International Journal of Education, Modern Management, Applied Science & Social Science (IJEMMASSS)
ISSN :2581-9925, Impact Factor: 7.555, Volume 07, No. 02(I), April- June, 2025, pp. 23-28

A Novel Hybrid Graph-Based Encryption Scheme Using Star and Wheel Graphs for Enhanced Data Security

Deena Ram Chhaba[1] | Binny Kakkar[2*]

[1]Scholar, Department of Mathematics, SKD University, Hanumangarh, Rajasthan, India.
[2]Department of Mathematics, SKD University, Hanumangarh, Rajasthan, India.

*Corresponding Author: binnykakkar29@gmail.com

ABSTRACT

This paper introduces an innovative symmetric encryption algorithm that utilizes a hybrid graph structure combining properties of star and wheel graphs. The proposed method enhances data security by applying a shift cipher over the plaintext and representing the resulting cipher text values as vertex labels within a structured graph. Edge weights are carefully calculated using algebraic transformations that hide the original values. The recipient decodes the message using inverse operations on graph weights and vertex arrangement. This technique increases encryption complexity, making brute-force decryption significantly more difficult. A case study on the message "HELLO" demonstrates the effectiveness and clarity of the method.

***Keywords**: Symmetric Encryption Algorithm, Hybrid Graph, Shift Cipher, Edge Weights, Data Security*.

Introduction

With the exponential rise of digital communication, the need for more robust encryption techniques has grown. Classical cryptographic methods, although useful, often suffer from vulnerabilities in the face of modern computational power. Graph theory, a field within discrete mathematics, offers structural frameworks that can encode complex relationships. In this work, we propose a new symmetric encryption algorithm using a hybrid star-wheel graph structure that leverages the algebraic properties of shift ciphers and the visual complexity of graph labelling. In the modern digital era, the exponential growth of internet-based communication has introduced new challenges in ensuring the confidentiality, integrity, and authenticity of data. As individuals, governments, and organizations increasingly rely on electronic communication for exchanging sensitive information, the need for robust cryptographic techniques has become paramount. Conventional encryption algorithms such as RSA, AES, and DES have proven effective in many applications; however, they often rely heavily on computational complexity and large key sizes, which may not be suitable for all systems—especially those with limited processing power, like IoT devices or embedded systems.

In recent years, researchers have begun exploring alternative methods for data encryption that are rooted in mathematical structures, particularly graph theory. Graph-based encryption provides a promising avenue for designing lightweight, symmetric cryptographic schemes that can achieve both security and efficiency. Graphs, by their very nature, can model complex relationships and structures, making them ideal for hiding information in both topology and labelling. A star graph ($K_1 \odot Kn$) is a simple but effective structure where a central node is directly connected to all other nodes. When combined with a cycle among the outer nodes, forming what is known as a wheel graph (Wn), the resulting structure becomes significantly more complex and capable of representing intricate data transformations. The hybrid graph formed by merging star and wheel structures can be used not only to encode data securely but also to disguise it using edge weights and label manipulation.

This paper introduces a novel encryption technique that utilizes a hybrid star-wheel graph for secure data transfer. The algorithm operates by applying a shift cipher on plaintext characters and mapping the encrypted values to graph vertices. Two types of edges—central (star) and peripheral (cycle)—are used to assign edge weights based on arithmetic transformations. These weights act as a

mask, concealing the actual character values and introducing obfuscation through structural complexity. Unlike traditional symmetric encryption, where a key directly transforms data, the proposed approach relies on the structure and labelling of a mathematical graph to carry out encryption and decryption. This not only increases resistance to common attacks but also allows the encrypted data to be shared in the form of a graph, which can be visually inspected or embedded into graphical systems.

In the sections that follow, we detail the encryption and decryption procedures using a worked-out example, analyse the security implications, and discuss possible extensions and applications of the proposed method in secure communications.

Literature Review

Graph theory has emerged as a significant mathematical tool in the design and analysis of cryptographic algorithms. Over the past few decades, researchers have explored its utility across diverse aspects of secure communication, from key generation and encryption to hash functions and fraud detection. Stinson [13] provides a comprehensive foundation in modern cryptography, which serves as a basis for integrating advanced mathematical structures like graphs. West [15], similarly, offers in-depth insight into the principles of graph theory, laying the groundwork for its applications in computer science, including security.

Frucht and Harary [3] introduced the concept of the corona of two graphs, an early mathematical exploration that would later find relevance in structural modelling for cryptographic applications. Rivest, Shamir, and Adleman's ground-breaking RSA algorithm [9] represents one of the earliest and most influential works in public-key cryptography, setting the stage for exploring new mathematical structures like graphs for secure communication.

The potential of graph-based techniques in cryptography has been further explored by Ustimenko [14], who discusses symbolic computations and the suitability of certain classes of graphs for cryptographic systems. Charles, Lauter, and Goren [2] extended this by proposing the use of expander graphs in designing cryptographic hash functions, leveraging their high connectivity and sparse structure. Recent works have focused more explicitly on encryption methods using specific graph types. Yamuna and colleagues [17,18] have demonstrated how bipartite graphs and fundamental circuits can be applied in data transfer and encryption schemes. Selvakumar and Gupta [11] also discussed the role of circuits and cut-sets in enhancing cryptographic systems. Meanwhile, Sinha and Sethi [12] proposed a hybrid technique using signed graphs and matrices for encryption, emphasizing algebraic representations.

Various innovative models have also emerged, including the use of Venn diagrams in conjunction with graph structures by Kedia and Agrawal [5], and Mahmoud and Etaiwi's [6] unique encryption algorithm leveraging general graph theory principles. Priyadarsini [16] provides a broader survey, outlining multiple graph-theoretic approaches and their relevance to cryptographic applications.

Research has also expanded into practical implementations such as mobile security. Hu, Liang, and Dong [4] proposed a bipartite graph-based approach for fraud detection in mobile advertising, demonstrating real-world applications beyond theoretical constructs. Similarly, Selim [10] examined direct techniques for encrypting graphs, suggesting novel data security models.

Substitution-box (S-box) construction — crucial in block ciphers — has also seen graph-based innovation. Razaq et al. [7,8] developed new methods of S-box generation using coset diagrams and symmetric group theory, resulting in highly nonlinear and secure cryptographic primitives. Lastly, Arunkumar [1] and others emphasized the broad relevance of bipartite graphs, including their application in cloud computing and secure communication infrastructures.

Preliminaries

- Star Graph ($K_1 \odot K_n$): A central node connected to n outer nodes.
- Wheel Graph (Wn): A star graph with an additional cycle connecting all outer nodes.
- Shift Cipher: A classical encryption technique where each letter is shifted by a fixed key value k mod 26.
- Vertex Labelling: Assigning numerical values to graph nodes representing the encrypted message.
- Edge Weights: Values derived using subtraction of powers of 10 from vertex labels to obfuscate data.

The Proposed Encryption Scheme: Let the message M = "HELLO".

- **Step 1:** Character to Number Conversion:
  Each character is replaced with its alphabetical index H=8, E=5, L=12, L=12, O=15
- **Step 2:** Apply Shift Cipher
  Let k = 5 (length of message). Add k to each value modulo 26: Encrypted values: 13, 10, 17, 17, 20
  Assign vertex labels: After encryption using a shift cipher with k = 5, the values are:

Character	ASCII Position	Shifted Value(mod26)	Node
H	8	13	V_1
E	5	10	V_2
L	12	17	V_3
L	12	17	V_4
O	15	20	V_5

- **Step 3:** Construct Graph: Create a wheel graph W_6 with one central node (V_0) and five outer nodes.
  Star Edges (Centre to each outer node) : (V_0,V_1), (V_0,V_2), (V_0,V_3), (V_0,V_4), (V_0,V_5)
  New shifted numeric values are 13, 10, 17, 17, 20 respectively. The corresponding graph is displayed in the Figure 3.

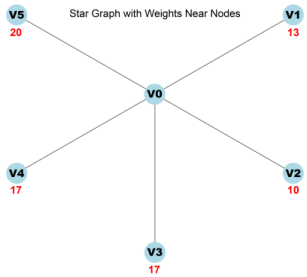


Figure 3

Next, assign weights w_i for all i ϵ {1,2,3,4} to the corresponding edges that connect the vertices:
$w_1(13) < w_2(10) < w_3(17) < w_4(17) < w_5(20)$
These are used to calculate special weights using powers of 10. For central edges (0 to each node), subtract increasing powers of 10. Example:

$w_1 = 13 - 10 = 3$

$w_2 = 10 - 100 = -90$

$w_3 = 17 - 1000 = -983$

$w_4 = 17 - 10000 = -9983$

$w_5 = 20 - 100000 = -99980$

Resulting star graph is shown in Figure 4.

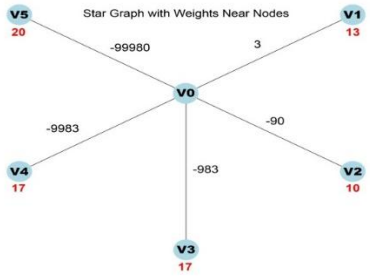


Figure 4

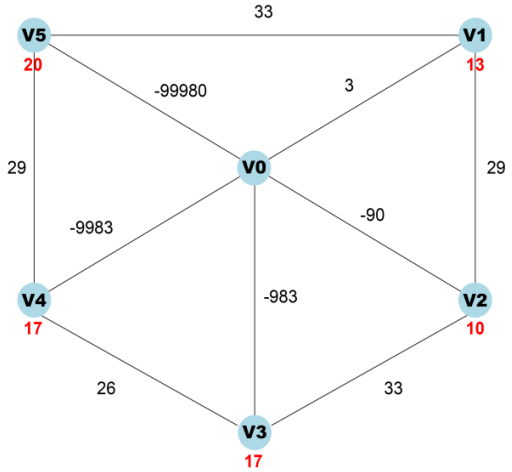
Cycle Edges (between outer nodes): (V_1, V_2), (V_2,V_3), (V_3,V_4), (V_4,V_5), (V_5,V_1)

These add extra security because they create a closed loop, making the graph less predictable. We can assign weights using the absolute difference between connected node values:

For cycle edges (node to node), use: weight = $|V_i - V_j|+26$

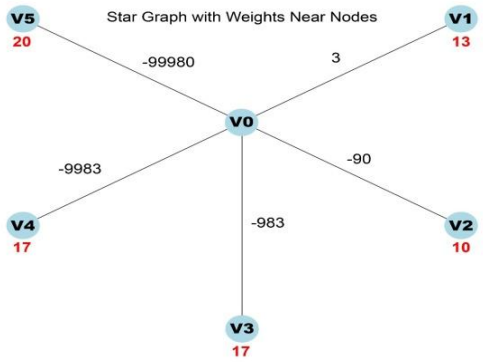
Edge	Weight
e_1	$\|V_1-V_2\|+26=\|13-10\|+26=29$
e_2	$\|V_2-V_3\|+26=\|10-17\|+26=33$
e_3	$\|V_3-V_4\|+26=\|17-17\|+26=26$
e_4	$\|V_4-V_5\|+26=\|17-20\|+26=29$
e_5	$\|V_5-V_1\|+26=\|20-13\|+26=33$

The cycle weights, although included with the encrypted graph, do not play a direct role in the decryption process. Instead, they function as a form of decoy or distraction, intended to mislead any potential attackers by adding complexity to the graph structure. The final wheel-shaped graph, incorporating both real and dummy edge weights, is illustrated in Figure 6.



In the decryption stage, the recipient begins by accessing the labelled graph received, which is illustrated in Figure 6. This graph represents the full encrypted structure that was forwarded to the second verifying authority for analysis. The initial step in this process involves deriving a minimum spanning tree from the wheel-shaped graph in Figure 6. This operation simplifies the structure, transforming it into a star topology, which forms the basis for further decryption steps. The subsequent step entails organizing the edge weights (refer to Figure 7) in increasing order based on their absolute magnitudes, meaning the values are sorted without considering their positive or negative signs.

|3| < |-90| < |-983| < |-9983| < |-99983|



Minimum Spanning Tree (Figure 7)

Add progressively increasing powers of 10 to each pair of neighbouring values.

|3+10| < |-90+100| < |-983+1000| < |-9983+10000| < |-99983+100000|

The values are obtained using this modulo operation:

13, 10, 17, 17, 20

Apply reverse shifting by estimating the number of edges in the star graph, which is 5. Subtracting 5 from each value gives us 13-5=8, 10-5=5, 17-5=12, 17-5=12, 20-5=15

This results in the values 8, 5, 12, 12 and 15. Referring to the encoding table, these correspond to the letters H, E, L, L, O revealing the hidden message.

This example illustrates how any kind of data can be concealed and remains protected until it reaches the intended recipient. The algorithm is based on star graph and wheel graph structures. The labelled graphs are transmitted to the receiver, making it an effective and secure method for data protection.

Conclusion

This paper proposed a symmetric encryption algorithm based on a hybrid star-wheel graph structure, blending the simplicity of shift ciphers with the structural complexity of graph theory. By mapping encrypted values to graph vertices and using edge weights for obfuscation, the method enhances data security while remaining lightweight and efficient. This makes it particularly suitable for systems with limited computational resources, such as IoT and embedded devices. The visual and structural encoding also opens doors for integration with graphical systems and steganography. Overall, the approach demonstrates the power of graph-based techniques in modern cryptographic applications.

References

1. B. R. Arunkumar, “Applications of Bipartite Graph in diverse fields including cloud computing,” International Journal of Modern Engineering Research, vol. 5, no. 7, p. 7, 2015.
2. 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.
3. R. Frucht and F. Harary, “On the corona of two graphs,” Aequationes Math, vol. 4, pp. 322–325, 1970.
4. J. Hu, J. Liang, and S. Dong, “A bipartite graph propagation approach for mobile advertising fraud detection,” Mobile Information Systems, vol. 2017, p. 12, Article ID 6412521, 2017.
5. P. Kedia and S. Agrawal, “Encryption using Venn-diagrams and graph,” International Journal of Advanced Computer Technology, vol. 4, no. 01, pp. 94–99, 2015.
6. W. Mahmoud and A. Etaiwi, “Encryption algorithm using graph theory,” Journal of Scientific Research and Reports, vol. 3, no. 19, pp. 2519–2527, 2014.
7. A. Razaq, H. Alolaiyan, M. Ahmad et al., “A novel method for generation of strong substitution-boxes based on coset graphs and symmetric groups,” IEEE Access, vol. 8, pp. 75473–75490, 2020.
8. A. Razaq, M. Awais Yousaf, U. Shuaib, N. Siddiqui, A. Ullah, and A. Waheed, “A novel construction of substitution box involving coset diagram and a bijective map,” Security and Communication Networks, vol. 2017, p. 16, Article ID 5101934, 2017.
9. R. L. Rivest, A. Shamir, and L. Adleman, “A method for obtaining digital signatures and public-key cryptosystems,” Communications of the ACM, vol. 21, no. 2, pp. 120–126, 1978.
10. G. A. Selim, “How to encrypt a graph,” International Journal of Parallel, Emergent and Distributed Systems, vol. 35, no. 6, pp. 668–681, 2020.
11. R. Selvakumar and N. Gupta, “Fundamental circuits and cutsets used in cryptography,” Journal of Discrete Mathematical Sciences and Cryptography, vol. 15, no. 4–5, pp. 287–301, 2012.
12. D. Sinha and A. Sethi, “Encryption using network and matrices through signed graphs,” International Journal of Computer Applications (0975–8887), vol. 138, no. 4, pp. 6–13, 2016.
13. D. R. Stinson, Cryptography: Theory and Practice, Chapman and Hall/CRC, Boca Raton, FL, USA, 4th edition, 2018.
14. V. A. Ustimenko, “On graph-based cryptography and symbolic computations,” Serdica Journal of Computing, vol. 1, pp. 131–156, 2007.

15. D. B. West, Introduction to Graph Theory, Pearson, London, UK, 2nd edition, 2001.
16. 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.
17. M. Yamuna and A. Elakkiya, “Data transfer using fundamental circuits,” International Journal of Computer and Modern Technology, vol. 2, no. 01, 2015.
18. M. Yamuna and K. Karthika, “Data transfer using bipartite graphs,” International Journal of Advance Research in Science and Engineering, vol. 4, no. 02, pp. 128–131, 2015.

❖◆❖