Hindawi
Journal of Mathematics
Volume 2023, Article ID 7588535, 9 pages
https://doi.org/10.1155/2023/7588535

Research Article

Block Encryption and Decryption of a Sentence Using Decomposition of the Turan Graph

C. Beaula [1], **P. Venugopal** [2] **and B. Praba** [1]

[1]*Department of Mathematics, Sri Sivasubramaniya Nadar College of Engineering, Kalavakkam 603110, India*
[2]*Mathematics, School of Science and Humanities, Shiv Nadar University Chennai, Kalavakkam 603110, India*

Correspondence should be addressed to C. Beaula; beaula_charles@yahoo.co.in

Received 8 March 2023; Revised 10 April 2023; Accepted 18 April 2023; Published 30 May 2023

Academic Editor: Xuanlong Ma

Encryption and decryption are the two processes in cryptography to conceal and convey important information to an authorized person without third-party interruption in a network. Cryptography is a branch of computer science in which the system has to be updated every second. It mainly depends on mathematical concepts like number theory and algebra. Recently, graph theory concepts are employed in cryptography to make it stronger. The usage of complex graphs in cryptosystems makes it difficult to hack. In this paper, we proposed a cryptosystem using the Turan graph which has a complex graph structure. The advantage of using a Turan graph is that it is a unique multipartite complete graph with more edges than other multipartite complete graphs. This adds robustness to the cryptosystem. The novelty of this paper is the decomposition of the Turan graph into paths and stars and applying edge labeling to them to encrypt and decrypt a sentence of k words. The algorithms for encryption and decryption are also proposed in this paper.

1. Introduction

Cryptography is a technique used in the communication system to transmit data more securely. Cryptography [1] is all about constructing and analyzing a system that helps in secret communication or authentication. Cryptography is the study of cryptosystems in engineering using mathematics. It is classified into symmetric key cryptography, public key cryptography, and hash functions. *Symmetric key cryptography* works with a single key for encryption and decryption. *Public key cryptography* runs on two keys, one with encryption and the other one with decryption. A *hash function* is designed as a one-way function with only one key used for encryption. This function is used in the authentication.

The data are communicated either block by block with a fixed length or as a whole data at a stretch. When the data are encrypted and decrypted block by block, it is called a *block cipher* [1]. If the encryption and decryption are done for a whole plaintext, then it is called a *stream cipher*. With all the studies on cryptography, it was found that symmetric key cryptography with block ciphering is more efficient and hard to break.

Cryptography is dominated by number theory and algebra. The combination of graph theory and cryptography [2] is another milestone in the research of cyber security. A lot of work has been carried out in cryptosystems using graph theory techniques. A few of them are listed here. The cryptosystem using stars, paths, and bipartite graphs is discussed in [3]. Yamuna and Karthika [4] applied a bipartite graph in the cryptosystem for data transferring. Their work shows that the application of graph theory in cryptography is very effective. Hash function in cryptosystem gives authentication of any shared message. Z′emor [5] modeled a hash function using a Cayley graph constructed from a group. Denis et al. [6] applied the expander graph for constructing hash functions. This gives a new dimension to the research of network security using graph theory. Cusack and Chapman [7] have given a study on graphic methods to challenge cryptographic performance. Ustimenko [8] used the graphs with the largest girths in constructing

a cryptosystem. A few more applications of graph theory in coding theory can be referred to in [9–12].

The graph theory techniques are used to visualize, analyze, and solve problems in engineering and science. Based on the nature of the problem, different graph techniques are applied to solve it. The *graphG* [13] is a tuple (V, E), where V is a nonempty set of vertices and E is a set of edges represented by unordered pairs of vertices. Two vertices in G are said to be *adjacent* if there is an edge between them. Two edges are said to be *adjacent* if the edges are incident on a common vertex. A sequence of alternative vertices and edges without repetition of vertices is called a *path*. A path that starts and ends with the same vertex is called a *cycle*. If there is an edge between every two pairs of vertices in a graph, then it is said to be a *complete graph*.

The decomposition of graphs is one of the research topics in graph theory with a lot of applications in engineering and science. The *decomposition* of a graph G [14] is the set of graphs $\{H_1, H_2, \ldots, H_k\}$, such that $\cup H_i = G$ and $\cap H_i = \varnothing$. In decomposition, we have a particular case called multidecomposition. The pair (S, T) is said to be a *multidecomposition* of G [15] if the edges of G can be partitioned into copies of S and T with at least one copy of S and one copy of T. The decomposition of a complete graph into a set L of cycles of different lengths is said to be *L-decomposition*. The problem of L-decomposition of complete graphs is the conjecture by Alspach [16]. Lin and Shyu [17] have extended this research by decomposing complete graphs into L sets of stars. For a complete multigraph, Tarsi [18] proposed path decomposition. Cycle decomposition of the product of graphs is discussed by Karunambigai and Muthusamy [19, 20]. The decomposition of a circulant-balanced complete multipartite graph is discussed in [21, 22].

If the vertex set can be partitioned into two disjoint nonempty subsets and every edge of the graph has its end vertices in different partitions, then the graph is said to be a *bipartite graph*. If the vertex set is partitioned into r disjoint subsets and every edge of the graph has its end vertices in different partitions, then the graph is said to be a *r-partite graph*. In a bipartite graph, if every vertex of one partition is adjacent to every other vertex in the other partition, then the bipartite graph is a *complete bipartite graph*. Similarly, if every vertex in r partitions is adjacent to every vertex in other partitions, then the graph is said to be a *completer-partite graph*.

Turan graph [23] is a r-partite complete graph with n vertices. It is denoted by $T_{n,r}$. An example of the Turan graph is given in Figure 1. In the Turan graph $T_{n,r}$, partite sets differ in size by at most 1. It has the highest number of edges in comparison to the other r-partite complete graphs [23]. The number of edges in $T_{n,r}$ is the difference between the number of edges of a complete graph with n vertices and all the edges of the complete graphs from each r-partition [24]. By the pigeonhole principle, some partite sets have a size of at least $\lfloor n/r \rfloor$ and some have a size of at most $\lceil n/r \rceil$. For our research purpose, we chose the Turan graph, as it is a complex structure to visualize without the knowledge of graph theory. The cryptosystem with more complex graph structures is stronger and more difficult to break compared to simple graph structures. In this paper, the Turan graph is decomposed into paths and stars to encrypt and decrypt a sentence.

The other graph techniques used in the construction of the cryptosystem in this paper are edge labeling and adjacency list. The *edge labeling* of the graph is the function of assigning numbers to edges in such a way that no two edges have the same labeling [25]. In general, graphs can be represented in two ways, namely, adjacency list and adjacency matrix [24]. The *adjacency list* representation of graph $G = (V, E)$ consists of an array of lists $|V|$, representing each row by a vertex in V. The *adjacency matrix* is the matrix with the edge weights corresponding to incident vertices listed in rows and columns. In this paper, the adjacency list is used to list all the edge weights as an encrypted message. For the cryptographic part, one can refer to [16] and graph theory [13, 23, 26].

There are many research works available in cryptography with number theory, algebra, and formal languages. Although stronger cryptosystems are constructed, the cryptosystems are hacked and many authentications are broken. This is a field that keeps growing in building and destroying the safety measures of the network and cyber security. To make the cryptosystem stronger, we apply new techniques in constructing the cryptosystem. Graph theory is a field with a lot of applications in engineering and science, which has different structures, simple and complex. This helps in meeting the demand of constructing a better cryptosystem.

This paper is organized as follows. In Section 1, we have discussed all the necessary terminologies and concepts of graph theory and cryptography needed for our study. Section 2 deals with the main result. In the main result, we employed block ciphering of length eight and used 8-partite Turan graphs as a symmetric key. The Turan graph is decomposed into paths and stars and then the edge labeling is applied in encrypting and decrypting the plaintext. The path connecting the 8 partitions is used for encryption. An algorithm for encryption and decryption based on the decomposition is also presented. Finally, the encrypted message is the edge weights of the Turan graph, in a defined format. The last section of the paper notes down the efficiency of the proposed cryptosystem and the future scope of work.

2. Main Results

In this paper, the multidecomposition technique is used to decompose the 8-partite Turan graph into copies of paths and stars. The edges of the paths and stars are labeled using edge labeling. We define a new cryptosystem using graph decomposition and edge labeling. This cryptosystem is safer and more secure as it is not possible to hack the system without understanding the structure of the Turan graph. We construct a cryptosystem to encrypt and decrypt a plaintext of a k-word sentence. A block cipher is assigned to encrypt the words one by one taking it as a block.

The novelty of this paper is the application of (P, S)-multidecomposition of a Turan graph, and the results are used in the construction of a cryptosystem.

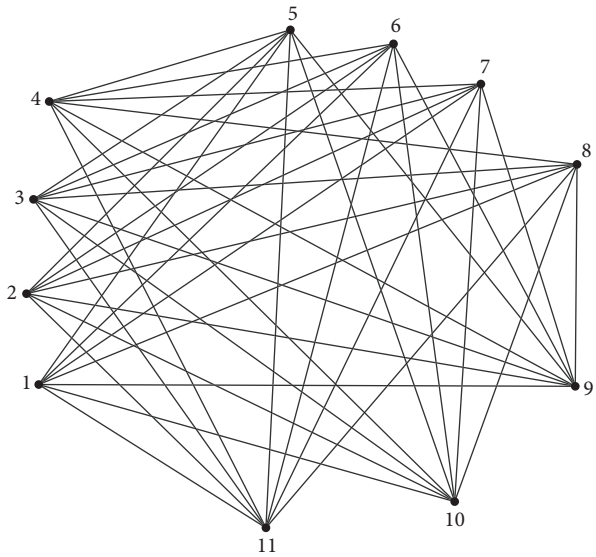

FIGURE 1: Turan graph $T_{11,3}$.

Definition 1. (**P**, **S**)-multidecomposition of a graph.

A graph $\mathcal{G}$ is said to be a (P, S)-*multidecomposition*, if $\mathcal{G}$ can be decomposed into at least one path of any arbitrary size and at least one star of any arbitrary size.

Theorem 1. *Turan graph with partitions of equal sizes is a* (P, S)*-multidecomposition.*

Proof. Let $T_{n,r}$ be a Turan graph with n vertices and r-partitions.

Let $X_1, X_2, \ldots, X_r$ be the r partitions of $T_{n,r}$. As all these partitions are of the same size, take $|X_1| = |X_2| = |X_3| \ldots = |X_r| = z$.

Therefore, $n = rz$. □

The procedure of decomposing the turan graph $\mathbf{T_{n,r}}$ into paths and stars.

Path Decomposition. Choose one vertex from each partition X_i, $1 \le i \le r$, and construct paths $P_{r1}, P_{r2}, P_{r3}, \ldots, P_{rz}$ of same size r as follows.

P_{r1}: $x_{11}, x_{21}, x_{31}, \ldots, x_{r1}, x_{12}$,

P_{r2}: $x_{12}, x_{22}, x_{32}, \ldots, x_{r2}, x_{13}$,

$\vdots$

P_{rz}: $x_{1z}, x_{2z}, x_{3z}, \ldots, x_{rz}, x_{11}$,

Star Decomposition. The star constructed from the k^{th} vertex in the i^{th} partition is denoted by S_{ik}, $1 \le i \le r, 1 \le k \le z$. Depending on the number of partitions r of $T_{n,r}$, we have the following two cases.

Case 1. Suppose r is even.

For $1 \le i \le r/2$, the stars are constructed by taking a vertex from X_i and connecting it with the vertices of $X_{i+1}, X_{i+2}, \ldots, X_{i+r/2}$. In Figure 2, the construction of star S_{11} from the vertex x_{11} is given.

For $i = r/2 + 1$, the stars are constructed by taking a vertex from $X_{r/2+1}$ and joining it with the vertices of $X_{r/2+2}, X_{r/2+3}, \ldots, X_r$. The construction of a star $S_{(r/2+1)1}$ from the vertex $x_{(r/2+1)1}$is given in Figure 3.

For $i = r/2 + 2$, the stars are constructed by taking a vertex from $X_{r/2+2}$ and joining it with the vertices of $X_{r/2+3}, X_{r/2+4}, \ldots, X_r, X_1$.

For $i = r/2 + 3$, the stars are constructed by taking a vertex from $X_{r/2+3}$ and joining it with the vertices of $X_{r/2+4}, X_{r/2+5}, \ldots, X_r, X_1, X_2$.

Proceeding like this, for $i = r$, the stars are constructed by taking a vertex from X_r and joining it with the vertices of $X_1, X_2, \ldots, X_{r/2-1}$.

In this construction, we have obtained $n/2$ copies of stars with $(rz/2 - 1)$ edges and $n/2$ copies of stars with $(rz/2 - z - 1)$ edges.

By [24], the total number of edges in $T_{n,r} = nC_2 - r(zC_2)$

$$= \frac{n^2}{2}\left(1 - \frac{1}{r}\right). \tag{1}$$

The number of edges in the decomposed graphs in our construction

= (number of edges in paths) + (number of edges in stars),

$$= rz + \frac{n}{2}\left(\frac{rz}{2} - 1\right) + \frac{n}{2}\left(\frac{rz}{2} - z - 1\right),$$

$$= n + \frac{n}{2}\left(\frac{n}{2} - 1\right) + \frac{n}{2}\left(\frac{n}{2} - z - 1\right),$$

$$= n + \left(\frac{n}{2}\right)^2 - \frac{n}{2} + \left(\frac{n}{2}\right)^2 - \frac{n}{2}z - \frac{n}{2},$$

$$= 2\left(\frac{n}{2}\right)^2 - \frac{n}{2}z,$$

$$= \frac{n}{2}\left(2\frac{n}{2} - z\right),$$

$$= \frac{n^2}{2}\left(1 - \frac{1}{r}\right),$$

= number of edges in $T_{n,r}$ (from (1)).

(2)

Case 2. Suppose r is odd.

For $1 \le i \le r + 1/2$, the stars are constructed by taking a vertex from X_i and connecting it with the vertices of $X_{i+1}, X_{i+2}, \ldots, X_{i+r-1/2}$. The construction of a star S_{11} from the vertex x_{11} is given in Figure 4.

For $i = r + 1/2 + 1$, the stars are constructed by taking a vertex from $X_{r+1/2+1}$ and joining with the vertices of $X_{r+1/2+2}, X_{r+1/2+3}, \ldots, X_r, X_1$.

For $i = r + 1/2 + 2$, the stars are constructed by taking a vertex from $X_{r+1/2+2}$ and joining with the vertices of $X_{r+1/2+3}, X_{r+1/2+4}, \ldots, X_r, X_1, X_2$.

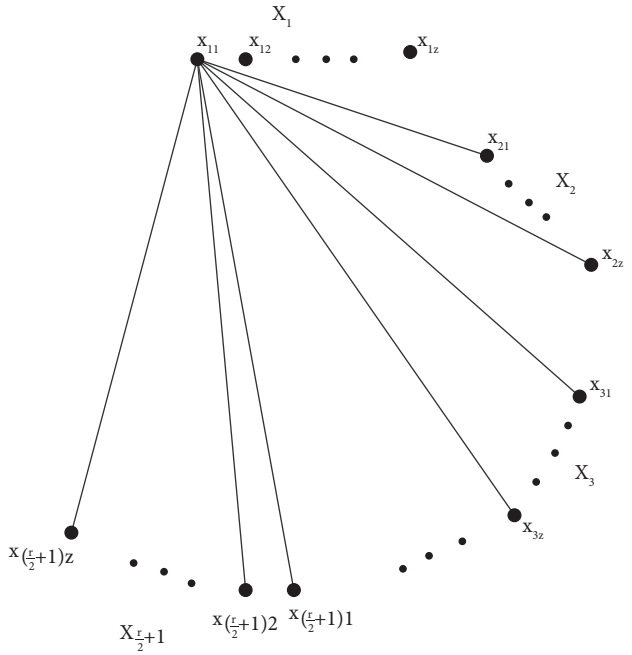

Figure 2: Star graph S_{11}.

Proceeding like this, for $i = r$, the stars are constructed by taking a vertex from X_r and joining it with the vertices of $X_1, X_2, \ldots, X_{(r-1/2)}$.

Therefore, by our construction, we obtained n copies of stars with $z(r - 1/2)_{-1}$ edges.

The number of edges in the decomposed graphs in our construction

$$\begin{aligned} &= (\text{number of edges in the paths}) \\ &\quad + (\text{number of edges in the stars}) \\ &= rz + n\left(\frac{rz - z}{2} - 1\right) \\ &= n + n\left(\frac{n - z}{2}\right) - n, \\ &= \frac{n^2}{2}\left(1 - \frac{1}{r}\right) \\ &= \text{number of edges in } T_{n,r} \ (\text{from } (1)). \end{aligned} \tag{3}$$

Hence, the Turan graph $T_{n,r}$ with partitions of equal sizes is a (P, S)-multidecomposition.

2.1. Turan Graph for the Proposed Cryptosystem. In this paper, an 8-partite Turan graph is used for the proposed cryptosystem.

Consider a Turan graph $T_{n,r}$ by taking $r = 8$ and $n = 8z$, where z is the number of vertices in each partition.

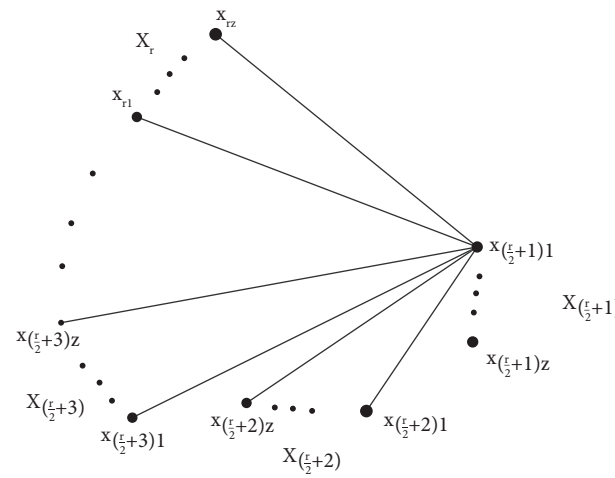

Figure 3: Star graph $S_{(r/2+1)1}$.

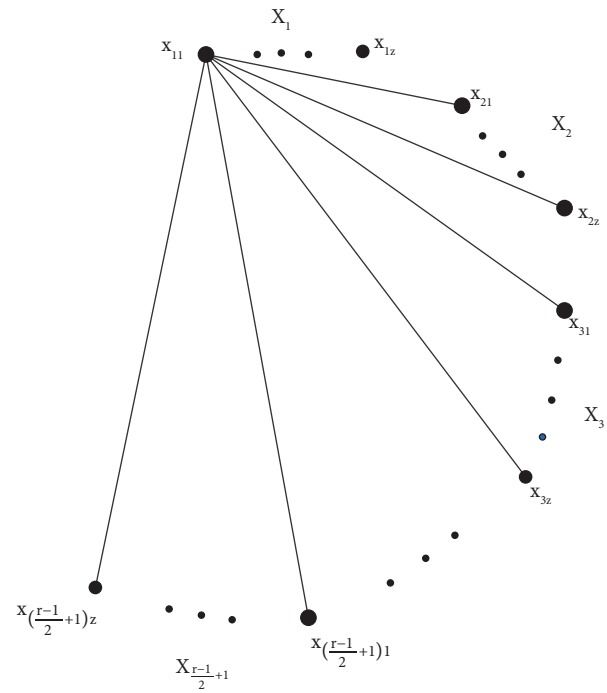

Figure 4: Star graph S_{11}.

Let $X_1, X_2, \ldots, X_8$ be the partitions of the vertex set V of size "n" which are given as follows.

$$\begin{aligned} X_1 &= x_{11}, x_{12}, \ldots, x_{1z}, \\ X_2 &= x_{21}, x_{22}, \ldots, x_{2z}, \\ &\vdots \\ X_8 &= x_{81}, x_{82}, \ldots, x_{8z}. \end{aligned}$$

2.2. Edge Labeling Lists for Turan Graph. The edges of the paths in encryption and decryption are labeled using the numbers in Table 1. The edges of stars are labeled using the distinct numbers not in Table 1. To label the edges of the stars, the following edge labeling lists are formed using

arithmetic progression with common difference 8 and different starting numbers.

List 1 = {9, 17, 33, 41, 57, 65, 81, 89, 105, 113, 121, 129, 137, 145, 153, 161, 169, 177, 185, 185, 193, 201, 209, . . .}

List 2 = {2, 18, 26, 42, 50, 66, 74, 90, 98, 114, 122, 130, 138, 146, 154, 162, 170, 178, 186, 194, 202, 210, . . .}

List 3 = {3, 11, 27, 35, 51, 59, 75, 83, 99, 107, 123, 131, 139, 147, 155, 163, 171, 179, 187, 195, 203, . . .}

List 4 = {12, 20, 36, 44, 60, 68, 84, 92, 108, 116, 124, 132, 140, 148, 156, 164, 172, 164, 172, 180, 188, 196, 204, . . .}

List 5 = {5, 21, 29, 45, 53, 69, 77, 93, 101, 117, 125, 133, 141, 149, 157, 165, 173, 181, 189, 197, 205, . . .}

List 6 = {6, 14, 30, 38, 54, 62, 78, 86, 102, 110, 118, 126, 134, 142, 150, 158, 166, 174, 182, 190, 198, 206, . . .}

List 7 = {15, 23, 39, 47, 63, 71, 87, 95, 111, 119, 127, 135, 143, 151, 159, 167, 175, 183, 191, 199, 207, . . .}

List 8 = {8, 24, 32, 48, 56, 72, 80, 96, 104, 120, 128, 136, 144, 152, 160, 168, 176, 184, 192, 200, 208, . . .}

2.3. Application of Turan Graph and Its Edge Labeling in the Cryptosystem. In this paper, we have modified the coding table [1] using arithmetic progression, with the first term $t_1 = a = 1$ and common difference $d = 3$ and the last term $t_{26} = a + d(26 - 1) = 76$. This modified table is named Table 1 and is used for converting any word to a number string.

2.3.1. Fixing the Length of Plaintext. Let "$M = R_1R_2 \ldots R_m$" be the plaintext, where R_i is a word.

Case 1. If $|R_i| = 8$, then the edges of the path P_{ri}, decomposed from Turan Graph $T_{n,8}$, are labeled by encoding the numbers of R_i using Table 1.

Case 2. If $|R_i| < 8$, then the edges of the path P_{ri}, decomposed from Turan Graph $T_{n,8}$, are labeled as in Case 1. The remaining edges $8 - |R_i|$ of this path are labeled with the numbers $t_{27}, t_{28}, t_{29}, \ldots$ (A.P) depending on the requirement.

Case 3. If $|R_i| > 8$, then break the word into parts, such that the first part is of length 8 and the remaining parts are of a length less than or equal to 8 so that Cases 1 and 2 can be applied for labeling.

By applying the above cases, the given sentence is transformed into number strings of multiples of length 8, say $8z$, where z is the number of "number strings."

(1) Repetition of Letters in. R_i If a letter L with encoding number n_{ij} is repeated in a word, then the letters repeated are labeled as follows. First L: n_{ij}, second L: $n_{ij} + t_{27}$, third L: $n_{ij} + t_{28}$, and so on as per the requirement (see Algorithm 1).

Remark 1. As the number of words (m) in the plaintext M increases, the construction of the Turan graph becomes tedious.

Table 1: Encoding table.

Alphabet	Coding number
A	1
B	4
C	7
D	10
E	13
F	16
G	19
H	22
I	25
J	28
K	31
L	34
M	37
N	40
O	43
P	46
Q	49
R	52
S	55
T	58
U	61
V	64
W	67
X	70
Y	73
Z	76

2.4. Illustration for a Block Cipher. Consider the plaintext "**I FLY HIGH**." The second block cipher is illustrated in the following.

Block Cipher: 22nd block for encryption: **FLY** (see Algorithm 2).

Step 1: $N_2 = 16, 34, 73, 79, 82, 85, 88, 91$, using Table 1.

Step 2: Label the path P_{82} of $T_{24,8}$ with the numbers of N_2 (see Figure 5).

Step 3: Label the edges of the stars from the second partition with **List 1**, and the edges of the stars from the third partition with **List 2**. Continue this process until the edges of the stars from the 8th partition are labeled, and finally, the edges of the stars from the 1st partition are labeled with **List 8**. The labels of the edges incident with the partition X_1 are listed in Table 2. Similarly, the labels of the edges incident with the partitions X_2, X_3, X_4, X_5, X_6, X_7, and X_8 are listed in Tables 3–9, respectively.

Step 4: Encrypted Message

Block Cipher 2: {4, 8, 24, 32, 48, 56, 72, 80, 96, 104, 120, 128, 77, 157, 221, 30, 118, 182, 15, 111, 55, 136, 16, 144, 152, 160, 168, 176, 184, 192, 200, 208, 216, 93, 157, 221, 38, 126, 190, 52, 119, 175, 224, 232, 7, 240, 248, 256, 264, 272, 280, 288, 296, 304, 101, 165, 229, 54, 134, 198, 23, 91, 183, 10, 9, 17, 33, 41, 57, 65, 81, 105, 113, 121, 129, 62, 142, 206, 39, 127, 191, 137, 34, 145, 153, 161, 169, 177, 185, 193, 201, 209, 217, 78, 150, 214, 47, 135, 199, 225, 233, 13, 241, 249, 257, 265, 273, 281, 281,

Input: Plaintext with m words
Output:z blocks
Procedure:
Step 1: Let M be the plaintext with m words. Convert this plaintext into z number strings using Section 2.4, which gives $N = N_1N_2 \ldots N_z$, where $N_j = n_{1j}n_{2j} \ldots n_{8j}$.
Step 2: Construct a Turan graph $T_{n,8}$ with $n = N$ vertices and 8 partitions of equal size z.
Step 3: Block cipher $j = 1$ to z
(a) Decompose the Turan graph into paths and stars (Theorem 1).
(b) Label the edges of the path P_{8j} with the numbers of the number string N_{8j}. Label the other paths $P_{81}, P_{82}, \ldots, P_{8(j-1)}, P_{8(j+1)}, \ldots, P_{8z}$ with the numbers of Table 1 that are not used in P_{8j}.
(c) Label the edges of stars using Section 2.3 as follows.
Step (i): Label the j^{th} partition stars with the numbers from **List 1**.
Step (ii): Label $(j+1)^{\text{th}}$ partition stars with the numbers from **List 2**.
Step (iii): Proceeding like this, label the 8^{th} partition stars with the numbers from **List** $(\mathbf{8} - (\mathbf{j} - \mathbf{1}))$.
Step (iv): Label the 1^{st} partition with **List** $(\mathbf{8} - (\mathbf{j} - \mathbf{2}))$.
Step (v): Proceeding like this, label the $(j-1)^{\text{th}}$ partition stars with the numbers from **List 8.**
Step (vi): If all the partition stars are labeled, go to (d).
(d) List the edge labels in the tables, for all the edges originating from each partition X_i, which are in the form of stars S_{ik}, $1 \le i \le 8, k \ge 1$, and some of the edges of the paths which do not belong to S_{ik}.
Step 4: Encrypted message: Block Cipher 1, Block Cipher 2, ..., Block Cipher z.
Block Cipher **j**
$\{(x_{11}, x_{21}), (x_{11}, x_{22}), \ldots, (x_{11}, x_{83}), (x_{12}, x_{21}), (x_{12}, x_{22}), \ldots, (x_{12}, x_{83}), (x_{13}, x_{21}), (x_{13}, x_{22}), \ldots, (x_{13}, x_{83}), (x_{21}, x_{31}), (x_{21}, x_{32}), \ldots,$
$(x_{21}, x_{83}), \ldots, (x_{71}, x_{81}), (x_{71}, x_{82}), (x_{71}, x_{83}), (x_{72}, x_{81}), (x_{72}, x_{82}), (x_{72}, x_{83}), (x_{73}, x_{81}), (x_{73}, x_{82}), (x_{73}, x_{83})\}$.

Algorithm 1: Encryption Algorithm.

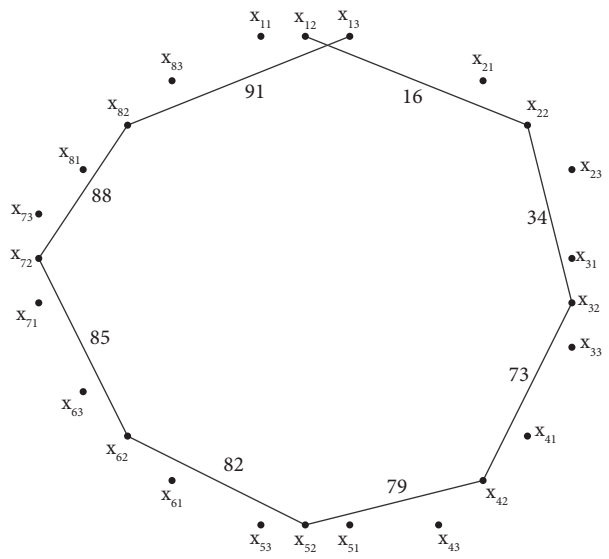

Figure 5: P_{82} of $T_{24,8}$.

Table 3: Labels of edges incident to X_2.

	x_{31}	x_{32}	x_{33}	x_{41}	x_{42}	x_{43}	x_{51}	x_{52}	x_{53}	x_{61}	x_{62}	x_{63}
x_{21}	10	9	17	33	41	57	65	81	105	113	121	129
x_{22}	137	34	145	153	161	169	177	185	193	201	209	217
x_{23}	225	233	13	241	249	257	265	273	281	289	297	305

Table 4: Labels of edges incident to X_3.

	x_{41}	x_{42}	x_{43}	x_{51}	x_{52}	x_{53}	x_{61}	x_{62}	x_{63}	x_{71}	x_{72}	x_{73}
x_{31}	19	2	18	26	42	50	66	74	90	98	114	122
x_{32}	130	73	138	146	154	162	170	178	186	202	210	218
x_{33}	226	234	22	242	250	258	266	274	282	290	298	306

Table 5: Labels of edges incident to X_4.

	x_{51}	x_{52}	x_{53}	x_{61}	x_{62}	x_{63}	x_{71}	x_{72}	x_{73}	x_{81}	x_{82}	x_{83}
x_{41}	25	3	11	27	35	51	59	75	83	99	107	123
x_{42}	131	79	139	147	155	163	171	179	187	195	203	211
x_{43}	219	227	28	235	243	251	259	267	275	283	291	299

Table 2: Labels of edges incident to X_1.

	x_{21}	x_{22}	x_{23}	x_{31}	x_{32}	x_{33}	x_{41}	x_{42}	x_{43}	x_{51}	x_{52}	x_{53}
x_{11}	4	8	24	32	48	56	72	80	96	104	120	128
x_{12}	136	16	144	152	160	168	176	184	192	200	208	216
x_{13}	224	232	7	240	248	256	264	272	280	288	296	304

Table 6: Labels of edges incident to X_5.

	x_{61}	x_{62}	x_{63}	x_{71}	x_{72}	x_{73}	x_{81}	x_{82}	x_{83}
x_{51}	31	12	20	36	44	60	68	84	92
x_{52}	108	82	116	124	132	140	148	156	164
x_{53}	172	180	37	188	196	204	212	220	228

Table 7: Labels of edges incident to X_6.

	x_{71}	x_{72}	x_{73}	x_{81}	x_{82}	x_{83}	x_{11}	x_{12}	x_{13}
x_{61}	40	5	21	29	45	53	77	93	101
x_{62}	117	85	125	133	141	149	159	157	165
x_{63}	173	181	43	189	197	205	213	221	229

Table 8: Labels of edges incident to X_7.

	x_{81}	x_{82}	x_{83}	x_{11}	x_{12}	x_{13}	x_{21}	x_{22}	x_{23}
x_{71}	46	6	14	30	38	54	62	78	86
x_{72}	102	88	110	118	126	134	142	150	158
x_{73}	166	174	49	182	190	198	206	214	222

Table 9: Labels of edges incident to X_8.

	x_{11}	x_{12}	x_{13}	x_{21}	x_{22}	x_{23}	x_{31}	x_{32}	x_{33}
x_{81}	15	52	23	39	47	63	71	87	95
x_{82}	111	119	91	127	135	143	151	159	167
x_{83}	55	175	183	191	199	209	207	215	223

Input: z block ciphers
Output: Decrypted Message
Procedure:
Step 1: Take each block cipher
$\{(x_{11}, x_{21}), (x_{11}, x_{22}), \ldots, (x_{11}, x_{83}), (x_{12}, x_{21}), (x_{12}, x_{22}), \ldots, (x_{12}, x_{83}), (x_{13}, x_{21}), (x_{13}, x_{22}), \ldots, (x_{13}, x_{83}), (x_{21}, x_{31}), (x_{21}, x_{32}), \ldots, (x_{21}, x_{83}), \ldots, (x_{71}, x_{81}), (x_{71}, x_{82}), (x_{71}, x_{83}), (x_{72}, x_{81}), (x_{72}, x_{82}), (x_{72}, x_{83}), (x_{73}, x_{81}), (x_{73}, x_{82}), (x_{73}, x_{83})\}$ and construct a Turan graph with $n = 8z$ vertices.
Step 2: Block decipher
For $j = 1$ to z
(a) Label the edges of the Turan graph $T_{n,8}$ with the j^{th} block cipher.
(b) Decompose the Turan graph $T_{n,8}$ into paths and stars (Theorem 1).
(c) List the edge labels of the path P_{8j} of the Turan graph $T_{n,8}$ and consider it as the j^{th} number string $N_j = n_{1j}\, n_{2j} \ldots n_{8j}$.
(d) If $n_{ij} > t_{26}$, subtract t_{27} from the first greater number, subtract t_{28} from the second greater number, and continue the process until the 8^{th} number.
(e) From the resulting 8 numbers, leave the zeros and convert the remaining numbers into alphabets using Table 1, which gives the j^{th} word of the plaintext.
Step 4: Decrypted Message.

Algorithm 2: Decryption Algorithm.

297, 305, 86, 158, 222, 63, 143, 209, 19, 2, 18, 26, 42, 50, 66, 74, 90, 98, 114, 122, 71, 151, 207, 130, 73, 138, 146, 154, 162, 170, 178, 188, 202, 210, 218, 87, 159, 215, 226, 234, 22, 242, 250, 258, 266, 274, 282, 290, 298, 306, 95, 167, 223, 25, 3, 11, 27, 35, 51, 59, 75, 83, 99, 107, 123, 131, 79, 139, 147, 155, 163, 171, 179, 187, 195, 203, 211, 219, 227, 28, 235, 243, 251, 259, 267, 275, 283, 291, 299, 31, 12, 20, 36, 44, 60, 68, 84, 92, 108, 82, 116, 124, 132, 140, 148, 156, 164, 172, 180, 37, 188, 196, 204, 212, 220, 228, 40, 5, 21, 29, 45, 53, 117, 85, 125, 133, 141, 149, 173, 181, 43, 189, 197, 205, 46, 6, 14, 102, 88, 110, 166, 174, 49}

3. Limitations of the Method

The construction of a cryptosystem using the Turan graph is a novel approach. The encryption and decryption algorithm is constructed purely on mathematical concepts. Hence, the time complexity, space complexity, and efficiency of the algorithms are yet to be studied further, to make the work an application to the real world.

4. Conclusion and Future Scope

There are numerous cryptosystems published and are used in banking and finance sectors, with the aid of programming

languages. Preserving and communicating confidential data is a big challenge as the technology is growing rapidly. The cryptosystem is strengthened regularly using different mathematical concepts. Applying graph theory in cryptography makes it strong and difficult to decrypt for any intruder. The knowledge of graph theory is needed to break the encrypted message. In this paper, sentences of z words are encrypted and decrypted using a Turan graph. As the Turan graph is the only multipartite complete graph with a maximum number of edges [26], the cryptosystem using the Turan graph is a new approach compared to those in [4, 5, 8]. More than one graph theory technique such as graph decomposition, adjacency list, and graph labeling is applied in this cryptosystem, which gives three levels of encryption. We have used an encoding table created with an arithmetic progression which is a different approach. As a cryptosystem is constructed using a block cipher with well-defined encoding, even for repeated letters, the cryptosystem is tough to break and collision resistant. This makes our cryptosystem better compared to other available cryptosystems. In this paper, the Turan graph with equal partitions is decomposed into paths and stars of different sizes.

The application of arithmetic progression to create an encoding table in this work opens a new approach in creating an encoding table using other progressions. In this paper, we have used a multipartite Turan graph for encryption. This work can be extended with different graph structures and for longer plaintext messages. We have decomposed the Turan graph into stars and paths. The decomposition of the Turan graph into other graph structures can be explored. In this paper, we have decomposed the Turan graph with equal partitions. The problem of decomposing the Turan graph with different-size partitions is still open.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors would like to thank the Management of the Shiv Nadar Foundation for their continuous support and encouragement to do research.

References

[1] W. Stallings, *Cryptography and Network Security*, Pearson Education Inc, 6th edition, New York, NY, USA, 2014.

[2] P. L. K. Priyadarsini, "A survey on some applications of graph theory in cryptography," *Journal of Discrete Mathematical Sciences and Cryptography*, vol. 18, no. 3, pp. 209–217, 2015.

[3] B. Ni, R. Qazi, S. U. Rehman, and G. Farid, "Some graph-based encryption schemes," *Journal of Mathematics*, vol. 2021, Article ID 6614172, 8 pages, 2021.

[4] M. Yamuna and K. Karthika, "Data transfer using bipartite graphs," *International Journal of Advance Research in Science and Engineering*, vol. 4, pp. 28–131, 2015.

[5] G. Z´emor, "Hash functions and Cayley graphs," *Designs, Codes and Cryptography*, vol. 4, pp. 381–394, 1994.

[6] D. X. Charles, K. E. Lauter, and E. Z. Goren, "Cryptographic hash functions from expander graphs," *Journal of Cryptology*, vol. 22, no. 1, pp. 93–113, 2009.

[7] B. Cusack and E. Chapman, "Using graphic methods to challenge cryptographic Performance," in *Proceedings of the 14th Australian Information Security Management Conference*, pp. 30–36, Edith Cowan University, Perth, Western Australia, December 2016.

[8] V. A. Ustimenko, "On graph-based cryptography and symbolic computations," *Serdica Journal of Computing*, vol. 1, no. 2, pp. 131–156, 2007.

[9] A. El-Mesady, O. Bazighifan, and H. M. Shabana, "On graph-transversal designs and graph-authentication codes based on mutually orthogonal graph squares," *Journal of Mathematics*, vol. 2022, Article ID 8992934, 10 pages, 2022.

[10] A. El-Mesady and O. Bazighifan, "Construction of mutually orthogonal graph squares using novel product techniques," *Journal of Mathematics*, vol. 2022, Article ID 9722983, 16 pages, 2022.

[11] A. El-Mesady, Y. S. Hamed, and K. M. Abualnaja, "A novel application on mutually orthogonal graph squares and graph-orthogonal arrays," *AIMS Mathematics*, vol. 7, no. 5, pp. 7349–7373, 2022.

[12] M. Higazy, A. El-Mesady, and M. S. Mohamed, "On graph-orthogonal arrays by mutually orthogonal graph squares," *Symmetry*, vol. 12, no. 11, pp. 1895–1913, 2020.

[13] J. A. Bondy and U. S. R. Murty, *Graph Theory with Applications*, Macmillan Press, London, UK, 1976.

[14] J. Bosak, *Decomposition of Graphs*, Kluwer Academic Publishers, Dordrecht, Netherlands, 1990.

[15] A. Abueida and M. Dave, "Multi-decomposition of several graph products," *Graphs and Combinatorics*, vol. 29, pp. 315–326, 2013.

[16] B. Alspach, "Research problems. Problem 3," *Discrete Mathematics*, vol. 333, p. 36, 1981.

[17] C. Lin and T. W. Shyu, "A necessary and sufficient condition for the star decomposition of complete graphs," *Journal of Graph Theory*, vol. 23, no. 4, pp. 361–364, 1996.

[18] M. Tarsi, "Decomposition of a complete multigraph into simple paths: non-balanced handcuffed designs," *Journal of Combinatorial Theory - Series A*, vol. 34, no. 1, pp. 60–70, 1983.

[19] M. G. Karunambigai and A. Muthusamy, "Cycle factorization of tensor product of complete graphs," *Bulletin Of The Institute Of Combinatorics And Its Applications*, vol. 73, pp. 89–100, 2008.

[20] D. G. Hoffman and D. Pike, "4-Cycle decomposition of the cartesian product of two complete graphs," *Journal of Combinatorial Mathematics and Combinatorial Computing*, vol. 28, pp. 215–226, 1998.

[21] A. El-Mesady and O. Bazighifan, "Decompositions of circulant-balanced complete multipartite graphs based on

a novel labelling approach," *Journal of Function Spaces*, vol. 2022, Article ID 2017936, 17 pages, 2022.

[22] A. El-Mesady, O. Bazighifan, and Q. Al-Mdallal, "On infinite circulant-balanced complete multipartite graphs decompositions based on generalized algorithmic approaches," *Alexandria Engineering Journal*, vol. 61, no. 12, pp. 11267–11275, 2022.

[23] B. W. Douglas, *Introduction to Graph Theory*, Pearson Publishing House, London, UK, 1995.

[24] N. Deo, *Graph Theory with Applications to Engineering and Computer Science*, Prentice-Hall, Hoboken, NJ, USA, 1974.

[25] J. A. Gallian, "A dynamic survey of graph labelings," *The Electronic Journal of Combinatorics*, vol. 19, 2019.

[26] F. Harary, *Graph Theory*, Narosa Publishing House, Delhi, India, 1988.