International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 09 | Sept - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Symmetric Encryption Using Graph Representation:Survey

Nandish M[1], Sinchana K G[1], Srujana R Patel[1], Uthpala R[1], Anagha C R[1]
[1]Dept. of CSE., JNNCE Shivamogga, Visvesvaraya Technological University, Belagavi – 590018

***Abstract*—The project explores a novel approach to symmetric encryption, leveraging graph representation techniques. The core idea is to transform plain text data into a graph structure, where nodes and edges encode information securely. Encryption involves applying graph-based algorithms to alter the structure, ensuring confidentiality and resistance against common cryptographic attacks. The symmetric key is used both to generate the initial graph and to guide the transformation process, allowing efficient encryption and decryption. This method aims to enhance traditional symmetric encryption by introducing complexity through graph theory, potentially improving security and performance. The experimental results demonstrate the feasibility and robustness of the approach in the protection of sensitive data.**

***Keywords*—Symmetric Cryptography, Graph Theory, Data Encryption, Network Security, Key Management**

I. INTRODUCTION

In today’s digital era, the secure transmission and storage of data have become critically important due to the increasing use of online platforms for communication, finance, healthcare, and more. Cryptography serves as a powerful tool for safeguarding information by converting it into an unreadable format to prevent unauthorized access. Among various cryptographic approaches, symmetric key encryption stands out for its simplicity and efficiency in encrypting large volumes of data, using the same key for both encryption and decryption. Recently, graph theory has emerged as an innovative tool in enhancing cryptographic techniques. Graphs, being mathematical structures used to model pairwise relations between objects, offer a rich and flexible framework for representing and manipulating data. In symmetric cryptography, graph-based methods utilize vertices and edges, along with adjacency matrices, to encode plaintext into complex graph structures that are hard to decipher without the appropriate key.

This paper surveys the various methods developed in the domain of symmetric encryption using graph representation, exploring how graphs are employed to improve encryption strength, increase key complexity, and provide novel encoding mechanisms. The integration of graph theory not only enriches the cryptographic process but also offers potential resistance against traditional cryptanalytic attacks due to the structural complexity introduced.

II. LITERATURE SURVEY

A. Symmetric Encryption Using Graph Representation

P.A.S.D. Perera et al. [1] proposed a novel symmetric encryption method that leverages graph theory to enhance data security. In this approach, plaintext is divided into blocks, and each character in a block is represented as a vertex in a graph. Edges are formed between the vertices, with weights determined by the difference in character indices from a predefined lookup table. To indicate the start of each block, a special character derived from the sum of values in the shared key matrix is added. The complete graph is then constructed by introducing additional edges with random weights, and the resulting graph is represented as an adjacency matrix. The final cipher text is produced by multiplying this matrix with the secret key. Decryption involves reversing the process using the inverse of the key matrix. This method increases resistance to brute-force attacks. Although the algorithm leads to a larger cipher text size and requires more computational operations, it introduces a unique and graphically structured approach to symmetric encryption.

B. Encryption Algorithm Using Graph Theory

Wael M. Al Etaiwi [2] provided a symmetric encryption technique where each character of the plaintext is converted into a graph vertex, with edges weighted based on their position in an encoding table. A complete graph is constructed and represented as an adjacency matrix, from which a minimum spanning tree is derived. These matrices are then multiplied and further encrypted using a shared key matrix to produce the final cipher text. The algorithm adds complexity and resistance against traditional attacks by introducing structure and mathematical transformations. Compared to existing methods, this graph-based approach offers a novel and secure encryption mechanism, particularly effective for small messages, though optimization is needed for large-scale applications.

C. Cryptography – A Graph Theory Approach

Uma Dixit [3] proposed a method in which characters of the plaintext are converted into graph vertices and linking them according to their sequence to form cycle and weighted graphs. Through encoding schemes and labeling using standard character-distance tables, a complete graph is formed. A public key is applied to the graph’s distance matrix, producing cipher text that is complex and difficult to reverse without the appropriate decryption key. This graph-based encryption model not only provides data confidentiality but also integrates key elements of one-way functions, which are essential for public-key systems.

DOI: 10.55041/IJSREM52525

D. An Application of Graph Theory in Cryptography

P. Amudha et al. [4] provided a model where each character in a message is transformed into a binary format through ASCII encoding, followed by XOR operations. These binary values are then mapped into graph structures, particularly Euler graphs, with their adjacency matrices used as encrypted representations. The Hamiltonian circuit derived from these graphs serves as a critical component for secure decryption. This approach significantly improves the security of the cryptographic system by increasing complexity and making unauthorized decryption exceedingly difficult. The method also highlights the suitability of graph-based encryption for digital communication, offering an innovative alternative to traditional cryptosystems.

E. A Study on Graph Theory in Cryptography Using Python

Dr. M. Lalitha et al. [5] proposed an innovative method where each character of a plaintext message is encrypted into an Euler graph. They utilize Python programming to implement an algorithm that converts each character into its ASCII value, transforms it into binary, performs an XOR operation with a fixed key (binary 32), and constructs an adjacency matrix based on the resulting binary values. These adjacency matrices represent Euler graphs, and a Hamiltonian circuit derived from each graph is used as the decryption key. To decode the message, the encoding steps are reversed using Python code. This graph-based encryption technique enhances the complexity and security of the encrypted message, making it more resistant to unauthorized access.

F. A New Symmetric Key Cryptographic Algorithm using Paley Graphs and ASCII Values

Zhour Oumazouz et al.[6] The proposed algorithm first converts plaintext into binary using ASCII values, then encrypts it by altering each bit based on its position and the neighborhood structure in a Paley graph of prime order, which acts as the secret key. Decryption is performed by solving linear equations over the field , essentially reversing the encryption process using the same graph structure. The strength of the algorithm lies in the properties of Paley graphs, including their strong regularity and the unpredictable behavior of quadratic residues and prime numbers. This makes brute-force attacks the only feasible method of cryptanalysis, thus enhancing security.

G. Algorithm of Encryption Using Graph Theory

Huda Anwar et al.[7] The approach involves representing plaintext as a complete graph, with each character encoded using a specialized encryption table and assigned to graph nodes. The edges between nodes are labeled using the differences in character values, forming an adjacency matrix (M1). A cycle matrix (M2) is then derived by removing internal edges, and is modified using an alphabet encoding table to get M2. This is further combined with M1 through matrix multiplication to generate M3. A shared upper triangular matrix (key K) is used to produce the final cipher matrix (C). For decryption, inverse matrices of the key and M1 are used in reverse operations to retrieve M2, from which the original message is decoded using the diagonal values. This method leverages complete and cyclic graphs along with matrix operations to ensure a complex and secure encryption scheme.

H. Symmetric Encryption Algorithm Using Graph Representation

Safaa Hraiz et al.[8] In this approach, plaintext is divided into blocks and each character in a block is represented as a vertex on a graph. Edges are formed between the vertices, with weights determined by the difference in character indices from a predefined lookup table. To indicate the start of each block, a special character derived from the sum of values in the shared key matrix is added. A complete graph is then constructed by introducing additional edges with random weights, and the resulting graph is represented as an adjacency matrix. The final cipher text is produced by multiplying this matrix by the secret key. Decryption involves reversing the process using the inverse of the key matrix. This method increases resistance to brute-force attacks. Although the algorithm leads to a larger cipher text size and requires more computational operations, it introduces a unique and graphically structured approach to symmetric encryption.

I. A Fuzzy Graph Theory Approach to Symmetric Key Cryptography

C.Ruby Sharmila et al. [9] explores the application of machine learning techniques to predict the thrust generated by solid propellant rocket engines. Recognizing the complexity and non-linearity involved in thrust generation, the algorithms such as Linear Regression, Decision Tree, Random Forest, and Gradient Boosting are used to develop predictive models based on input parameters like chamber pressure, nozzle dimensions, and burn time. The dataset used for the study is derived from empirical rocket engine data. Performance metrics such as R^2 score and mean squared error are used to evaluate and compare the models. Among the models tested, gradient boost emerged as the most accurate, indicating its suitability to predict the thrust characteristics of solid propellants. The study demonstrates how machine learning can help optimize and model the behavior of the propulsion system efficiently.

J. Cryptography: Graph Theory Implementation in Text Encryption and Decryption

Albert Ghazaly [10] The approach involves converting plaintext characters into ASCII values, followed by binary encoding. These encoded values are then integrated into graph structures, particularly directed graphs, where vertices represent characters and edges signify relationships based on binary transitions. Adjacency matrices and graph traversals, such as Depth-First Search (DFS), are utilized to transform and obscure the original message. The encrypted output depends on the specific graph path taken, making decryption without the original graph model computationally challenging. This approach illustrates how graph theory can enhance encryption methods by introducing structural complexity and data obfuscation in secure communications.

K. Data Security: A New Symmetric Cryptosystem based on Graph Theory

K. Bekkaoui et al. [11] The method constructs a unique labeled graph from the plaintext message, where each character corresponds to a graph node. Encryption is achieved through transformations in the structure of the graph, such as reordering vertices, modifying adjacency relations, and encoding graph properties into a cipher text. The decryption process requires reconstructing the original graph using a shared secret key and performing reverse transformations. This graph-based approach increases resistance against brute-force and statistical attacks due to the complex inter-node relationships and dynamic encoding, highlighting graph theory's strength in modern cryptographic design.

L. Implementation of Data Encryption and Decryption Technique using Graph Theory

D.Janardhan et al. [12] The approach involves representing each character of the plaintext as a node in a graph, with edges determined by the relational logic of ASCII values and positional patterns. The graph is then transformed through edge manipulation, node renaming, and encoding of adjacency matrix values, generating a secure ciphertext. The decryption phase reverses these operations based on a shared key and known transformation rules. This method enhances security by leveraging the inherent complexity and flexibility of graph structures, ensuring data confidentiality and integrity while offering resistance to conventional cryptanalytic attacks.

M. Some Graph-Based Encryption Techniques

D.Narayan H et al. [13] introduces encryption schemes using bipartite, corona, and star graphs. Plaintext characters are encoded numerically and encrypted using RSA-based public key cryptography. These encrypted values are then embedded into structured graphs where vertex and edge properties represent cryptographic transformations. Each method employs modular arithmetic, prime-based key generation, and inverse operations for secure decryption. The graph-based approach introduces structural complexity, making cryptanalysis more challenging. This layered encryption model ensures robust security in modern digital communication systems.

N. Advancing Cryptographic Security through Graph Theory: A Comprehensive Review

H.Gena et al.[14] proposed a graph-based encryption scheme that enhances cryptographic security using graph theory. Characters of the plaintext are encoded into numeric values using a custom encryption table and assigned to vertices of a complete graph. Edge weights are calculated based on differences in values, forming matrices used for encryption. A modified cycle matrix incorporates alphabet encoding, and matrix operations involving a triangular key matrix and Hamiltonian cycle generate the final cipher text. The decryption process reverses these steps using matrix inverses and decoding rules. The method ensures strong encryption through layered mathematical transformations and is well-suited for secure digital communication.

O. Symmetric Key Encryption Using Modular Traversal of Ananta-Graphs

V.H.R.Vidyashree et al. [15] In this approach, plaintext characters are mapped onto graph nodes, and encryption is performed using a modular arithmetic-based traversal pattern that dynamically alters the graph's structure based on the key. The adjacency matrix derived from this traversal serves as the cipher representation. Decryption involves reversing the traversal using the same key parameters. This approach introduces a high degree of nonlinearity and structural variation, enhancing resistance against brute-force and pattern-based attacks, and demonstrating the potential of advanced graph constructs in secure symmetric cryptography.

P. Trustless Distributed Symmetric-key Encryption

F.Le Moue¨l et al. [16] proposed a trustless threshold symmetric-key encryption scheme (TDiSE) by eliminating the need for a trusted third party. It uses a Distributed Key Generation (DKG) protocol based on Shamir's Secret Sharing Scheme (SSSS) to securely distribute secret keys among participants. Commitment schemes like Pedersen and hash-based commitments are employed to ensure verifiability and integrity of key shares. A strongly-secure Distributed Pseudo-Random Function (DPRF), under the Decisional Diffie-Hellman (DDH) assumption, is used for encryption operations. The scheme adopts a k-of-n threshold model for collaborative encryption/decryption. A test encryption process validates key correctness, and the protocol is implemented in Python with performance benchmarking against DiSE.

TABLE I: Summary of Techniques used in Symmetric Encryption model

Reference / Author	**Method / Approach**	**Key Features / Techniques**	**Findings / Inference**
P.A.S.D. Perera et al. (2021)	Symmetric Key Cryptography using Graph Theory	Graph-based encryption; matrix as secret key; complete weighted graphs; adjacency matrix; encoding table-based mapping	Generates ciphertext larger than plaintext; hard-to-guess matrix key; resistant to cryptanalysis
Wael M. Al Etaiwi (2014)	Symmetric Encryption using Graph Theory	Cycle Graph, Complete Graph, Minimum Spanning Tree, Adjacency Matrices, Shared Key, Matrix Multiplication, Encoding Table	Novel symmetric encryption method; efficient for small plaintexts; accuracy proven via experiments; increased cipher size and time with text size.
Uma Dixit (2025)	Graph-Theory-Based Public Key Encryption using Spanning Tree	Cycle and complete graphs constructed;Edge weights via encoding table;Minimal Spanning Tree (MST);Matrix encryption using public key	Graph transformations and spanning trees; showed how graph theory enhances message security and reversibility.
P. Amudha et al. (2018)	Euler Graph-based Encryption Algorithm	XOR operation with binary values, adjacency matrix construction, Hamiltonian circuit as decryption key, ASCII to binary conversion	Each character converted into an Euler graph; highly secure due to graph complexity; resistant to interpretation without knowledge of the Hamiltonian circuit and encoding rules
Dr. M. Lalitha et al. (2023)	Graph-based Encryption with Python	Euler and Hamiltonian Graphs, Adjacency Matrix, XOR cipher, ASCII to binary conversion, Python encoding/decoding	A novel cryptographic algorithm using graph theory; each character is encrypted into an Euler graph; secure due to complex adjacency matrices and circuits; Encryption and decryption using Python code.
Zhour Oumazouz et al. (2021)	Symmetric Key Encryption using Paley Graphs	Paley Graphs over finite field Zp;ASCII to Binary conversion; Neighborhood-based bit transformation; Decryption via linear equations over Z	Achieved secure encryption through unknown behavior of quadratic residues and Paley graph properties; Resistant to cryptanalysis except brute-force attack
Huda Anwar et al. (2023)	Symmetric encryption using complete graph theory	Complete Graphs (K), Adjacency Matrices, Cycle Matrix, Alphabet Encoding Table, Matrix Operations, Upper Triangular Key Matrix (K)	A robust encryption algorithm using graph structures and matrix operations; ensures data complexity and effective decryption.
Safaa Hraiz et al. (2017)	Symmetric encryption using graph representation	Graph-based encryption, adjacency matrix, complete graphs, special character indexing,key-based transformation	Secure against brute force; encryption complexity is $O(n^3)$; ciphertext size is $O(n^2)$; extendable to multimedia
C. Ruby Sharmila et al. (2025)	ECC-based Lightweight Cryptography for IoT	Elliptic Curve Cryptography (ECC), Key exchange protocol, IoT application	Proposed lightweight ECC algorithm with reduced computation suitable for IoT devices

III. RECOMMENDATIONS AND FUTURE DIRECTION

To enhance the robustness and adaptability of graph-based encryption, future work should focus on developing quantum-resistant designs by leveraging highly complex and non-linear graph structures capable of withstanding quantum computational attacks. Integrating artificial intelligence for dynamic key management can further strengthen security by enabling real-time generation and adaptation of encryption keys through evolving graph patterns. In parallel, lightweight encryption models tailored for IoT and wearable devices must be designed with minimal graph structures to ensure low-latency, energy-efficient performance. Regular cryptanalysis testing against classical, AI-assisted, and quantum attack vectors will be essential to validate the long-term viability and security of these models. Moreover, designing scalable graph encryption frameworks will facilitate their integration into emerging domains such as smart cities, autonomous systems, and real-time analytics, ensuring future-ready, secure data communication architectures.

IV. CONCLUSION

Symmetric encryption using graph representation presents a promising advancement in the field of cryptography. By leveraging the structural complexity of graphs and matrix operations, this approach enhances data security and makes unauthorized decryption significantly more difficult. It combines mathematical robustness with flexibility, offering potential resistance against traditional and modern cryptanalytic techniques. As digital threats evolve, especially with the rise of quantum computing and AI, graph-based symmetric encryption offers a scalable and adaptable solution for securing sensitive information in diverse applications.

REFERENCES

[1] P.A.S.D. Perera, G.S. Wijesiri, “Encryption and Decryption Algorithms in Symmetric Key Cryptography Using Graph Theory”, Psychology and Education, vol. 58, issue 1, pp.12-27, Feb 2021.

[2] Wael Mahmoud AI Etaiwi, “Encryption Algorithm Using Graph Theory”, journal of Scientific Research and Reports, vol.3, issue 19, pp.12-27, Jan 2014.

[3] Uma Dixit, “Cryptography – A Graph Theory Approach”, Proceedings of University Post Graduate College, Osmania University, pp.12-27, 2021

[4] P. Amudha, A.C. Charles Sagayaraj, A.C. Shantha Sheela “An Application of Graph Theory in Cryptography”, International Journal of Pure and Applied Mathematics, vol. 119, issue 13, pp.12-27, 20182.

[5] M. Lalitha and S. Vasu, “A Study on Graph Theory in Cryptography Using Python,” Journal of Emerging Technologies and Innovative Research (JETIR), vol. 10, issue 4, pp. b97–b107, Apr. 2023.

[6] Z. Oumazouz, D. Karim, “A New Symmetric Key Cryptographic Algorithm using Paley Graphs and ASCII Values”, E3S Web of Conferences, vol. 297, issue 01046, pp.12-27, 2021.

[7] H. Anwar, Z. Shams, “Algorithm of Encryption Using Graph Theory”, Mathematical Sciences and Applications, vol. 2, issue 2, pp.12-27, Dec. 2023.

[8] Safaa Hraiz, Wael Etaiwi, “Symmetric Encryption Algorithm Using Graph Representation”, 2017 8th International Conference on Information Technology (ICIT), pp.12-27, 2017.

[9] C. Ruby Sharmila, S. Meenakshi, “A Fuzzy Graph Theory Approach to Symmetric Key Cryptography”, Journal of Propulsion Technology, vol. 45, no. 1, pp. 467–477, Jan. 2024

[10] Albert Ghazaly, “Cryptography: Graph Theory Implementation in Text Encryption and Decryption”, Makalah IF2120 Matematika Diskrit – Semester I Tahun 2023/2024, Institut Teknologi Bandung, pp. 1–14, Dec. 2023

[11] K. Bekkaoui, S. Ziti, F. Omary, “Data Security: A New Symmetric Cryptosystem based on Graph Theory”, International Journal of Advanced Computer Science and Applications, vol. 12, no. 9, pp. 742–750, Sep. 2021.

[12] Mr. D. Janardhan, Dr. G. Neelima, “Implementation of Data Encryption and Decryption Technique using Graph Theory”, Journal of Engineering Sciences, vol. 15, no. 02, pp. 337–346, Feb. 2024.

[13] D. Narayan H., S. R. Bhat, R. Bhat, S. G. Bhat, “Some Graph Based Encryption Techniques”, IAENG International Journal of Applied Mathematics, vol. 54, no. 12, pp. 2727–2734, Dec. 2024.

[14] H. Gena, B. Kakkar, “Advancing Cryptographic Security through Graph Theory: A Comprehensive Review”, International Journal of Education, Modern Management, Applied Science and Social Science (IJEMMASSS), vol. 7, no. 02(I), pp. 41–50, Apr.–Jun. 2025.

[15] V. H. R. Vidyashree, S. Lakshminarayana, “Symmetric Key Encryption Using Modular Traversal of Ananta-Graphs”, Journal of Information Systems Engineering and Management, vol. 10, no. 44s, pp. 823–831, 2025.

[16] F. Le Moue¨l, M. Godon, R. Brien, E. Beurier, N. Boulahia-Cuppens, and F. Cuppens, “Trustless Distributed Symmetric-key Encryption,” arXiv preprint arXiv:2408.16137, Aug. 2024.

[17] M. U. Bokhari and Q. M. Shallal, “A Review on Symmetric Key Encryption Techniques in Cryptography,” International Journal of Computer Applications, vol. 147, no. 10, pp. 43–49, Aug. 2016.

[18] M. Ebrahim, S. Khan, and U. B. Khalid, “Symmetric Algorithm Survey: A Comparative Analysis,” Int. J. Comput. Appl., vol. 61, no. 20, pp. 12–19, Jan. 2013.

[19] M. N. Alenezi, H. Alabdulrazzaq, and N. Q. Mohammad, ”Symmetric Encryption Algorithms: Review and Evaluation Study,” Int. J. Commun. Netw. Inf. Secur., vol. 12, no. 2, pp. 256–272, Aug. 2020.

[20] A. Khompysh, D. Dyusenbayev, and M. Maxmet, “Development and analysis of symmetric encryption algorithm,” International Journal of Electrical and Computer Engineering (IJECE), vol. 15, no. 2, pp. 1900–1911, Apr. 2025,