IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

Graph Theory in the Era of Modern Computer Science Applications: A Critical Review

Mangal Pati[1] Uma Shankar[2] Prodip Karmakar[3] Shambhu Charan Barman[4]

[1]Assistant Professor, Monalisa B.Ed. Collage & Higher Education & Research Scholar, Asian International University, Imphal West, Manipur, India
[2]Associate Professor, Asian International University, Imphal West, Manipur, India
[3]Assistant Professor, Sponsored Teachers' Training College, Purulia, West Bengal, India
[4]Assistant Professor, Department of Mathematics, Shahid Matangini Hazra Government General College for Women, Nimtouri, West Bengal, India

Abstract:Graph theory is rapidly growing in the epoch of computer science application and technology. Many applications of Graph theory, including coding theory, astronomy, circuit design, communication network addressing, database management, computer science (algorithms, data structures and computation), transportation (routing and traffic flow) and operations research (scheduling) are currently being used in the fields of biochemistry, chemistry, communication networks, mathematics and information technology. In addition to focused on computer science applications that make use of graph theoretical principles and effective algorithms, this study also critically explores the key applications of graph theory in a variety of domains. In computer science, graph theory has a wide range of applications, from modelling social networks to optimizing computer networks and algorithms.Various papers have beenstudied and critically analyzed through meta-analysis based on various useful algorithms of graph theory incomputer scienceapplications to find some valuable insights.

Keywords: Graph theory, network, application of graphs, graph algorithms, computer science applications

1.INTRODUCTION

Today, graph theory plays asignificant role in various fields such as mathematics, computer science, biology, chemistry, bio-informatics, social sciences,engineering and technology. Graph theory is widely used to model relationships in social networks, the interactions between proteins in biological systems,modeling ecosystems,analyzing internet traffic patterns, optimize transportation routes, effective urban planning and even improve web search algorithms.In fact, the ability to represent and solve problems with graphs is one of the reasons why graph theory has become so valuable in the modern world.Graph theory is a branch of mathematics that deals with the study of graphs, which are structures made up of vertices (or nodes) connected

by edges. Graph theory helps us understand how things are connected and find efficient solutions to problems like routing, scheduling, and analyzing networks. Here are some of the ways in which graph theory is used in computer science.In this context, the main aim of this study is to explore the applications of graph theory in optimizing complex networks and finds such valuable insights in this regard.

2. OBJECTIVES OF THE STUDY

The main objective of this study is to explore the key applications of graph theory in modern computer science applications.

3. UNDERSTANDING THE BASIC GRAPH THEORY

To understand the basis graph theory, here we present two separate sections. **(i)Definitions Some Important Terms on Graph Theory**and **(ii) Some Common Algorithms on Graph Theory**

3.1 DEFINITIONS SOME IMPORTANT TERMS ON GRAPH THEORY

A graph G is a collection of two sets V & Ewhere V is the collection of vertices i.e., $v_0,v_1,v_2.....v_{n-1}$ also called nodes and E isthecollection of edges e_1, e_2,.....e_n where anedge is an area which connects two nodes. This can be represented as –

$G=(V,E)$

$V(G)=(v_0,v_1,..... v_{n-1})$or set of vertices

$E(G)=(e_1,e_2,.....e_n)$or set of edge

Path: A path is a simple graph whose vertices can be ordered so that two vertices are adjacent if and only if they are consecutive in the list.

Undirected Graph: A graph in which each edge symbolizes an unordered, transitive relationship between two nodes. Such edges are rendered as plain lines or arcs.

Directed Graph: A graph in which each edge symbolizes an ordered, non-transitive relationship between two nodes. Such edges are rendered with an arrowhead at one end of a line or arc.

Loop: A loop is a special type of edge that connects a vertex to itself. Loops are not used much in street network graphs.

Degree: The number of edges which connect a node.

In Degree: Number of edges pointing to a node.

Out Degree: Number of edges going out of a node.

Un-weighted edge: A graph in which all the relationships symbolized by edges are considered equivalent. Such edges are rendered as plain lines or arcs.

Weighted edge: Weighted edges symbolize relationships between nodes which are considered to have some value, for instance, distance or lag time. Such edges are usually annotated by a number or letter placed beside the edge. If edges have weights, we can put the weights in the lists. Weight**:** W: E → R

Tree: An undirected connected graph T is called tree if there are no cycles in it. There is exactly one simple path between any vertices u and v.

Simple Path: Simple path is a path in which all the vertices are distinct.

Spanning Tree: A sub graph T of a connected graph G, which contains all the vertices of G and T is called a spanning tree of graph G. It is called spanning tree because it spans over all vertices of graph G.

Hamiltonian Graph: A graph that has a Hamiltonian cycle—a closed path that visits each vertex exactly once, except the starting and finishing vertices—is considered to be HamiltonianGraph.

Bipartite Graph: If the vertices in V(G) can be separated intotwo non-overlapping subsets in such a way that every edge connectsa vertex from one subset to a vertex in the other, the graph G isclassified as a bipartite graph.

Complete-bipartite Graph: A bipartite graph becomes completebipartite when each vertex in one subset is linked to every vertex inthe other subset.

3.2SOME COMMON ALGORITHMS IN GRAPH THEORY

Graph theory is used for the study of algorithms in Computer Science application. Some of these essential algorithms in Graph Theory are presented herewith.

3.2.1Shortest Path Algorithms

Shortest path algorithms help find the best routes in transportation networks, such as road systems, public transport, or logistics, to reduce traffic and improve flow.

(A) Dijkstras Algorithm: Finds the shortest route between two points in a network. It helps planners identify the fastest or most efficient routes in a city's road network, taking into account factors like distance and traffic conditions. This algorithm finds the shortest paths from a starting node to all other nodes in a graph, using a formula that iteratively updates distances as follows:

Dist(v) = Min(Dist(v), Dist(u) + Weight(u,v)).

where, Dist(v): Represents the current shortest distance from the source node to node v.

Dist(u): Represents the current shortest distance from the source node to node u.

Weight(u, v): Represents the weight (or cost) of the edge connecting nodes u and v.

Min(Dist(v), Dist(u) + Weight(u, v)): It compares the current shortest distance to node v (Dist(v)) with a potential shorter path through node u (Dist(u) + Weight(u, v)) and keeps the smaller value.

(B) A* SearchAlgorithm:An improved version of **Dijkstra's** that uses heuristics to guide the search, making it faster and more efficient for larger urban planning problems.A*Search algorithm is incorporated to enhance efficiency by introducing aheuristic function:

$f(n)$ = r(n) + e(n)

Where,r(n) is the cost to reach node n from the start node, and e(n) is the estimated cost from node $n$to the goal.

(C) Floyd-Warshall's Algorithm: It finds all-pair shortest path in weighted graph.It uses Adjacency matrix. This algorithm compares all possible paths through the graph between each pair of vertices. Negative weights are allowed but Negative cycle is not allowed. The time complexity of this algorithm is O(V^3) and it is slower. This algorithm enables optimized short pathcalculations between every pair of nodes in networked environments by establishing all-distances.

(D) Bellman-Ford Algorithm: It finds single-source shortest path in weighted graph and detects negative cycles. Its basic structure is very similar to **Dijkstra's algorithm**, but instead of greedily selecting the minimum-weight node not yet processed to relax, it simply relaxes all the edges, and does this $|V| - 1$ times, where |V| is the number of vertices in the graph. The repetitions allow minimum distances to accurately propagatethroughout the graph, since, in the absence of negative cycles; the shortest path can only visit each node at most once. Bellman-Ford cannot find the shortest path that does not repeat any vertex in such a graph. The runtime: Bellman–Ford runs in O(V*E) time.

3.2.2 Network Flow Algorithms

Network flow algorithms help determine the best way to distribute resources like electricity, water, and traffic across urban systems.

(A) Ford-Fulkerson Algorithm: Helps find the maximum flow in a network, such as optimizing the distribution of electricity or water from suppliers to consumers.In network flow optimization, this algorithm is used to determine the maximum flow $f$in a given network

$f_{max} = \sum_{u \in V} f(s, u)$

where,$f(s, u)$ represents the flow from the source node s to other nodes in the network. The residual capacity of an edge (u, v) is updated as follows:

$c'_{uv} = c_{uv} - f_{uv}$

This ensures that flow conservation and capacity constraints are maintained. This algorithm Maximizing flow management needs during logistics operations requires the application of the Ford-Fulkerson method which solves maximum flow problems [1].

(B) Edmonds-Karp Algorithm: An efficient version of Ford-Fulkerson using Breadth-First Search (BFS) to improve maximum flow problems in utility networks.

3.2.3 Louvain algorithm

The most popular community detection algorithm in the space, the Louvain algorithm is based on the idea of graph (component) density i.e. something related to edges/connections frequency within a component compared to other components in the same graph. It can handle very large networks by enabling multi-level refinement and aggregation [2].A node is added to a community if it improves its density else not. We call this term closely related to density as Modularity.

It has a very complex formula

$$Q = 1/(2m) * \Sigma_i([A_i - k_i k_j / (2m)] * \delta(c_i, c_j))$$

The above formula signifies as follows -

Q= Modularity

m = Weights of all edges in the graph (if unweighted graph, the count of edges in the entire graph)

The summation has 3 terms

A_i = weight of the edge between node ‘i’ & node ‘j’ if weighted graph else 1 if edge between node i & j exists else 0 (in case of unweighted graph)

K_i is the degree (total connections) of some node i.

$\delta(c_i, c_j) = 1$ if both node i & node j are in same community else 0.

So, the above summation term is 0 if i & j are from different communities.

Modularity (Q)measures the density of edges within communities comparedto random expectations. It is often maximized via heuristicmethods such as the Girvan–Newman algorithm or spectraloptimization [3].

3.2.4Kruskal’s Algorithm

Kruskal’s algorithm is also a greedy algorithm but takes a different approach. It begins with all the vertices and no edges, and it adds edges one by one in increasing order of weight, ensuring no cycles are formed until the MST is complete.

3.2.5Prim's Algorithm

Prim's algorithm is a greedy algorithm that builds the Minimum Spanning Tree(MST) incrementally. It starts with a single vertex and grows the MST one edge at a time, always choosing the smallest edge that connects a vertex in the MST to a vertex outside the MST.

4. METHODOLOGY OF THE STUDY

This study is based on secondary sources of data such as articles, books, journals, research papers, websites and other sources. Also Meta analysis was done based on graph theoretical concepts and efficient algorithms.

5. REVIEW OF ALLIED LITERATURE

In order to identify the different Key Applications of Graph Theory principles and effective algorithms in the field of computer science, the researcher has made an attempt to present the review ofallied literature.

5.1 Network Analysis and Optimization

Networks including computer networks, social networks, and transportation networks are frequently modelled using graph theory. Researchers can explore these networks characteristics and enhance their functionality by expressing them as graphs. Graph theory, for instance, can be used to optimize the flow of traffic in transportation networks or to find inefficiencies and shortcomings in computer networks. The primary utilitarian component of graph theory, according to the researchers, is optimization, which looks for the optimum answer among limited possibilities. Several optimization problems, including network flow optimization challenges, minimum spanning tree problems, and shortest path problems, are represented graphically. [4].

5.2 Data Structures and Algorithms

Graphs can be used to model a variety of computer science data structures and algorithms. Since overwhelming network data structures exist simultaneously with social network applications in big data systems, managing these structures effectively required the use of strong computational algorithms [5]. The time complexity of data structures for the **Dijkstra algorithm** is O(|E|*log|V|), and for the Bellman-Ford algorithm, it is O(|V||E|). These findings have been reported in a different study on data structures and algorithms.

5.3 Computer Graphics and Visualization

The creation of 3D models and visual representation of complex data sets are only two examples of the many applications of graph theory in computer graphics and visualisation. A winged edge representation [6] is far more prevalent in 3D modelling applications because it includes pointers to adjacent geometric primitives, which simplifies many kinds of adjacency queries. Graph algorithms can be used to modify and visualize data and objects by representing them as graphs. The **Catmull-Clark** subdivision algorithm [7], which recursively

creates smooth subdivision surfaces from a low-resolution mesh, is one of the most widely used algorithms in computer graphics.

On arbitrary networks, another technique developed by **Doo and Sabin** [8] performs comparably. Later, **Loop** [9] developed a more simplified algorithm that was meant to be effective to evaluate. Indeed, the approach was demonstrated to be effective to evaluate on programmable GPU tessellation units after programmable GPUs became widely available [10]. Graph algorithms, for instance, can be used to find patterns in big data sets or to determine the shortest path between two locations in a 3D model.

5.4 Network Design

Networks are modelled using graph theory; in which nodes stand for indicating devices (such as computers or routers) and edges for connections. Network routing is optimized and the shortest paths are found using algorithms like **Bellman-Ford** and **Dijkstra's**. Numerous optimization problems pertaining to routing protocols and network monitoring were given by the researchers, who have conducted substantial study in relation to complex communication networks. They demonstrated that many of these optimization problems are either NP-Complete or NP-Hard. Lastly, they described a few popular tools for creating graph-theoretic network topologies.[11].

5.5Social Network Analysis (SNA)

One of the most popular uses of graph theory is in social networks, which enable the modelling and examination of complicated user relationships. Social network analysis (SNA) is made possible by graph theory using metrics that assess connection, user influence, and network structure [12]. Link prediction algorithms, for instance, use patterns in the network to predict possible connections, recommending friends or followers [13]. These models facilitate applications in impact analysis, anomaly detection, and community discovery by capturing crucial social connections.

In a social network theory, entities or groups are called "nodes," connected by one or more particular dependencies like friendship, resemblance, gender, etc., as explained by the researcher**Scott** [14]. In the context of a graph, the links that connect these nodes are referred to as connections or edges. Social Network Analysis (SNA), which is popular in anthropology, economics, geography, and sociolinguistics, is closely related to social networks. SNA is essential to contemporary sociology.

5.6AI and Machine Learning

For AI-based graph models to interact well with dynamic and changing network systems, scientists should focus on designing more swiftly computational processes [15]. Applications for Graph Neural Networks (GNNs) include molecular chemistry, recommendation systems, and social network analysis. The potential of machine learning and artificial intelligence technologies to improve graph-based optimization techniques led to their further development. The detection of subtle danger patterns that conventional rule-based systems can overlook is made possible by recent developments in Graph Neural Networks (GNNs), which apply deep learning paradigms to irregular graph structures [16].

Applications of Graph Neural Networks (GNNs) are widely used in fraud prevention systems, molecular structure analysis, and predictive modelling for the recommendation industry. Graph theory has become increasingly important in today's environment, as evidenced by contemporary research and industrial activities [17–20].

5.7Graph Database Systems

Popular graph database systems include **Neo4j, Amazon Neptune,** and **ArangoDB**are made to store and query data in graph form, they can handle complicated queries regarding connections and relationships that are more challenging to describe in relational databases. As specialized systems created to manage data with complicated, linked relationships, graph databases have gained popularity. These databases are useful in domains like fraud detection, recommendation systems, and social networking because they facilitate advanced analytics, pathfinding, and pattern matching [21].

Neo4j leverages a labelled property graph model to organize data, allowing characteristics to be stored as key-value pairs for both nodes and edges. **Neo4j** can effectively execute graph traversals thanks to this architecture, which is an essential capability for applications such as recommendation engines, social network analysis, and fraud detection [22].

Amazon Neptune works especially well for applications that need to handle massive amounts of dynamic data quickly and efficiently. Its applications include knowledge graphs, fraud detection, and recommendation engines, where quick data retrieval and intricate relationship analysis are crucial [21].

A distinctive graph database, **ArangoDB** supports document, key-value, and graph data models in a single database and has multi-model capabilities. For applications with intricate and varied data requirements, this flexibility helps users manage a variety of data kinds and formats. ArangoDB makes use of the Arango Query Language (AQL), a uniform query language that streamlines intricate queries involving many data kinds and is intended to function across various data structures [23].

5.8 Compiler Design

In compiler design, a Control-Flow Graph (CFG) simulates the control flow between a program's fundamental components. A directed graph, G = (N, E), is called a CFG. A fundamental block is represented by each node n ∈ N. Every edge e = (n_i, n_j) ∈ E represents a potential control transfer from block n_i to block n_j. CFGs are used in compilers to depict the program's control flow [28]. Using their Control Flow Graphs, **Eskandari and Hashemi (2011)** [24] offered a method for identifying metamorphic malware. They were employed for software execution, route comprehension and optimization.

5.9Graph-Based Cryptography

Graph-theoretic ideas are used by some cryptographic algorithms to develop secure communication techniques; these algorithms frequently take use of the intricacy of graph problems to improve security. Interest in using graphs to suggest novel approaches in a variety of cryptographic fields has increased recently **(Selim, 2020) [25].**In another study the researchers presented a unique coding and decryption algorithm using linked graphs **(Ali et al., 2024a) [26]**. A graph-based encryption technique was suggested by **Yamuna and Elakkiya**

(2015) [27], in which appropriate edge weights are used to choose essential circuits. They used a numeric table for alphabet representation and bipartite graphs to provide a novel data transport technique.

6. DISCUSSION

Inthe field of**Network Analysis and Optimization,** Several optimization problems in transportation networks traffic flow optimizationare extensively used to explore and analyze the network structure graphically by using the concept of Minimum Spanning Tree (MST)algorithms, Network Flow algorithms, Breadth-First Search (BFS) algorithm and Shortest Path Algorithms etc.

Incase of**Data Structures and Algorithms,**several algorithmssuch asDijkstra's algorithm, Kruskal's algorithm, Bellman-Ford algorithm, Floyd-Warshall algorithm areextensively usedto form and managing a model of variety of datastructuresin big data system.

Incase of**Computer Graphics and Visualization,** Several Graph algorithms such as **Catmull-Clark** subdivision algorithm, Breadth-First Search (BFS) and Depth-First Search (DFS) algorithms, Prim's algorithm, A* Search algorithm, Floyd-Warshall algorithm and Graph coloring are broadly used to explore, modify, manipulate, animate, optimize and visualize data for 3D modeling applications and visual representation of complex data set.

In case of**Network Design,**basic graph theory concepts such as nodes, edges, paths and useful algorithms such as Bellman-Ford, Dijkstra's, Network flow, Louvain algorithm,Prim's and Kruskaal's algorithm are broadly used for complex network modelling, optimizing, traversal, creating graph-theoretic network topologies, routing protocols and network monitoring.

In case of**Social Network Analysis (SNA),**basic graph theory concepts such as nodes, edges, centrality and useful algorithms such as community detection algorithms and Dijkstra's algorithmare broadly used to understand, explores and measures of centrality, relationships and influential connections of social network structures.

Incase of**AI and Machine Learning,**mostuseful algorithms such as Graph Neural Networks (GNNs), A* Search algorithm and Dijkstra's algorithm are extensively used to leveraging, clustering in complex data set, the structure and relationships in various domains including recommendation system, knowledge graph representation, robotics, image and video processing etc.

Incase of**Graph Database Systems,**mostuseful algorithms such as A* Search algorithm, Dijkstra's algorithm, Prim's and Kruskal's algorithm, are heavily used to explore a property graph databases, minimum costing of connected components and linked complex relationships for efficient data traversal.

In case of**Compiler Design,**basic graph theory concept like Control-Flow Graph (CFG) and mostuseful algorithms such as Breadth-First Search (BFS) and Depth-First Search (DFS) algorithms,Dijkstra's algorithm and Floyd-Warshall's Algorithm are used to a great deal in compiler design for tasks such as code optimization control flow analysis and register allocation.

Incase of**Graph-Based Cryptography,**basic graph theory concepts such as spanning trees, bipartite graphs and Hamiltonian graphs and also algorithms likeFloyd-Warshall's algorithm,Dijkstra'sand Bellman-Ford algorithm are extensively used for unique coding with encryption and decryption to secure communication.

7. CONCLUSION

Based on the findings of numerous useful applications of graph theory in the era of modern computer science applications,it is reflected that graph theory offers so many essential tools and frameworks for modelling linkages and interactions in a variety of fields within the computer science application. Its uses range from theoretical underpinnings to real-world applications, impacting fields like machine learning, networking, data analysis, data management systems, and more. Computer scientists can effectively use graph-theoretic topologies in their work by being aware of these applications. Additionally, it made easy to solve real-world problems for data analysts, scientists, engineers, and researchers by using various useful algorithms and basic concepts of graphtheory.

REFERENCES

[1] Akbarpour,M. (2023). "Network Theory in Engineering Project Management: A Review of Graph-Based Models for Optimization," Management Strategies and Engineering Sciences, 5(2):32–41, 2023.

[2] Blondel, V. D., Guillaume, J. L., Lambiotte, R., and Lefebvre, E. (2008). Fast unfolding of communitiesin large networks. Journal of Statistical Mechanics: Theory and Experiment, 10, P10008.

[3] Newman, M. E. J. (2010). Networks: An Introduction. Oxford University Press. Available:https://doi.org/10.1093/acprof:oso/9780199206650.001.0001

[4] Chen, W. K. (1980) "Some Applications of Graph Theory to Network Analysis and Synthesis," IEEE Transactions on Circuitsand Systems, 27(6):523–533. Available: https://doi.org/10.1109/TCS.1980.1084710

[5] Meghanathan, N. Review of Graph Theory Algorithms, Department of Computer Science, Jackson State University, Jackson, MS 39217.

[12] Gunjal, B. (2003). Database System: Concepts and Design.

[6] Bruce G. B.(1975).A polyhedron representation for computer vision. In Proceedings of the May 19-22, 1975, nationalcomputer conference and exposition on – AFIPS '75, page589, New York, New York, USA, ACM Press.

[7] Catmull, E. and Clark, J. (1978). Recursively generated B-splinesurfaces on arbitrary topological meshes.Computer-Aided Design, 10(6):350–355.

[8] Doo, D. and Sabin, M. (1998). Behaviour of recursive division surfaces near extraordinary points. InSeminal graphics, 177–181. ACM, New York, NY, USA.

[9] Loop, C. (1987). Smooth subdivision surfaces based on triangles. Master's thesis, University of Utah.

[10] Loop, C. and Schaefer, S. (2008). Approximating Catmull-Clark subdivision surfaces with bicubic patches. ACM Tran-sactions on Graphics, 27(1):1–11.

[13] Ahmat, K.(2009). Graph Theory and Optimization Problems for Very Large Networks. City University of New York/Information Technology New York, USA

[12] **Majeed, A. and Rauf, I. (2020).** Graph theory: A comprehensive survey about graph theory applications in computer science and social networks. Inventions, 5(1):10.

[13] Daud,N. N., Hamid, S. H. A., Saadoon,M., Sahran, F., and Anuar, N. B.(2020) Applications of link prediction in social networks: A review. Journal of Network and Computer Applications, 166:102716.

[14] Scott, J. (2023). Social Network Analysis. Journal of SAGE Publications.

[15] Boykov, Y. Veksler,O. and Zabih,R. (2001). "Fast Approximate Energy Minimization via Graph Cuts," IEEE Transactions onPattern Analysis and Machine Intelligence, 23(11):1222–1239.

[16] Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., and Philip, S. Y. (2021). A comprehensive survey ongraph neural networks. IEEE Transactions onNeural Networks and Learning Systems, 32(1), 4–24.

[17] Kolmogorov, V. and Zabin, R. (2004). "What Energy Functions Can Be Minimized via Graph Cuts?," IEEE Transactions onPattern Analysis and Machine Intelligence, 26(2):147–159,

[18] Owerko, D. Gama, F. and Ribeiro,A. (2019). "Optimal Power Flow Using Graph Neural Networks," arXiv preprintarXiv:1910.09658, Available: https://arxiv.org/abs/1910.09658

[19] Faruqi, M. A. A. and Baig, M. S. (1984). "Graph Theory Applied to Optimal Connectivity in Computer Networks," ACMSIGMETRICS Performance Evaluation Review, 12(1):268–272.

[20] Ishizaki, T. Chakrabortty, A.and Imura, J. I. (2018)."Graph-Theoretic Analysis of Power Systems," Proceedings of the IEEE, vol. 106, no. 5, pp. 818–839. Available: https://doi.org/10.1109/JPROC.2018.2812218

[21] Maciej Besta, Robert Gerstenberger, Emanuel Peter, Marc Fischer, Michał Podstawski, Claude Barthels, Gustavo Alonso, and Torsten Hoefler. (2023) Demystifying graph databases: Analysis and taxonomy of data organization, system designs, and graph queries.ACM Computing Surveys, 56(2):1–40, 2023.

[22] Kornelije Rabuzin, Maja Cerjan, and Snježana Križanić.(2022). Supporting data types in neo4j.In *European Conference on Advances in Databases and Information Systems*, 459–466, Springer.

[23] Rajat Belgundi, Yash Kulkarni, and Balaso Jagdale. (2022). Analysis of native multi-model database using arangodb. In Proceedings of Third International Conference on Sustainable Expert Systems: ICSES 2022, 923–935, Springer.

[24] M. Eskandari, S. Hashemi (2011). Metamorphic malware detection using control flow graph mining, International Journal of Computer Science and Network Security, 11 (12).

[25] Selim, G. A. (2020). How to encrypt a graph. Int. J. Parallel Emergent Distributed Syst.35, 668–681. doi: 10.1080/17445760.2018.1550771

[26] Ali, N., Kousar, Z., Safdar, M., Safdar, J., and Tolasa, F. T. (2024a). A mathematical analysis of concealed non-Kekulean benzenoids and subdivided networks in associated line graphs. Acadlore Trans. Appl Math. Stat 2, 72–80. doi: 10.56578/atams020202

[27] Yamuna, M., and Elakkiya, A. (2015). Data transfer using fundamental circuits. Int. J.Comp. Modern Technol. 2.

[28] A.V. Aho, R. Sethi and J. D. Ullman,(2007)*Compilers Principles, Techniques, & Tools*, (2nd Ed.), Pearson Education.