GRAPH THEORY AND ITS APPLICATIONS TO SOLVE REAL WORLD PROBLEMS

Pragya Dubey

Assistant Professor

Department Of Engineering Mathematics

Gyan Ganga College Of TECHNOLOGY, JABALPUR M.P. India

$∎ $**Abstract:**

The field of graph theory is expanding due to its applications in science, technology, and mathematics. It is actively employed in the communication, chemistry, and biochemistry disciplines. Computer science (algorithms and computation), networks and coding theory, and operations Research (scheduling) as well as several applications such as circuit design, communication network addressing, astronomy, coding theory, x-ray crystallography, radar, and database management. This paper primarily focuses on applications of graph theory in computer science and chemistry, but it also provides a broad review of graph theory's applicability in heterogeneous domains. An overview of the numerous publications on graph theory that have been examined in relation to scheduling ideas and computer science applications is given here.

$∎ $**Keywords:**

 Graphs, edges, vertex, path, degree , types of Graphs.

$∎ $**Introduction:**

Graph theory is a branch of mathematics that deals with a network of points that are connected via lines. The study of the relationships between a set of nodes that are connected by edges is the essence of graph theory. An excellent abstraction for many real-world issues is nodes and edges. Examples include organization, networking, and optimization issues with regard to political campaigns, social media reach, city layouts, and navigation. First, let's familiarize ourselves with some key ideas in graph theory.

$∎ $ **Vertex definition:**

An object that is connected to other objects in a graph is called a vertex, also known as a node. The linkages, sometimes known as edges, that connect the graph.

. 

$∎ $**Graph:**

A graph is a type of data structure made up of a finite number of vertices, also known as nodes, and the directed and undirected edges that connect them. A edge is shown in the (U, V) format. In this case, the edge joins U and V. A directed graph, also known as a digraph, is created when there are directed edges between the source and destination nodes, U and V, respectively. It is referred to as an undirected graph if they aren't directed.A simple graph is any graph with no loops and any two nodes connected by a maximum of one edge. A basic graph can have connections or disconnections. A multigraph is formed when two nodes are connected by more than one edge. We can presume a graph is a simple graph unless otherwise noted. The graph is referred to as complete if every node has an edge connecting it to every other node. Therefore, we can determine the general structure of a graph if we know how many nodes are present in it.[1]

 

$∎ $**Degree:**

The degree of a node in a directed graph is the total number of edges that enter or exit the node. The terms "in-degree" and "out-degree" refer to the quantity of entering and exiting edges, respectively. The degree of a node in an undirected graph is again the number of edges that are connected to that node However, in an undirected graph, the edges are bidirectional, and all edges are counted towards the degree Based on this definition, we may therefore draw the following conclusions:

▪ In a connected simple graph, all the nodes will have a degree of at least 1In a complete graph having N nodes, the degree of each node will be N-1.

▪To the degree of a node that is a part of a self-loop, the loop contributes +2 to the degree of the node.

$∎ $**Path:**

 A path is a collection of edges that joins a series of nodes. A path can consist of one edge joining two nodes or several edges joining several nodes together.

 • A graph is referred to as linked if there is an unbroken path connecting every point to every other point in the network.
• A path is referred to as a circuit if it begins and finishes at the same node and no edge is crossed more than once.

Thus, a circuit is a closed path. once and every node is traversed as well is referred to as an Eulerian circuit. In this instance, the graph is referred to as an Eulerian graph, indicating that it is connected and that every node has only even degrees. A path that visits every node once with no repeat visits A circuit where each edge is traversed exactly is called a Hamiltonian path. A Hamiltonian circuit is a circuit that starts and finishes at the same node and visits each node once without making repeat visits.

$∎ $**The History of Graph Theory:**

 Although Euler was able to demonstrate his result, he did not adequately create the methods and tests that would have allowed him to support his claim with sound mathematical reasoning. Thus, graph theory made little to no progress for decades. Now, with the tools and techniques we have available in modern times, we’re really starting to explore the full potential Graph theory’s origins have been traced to be in the 18th century, around the year 1735. It all began as arithmetic puzzles for fun, but as they acquired popularity and significance, they developed into a significant field of mathematical study. One such historical riddle, the Königsberg bridge problem, was solved by the Swiss mathematician, physicist, astronomer, geographer, logician, and engineer Leonhard Euler. The first theorem of graph theory was also demonstrated by this Euler's proof. For many, graph theory began with this proof.

 $∎ $**Directed Graph:**

 A graph with directed edges, often called a digraph, is one whose edges have orientations. In one restricted but very common sense of the term a directed graph is a pair

 G = (V, E) comprising:
V, a collection of vertices (also known as points or nodes);

 

E, a set of edges (also called directed edges, directed links, directed lines, arrows, or arcs), which are ordered pairs of distinct vertices.
This kind of object is more accurately referred to as a directed simple graph to prevent confusion.

$∎ $**Mixed Graph:**

A graph with both directed and undirected edges is called a mixed graph.It is an ordered triple *G* = (*V*, *E*, *A*) for a *mixed simple graph* and For a mixed multigraph with V, E (the undirected edges), A (the directed edges), $ϕ$E, and ϕA defined as before, G = (V, E, A, $ϕ$E, $ϕ$A).Directed and undirected graphs are special cases.

$∎ $**Weighted Graph:**

A graph in which each edge is given a number, known as the weight, is referred to as a weighted graph or network. These weights could be used to indicate lengths, capacities, or expenses, depending on the issue at hand. These graphs appear in a variety of situations, including reduced path issues like the traveling salesman problem [2].

 

 (10 vertices and twelve edges in a weighted graph)

$∎ $**Types of Graphs:**

**1) Oriented Graph:**

An oriented graph can be defined as a directed graph with a maximum of one edge that can be either (x, y) or (y, x). It is a directed graph, in other words, that can be created by orienting an undirected (basic) graph. "Oriented graph" and "directed graph" are used interchangeably by certain writers. Certain writers define a "oriented graph" as any orientation of an undirected graph or multigraph.

**2) Regular Graph:**

 A regular graph is one in which every vertex has the same degree, or the same number of neighbours. A k regular graph, also known as a regular graph of degree k, is a regular graph having vertices of degree k

**3) Complete Graph:**

 A whole graph with ten edges and five vertices. Every vertex is connected to every other vertex by an edge

. ▪ A graph is said to be complete if every pair of vertices is connected by an edge.. Every potential edge is present in a complete graph.

 

**4) Finite Graph:**

A graph is considered finite if both the vertex and edge sets are made up of finite sets. If not, it's referred to as an infinite graph. In graph theory, it is most frequently assumed that the graphs under discussion are finite. It is usually stated explicitly if the graphs are infinite.

**5) Connected Graph:**

An unordered pair of vertices {x, y} in an undirected graph is said to be connected if a path connects x and y. If not, the unordered pair is referred to as disconnected.

**6) Bipartite Graph:**

A simple graph known as a bipartite graph is one in which the vertex set can be divided into two sets, W and X, such that no two vertices in W or X share an edge. Alternatively, it is a graph with a chromatic numbers of 2.

 **7) Path Graph:**

 A *path graph* or *linear graph* of order *n* ≥ 2 is a graph in which the vertices can be listed in an order *v*1, *v*2, …, *vn* such that the edges are the {*vi*, *vi*+1} where *i* = 1, 2, …, *n* − 1. Path graphs can be characterized .A *path graph* or *linear graph* of order *n* ≥ 2 is a graph in which the vertices can be listed in an order *v*1, *v*2, …, *vn* such that the edges are the {*vi*, *vi*+1} where *i* = 1, 2, …, *n* − 1. Path graphs can be characterized as connected graphs in which the degree of all but two vertices is 2 and the degree of the two remaining vertices is 1. A path graph is a path in another graph if it appears as a subgraph of that other graph[3].

**8) Planar Graph:**

A graph that has vertices and edges that can be drawn in a plane without any edges intersecting is said to be planar.

 **9) Cycle Graph:**

 When the vertices of a graph are enumerated in the order v1, v2,..., vn, the resulting graph is called a cycle graph or circular graph of order n ≥ 3.the edges are the {*vi*, *vi*+1} where *i* = 1, 2, …, *n* − 1, plus the edge {*vn*, *v*1}. Cycle graphs are defined as connected graphs where every vertex has a degree of two. A cycle or circuit is present in another graph if it appears as a subgraph of that other graph.

**10) Tree:**

 A tree is an undirected graph, or more accurately, a linked acyclic undirected graph, where any two vertices are connected by precisely one path. A forest can be defined as an acyclic undirected graph, a disjoint union of trees, or an undirected graph with any two vertices connected by a maximum of one path.

**11) Poly tree:**

A poly tree, also known as a directed tree, oriented tree, or singly linked network, is an underlying undirected graph that is a tree in a directed acyclic graph (DAG).A *poly forest* (or *directed forest* or *oriented forest*) is a directed acyclic graph whose underlying undirected graph is a forest.

* **Application of graph theory to solve real problems:**

**● Studying Different Graphs in Decision Mathematics**

 Allocating resources, resolving optimization issues, and reaching judgments based on mathematical models and analysis are all included in decision mathematics. Studying different types of graphs is crucial in providing powerful tools and techniques for addressing real-world problems. Examples of areas in decision mathematics where different types of graphs play a significant role include[4]:

 **• Network Design:** Graphs can be used to depict logistical, transportation, or communication networks, which can be used to spot any bottlenecks and create effective routing plans.

**• Project Management:** Task dependencies, scheduling, and resource allocation are modeled during project management with the use of graphs such as Activity-on-Node (AON) and Activity-on-Arc (AOA).

**Job Assignment**: By analyzing bipartite graphs, particularly maximum-matchings, it is possible to allocate employees to tasks in a way that maximizes output while minimizing expenses.

• **Graph Colouring**: This effective method aids in work scheduling and resource allocation while preventing disputes. Frequency allocation in wireless communication networks is one example of an application to reduce signal interference.

. • **Social Network Analysis**: Key persons or communities within the network are identified and their properties are assessed using graphs to simulate social network structures.[5]

Understanding and analyzing diverse graph types in the context of decision mathematics enables the development of creative solutions that can resolve challenging issues in a variety of academic fields.

### ■ Overcoming Challenges in Graph Theory Applications

### The actual use of graph theory presents a number of unique obstacles. To guarantee successful implementation and obtain the intended results, it is critical to acknowledge and address these obstacles. Here are some typical obstacles and strategies for overcoming them:

**1. Data Representation**: Selecting the appropriate graph model to represent the relationships and entities found in the actual world can be difficult. When applying different graph structures (e.g., directed or undirected, weighted or unweighted), be sure to grasp the subtleties of the issue domain.

1. **Scalability:** Large datasets and intricate interactions are common in real-world applications, which provide computing difficulties.[6] These scalability problems can be resolved by using effective algorithms, parallel processing, or approximation techniques.
2. **Noise and Uncertainty :**Data may include mistakes or insufficient information in real-world applications. Create reliable models and algorithms that can manage uncertainties and flaws in the data while still yielding insightful outcomes.
3. **Interpretation and Evaluation:** It is necessary to convert the solutions obtained from graph theory models into workable action plans. Make sure the results are comprehensible and applicable to the actual issue. Conduct thorough assessments to verify these solutions based on criteria particular to the domain.
4. **Adaptability:** Graph models and solutions must be modified in response to changes in real-world scenarios. Create flexible models and algorithms that can adjust to changing conditions or, if necessary, take into account more data.

One way to guarantee a smooth transition from theoretical to practical problem-solving is to identify the obstacles that come with applying graph theory in practical settings and devise suitable solutions to surmount them.

**■ Examples of Graph Theory Applications**

Graph theory has attracted a lot of interest because it can be applied to a wide range of situations and provides fresh perspectives on challenging issues.[7] Graph theory offers a mathematical framework to depict connections and interconnections in a wide range of real-life contexts, from communications networks to social networks and urban planning.

### ■ Using Graph Theory in Modern-Day Problem Solving

Graph Theory serves as a powerful tool for modern-day problem solving in multiple disciplines, such as:

* **Computer Science :**Graph theory is a fundamental idea in computer science that is used to network architecture, algorithms, and data structure management. For instance, eigenvector centrality in networks serves as the foundation for Google's well-known Page Rank algorithm.
* **Operations Research:** Graph models are widely employed in the optimization of network flow issues, supply chains, and route planning. One well-known example is the Travelling Salesman Problem (TSP), in which the objective is to determine the shortest path that stops at a predetermined number of destinations before returning to the starting point.
* **Social Network Analysis:** Understanding the dynamics, structure, and influence of social networks like Face book and Twitter is made easier with the aid of graph theory. Vertex (node) rankings are determined by centrality metrics such as between’s and proximity, which take into account the nodes' relative importance in the network.
* **Urban Planning and Transportation:** Road networks, public transportation, and traffic flow can be represented as graphs, assisting in the design and analysis of efficient transportation systems. Techniques like the shortest path algorithm help to improve route planning and optimise transit times [8].
* **Biology and Ecology**: Graph theory has applications within biology and ecology, such as representing gene or protein interactions and examining ecosystem structures or food webs. This information can reveal important insights into the stability and complexity of such systems.

Graph Theory's versatility extends far beyond these examples and continues to provide new opportunities for real-life problem-solving across a range of disciplines.

■ **A real life applications of graph theory:**

 This section includes five distinct graph theory problems with examples from real-world situations. Since their solution can sometimes be calculated using a number of methods that are too difficult for this introductory blog, I recommend the reader to look them up. Furthermore, there's a chance that these difficulties' remedies won't be exact or unique. Graph theory techniques rely on the graph's size and complexity, therefore certain solutions might only be excellent approximations of the real answer. Furthermore, approximations are the ideal solution because certain problems have not even been solved [9]

**● Airline Scheduling (Flow problems)**

 Among the most widely used applications of graph theory are flow issues, which include situations from everyday life such as airline scheduling. Every airline flight needs an operating crew, and these flights take place all over the world. Some people may be stationed in a specific city, so not all personnel are available on every flight. Graph theory is utilized to schedule the flight crews. In this issue, a directed graph is created using flights as the input. Every serviced city serves as a vertices, and a directed edge links the flight's departure and arrival locations. One way to see the generated graph is as a network flow. The weights, or flow capacity, at the edges match the amount of crew members needed for the trip. A source and a sink vertex must be added for the flow network to be finished. The sink vertex is connected to every destination city, whereas the source is connected to the airline's base city that supplies the workers.The airline may then determine the minimum flow that covers all vertices and, consequently, the minimum staff size required to run all flights by applying graph theory. Additionally, the airline can create a timetable for a smaller crew that may not visit every city by assigning weights to the locations based on their significance.

**● Directions in a map (Shortest path)**

These days, we never leave our smart phones to assist us in our daily lives. It's helpful to me because it provides directions on how to ride a bike from where I am to a restaurant or bar. However, how are these directions determined? This problem, which belongs to the category of defining the shortest path, can be solved using graph theory. Creating a graph from a map is the first stage. In this case, the streets that connect crossings are regarded as edges, and all-street intersections as vertices. Weights assigned to the edges may indicate the actual distance or the time required to go between vertices. You may direct this graph to show the city's one-way streets as well. An algorithm now simply has to determine the path with the lowest sum of edge weights between the two corresponding vertices in order to determine the direction between two points on the map. For small graphs, this can be easily solved; however, this becomes a challenging challenge for graphs generated from large cities. Luckily, there are numerous methods available that, while they might not provide an exact answer, will provide a highly accurate approximation. Examples of these algorithms include the Dijkstra's algorithm and the A\* search algorithm One of the most common uses of graph theory is determining the fastest or shortest path between two points on a map.[10] The shortest path problem has further uses, though. It can be used, for instance, to determine the "six degrees of separation" between individuals in social networks or to determine the network's minimum delay time in telecommunications networks.

**● Solving Sudoku’s puzzles (Graph colouring)**

The numbers in the 9x9 grid of the well-known puzzle Sudoku must be filled in from 1 to 9. A handful of digits serve as hints, and the remaining numbers must be filled in according to a straightforward rule: they cannot be repeated in the same row, column, or area. Despite utilizing numbers, this puzzle is combinational rather than mathematical and can be solved with the aid of graph colouring.

The puzzle can be made into a graph. Here, a vertex represents each place on the grid. If two vertices are in the same row, column, or region, they are linked. Given that there is a bidirectional interaction between the vertices, this graph is undirected. The labelling of each vertex in the graph is a crucial component. The number used in that position and the label match. The labels of vertices in graph theory are referred to as colours. It is necessary to give each vertex a colour in order to complete the puzzle. The primary Sudoku rule states that no row, column, or region may contain two of the same number. As a result, no two connected vertices may have the same colour. Graph colouring is the name of this problem. Similar to other graph theory problems, it may be solved using a variety of methods (such as the D Satur algorithm or Greedy colouring), although the effectiveness of each strategy is largely dependent on the graph in question.

Typically, the colouring problem is applied to quite basic issues. But there are other real-world issues, like task scheduling, that can be converted into colouring problems. For instance, booking test rooms. If multiple exams are given at the same time, they are each a vertex with an edge linking them. The generated graph is known as an interval graph, and the bare minimum of exam rooms can be obtained by working out the graph's minimum colouring issue. Tasks that share the same resources, like radio station bandwidth allocation or language compilers, can be generalized to include this.

**● Search Engine Algorithms (Page Rank algorithm)**

We have no trouble navigating the World Wide Web thanks to search engines like Google. When a user queries a certain set of terms, the search engine finds WebPages that match the query. How does the engine sort millions of matches to display the most popular ones first. By initially building a web graph, a graph containing websites as its vertices and hyperlinks inside them as its directed edges, the search engine uses graph theory to tackle this problem. A directed graph displaying every relationship between websites is the end product. Moreover, weights can be added to the vertices to prioritize websites that are more significant or well-known. There are various algorithms that can be used to categorize the most popular websites. Page Rank is one of the first ones that Google used. In this case, the engine generates a probability distribution by iteratively adding the probabilities assigned to click on a hyperlink. The probability that someone will arrive at a specific website at random is represented by this distribution. Subsequently, the search engine presents the websites in order of this distribution, highlighting the top ones.This was a really flawed algorithm. One can take advantage of it by purchasing hyperlinks on websites with larger weights or by creating blog websites with numerous links to a certain website in order to improve the likelihood that a user will click on it. Even while more complex algorithms are now available that take sponsored advertisements into account, graph theory and the relationships between websites still form the foundation.

**● Social Media Marketing (Community detection)**

Face book boasted 2.9 billion active users as of January 2022. The majority of the network's income as a social media platform comes from advertising. Due to the large number of users, it will be exceedingly costly for marketers to place their ads where everyone can see them. You may, alternatively, merely focus on those who you think would be interested in your offering. What constitutes a target audience of this kind. By giving each individual a vertex, you may use graph theory to create a social network graph. If there is a relationship between the people, like friends on Face book, you connect vertices with edges. Thus, an undirected graph is produced. Even though this enormous graph would seem chaotic at first, there are always patterns to be found. By breaking the graph down into smaller sub-graphs, you can determine the perfect target audience. This can be accomplished by a variety of algorithms, including minimal cut techniques like the Karger's algorithm and hierarchical clustering algorithms. As a result, the graph is split up into groups of people who are very related to one another but not as much to other groups of people. These collectives, which go by the name of communities, have similar tastes in music, fashion, and even political parties. Advertising can benefit from identifying these communities because they are more inclined to vote for similar parties, purchase similar goods, or follow similar artists. Aside from advertising, there are additional uses for community detection. One can compare relationships between groups or even within groups after identifying the communities. An indication of intrusion may be seen if a group or a vertex inside the group behaves differently from its peers. This is a controllable security measure. Attacks against a machine or program within a network, for instance, could be the source of odd behaviour. Network security can be enhanced by spotting odd connections.

$∎ $**Conclusion:**

 In conclusion, graph theory is a foundational and versatile field of mathematics and computer science with wide-ranging applications. It provides a powerful framework for modelling and analyzing complex relationships and structures in various domains. We’ve looked at important representations, techniques, and ideas in graph theory, enabling you to use these ideas in real-world problem-solving situations.

$∎ $**Refrence:**

1. Bender, Edward A.; Williamson, S. Gill (2010). [*Lists, Decisions and Graphs. With an Introduction to Probability*](https://books.google.com/books?id=vaXv_yhefG8C).Berge, Claude (1958). Théorie des graphes et ses applications. Paris: Dunod. English edition, Wiley 1961;
2. Methuen & Co, New York 1962; Russian, Moscow 1961; Spanish, Mexico 1962; Roumanian, Bucharest 1969; Chinese, Shanghai 1963; Second printing of the 1962 first English edition, Dover, New York 2001.
3. Biggs, N.; Lloyd, E.; Wilson, R. (1986). Graph Theory, 1736–1936. Oxford University Press.Bondy, J. A.; Murty, U. S. R. (2008). Graph Theory. Springer. [*ISBN*](https://en.wikipedia.org/wiki/ISBN_%28identifier%29) [*978-1-84628-969-9*](https://en.wikipedia.org/wiki/Special%3ABookSources/978-1-84628-969-9).
4. Bollobás, Béla; Riordan, O. M. (2003). Mathematical results on scale-free random graphs in "Handbook of Graphs and Networks" (S. Bornholdt and H.G. Schuster (eds)) (1st ed.). Weinheim: Wiley VCH.
5. Chartrand, Gary (1985). [*Introductory Graph Theory*](https://archive.org/details/introductorygrap0000char). Dover. [*ISBN*](https://en.wikipedia.org/wiki/ISBN_%28identifier%29) [*0-486-24775-9*](https://en.wikipedia.org/wiki/Special%3ABookSources/0-486-24775-9).Deo, Narsingh (1974). [*Graph Theory with Applications to Engineering and Computer Science*](https://www.edutechlearners.com/download/Graphtheory.pdf) *(PDF)*. Englewood, New Jersey: Prentice-Hall. [*ISBN*](https://en.wikipedia.org/wiki/ISBN_%28identifier%29) *0-13-363473-* [*Archived*](https://web.archive.org/web/20190517165158/http%3A/www.edutechlearners.com/download/Graphtheory.pdf) *(PDF)* from the original on 2019-05-17.
6. Gibbons, Alan (1985). Algorithmic Graph Theory. [*Cambridge University Press*](https://en.wikipedia.org/wiki/Cambridge_University_Press).Golumbic, Martin (1980). Algorithmic Graph Theory and Perfect

**7**  .Gross J, Yellen J (1999) Graph theory and its applications. CRC Press, LondonPrasanna NL (2014) Applications of graph labeling in communication networks. Int Res J Comp Sci Technol 7(1)

 **8**. West DB (2000) Introduction to graph theory, 2nd edn. Prentice-Hall, Englewood Cliffs, NJAbirami P, Parthiban A, Srinivasan N (2020) On 2-odd labeling of graphs. Eur J Mole Clin Med 07:3914–3918

**9.** Laison JD, Starr C, Walker A (2013) Finite prime distance graphs and 2-odd graphs. Discret Math 313:2281–2291

**10.** Parthiban A, Gnanamalar David N (2017) On prime distance labeling of graphs. In: Arumugam S, Bagga J, Beineke L, Panda B (eds) Theoretical computer science and discrete mathematics. ICTCSDM 2016. Lecture notes in computer science 10398. Springer, Cham, vol 31, pp 238–241