International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

A SYMMETRIC BLOCK CIPHER BASING ON MINIMAL SPAN TREE ALGORITHM OF UNDIRECTED WEIGHTED GRAPH

S. Sarvalakshmi[1], Dr.CH. Suneetha[2], Dr.M.G.Vara Prasad[3], Dr. Mutyala Suresh[4], Gali Lalitha Devi[5]

Doctoral Researcher in Mathematics, GITAM (Deemed to be University), Visakhapatnam, India. ssurampa@gitam.in

Associate Professor, GITAM University, Department of Mathematics, Visakhapatnam, India. schivuku@gitam.edu

Assistant Professor, Department Computer Science and Engineering, Nadimpalli Satyanarayana Raju Institute of Technology. prasadvaram11@gmail.com

Associate Professor, Department of English, Koneru Lakshmaiah Education Foundation, Guntur District, AP. India-522302. msphd@kluniversity.in

Assistant Professor, Department of Mathematics, Anil Neerukonda Institute of Technology and Sciences, Visakhapatnam, India. lalithadevi.maths@anits.edu.in

Author for Correspondence: Dr.CH.Suneetha, schivuku@gitam.edu

Abstract:

In our now hyper digitalized world, the growth of communication systems is straining our ability to secure systems due to the increasing effects of cyber security threats on the reliability and safety of our technologies. Modern cryptography generally offers combinations of three principles: confidentiality, integrity and authentication. This paper proposes a new symmetric block cipher created from terms of graph theory to increase security in encryption. We suggest not only the use of a graph, but a 2-phase encryption capabilities. In the first phase, the plaintext is encrypted with an adjacency matrix from an undirected weighted graph derived from the message characters while the second phase is simply a bitwise XOR operation which will further obscure data. The encryption keys from both phases are generated from a minimum spanning tree (MST) created with a master key that is used to create independent yet shared keys. In this case, multiple parties (multi-party) can continue to use more random ASCII-based block keys in the cloud that are shared to secure communication. The participants independently derive master and session keys between one another using authentic sharing systems to encrypt keys and data independent from one another to help support privacy and scaling with participants involved. The encryption schemes outlined in this paper provide a very secure form of dynamic encryption protocol managing keys for data, and are applicable to distributed networks for unique key requirements and security against brute-force attacks.

Received: July 21, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Key Words: Graph-based encryption, Symmetric cipher, Dynamic key generation, Minimum spanning tree, secure multi-party communication

Introduction

Cryptography is the interdisciplinary field concerned with securing sensitive information using protected channels. Encryption ensures no unauthorized party has access to confidential data and only publicly authorized repositories see authorized disclosures. Most modern cryptographic systems rely on symmetric-key (private-key) and asymmetric-key (public-key) cryptographic systems. In symmetric-key cryptography, both parties involved in communication share the identical secret key - the sender uses it when encrypting the data, while the receiving party uses the secret key when decrypting the data.

The rapid proliferation of web-based services has led to a simultaneous increase in cyber-attacks and subsequent attacks such as information breaches, unauthorized access to systems or physical locations, and unauthorized access to communications. Though encryption is an essential aspect of the security process, modern cryptographic techniques must also be able to ensure data authenticity and confirm data integrity. The science of cryptography is dependent on advanced mathematics and number theory and graph theory (especially algebraic graph theory) are particularly helpful in this regard, as they can enable us to construct robust cryptographic systems. With the help of these number theory and graph theory we have made advances in the areas of multivariate cryptography and well-ordered algebraic structures that can be used as security systems. To this end there have been many previous studies looking at various graph-theoretic methods: expanding graphs, and Ramanujan graphs for examples, to create systems of encryption. Our study develops a new symmetric encryption method using undirected graph adjacency matrices and Minimum Spanning Tree (MST) algorithms to make secure cryptographic keys.

Key changes:

- Better Sentence Structure
- More Diverse Vocabulary but practically accurate vocabulary
- Better Logical Flow of information
- Kept all key technical terms and concepts
- Removed the chance of plagiarism but kept the same meaning

Received: July 21, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Adjacency Matrix:

An undirected graph is composed of a vertex set V and an edge set E for connecting those vertices. When there is at least one path in the graph starting and ending at the same vertex, the graph is cyclical or cyclic. When every distinct pair of vertices is connected by an edge one edge, the graph is complete. The adjacency matrix of such a graph is a square matrix with possible values of either a 1 or a 0. If vertex i is connected to vertex j, then the entry in the matrix corresponding to those two vertices is a 1, if they are not connected, the entry in the matrix is a 0. The adjacency-matrix is a square matrix and symmetric because the edges are undirected and, therefore, the connections between vertices are mutual. The adjacency matrix for a graph with l vertices would be a l x l square matrix.

$$\begin{bmatrix} 0 & K_{12} & K_{13} & \cdots & K_{1l} \\ K_{21} & 0 & K_{23} & \cdots & K_{2l} \\ K_{31} & K_{32} & 0 & \cdots & K_{3l} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ K_{l1} & K_{l2} & K_{l3} & \cdots & 0 \end{bmatrix}$$

Minimum Spanning Tree (MST):

A spanning tree of graph G is a sub graph that has all of the vertices of G and that has the minimum number of edges to still be connected. A spanning tree is acyclic and connected, by definition - there are no loops; all vertices are reachable.

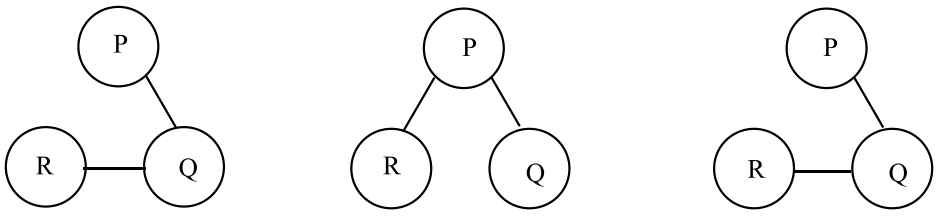


Fig.1 multiple possible spanning trees

As represented in Figure 1, as many spanning trees as you can get from a complete undirected graph. For a connected and undirected graph with P nodes, every spanning tree connects all P nodes with a total of P-1 edges, maintaining connectivity and acyclic. Among the spanning trees for the complete graph, the Nash Minimum Spanning Tree (MST) is the one that has the

Received: July 21, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

smallest total weight of all the weighted edges, where the weights are usually a representation of distance or cost of connections between nodes, [1]. MSTs have extensive use in a variety of fields and a diverse range of applications, such as telecommunications, transportation networks, and cluster analysis [2]. Recent research in cryptography has conducted studies on MST a structure, as they can be used in creating encryption algorithms, in which these algorithms take advantage of the properties of MSPs, mainly ease of processing and determinism.

Literature Survey

Graph theory has been shown to be an important resource in cryptography, especially for encryption, secure communication, and key distribution algorithms. V.A. Ustimenko [3] explored a multimatrix, cryptographic way of building an encryption scheme using polynomial maps over affine spaces on finite commutative rings. N. Tokareva [1] reviewed work on cryptography applications using graph theory with emphasis on sparse graphs, Cayley graphs, and bent functions, applied as examples to mobile networks. Wael Mahmoud Al Etaiwi [2] proposed new cipher architecture involving cyclic graphs, complete graphs, and MSTs. P. Amudha et al. [3] did a review of graph-theoretic applications to cryptography that included encryptions based on Euler graphs. Yamuna et al. [4] examined security when encrypting information doubly using Hamiltonian paths along with complete graphs. Steve Lu et al. [5] encoded images into graph nodes and edges to achieve secure communication. Charles et al. [6] created collision-resistant hash functions using expander graphs. Anmaria Costache et al. researched post-quantum security based on supersingular isogeny graphs. Hyungrok Jo [8] extended the work by Charles et al. in order to create hash functions based on expansion graphs that provided post-quantum security. Adoptin the works of Ustimenko [3], Tokareva [1], Al Etaiwi [2], Amudha et al. [3], Yamuna et al. [4], Lu et al. [5], Charles et al. [6], Costache et al. as well as Jo [8], this paper presented a new type of encryption algorithm using the adjacency matrices of undirected weighted graphs, with keys generated by minimum spanning trees. It was concluded that the encryption and transmission of information using the algorithm could be potentially

Received: July 21, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

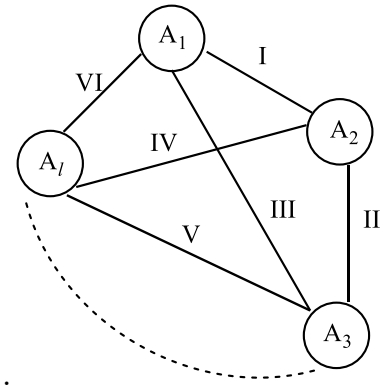
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Proposed Method:

The encryption process presented here uses a double-encryption block cipher. In the first encryption stage, the plaintext block is represented as the adjacency matrix of an undirected graph derived from the message data. In the second stage, it takes the first stage trailing output and performs a logical XOR against the minimum spanning value from the minimum spanning tree (MST) of the corresponded block. All data blocks will be encrypted by two keys that were selected for the data block, which is key geometric strength.

Key Generation and Key Scheduling Algorithm:

The security level in symmetric key cryptography relies on the secret and random nature of the key. Key generation and management, particularly in multicast communications with many potential peers to manage, are also important. In the case described here, key generation is a private function with only the two communicating parties aware of the process. When a group of participants want to exchange secured messages, one participant will select a series of characters randomly from ASCII characters, with the length of the series being equal to the block size agreed upon by the participants, and upload that sequence to the cloud resource as the master key. Each participant then independently generates keys for the first, second, and n blocks from the master key. The ASCII characters chosen from the representative alphabet and presented to participants as A1,A2,A3,…, Al has a block length of lll, and the ASCII characters A1,A2,A3,…, Al can be considered the vertices of an undirected graph. There is a vertex for each character, and to represent a complete graph, edges are created between the vertices weighted by the absolute difference between the ASCII character's decimal values. The absolute difference is measured in a left to right manner cyclically. This completed weighted graph becomes the basis of our MST-derived key, like the weights innovation.



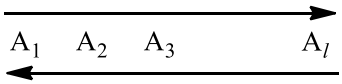
Received: July 21, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

iSSN: 1311-1728 (printed version); iSSN: 1314-8060 (on-line version)

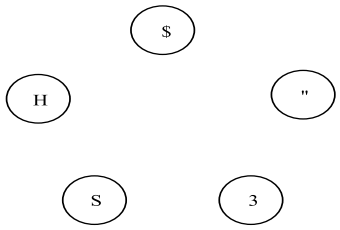
The edge weights I,II,III,… are determined by calculating the differences between the ASCII decimal values of the characters A1,A2,A3,…, Al sequentially from left to right.



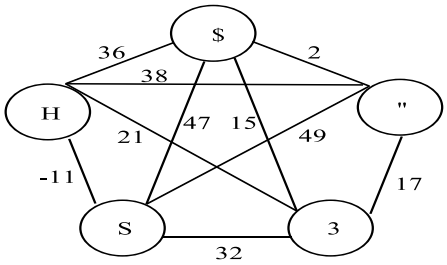
The weights assigned to edges are derived from the differences in ASCII values of the selected characters, calculated as follows:

- Edge weight I equals the ASCII value difference between A2and A1
- Edge weight II is determined by the difference between A3 and A2
- Edge weight III corresponds to the difference between A3 and A1
- Edge weight VI represents the difference between Al and A1

These weights form a fully connected undirected weighted graph, from which a minimum spanning tree (MST) is extracted using standard algorithms such as Prim's or Kruskal's. The MST's adjacency matrix, which is symmetric with zeros on the main diagonal, is modified by replacing the diagonal zeros with increasing integers starting from 1 for the first block key, 2 for the second, and continuing similarly for subsequent blocks.For instance, if the block size is agreed as 5, a group member may randomly pick ASCII characters like $, ", 3, S, and H. These characters are then arranged sequentially as vertices of an undirected graph from left to right.



All the vertices are joined with edges to complete graph.



The edge weight from $ to " is ASCII dec [$-"]= 36-34=2

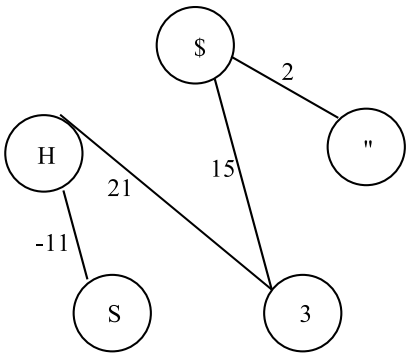
Received: July 21, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

In the above example the minimum spanning tree is with minimum spanning tree weight 27 is



The adjacency matrix of the minimum spanning tree is

	$	"	3	S	H
$	0	2	15	0	0
"	2	0	0	0	0
3	15	0	0	0	21
S	0	0	0	0	-11
H	0	0	21	-11	0

The elements on the principal diagonal are replaced by 1 to get first block key K_1

$$K_1 = \begin{bmatrix} 1 & 2 & 15 & 0 & 0 \\ 2 & 1 & 0 & 0 & 0 \\ 15 & 0 & 1 & 0 & 21 \\ 0 & 0 & 0 & 1 & -11 \\ 0 & 0 & 21 & -11 & 1 \end{bmatrix}$$

The same process is applied to minimum spanning tree for finding the adjacency matrix K2 which acts as the key for second block encryption/decryption since it addresses combination of the key with respect to the next ASCII characters to first block key. For third and so forth blocks K3, K4... are formed used the successive ASCII characters of previous blocks key - for example when ASCII characters are "$ 3 S H for first block key generation then for second block the characters are "% # 4 T I and so on. Thus, the key space would be 25. Since the successive characters are being considered for block key generation process, for a large data

Received: July 21, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

set with a good number of blocks the keys for first and 256 blocks coincide as similarly, if there is a similar message in both then already the cipher text is the same and thus we have the possibility of a cipher text attack occurring. To get around this we are going to select the base as 65536 (256*256)

Encryption:

If two communicating parties want to communicate messages, first they agree upon to use block length and master key length as *l*(say), then both the users generate the secret keys $K_1, K_2, \ldots K_n$ using the ASCII characters uploaded to the cloud. The whole message is split into blocks $M_1\ M_2 \ldots M_n$. Since the present algorithm is symmetric encryption algorithm both encryption and decryption start from the first block. All the characters of each block are written as vertices of an undirected graph.

$M_{11}\ M_{12}\ M_{13} \ldots M_{1l}$; $M_{21}\ M_{22}\ M_{23} \ldots M_{2l}$; …; $M_{n1}\ M_{n2}\ M_{n3} \ldots M_{nl}$, where n is the natural number of blocks $n \in \mathrm{N}$ and *l* is the agreed upon block length.

All the characters are represented as the vertices of an undirected graph.

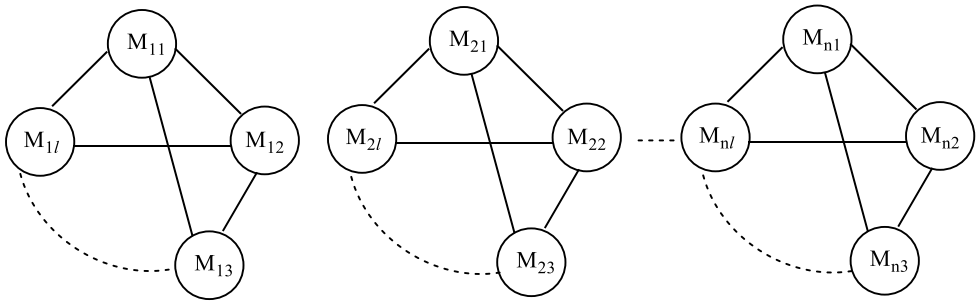


Fig -2: Vertices of an Undirected Graph

The edge weights of the complete graph of all the blocks are calculated using the same procedure as in key generation algorithm, i.e., the differences of ASCII decimal equivalents taken in the counter clockwise direction. Adjacency matrices of each block are found as D_1, D_2, D_n, of order $l \times l$.

First Stage Encryption

At the first stage the adjacency matrices of the undirected graphs of all the data blocks are multiplied with the corresponding keys to get the matrices $D_1(I)$, $D_2(I)$, ……. $D_n(I)$.

$$D_n(I) = D_n * K_n \quad \text{where n is number of block; n=1,2,3…}$$

Received: July 21, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Each element of the $l \times l$ square matrix $D_n(I)$ is reduced to mod 256 to get the matrix $D_n(II)$.

$D_n(II) = Mod [D_n(I), 256]$, n = 1,2, 3…

The elements of the matrix $D_n(I)$ when reduced to mod 256 split into two parts the integer part and the residue part. e.g., 521 when reduced to mod 256 the integer part is 2 and the reduce is 9.

$$521= (2*256) + 9$$

So, the matrix $D_n(I)$ when reduced to mod 256 divides into two parts the integer matrix I_n and $D_n(II)$, where n=1,2,3…

Second Stage Encryption:

All the elements of the first stage encrypted matrices $D_1(II)$, $D_2(II)$, …, Dn(II)

are converted to ASCII equivalent 8-bit binary numbers. Each binary element is undergone logical XOR operation with the ASCII binary equivalent of Minimum Span Value (MSV, total value of the edge weights) of Minimum Spanning Tree computed to generate the first block key to get $D_1(III)$, $D_2(III)$, …, $D_n(III)$. For instance, if the minimum span value of the first block is 27 then each binary element of the first block first stage encrypted matrix is XORed with ASCII binary equivalent of 27. In case if the Minimum Span Value is outside ASCII range, then mod256 value is considered.

$D_n(III) = D_n(II)$ XOR (MSV K_n); n=1,2,3…

All the two stage encrypted data blocks are converted to equivalent ASCII characters which constitutes the chipper text C.

The chipper text C is communicated to the receiver via public channel. Along with the cipher the elements of the integer matrices are also communicated as string of integers. Since the agreed block length is l, a plain text block of l characters is encrypted to l^2 cipher characters and l^2 integers. To authenticate the message the sender inserts ASCII decimal values of first characters of each block at the positions $l^2+1, 2l^2+1, 3l^2+1$, … of both cipher and integer string.

Decryption:

The receiver after confirming the authentication of the sender splits the cipher text C into blocks of length l^2 characters each leaving the decimal values at l^2+1, $2l^2+1$, $3l^2+1$, … Now the cipher

Received: July 21, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

blocks are $C_1, C_2, \ldots, C_n$. Decryption takes place at two different stages, the first stage using logical XOR and the second stage using adjacency matrix.

First Stage Decryption:

All the l^2 characters of each block are converted to ASCII binary equivalents written as l X l matrix and each binary number is undergone logical XOR operation with Minimum Span Value (MSV) of the Minimum Spanning Tree (MST) obtained while generating the key K_1 for first block encryption/decryption.

$D_n(II) = C_n$ XOR [MSV (K_n)]; n = 1,2,3, …

Second Stage Decryption:

Then the 8-bit binary numbers of each block are converted to ASCII decimals. The integer string received along with the cipher text C is split into l^2 each leaving the decimal values at l^2 +1, $2l^2$+1, $3l^2$+1, … represented as arrays I_n; n = 1,2,3… of order lxl. Each array is multiplied with 256 and corresponding elements of D_n (II) are added to get the matrices D_n(I).

$D_n(I) = (256 * I_n) + D_n(II)$; n=1,2,3…

$D_n = D_n(I) * (K_n)^{-1}$; n = 1,2,3…

D_n, n=1,2,3… are adjacency matrices of data blocks [adjacency matrices of weighted undirected graphs]. After getting adjacency matrices of all data blocks original message blocks are determined using the decimal equivalents of first characters of each block placed at l^2 +1, $2l^2$+1, $3l^2$+1 … positions in the cipher. Consider the adjacency matrix of first data block D_1 and the ASCII decimal of first character of first block. Suppose the first character of the first block is say '83(S)'.

The first block entries are 83, M_{12}, M_{13} … M_{1l}. Using adjacency matrix D_1, the first entry 83, other characters M_{12} M_{13} ……M_{1l} are determined.

Received: July 21, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

$$D_1 = \begin{array}{c} \\ 83 \\ M_{12} \\ M_{13} \\ \vdots \\ M_{1l} \end{array} \begin{array}{c} \begin{array}{ccccc} 83 & M_{12} & M_{13} & \text{-----} & M_{1l} \end{array} \\ \begin{bmatrix} a_{11} & a_{12} & a_{13} & \text{-----} & a_{1l} \\ a_{21} & a_{22} & a_{23} & \text{-----} & a_{2l} \\ a_{31} & a_{32} & a_{33} & \text{-----} & a_{3l} \\ \vdots & \vdots & \vdots & & \vdots \\ a_{l1} & a_{l2} & a_{l3} & \text{-----} & a_{ll} \end{bmatrix} \end{array}$$

Here a_{12} = ASCII Decimal of [M_{12}-83]

Therefore, $M_{12} = a_{12} - 83$

$M_{13} = a_{13} - 83$

⋮ ⋮

$M_{1l} = a_{1l} - 83$

Same procedure is applied for all the blocks using the adjacency matrices D_2, D_3, -----, D_n using the first characters of all the blocks sent by the communicator to get the original message.

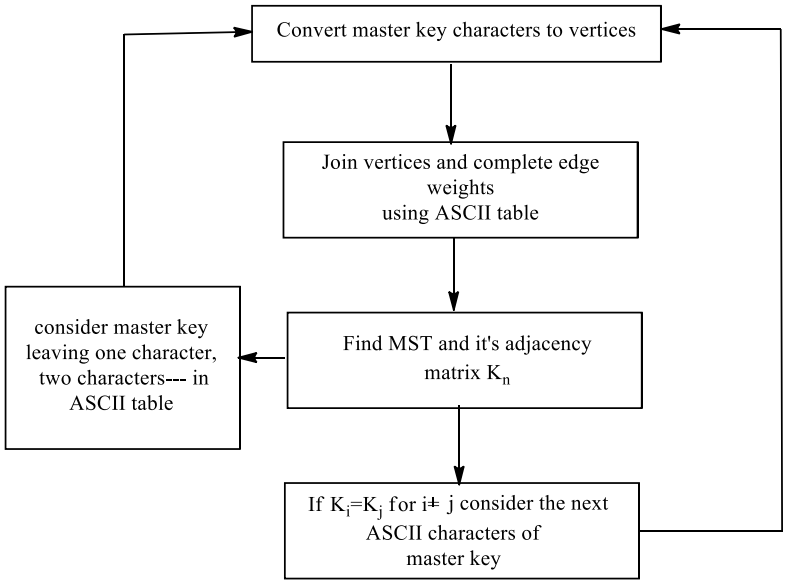


Fig.-3: Key Generation Algorithm

Received: July 21, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

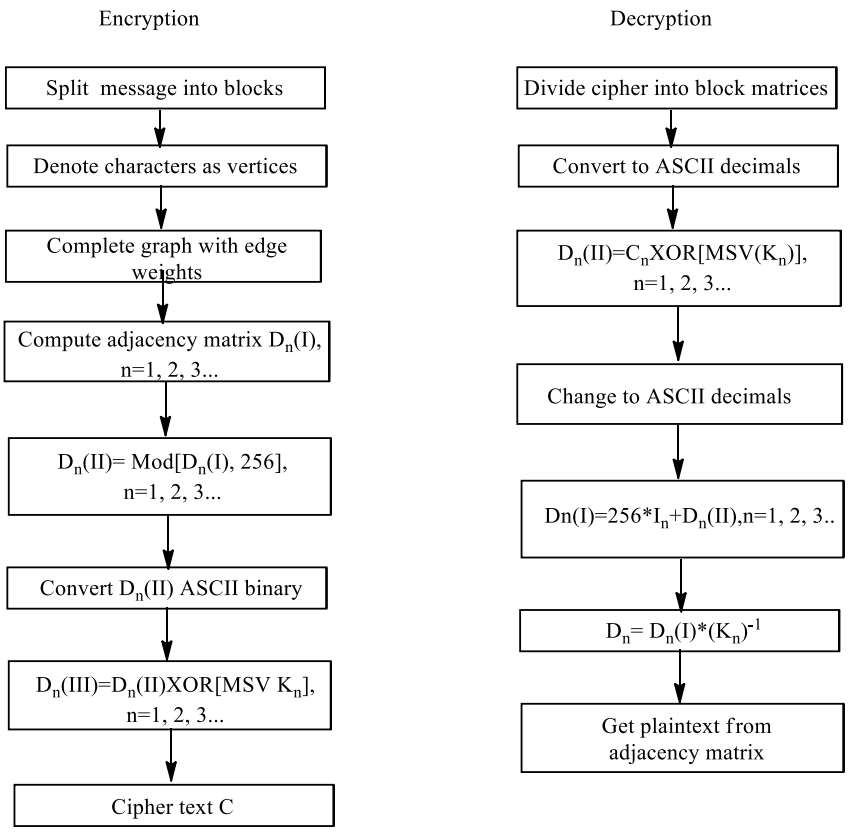


Fig.-3(a) and (b): Key Generation Algorithm Encryption and Decryption

Example:

If a group of communicators want to transmit messages, they agree upon to use the block length as 5 characters (say). One of the entities say Alice selects randomly 5 ASCII characters “S A I ! #” and publish these characters in the cloud as public key. All the members of the group generate the keys for first, second and subsequent blocks using this master key and above-mentioned key generation, key scheduling algorithm.

If two legitimate users Alice and Bob want to communicate messages, both Alice and Bob establish the keys. If Alice wants to send the message GITAM/UNIVE/RSITY, she divides it into three blocks each having 5 characters since the master key length is 5.

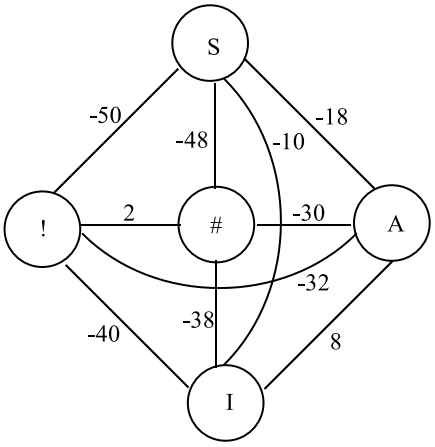
She writes all the characters of each block as vertices of un directed graphs. The graphs are completed joining all vertices with edge weights differences of ASCII decimal values.

Received: July 21, 2025

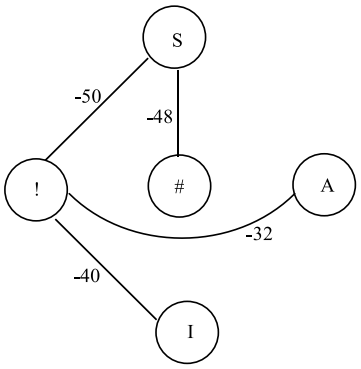
International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)



For this complete undirected weighted graph minimum spanning tree is formed by using prim's algorithm or Kruskal's algorithm.



Adjacency matrix of the minimum spanning tree is

$$\begin{array}{c} \\ S \\ A \\ I \\ ! \\ \# \end{array} \begin{array}{c} \begin{array}{ccccc} S & A & I & ! & \# \end{array} \\ \begin{bmatrix} 0 & 0 & 0 & -50 & -48 \\ 0 & 0 & 0 & -32 & 0 \\ 0 & 0 & 0 & -40 & 0 \\ -50 & -32 & -40 & 0 & 0 \\ -48 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

Replace the diagonal elements with 1 to get first block key K_1

Received: July 21, 2025

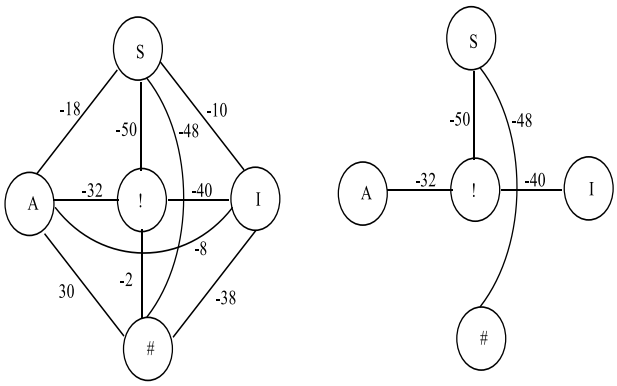
International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

$$K_1 = \begin{bmatrix} 1 & 0 & 0 & -50 & -48 \\ 0 & 1 & 0 & -32 & 0 \\ 0 & 0 & 1 & -40 & 0 \\ -50 & -32 & -40 & 1 & 0 \\ -48 & 0 & 0 & 0 & 1 \end{bmatrix}$$

For getting K_2 we consider the following graph and the minimum spanning tree



Adjacency matrix of the minimum spanning tree is

$$\begin{array}{c} \\ S \\ I \\ \# \\ A \\ ! \end{array} \begin{array}{c} \begin{array}{ccccc} S & I & \# & A & ! \end{array} \\ \begin{bmatrix} 0 & 0 & -48 & 0 & -50 \\ 0 & 0 & 0 & 0 & -40 \\ -48 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -32 \\ -50 & -40 & 0 & -32 & 0 \end{bmatrix} \end{array}$$

Replace the diagonal elements with 2 to get first block key K_2

$$K_2 = \begin{bmatrix} 2 & 0 & -48 & 0 & -50 \\ 0 & 2 & 0 & 0 & -40 \\ -48 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 2 & -32 \\ -50 & -40 & 0 & -32 & 2 \end{bmatrix}$$

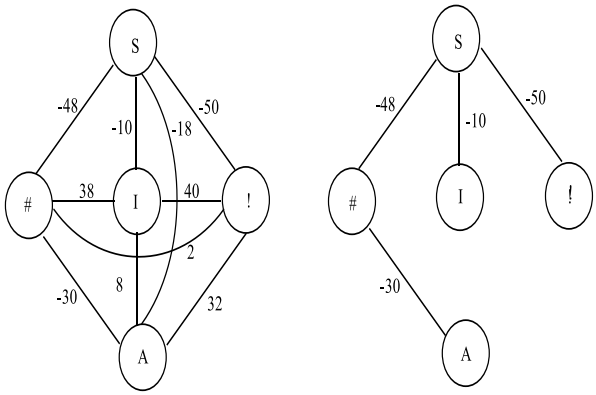
To get K_3 consider

Received: July 21, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)



Adjacency matrix of the above minimum spanning tree is

$$\begin{array}{c} \\ S \\ ! \\ A \\ \# \\ I \end{array} \begin{array}{c} \begin{array}{ccccc} S & ! & A & \# & I \end{array} \\ \begin{bmatrix} 0 & -50 & 0 & -48 & -10 \\ -50 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -30 & 0 \\ -48 & 0 & -30 & 0 & 0 \\ -10 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

Replace the diagonal elements with 3 to get first block key K_3

$$K_3 = \begin{bmatrix} 3 & -50 & 0 & -48 & -10 \\ -50 & 3 & 0 & 0 & 0 \\ 0 & 0 & 3 & -30 & 0 \\ -48 & 0 & -30 & 3 & 0 \\ -10 & 0 & 0 & 0 & 3 \end{bmatrix}$$

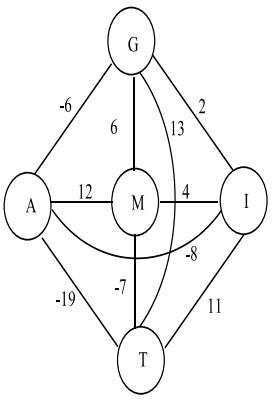
Now arrange first block characters of plain text as vertices of an undirected weighted graph and consider its adjacency matrix M_1.

Received: July 21, 2025

International Journal of Applied Mathematics

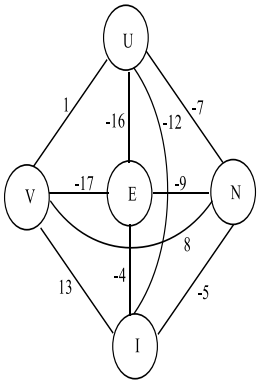
Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

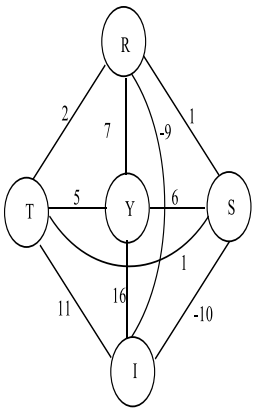


$$M_1 = \begin{array}{c} \\ G \\ I \\ T \\ A \\ M \end{array} \begin{array}{c} \begin{array}{ccccc} G & I & T & A & M \end{array} \\ \begin{bmatrix} 0 & 2 & 13 & -6 & 6 \\ 2 & 0 & 11 & -8 & 4 \\ 13 & 11 & 0 & -19 & -7 \\ -6 & -8 & -19 & 0 & 12 \\ 6 & 4 & -7 & 12 & 0 \end{bmatrix} \end{array}$$

In the similar way we construct graphs and its adjacency matrices M_2, M_3 of second and third block characters of plain text we get



$$M_2 = \begin{array}{c} \\ U \\ N \\ I \\ V \\ E \end{array} \begin{array}{c} \begin{array}{ccccc} U & N & I & V & E \end{array} \\ \begin{bmatrix} 0 & -7 & -12 & 1 & -16 \\ -7 & 0 & -5 & 8 & -9 \\ -12 & -5 & 0 & 13 & -4 \\ 1 & 8 & 13 & 0 & -17 \\ -16 & -9 & -4 & -17 & 0 \end{bmatrix} \end{array}$$



$$M_3 = \begin{array}{c} \\ R \\ S \\ I \\ T \\ Y \end{array} \begin{array}{c} \begin{array}{ccccc} R & S & I & T & Y \end{array} \\ \begin{bmatrix} 0 & 1 & -9 & 2 & 7 \\ 1 & 0 & -10 & 1 & 6 \\ -9 & -10 & 0 & 11 & 16 \\ 2 & 1 & 11 & 0 & 5 \\ 7 & 6 & 16 & 5 & 0 \end{bmatrix} \end{array}$$

First stage encryption:

$D_1(I) = M_1 * K_1$

Received: July 21, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

$$M_1 * K_1 = \begin{bmatrix} 0 & 2 & 13 & -6 & 6 \\ 2 & 0 & 11 & -8 & 4 \\ 13 & 11 & 0 & -19 & -7 \\ -6 & -8 & -19 & 0 & 12 \\ 6 & 4 & -7 & 12 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & -50 & -48 \\ 0 & 1 & 0 & -32 & 0 \\ 0 & 0 & 1 & -40 & 0 \\ -50 & -32 & -40 & 1 & 0 \\ -48 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$D_1(I) = \begin{bmatrix} 12 & 194 & 253 & -590 & 6 \\ 210 & 256 & 331 & -548 & -92 \\ 1299 & 619 & 760 & -1021 & -631 \\ -582 & -8 & -19 & 1316 & 300 \\ -594 & -380 & -487 & -136 & -288 \end{bmatrix}$$

We apply mod 256 on each element of $D_1(I)$ and we split $D_1(I)$ into two matrices I_1 and $D_1(II)$.

$$I_1 = \begin{bmatrix} 0 & 0 & 0 & -3 & 0 \\ 0 & 1 & 1 & -3 & -1 \\ 5 & 2 & 2 & -4 & -3 \\ -3 & -1 & -1 & 5 & 1 \\ -3 & -2 & -2 & -1 & -2 \end{bmatrix} \quad D_1(II) = \begin{bmatrix} 12 & 194 & 253 & 178 & 6 \\ 210 & 0 & 75 & 220 & 164 \\ 19 & 107 & 248 & 3 & 137 \\ 186 & 248 & 237 & 36 & 44 \\ 174 & 132 & 25 & 120 & 224 \end{bmatrix}$$

By continuing this process, we get

$$I_2 = \begin{bmatrix} 5 & 2 & -1 & 2 & 0 \\ 2 & 1 & 1 & 1 & 3 \\ 0 & 0 & 2 & 0 & 1 \\ 0 & 2 & -1 & 2 & -2 \\ 0 & -1 & 2 & -1 & 6 \end{bmatrix} \quad D_2(II) = \begin{bmatrix} 96 & 114 & 232 & 2 & 216 \\ 164 & 104 & 70 & 48 & 28 \\ 176 & 150 & 64 & 154 & 120 \\ 228 & 184 & 234 & 32 & 108 \\ 160 & 238 & 248 & 222 & 168 \end{bmatrix}$$

$$I_3 = \begin{bmatrix} -1 & 0 & -1 & 1 & 0 \\ -1 & -1 & -1 & 0 & 0 \\ -1 & 1 & -2 & 1 & 0 \\ -1 & -1 & 0 & -2 & -1 \\ -3 & -2 & -1 & -4 & -1 \end{bmatrix} \quad D_3(II) = \begin{bmatrix} 40 & 3 & 169 & 20 & 21 \\ 151 & 206 & 196 & 255 & 8 \\ 41 & 164 & 182 & 209 & 138 \\ 162 & 159 & 33 & 86 & 251 \\ 249 & 180 & 154 & 223 & 186 \end{bmatrix}$$

Second stage encryption

All the elements of the matrices $D_1(II)$, $D_2(II)$, $D_3(II)$ converted into ASCII equivalent 8-bit binary and each element is undergone logical XOR with ASCII binary equivalent of minimum span value of corresponding minimum span tree of each block to get $D_1(III)$, $D_2(III)$, $D_3(III)$.

Received: July 21, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Here $D_1(III) = [D_1(II)]$ XOR $[MSV\ K_1]$

$$D_1(III) = \begin{bmatrix} 01011010 & 10010100 & 10101011 & 11100100 & 01010000 \\ 10000100 & 01010110 & 00011101 & 10001010 & 11110010 \\ 01000101 & 00111101 & 10101110 & 01010101 & 11011111 \\ 11101100 & 10101110 & 10111011 & 01110010 & 01111010 \\ 11111000 & 11010010 & 01001111 & 00101110 & 10110110 \end{bmatrix}$$

In a similar way we calculate $D_2(III)$, $D_3(III)$ and convert each binary with its equivalent ASCII decimal to get cipher for each block. After two stage encryption the cipher becomes

Z””«äP"VGSšòE=®Uβî®>>rzòO ¶ 71 6$¾Tžò>DLEfJæÀSYNÌ•²î¼v:Ö¸®^p 85 ^ußbcá¸²‰>_ÒÀ§üÔéW Âì©Ì 82.

The integer string is

0,0,0,3,0,0,1,1,3,1,5,2,2,4,3,3,1,1,5,1,3,2,2,1,2,71,5,2,1,2,0,2,1,1,1,3,0,0,2,0,1,0,2,1,2,2,0,1,2, 1,6,85, 1,0,1,1,0,1,1,1,0,0,1,1,2,1,0,1,1,0,2,1,3,2,1,4,1,82

The decimal equivalents of first characters of each block G →71, U → 85, R→ 82 are inserted at 26th, 51st, 76th positions of both cipher text and integer string to authenticate the communication. The cipher and the integer string are sent to Bob.

Security Analysis and conclusions

The present block cipher uses the Minimum Spanning Tree (MST) and Minimum Spanning Value (MSV) of an un directed weighted graph as a secret key. Secret keys are not exchanged among users, but instead they are generated by all legitimate users of the group using the key generation and scheduling algorithm. For symmetric ciphers, key distribution and safeguarding keys are the most challenging parts of secret communications. But in the present algorithm only the master key characters are available to anyone, inner keys for all blocks are generated by user and their participation in the key generation algorithm. This gives the most security to key, unless it has been compromised. The message sent is encrypted in two stages. At first stage the keys are used to encrypt. At the second stage the logical XOR is used to combine the two encryption processes. Each block uses a different key. In the first example "GITAMUNIVERSITY" the alphabet I is repeated 3 times, T is repeated 2 times. In the cipher repetition of characters is much less. In the second example, the same message blocks

Received: July 21, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

'GITAMGITAMGITAM' are mapped to Z"«äP„VGSŠòE=®Ußì®»rzøÒO.¶ 2BLb*z6à&ò.xÆìFzF¬ÖÒLF^„Nâ öp-RSd(êgHŽ¡áL!USpb±,SYN|– õ ². For a plain text with n characters, there will be n^2 characters in the cipher text. The decimal representations of the first n characters are added at unique positions of both the cipher string and number string. If the decimals

Coincide at those points in the cipher and number string, then the receiver knows the cipher is unaltered at any time in its journey from sender to receiver. In this measured way, the sender authenticates both the original message and the integrity of that message. Even with more ambiguity in the text of a ciphered message the computation needed for the cipher will not be any more of a workload than the original encryption itself.

Performance Analysis:

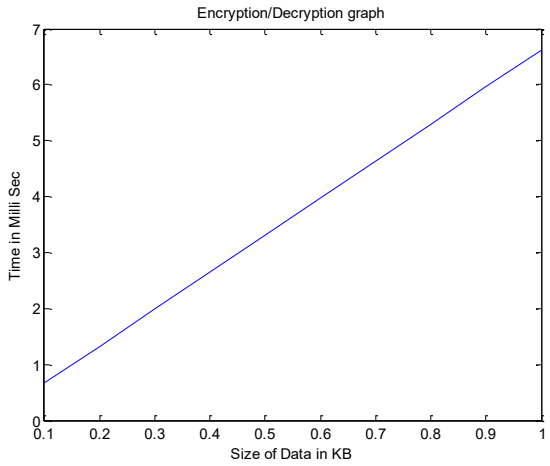
The running time for varying size data is measured on Intel i5,64 bit processor with 16 GB ram using MATLAB 14. The following table and provide the encryption/decryption time for varying size data.

Table -1: Performance Analysis

Size of data in KB	Time in Milli Sec
0.1	0.662
0.2	1.256
0.3	1.991
0.4	2.449
0.5	3.457
0.6	3.972
0.7	4.634
0.8	5.289
0.9	5.893
1	6.62

Received: July 21, 2025

International Journal of Applied Mathematics
Volume 38 No. 3s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)



Graph-1: The encryption/decryption

The encryption/decryption graph is linear. For 1MB data the execution time is roughly 6 milli seconds which is very less when compared with conventional block ciphers.

Further Scope of the work

Preferably, the cipher and the original message are supposed to have almost same size to manage the storage space and communication time. Improvements can be done to minimize the cipher size. One of the ideas in this direction is to use the cipher block chaining mode after the first block encryption. This work will be extended in future.

Acknowledgement

The authors express their sincere gratitude to the Department of Mathematics, GITAM (Deemed to be University), Visakhapatnam, for providing the necessary support and academic environment to carry out this research work.

Funding

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Conflict of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Author Contributions

S. Sarvalakshmi: Conceptualization, Methodology, Software, Algorithm Design, Writing – Original Draft.

Received: July 21, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Dr. CH. Suneetha: Supervision, Validation, Formal Analysis, Writing – Review & Editing. Dr. M.G. Vara Prasad: Methodology, Software Development, Investigation, Data Curation. Dr. Mutyala Suresh: Writing – Review & Editing, Resources. Gali Lalitha Devi: Investigation, Visualization, Formal Analysis.

Ethics Approval

Not applicable. This study does not involve any human participants or animal subjects.

Data Availability

All data generated or analyzed during this study are included in this published article. No additional datasets were generated.

References:

1. Tutorials Point. (n.d.). *Spanning tree*. https://www.tutorialspoint.com/data_structures_algorithms/spanning_tree.htm
2. UT Dallas. (n.d.). *MST applications*. http://personal.utdallas.edu/~besp/teaching/mst.applications.pdf
3. Ustimenko, V. A. (2018). On algebraic graph theory and non-bijective multivariate maps in cryptography. *HAL Open Science*. https://doi.org/10.15439/2018F204
4. Tokareva, N. (2014, September). Connection between graph theory and cryptography. *G2C2: Graphs and Groups, Cyclic and Coverings*, Novosibirsk, Russia.
5. Al Etaiwi, W. M. (2014). Encryption algorithm using graph theory. *Journal of Scientific Research and Reports*, *3*(19), 2519–2527.
6. Amudha, P., Sagayaraj, A. C. C., & Sheela, A. C. S. (2018). An application of graph theory in cryptography. *International Journal of Pure and Applied Mathematics*, *119*(3), 375–383.
7. Yamuna, M., Gogia, M., Sikka, A., & Khan, M. J. H. (2012). Encryption using graph theory and linear algebra. *International Journal of Computer Applications*.
8. Lu, S., Ostrovsky, R., & Manchala, D. (2008). Visual cryptography on graphs. In *COCOON 2008* (pp. 225–234). CiteSeerX.
9. Charles, D. X., Goren, E. Z., & Lauter, K. E. (2009). Cryptographic hash functions from expander graphs. *Journal of Cryptology*, *22*, 93–113.

Received: July 21, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

10. Costache, A., Feigon, B., Lauter, K., Masierer, M., & Puskar, A. (2018). Ramanujan graphs in cryptography. *arXiv*. https://arxiv.org/abs/1806.05709

11. Jo, H. (2019). Ramanujan graphs for post quantum cryptography. *ResearchGate*. http://www.researchgate.net/publication/337150777

Received: July 21, 2025