

A Study on Graph Theory and Its Application

Nanjundaswamy M.*

Assistant Professor of Mathematics, Sri Mahadeshwara Government First Grade College, Kollegal,
Chamarajanagar, District-571440

Abstract – The main purpose of this paper is to present the main principles of Graph theory and analysis regarding linked graph, Eulerian graph, Hamiltonian graph etc. Graph analysis is a theory of mathematics which has wide application in the field of mathematics as well as in other branches of research. This paper intends to highlight the uses of graph theory in our everyday life, in Computer science, Operation Research.

Keywords: Hamiltonian Graphs, Eulerian Graph, Graph, Operational Research;

----- X -----

INTRODUCTION

The definition of a diagram is very modern since it only emerged formally throughout the 20th century. In many fields, particularly in applied and fundamental computer sciences, optimization and algorithmic complexity, it has become important today. Therefore, the analysis of graphs and their implementations offers an ability to work with multiple applications in a number of ways. For example, it can be used to build task scheduling methods from optimal graph paths or communication networks' properties in relation to graph connectivity.

The theory of graph is a subset of discrete mathematics. Graph theory is the analysis of mathematical diagrams used to form pair-size relationships between artefacts in mathematics and in computer science. Graphs are commonly used to include problem solving methods since it provides an intuitive way before the formal description is presented.

GRAPH

A simple graph X is a pair of sets ordered $X = (V, E)$. V elements are known as X vertices and E elements are known as X edges. An edge joins two vertices, the vertices of the end. We can formally consider the E elements as V sub-sets of size. A simple graph is therefore an undirected graph without loops or several edges.

If $u \neq v$ Are vertices of a single graph X and $\{u, v\}$ is a border of X , then that border is said to occur to u, v . Likewise, u and v are said to be adjacent or neighbors, and we write $u \sim v$ as well. Phrases like, 'a border joins u and V ' and 'a boundary between u and v ' are also common.

Graphs can be nicely defined by diagrams of dots with vertices and edge lines (see Fig. 1).

A simple graph contains at most one edge that connects a couple of vertices. Multiple edges between pairs of vertices are allowed in a multigraph. Edges, known as loops, can also attach a vertex (see Fig. 2).

Unlike the simplest diagram, in which edges are undirected, a directed diagram (in short, digraph) is an ordered pair of sets (V, E) , where V consists of a set of vertices and E consists of a subset of ordered pairs of V vertices.

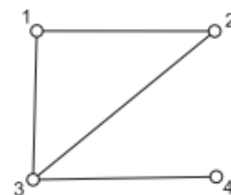


Fig. 1 A graph $X = (V, E)$, in which $V = \{1, 2, 3, 4\}$ and $E = \{\{1, 2\}, \{1, 3\}, \{2, 3\}, \{3, 4\}\}$ are expressed.

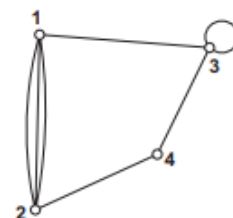


Fig. 2 An example of a multigraph between vertices 1 and 2 with three edges and a circuit on vertex 3

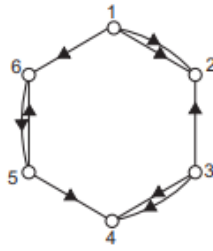


Fig. 3 A directed graph

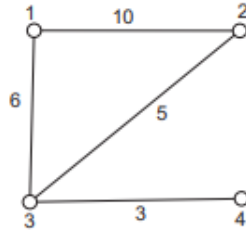


Fig. 4 A weighted graph

Sometimes it is useful to associate a number with each edge in a graph, often called its weight. These graphs are referred to as border-weighted or simply weighted graphs; they can be simple, directed, and so forth (see figure 4).

From here on, a graph is a simplistic, finite, undirected, and connected graph, unless otherwise specified. Let $X = (V, E)$ be a plot. The degree $d(v)$ of a vertex $v \in V$ is the number of edges it is occurring. The neighboring set $N(v)$ of a vertex v is a set of nodes adjacent to v , $d(v) = |N(v)|$ (See Fig. 5). The index and the outdegree of the vertex $v \in V$ in the case of directed graph $X = (V, E)$ is the number of rims v as a head vertex and as a tail vertex, including both. For instance, the vertex 5 in the figure shown in Fig. Three has both Indegree 1 and Outdegree 2.

When people shake hands at one party, the total number of hands shaken is twice as high. This fact, which translates into graph-theoretical terminology, represents the party through a graph (each person is represented by a vertex and a handshake between two persons, represented by a rim between the corresponding vertices).

Proposition 1 (Lemma handshaking). The number of all vertices in a single graph is twice as many edges.

Proof. Each edge is two for the sum of the degrees, one for each of the edges.

As a result, we get the expected result.

Result 1. Each graph contains a number of peculiar vertices. Proof. Proof. We know that the vertices are divided into those of even and strange grades

$$\sum_{v \in V} d(v) = \sum_{d(v) \text{ is odd}} d(v) + \sum_{d(v) \text{ is even}} d(v)$$

The left side of the figure has an even value and the second summand on the right side is even as it is a total sum of equal values. Therefore, the first summand on the right should be even. However, because it is completely a sum of strange values, it has to contain just a few terms. In short, the number of vertices with a strange degree must be equal.

If all vertices have same degree of d , a graph X is d -regular. Usually a 3-regular simple chart is called a cubic chart.

Result 2. There are even numbers of vertices in each cubic graph.

A graph X is a graph with all its vertices and edges in X (see Fig. 6). A subgraph of X that covers all vertices of X is a subgraph. Graph X complements have the $V(X)$ set and two vertices are relatively close if and only if not adjacent to X (see Fig. 7).

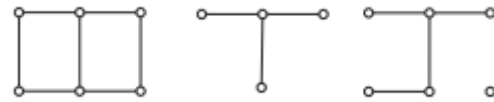


Fig. 6 A graph and two of its subgraphs

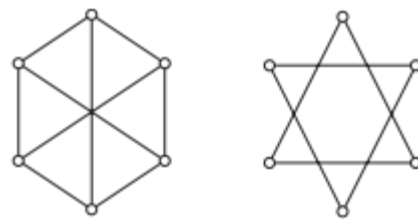


Fig. 7 A graph and its complement

TRAVERSABILITY

A walk (or $v_0 v_k$ - Walk) a graph X is a vertical sequence

$$v_0, v_1, v_2, \dots, v_k$$

such that v_i is adjacent to v_{i+1} for every $i \in \{0, 1, 2, \dots, k-1\}$. It is closed if $v_0 = v_k$. And otherwise it's open. It is a path that is separate from all the vertices. It is a path when all borders are distinct. A graph is connected if a path connects two edges and otherwise it will be disconnected. A cycle in a graph (also called a circuit) is a closed path $uv_1, \dots, v_{k-1}u$ in which all $u, v_i, i \in \{1, \dots, k-1\}$, Are distinguishable. A tree is a cycle-free graph connected.

Let us offer some examples of well-known graphs. A cycle C_n is a graph with n vertices in which each vertex has a degree of 2. A complete graph K_n is a diagram with n vertices adjacent to two vertices. A two-part graph is a graph whose axis V could be

separated into two subsets Y and Y'' in just such a way that any corner has one end in Y as well as the other end in Y'' . Such a partition (Y, Y'') is regarded as a graph bipartition. A complete two-part graph is a simple two-part diagram with a two-party graph (Y, Y'') in which each Y vertex is connected with each Y'' vertex. If $|Y| = m$ and $|Y''| = n$, that diagram will be marked with $K_{m,n}$ (see Fig. 8).

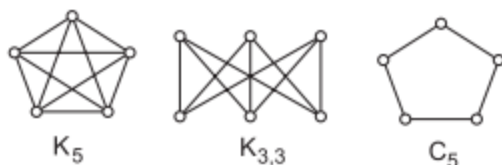


Fig. 8 Examples

Proposition 2.

The following are equivalent to a simple graph $X = (V, E)$:

- (1) X is associated and connected $|E| = |V| - 1$.
- (2) X is attached, and acyclic (X is a tree) is attached.
- (3) X is related, but any edge removal from X leaves a graph detached.
- (4) The path between these two distinct vertices of X is special, easy.

Graphs and Matrices

Let X be a graph. The adjoining X matrix in reference to the vertex symbol $v_1, v_2, \dots, v_n$ is the n default matrix $A(X)$.

$$a_{ij} = \begin{cases} 1, & \text{if } v_i \text{ and } v_j \text{ are adjacent} \\ 0, & \text{otherwise} \end{cases}$$

For e.g., the adjacencies matrix of graph K_4

$$A(K_4) = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}$$

Notice that the undirected graph adjacency matrix is symmetrical. There is an important use of the forces of the graph's adjacency matrix (see for example [12, 40]).

Proposal 3. If A is the neighboring matrix of graph X in order n , relative to the top of vertex marking $v_1, v_2, \dots, v_n$, the (i, j) entry is the number of separate r -walks from vertex v_i to vertex v_j in the graph.

The above matrix square $A(K_4)$ gives

$$A^2 = A(K_4)^2 = \begin{pmatrix} \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} \end{pmatrix}^2 = \begin{bmatrix} 3 & 2 & 2 & 2 \\ 2 & 3 & 2 & 2 \\ 2 & 2 & 3 & 2 \\ 2 & 2 & 2 & 3 \end{bmatrix}$$

The matrix provides the number of paths between both the vertices of graph K_4 using two edges

Eulerian Graphs

The little bit of history: in Prussia, Königsberg was on the Pregel River and was a home for the Prussian Dukes in the seventeenth century. The town is now renamed Kaliningrad, and is one of western Russia's main manufacturing and commercial centers. The Pregel river passed through the city and split it into four areas. Seven bridges linked the four regions in the 18th century. On Sundays, residents of Königsberg took long walks around the area. They asked whether it was feasible to start at a position in the area, cross both bridges twice without crossing any bridge, and then return to the starting point. This issue was first discussed by the prominent Swiss mathematician Leonhard Euler, who developed the field of mathematics now referred to as graph theory as a result of his solution. The response to Euler was a graph with the four regions of four vertices and the seven bridges of seven sides, as seen in the figure. 9. The task defined as a general graph theory problem is to create a closed path of the diagram that crosses each edge exactly once. It proceeds from Proposition 4 that the Königsberg Bridges Trail is not feasible.

A Eulerian trail of an undirected graph X is a closed walk around each corner of X exactly once.

Proposition 4. If an undirected graph X has a Eulerian trail, X is associated and each vertex is even. Proof. Naturally, the Eulerian trail leads to every tip. As many times as we exit a vertex, we join it. Each vertex must be identical to the outdegree. In fact, all classes are equivalent.

Proposition 5. If X is a linked graph and each vertex is even, X would have a Eulerian path.

Proof. If X only has one vertex, it has an incomplete Eulerian trail (1-vertex). Suppose then that in X , each vertex has at least 2 (even) degrees. The consequence is apparent if all vertices are grade 2, then X is a cycle at the moment. By induction, suppose that for all graphs the consequence is smaller than the "normal" grade of X . Because X is not a tree, cycle C can be contained in X . Then $X - C$ has a Eulerian trail with C which provides us with the required Eulerian trail in X .



Fig. 9 Konigsberg bridges in a graph

Hamiltonian Graphs

A basic cycle that passes through each vertex precisely once is known as the Hamiltonian cycle. Similarly, the Hamiltonian path is a straightforward path that crosses any vertex exactly once. A graph in Hamilton is a graph with a Hamiltonian cycle.

Proposition 6 (Dirac theorem). Any graph with n vertices $n \geq 3$, Where the minimum degree is at least $n/2$, each vertex has a Hamiltonian cycle.

Proof. Assume that any non-Hamiltonian graph has n vertices, by implication, $n \geq 3$. And that each vertex 's minimum degree is at least $n/2$. If we connect edges at a time to this line, we end up with a complete graph with a Hamiltonian cycle. We get a line, name it X anywhere in this process and don't get a Hamiltonian loop, but inserting an edge uv gives a Hamiltonian graph and naming it X . For the graph X , we now get a paradox.

As X is Hamiltonian, X must have a Hamiltonian way $u = u_1, \dots, u_n = v$. Adding u and v . This direction contains by definition all X vertices. Let us now play on this journey and turn it into a cycle and get a paradox. We use the fact that u and v have at least $n/2$ to generate two edges in order to replace one edge (u_i, u_{i+1}) on this path.

Let's qualify: of the $n-2$ vertices on the way $u_2, \dots, u_{n-1}$. We know that at least $n/2$ are neighbors of u and at least $n/2$ are neighbors of v . Similarly, there are two neighboring uprights, u_i and u_{i+1} , which are neight of v and $i+1$ is neighbor of u , based on the Pigeon Hole theory. In other terms, let's $S = \{i \mid u_{i+1} \sim u\}$ and $T = \{i \mid u_i \sim v\}$. Then every S and T has $n/2$ elements at least. Since there are only $n-2$ valid i values, certain i have to be in both sets. In other terms, u_i is a v neighbor, and u_{i+1} a u neighbor. In X , generating a Hamiltonian cycle is now a simple process, offering us the contradiction we seek.

In order to complete this subsection, we give Hamilton exercises

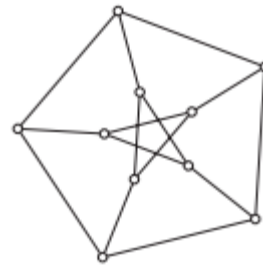


Fig. 10 The Petersen graph

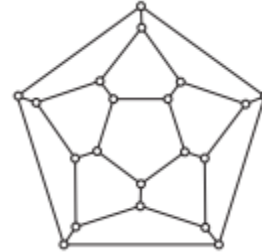


Fig. 11 The dodecahedron

Factorizations, Colorings and Tournaments

A graph X factor is an X -spreading subgraph that is not fully disconnected. (A fully isolated order n graph is the complement $nK_1 = \bar{K}_n$ of the maximum graph K_n .) We state that X is the sum of the factors X_i if it is their edge disjoint union and a factorization is called X . An n -factor is an n -regular graph factor. If X is the number of n -factors, its union will be labelled n -factorization and X itself will be n -factoring.

When $X = (V, E)$ It's obvious that the $|V|$ is even and the edges of the 1 factor are disjoint vertex. K_{2n+1} can't have a 1 element in particular, but K_{2n} can definitely be seen in the proposal below.

Proposition 7. The complete graph K_{2n} is 1-factorable.

Proof. We just need to display the E-set partition of the K_{2n} edges into $(2n-1)$ 1-factor. Denote the hills of K_{2n} by $v_1, v_2, \dots, v_{2n}$. Define, for $i \in \{1, 2, \dots, n-1\}$, the sets of edges

$$E_i = \{v_i v_{2n}\} \cup \{v_{i-j} v_{i+j} \mid j \in \{1, 2, \dots, n-1\}\},$$

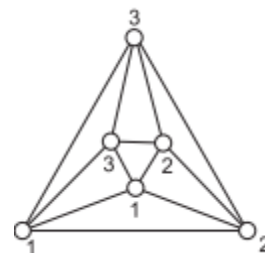


Fig. 12 Not a good scheduling

Where each of the $l - j$ and $l + j$ subscriptions are described as one of the $1, 2, \dots, (2n - 1)$ modulo number $(2n - 1)$. The $\{E_i\}$ set conveniently reveals an effective E partition, and the number of X_i subgraphs caused by E_i is a K_{2n-1} -factor.

An example of the dilemma of scheduling is where vertex coloring can be used. For each final review, the Schedules Office shall reserve a time period. This is not simple since several students take multiple final classes and a student may only take one exam on a certain time period. The Schedules Office tries to prevent some tension but aims to guarantee that the review time is as brief as practicable. Let any vertex be a path for which final examinations are performed. Place a corner between two vertices if any students take all classes. Identify the colour of each potential time slot. For example, colour Monday 9–12, colour 1, colour from Monday 1–4, colour 2 and colour 3, colour 9–12, etc. If there are two vertices of the same colour, a dispute assessment may be arranged when a student needs to be examined for the courses represented by vertices but the tests are scheduled at the same period (see Fig. 12). All needs dispute tests to be stopped. The registrar needs to colour every vertex of the graph to guarantee that there are no neighboring vertices of the same colour. To make the test time as brief as practicable, the Registrar needs to use the least amount of colours possible.

The minimum of colours needed to colour graph X vertices such that no two neighboring vertices are of the same colour shall be named the chromatic number and be labelled with $\chi(X)$. Examples include: $\chi(C_n) = 3$ if n is odd and $\chi(C_n) = 2$ if n is even. The following ideas grant the chromatic number the upper limit.

SYMMETRIES IN GRAPHS

Digraphs $X_1 = (V_1, E_1)$ and $X_2 = (V_2, E_2)$ are invertible if and only if a bijection occurs $f: V_1 \rightarrow V_2$ such that for all $u, v \in V_1$

$$(u, v) \in E_1 \Leftrightarrow (f(u), f(v)) \in E_2.$$

The bijection f is considered a graph isomorphism.

Example 1. Table 1. Let X_1 be a simple graph with vertices $1, \dots, 2n$ with an edge between vertices, given they are of the same parity. Let X_2 be a simple graph with integer vertices $-1, \dots, -2n$ with an edge between two vertices, only if the two vertices are less than $-n$ or both are higher or similar to $-n$. The role then $f: \{-1, \dots, -2n\} \rightarrow \{1, \dots, 2n\}$,

where $f(k) = -2k$ if $-n \leq k \leq -1$ and $f(k) = 2(2n+k)+1$ if $-2n \leq k < -n$, is an isomorphism between X_1 and X_2

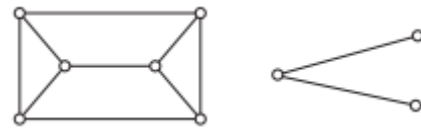


Fig. 18 The smallest graph which is transitive to vertex but not transitive to the edge (3 Prisma $K_3 K_2$) and the smallest graph, which is transitive to the edge but not transitive to vertex (star $K(1,2)$).

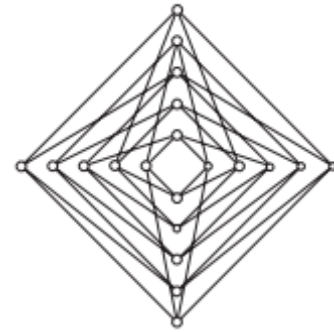


Fig. 19 The Folkman graph

Instances of edge-transitive even if not vertex-transitive graphs $K_{m,n}$, $n \neq m$ (see also Fig. 18). A regular transitive edge but not a transitive vertex graph is considered semi symmetric. Work has started on these graphs and laid the groundwork for Jon Folkman's theory. The smallest of these diagrams has 20 vertices and is now referred to as the Folkman Graph (see Fig. 19). The Gray graph in Fig is the smallest cubic semi-symmetric graph. 20 (See [24] as well).

APPLICATION OF GRAPH THEORY IN COMPUTER SCIENCE

In the world of computer science, graph theory plays an extremely significant role. Any software to be created and any application to be checked render graphics simple to use. Its significance arises from the fact that data flow of control and flow can be represented in guided graphs with every programme. Graph theory is often used for microchip architecture, circuits, operating system scheduling concerns, database management system management, network flow control between networks. Graphic theory has helped the computing world build its own computational graph algorithms. These algorithms are used to devise strategies for many applications of computer science. Certain algorithms are:

- Shortest network route algorithm
- Kruskal's-minimum tree time
- Euler's planarity graph

- Algorithms to locate matrices adjacent to them.
- algorithms for connectivity finding
- Algorithms for graph loops
- Search algorithms for a data structure function (DFS, BFS) and so on.

The development of graph algorithms is the key task of graph theory in computer applications. Many algorithms are used to solve problems modelled in graph form. These algorithms are used to solve theoretical graphic concepts which in turn solve the corresponding problems in the field of computer science. Certain algorithms are:

- (1) Network's shortest path algorithm.
- (2) Finding the planarity of the graph.
- (3) Check for a minimal forest.
- (4) Algorithms for connectivity finding.
- (5) Algorithms in a diagram to find the cycles.
- (6) Algorithms to locate matrices adjacent to them.
- (7) Algorithms to check for a data structure function and so on.

Different programming languages endorse the principles of graph theory. The key aim of these languages is to allow the consumer to compactly and naturally formulate operations on graphs.

Graph theory in OR

Chart theoretical principles in organizational analysis are commonly used. It is used in simulation, transport networks, networks of operation and game theory. The network operation is used to address several combinatory issues. The most famous and effective network technologies in OR are the preparation and preparation of big complex projects. PERT (project evaluation Technique) and CPM (Critical Path Method) are the two popular issues. Furthermore, game theory refers to physics, economics and war science questions in order to find the perfect way to execute such functions in competitive conditions. A bigraph is used to describe the finite game method. The vertices are the locations, and the boundaries are the movements.

Uses of Graph Theory in Day to Day Life

Perhaps we don't note why we use graph theory in our everyday lives. Infact graph theory is included in all our everyday work. For instance:

- (1) To decide a path based on user settings using GPS or Google Maps / Yahoo maps (the fastest route / shorter route), or to locate the cheapest airfare between 2 destinations. The destinations are vertices with edges holding details such as distance or air fare. Based on user settings the app finds the essential path (optimal route).
- (2) Link with friends via social media or a viral video. Each user is a vertex which generates an edge if users connect. Videos are considered to be viral after a range of connections / views have been achieved.
- (3) Google is used to scan web sites. Hyperlinks connect internet sites; each page is a vertex, and the link between the pages is a boundary. Algorithms are used by PageRank and Googlebot to help the connectivity process.
- (4) School administration establishes bus routes for school pick-ups. Each stop is an edge and a vertex. The utility of including any vertex on the route is a Hamiltonian path.
- (5) Visit a zoo, amusement park or theme park and wish to see a range of attractions or plan a viable route for seeing all attractions. A Hamiltonian path or circuit comprising each vertex in the map is used.
- (6) Town preparation to bring salt on the roads as ice rises. Euler routes or loops are used most easily to traverse the streets.

CONCLUSION

Some simple graphics technical principles and some open problems are discussed in this brief introductory lecture on graph theory, probably one of the most propulsive fields in modern mathematics. This paper includes a short introduction to graph theory and its implementations. It is ideal for students in multiple streams. The topic of arranging timetables has been debated extensively, which is very useful for any kindergarten. Graph theory implementations are useful both in information science, mathematics and daily life.

REFERENCES

1. Allen Dickson (2016) on Introduction to Graph Theory
2. Paul Van Dooren (2009) on Graph Theory & Applications,
3. Ashay Dharwad, (2006) The Vertex Coloring Algorithm,

4. Tom Davis (2006) on graph Theory Problems and Solutions.
5. Mr. Lovejit singh (2016) on concepts of graph theory and its applications
6. K. Kutnar, D. Maruši (2008) on Some Topics in Graph Theory
7. LOWELL W. BEINEKE, ROBIN J. WILSON, PETER J. CAMERON (2007) on Topics in Algebraic Graph Theory
8. Jean-Claude (2009) on Graph Theory and Applications
9. Badwaik Jyoti S (2015) Recent Advances in Graph Theory and its Applications
10. Rishi Pal Sing, Vandana, (2014) on "Application of Graph Theory in Computer Science and Engineering,"
11. Dr. C Shoba Bindu* and K Dhanasree," Graph Theory: Applications in Various Fields,"
12. Veena Rani (2017) on USES OF GRAPH THEORY IN DAY TO DAY LIFE

Corresponding Author

Nanjundaswamy M.*

Assistant Professor of Mathematics, Sri Mahadeshwara Government First Grade College, Kollegal, Chamarajanagar, District-571440