2, 15, 25.$$

$$\therefore \text{The message is } 1 \rightarrow A\,,\ 2 \rightarrow B, 15 \rightarrow O,\ 25 \rightarrow Y. \text{ i.e., A BOY.}$$

5.3 Using Domino Graph

Suppose that User A(sender) wants to send the message "NIGHT" to another user (User B(receiver)) using the technique which is explained Section 4, using Domino and its adjacency matrix, the self-invertible key matrix that has been explained in Section 2.

Encryption- User A (The sender): Encryption is done by the following,

Initially, we should add a special character A to the beginning letter of the given plaintext message units and then convert the given message" A NIGHT" as the vertices of a Domino graph. The vertices are joined from v1 to v6 by connecting sequential letters in the given message units.

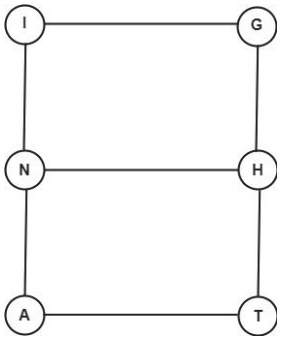

Fig. 12. Domino Graph for given plaintext.

Using the encoded table (Table 1) we get, $A \rightarrow 1,\ N \rightarrow 14,\ I \rightarrow 9, G \rightarrow 7,\ H \rightarrow 8,\ T \rightarrow 20$.

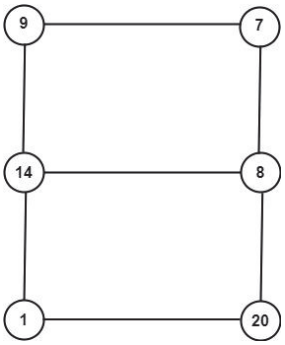

Fig. 13. Encoded Domino graph.

Weights of the edges of this graph are assigned by finding the numerical distance between the consecutive two connected vertices and then take addition modulo 29 as we are using 29 characters in the given encoded table (e1= Code N – Code A, e2 = Code I – Code N, …)

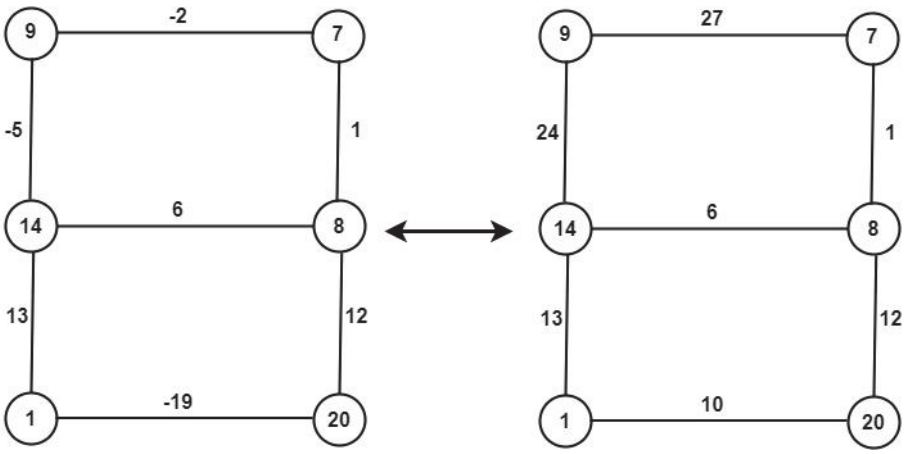

Fig. 14. Weighted Domino graph.

The corresponding adjacency matrix of the above graph was computed, name it as 'A'

$$A = \begin{bmatrix} 0 & 13 & 0 & 0 & 0 & 10 \\ 13 & 0 & 24 & 0 & 6 & 0 \\ 0 & 24 & 0 & 27 & 0 & 0 \\ 0 & 0 & 27 & 0 & 1 & 0 \\ 0 & 6 & 0 & 1 & 0 & 12 \\ 10 & 0 & 0 & 0 & 12 & 0 \end{bmatrix}$$

The Self-invertible key matrix, $R = \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix} = \begin{bmatrix} 28 & 24 & 28 & 2 & 5 & 1 \\ 25 & 20 & 23 & 4 & 10 & 6 \\ 27 & 25 & 28 & 2 & 4 & 2 \\ 0 & 24 & 28 & 1 & 5 & 1 \\ 25 & 21 & 23 & 4 & 9 & 6 \\ 27 & 25 & 0 & 2 & 4 & 1 \end{bmatrix}$

Finally, the encrypted matrix was computed by multiplying A and R

$$C = A \cdot R = \begin{bmatrix} 0 & 13 & 0 & 0 & 0 & 10 \\ 13 & 0 & 24 & 0 & 6 & 0 \\ 0 & 24 & 0 & 27 & 0 & 0 \\ 0 & 0 & 27 & 0 & 1 & 0 \\ 0 & 6 & 0 & 1 & 0 & 12 \\ 10 & 0 & 0 & 0 & 12 & 0 \end{bmatrix} \cdot \begin{bmatrix} 28 & 24 & 28 & 2 & 5 & 1 \\ 25 & 20 & 23 & 4 & 10 & 6 \\ 27 & 25 & 28 & 2 & 4 & 2 \\ 0 & 24 & 28 & 1 & 5 & 1 \\ 25 & 21 & 23 & 4 & 9 & 6 \\ 27 & 25 & 0 & 2 & 4 & 1 \end{bmatrix}$$

$$C = \begin{bmatrix} 595 & 510 & 299 & 72 & 170 & 88 \\ 1162 & 1038 & 1174 & 98 & 215 & 97 \\ 600 & 1128 & 1308 & 123 & 375 & 171 \\ 754 & 696 & 779 & 58 & 117 & 60 \\ 474 & 444 & 166 & 49 & 113 & 49 \\ 580 & 492 & 556 & 68 & 158 & 82 \end{bmatrix}$$

The encrypted matrix can be transformed into a row or column matrix and delivered to another user over any type of median with specifying the order of the matrix, the matrix which aids in computing the self-invertible matrix.

[6, 595, 510, 299, 72, 170, 88, 1162, 1038, 1174, 98, 215, 97, 600, 1128, 1308, 123, 375, 171, 754, 696, 779, 58, 117, 60, 474, 444, 166, 49, 113, 49, 580, 492, 556, 68, 158, 82; 1, 5, 1, 4, 9, 6, 2, 4,1].

By using the reverse process to the received information the receiver is be able to retrieve the original message.

6 Conclusion

These days, keeping our information secure is crucial to accomplish this, numerous publications use various symmetric encryption techniques, including the Caesar cipher, the Hill cipher, graphical approaches, and others. This study proposes a novel technique to cryptosystem encryption in order to enhance the security of our data. It employs an even order self-invertible matrix as the key matrix together with adjacency matrices of several graphs, including the A-graph, Centipede C2, and Domino Graph. This recommended approach can get around the intermediate and is more efficient, any graph with an even number of vertices may utilise the proposed algorithm. The suggested method uses a simple encryption and decryption technique with better security; we just exchange a $\frac{n}{2} \times \frac{n}{2}$ matrix that helps create the self-invertible matrix, which reduces the complexity involved in sharing the common key, increasing the security against hacking the key. We are using a self-invertible matrix as a key matrix, which eliminates the need to figure out the inverse of the key matrix while decrypting the ciphertext. This method of message encryption and decryption is employed in this study along with a few graph theory ideas. In addition, the key matrix is an even order self-invertible matrix. In the future, this approach will be improved and used to a variety of other difficult graph theory concepts, self-invertible matrices of any order, and more encryption methods such as image and video encryption, among others.

References

1. B. Acharya, G.S. Rath, S.K. Patra, S.K. Panigrahy, International Journal of Security 14-21 (2007)
2. P. Amudha, J. Jayapriya, J. Gowri, Journal of physcis **1770(1),** 012072 (2021) https://iopscience.iop.org/article/10.1088/1742-6596/1770/1/012072

3. Alastair Farrugia (1999) Self-complementary graphs and generalisations: a comprehensive reference manual (University of Malta, 1999)
4. S. Arumugam, S. Ramachandran, Invitation to Graph theory, Scitech Publications, (2015)
5. W. Diffie, M. Hellman, IEEE Trans. Inf. Theory **22(6),** 644-654 (1976)
6. P. Mohan, K. Rajendran, A. Rajesh, Journal of Algebraic statistics **13(3),** 2022. https://publishoa.com/index.php/journal/article/view/816, pp.1821-1826
7. P. Mohan, K. Rajendran, A. Rajesh, Indian Journal of Science and Technology **15(44),** 2351-2355, 2022.
8. R. Nandhini, V. Maheswari, V. Balaji, Journal of Computational Mathematics **2(1),** 97-104 (2018) https://doi.org/10.26524/jcm32
9. Neal Koblitz, A course in Number Theory and Cryptography, second edition, Springer
10. Saniah Sulaiman Zurina Mohd Hanpi, Journal of Computer Science **17(3)**, 221-320 (2021) https://doi.org/10.3844/jcssp.2021.221.230
11. Uma Dixit, International journal of Advance Research in Science and Engineering **6(01)**, 218-221 (2017)
12. Weal Mahmoud AI Etaiwi, Journal of Scientific Research and Reports **3(19)**, 2519-2527 (2014)
13. Weisstein, Eric W., Bull Graph, Math World
14. M. Yamuna, Gogia Meenal, Ashish Sikka, Md. Jazib Hayat Khan, International Journal of Computer Application **2(5),** 102-107 (2012).
15. Ziad E. Dawahdeh, Shahrul N. Yaakob, Rozmie Razif bin Othman, Journal of King Saud University - Computer and Information Sciences **30(3),** 349-355 (2018).