AN INTRODUCTION TO INTERSECTION GRAPHS

Project Report Submitted To

MAHATMA GANDHI UNIVERSITY

In Partial Fulfillment Of The Requirement For The Award Of

THE MASTER DEGREE IN MATHEMATICS

BY

SIMNA SONY

M.Sc.(IV Semester) Reg No. 180011015189

DEPARTMENT OF MATHEMATICS

ST. PAUL'S COLLEGE, KALAMASSERY

2018 - 2020

CERTIFICATE

This is to certify that the project entitled **"AN INTRODUCTION TO INTERSECTION GRAPHS"** is a bonafide record of studies undertaken by SIMNA SONY (Reg no. 180011015189), in partial fulfillment of the requirements for the award of M.Sc. Degree in Mathematics at Department of Mathematics, St. Paul's College, Kalamassery, during 2018 – 2020.

Ms. NISHA V M
Department of Mathematics

Dr. SAVITHA K S
Head of the Dep of Mathematics

Examiner :

DECLARATION

I **SIMNA SONY** hereby declare that the project entitled **"AN INTRODUCTION TO INTERSECTION GRAPHS"** submitted to department of Mathematics St. Paul's College, Kalamassery in partial requirement for the award of M.Sc Degree in Mathematics, is a work done by me under the guidance and supervision of **Ms. Nisha V M ,** Department of Mathematics, St. Pauls's College , Kalamassery during 2018 –2020.

I also declare that this project has not been previously presented for the award of any other degree, diploma, fellowship, etc.

Date SIMNA SONY

Kalamassery

ACKNOWLEDGEMENT

I express my heartfelt gratitude to my guide **Ms. Nisha V M** for providing me necessary stimulus for the preparation of this project.

I would like to acknowledge my deep sense of gratitude to **Dr. Savitha K S**, Head of the department and all other teachers of the department and classmates for their help at all stages.

I also express my sincere gratitude to **Prof. Valentine D'Cruz** , Principal, St. Paul's College, Kalamassery for the support and inspiration rendered to me in this project report.

Kalamassery **SIMNA SONY**

CONTENTS

CHAPTER 1

1. INTRODUCTION.

It is well known that graph is a very useful tool to model problems originated in almost all areas of our life. The geometrical structure of any communication system including Internet is based on graph. The logical setup of a computer is designed with the help of graph. So graph theory is an old as well as young topic of research. Depending on the geometrical structures and properties different type of graphs have emerged, viz. path, cycle, complete graph, tree, planar graph, chordal graph, perfect graph, intersection graph, etc.

Here, we concentrate our discussion on intersection graphs.

suppose $S = \{S_1, S_2, \dots\}$ be a set of sets. Draw a vertex (v_i) for each S_i and two vertices v_i and v_j are joined by an edge if the corresponding sets have a non-empty intersection, i.e. the edge E is given by $E = \{(v_i, v_j) \mid S_i \cap S_j \neq \emptyset\}$. An undirected graph $G = (V, E)$ is said to be χ–perfect if $\omega(G(A)) = \chi(G(A))$,For all A⊆ V,and G is said to be α-perfect if $\alpha(G(A)) = \kappa(G(A))$ for all A ⊆ V where G(A) is a sub graph induced by a subset A of vertices. A graph is called perfect if it is either χ -perfect or α -perfect. It was proved in the famous Perfect Graph Theorem that a graph is χ -perfect if and only if it is α- perfect.

An undirected graph G is called p-critical if it is minimally imperfect, i.e. G is not perfect but every proper induced subgraph of G is a perfect graph.

An undirected graph G is called triangulated if every cycle of length strictly greater than three possesses a chord. In the literature, triangulated graphs are also called as chordal , rigid-circuit, monotone transitive and perfect elimination graphs.

The clique graph C(G) of a graph G is the intersection graph of the family of all cliques of G . The intersection graphs of conformal hypergraphs are just the clique graphs. Cographs (also called complement reducible graphs) are defined as the graphs which can be reduced to single vertices by recursively complementing all connected subgraphs.

A graph is a comparability graph if its edges can be given a transitive orientation. A cocomparability graph is the complement of a comparability graph.

1.1. INTERSECTION GRAPHS

Intersection graphs are very important in both theoretical as well as application point of view. we define intersection graphs as follows.

A graph $G = (V, E)$ is called an intersection graph for a finite family F of a non-empty set if there is a one-to-one correspondence between F and V such that two sets in F have non-empty intersection if and only if their corresponding vertices in V are adjacent. We call F an intersection model of G. For an intersection model F, we use G (F) to denote the intersection graph for F .

Depending on the nature or geometric configuration of the sets $S_1, S_2, \ldots$ different types of intersection graphs are generated. The most useful intersection graphs are

- Interval graphs (S is the set of intervals on a real line)
- Tolerance graphs
- Circular-arc graphs (S is the set of arcs on a circle)
- Permutation graphs (S is the set of line segments between two line segments)
- Trapezoid graphs (S is the set of trapeziums between two line segments)
- Disk graphs (S is the set of circles on a plane)
- Circle graphs (S is the set of chords within a circle)
- Chordal graphs (S is the set of connected subgraphs of a tree)

• String graphs (S is the set of curves in a plane)

• Graphs with boxicity k (S is the set of boxes of dimension k)

• Line graphs (S is the set of edges of a graph).

It is interesting that every graph is an intersecting graph. For each vertex v_i of , G form a set S_i consisting of edges incident to v_i,the two such sets have a non-empty intersection if and only if the corresponding vertices share an edge.

An example of intersection graph is depicted in Figure 1.

Let $S_+ = \{1,2,3, \dots\}$, $S_- = \{-1, -2, -3, \dots\}$,$S_P = \{2,3,5,7,11, \dots\}$, $S_f = \{1, ,2,3,5,8, \dots\}$,$S_e = \{0, \pm 2, \pm 4, \dots\}$,$S_o = \{\pm 1, \pm 3, \dots\}$
be the sets of positive integers, negative integers, primes, Fibonacci numbers, even integers and odd integers respectively.

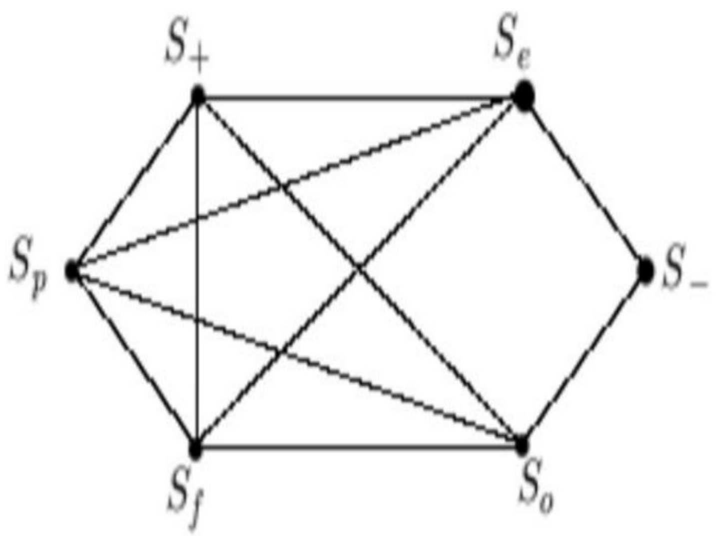


FIGURE 1: An intersection graph

Some of them are discussed in the subsequent sections.

CHAPTER 2

2. INTERVAL GRAPHS

An undirected graph ,G=(V,E) is said to be an interval graph if the vertex set V can be put into one-to-one correspondence with a set I of intervals on the real line such that two vertices are adjacent in G if and only if their corresponding intervals have non-empty intersection. That is, there is a bijective mapping $f : V \to I$.

The set I is called an interval representation of G and G is referred to as the interval graph of I.

Interval graphs arise in the process of modelling many real life situations, specially involving time dependencies or other restrictions that are linear in nature. This graph and various subclass thereof arise in diverse areas such as archeology, molecular biology, sociology, genetics, traffic planning, VLSI design, circuit routing, psychology, scheduling, transportation, etc. Also, interval graphs have found applications in protein sequencing, macro substitution , circuit routine , file organization , job scheduling , routing of two points nets and so on. In addition to these, interval graphs have been studied intensely from both the theoretical and algorithmic point of view.

In the following an application of interval graph to register allocation is presented. A computer program stores the values of its variables in memory. For arithmetic computations, the values must be entered in easily accessed locations called registers. Registers are expensive, so we want to use them efficiently. If two variables are never used simultaneously, then we can allocate them to the same register, one after use of other. For each variable, we compute the first and last time when it is used. A variable is active during the interval between these times.

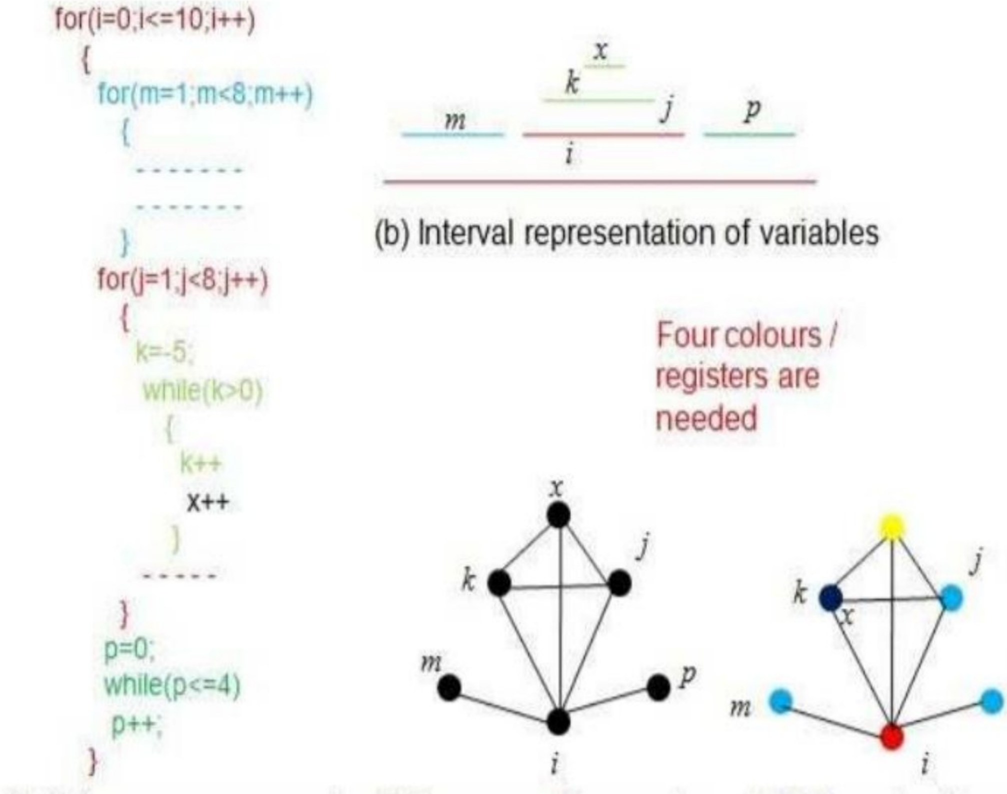
We define a graph whose vertices are the variables. Two vertices are adjacent if they are active at a common time. The number of registers needed is the chromatic number of this graph. The time when a variable is active is an interval, so we obtain a special type of representation called interval graph. Let us consider a program segment shown in Figure 2(a).

The corresponding interval representation and interval graph are shown in Figure 2(b) and 2(c) respectively. Note that the chromatic number of the graph of Figure 2(c) is 4. That is, to execute this program segment of Figure 2(a), only 4 registers are required. Figure 2(d) shows the allocation of registers/colours. Note that an interval graph can be coloured in O(n) time, where n is the number of vertices .

Let us consider another example.

Suppose a company or an organization is interested to run its advertisement through television channels. The constraint is that only one programme slot is selected for advertisement at any instant. The objective of the problem is to select the programme slots such that the sum of the number of viewers of the selected programmes is maximum.

In this problem, a television programme slot is represented as a subinterval of an interval of length of 24 h on the real line. Each programme slot, i.e. each interval is regarded as a vertex of the interval graph G . That is, all the programme slots of all television channels constitute the set of vertices V of the interval graph G . If there exists an intersection of timings in between two programme slots, there is an edge between the vertices corresponding to these programme slots. If the finishing time of a programme is the starting time of another programme then we assume that these programmes are non-intersecting. It may be noted that any two programmes in a particular television channel are always non-intersecting. The number of viewers of each programme is considered as the weight of the corresponding vertex of the graph. All the programme slots of all television channels are modelled as an interval graph.



(a) **A C program segment** (c) **corresponding graph** (d) **after colouring**

FIGURE 2: An application of interval graph in register allocation

A colouring of a graph is an assignment of colours to its vertices so that no two adjacent vertices have the same colour. The vertices of same colour form a colour class. Any two vertices of a colour class are not adjacent. The maximum weight colouring problem is to find a subset S of V such that no two vertices of S are adjacent and sum of the weights of the vertices of S is maximum. This problem is also known as maximum weight 1-colouring problem. Thus, the above problem is equivalent to the 1-colouring problem on interval graph. As mentioned above, all the programme slots of all channels in a geographical area can be represented as an interval graph.

Interval graphs satisfy a lot of interesting properties. The first one is the hereditary property.

LEMMA 2.1. An induced subgraph of an interval graph is an interval graph.

PROOF: If $\{I_v\} v \in V$ is an interval representation for a graph G=(V,E), then $\{I_v\} v \in X$ is an interval representation for the induced subgraph $G_X = (X, E_X)$.

The next property of interval graphs is also a hereditary property, called triangulated graph property, which is stated below.

Every simple cycle of length strictly greater than 3 possesses a chord. The graphs which satisfy this property are called triangulated graphs. So we have the following lemma.

LEMMA 2.2. An interval graph satisfies the triangulated graph property.

Another important property on graphs is transitive orientation property stated below:
Each edge can be assigned a one-way direction in such a way that the resulting oriented graph (V,E) satisfies the following condition:

$$(u,v) \in E \text{ and } (v,w) \in E \Rightarrow (u,w) \in E, u,v,w \in V.$$

LEMMA 2.3. The complement of an interval graph satisfies the transitive orientation property.

PROOF: Let G=(V,E) be graph and let, $\{I_v\}_{v \in V}$ be an interval representation for G . Let us assign an orientation A of the complement $G^c = (V, E^c)$ as follows:

$$V_i v_j \in A \Leftrightarrow I_{vi} < I_{vj} \text{ for all } v_i v_j \in E^c$$

Since $I_{vi} < I_{vj}$, the interval I_{vi} lies entirely to the left of the interval I_{vj} . Trivially, the condition for transitive orientation property is satisfied. It follows that $I_{vi} < I_{vj} < I_{vk}$. Hence, A is a transitive orientation of G^c. ●

Let G=(V,E) be a graph. A set of vertices $C \subseteq V$ forms a clique in G if every pair of vertices in C are adjacent. A maximal clique is a clique to which no further vertex of the graph can be added so that it remains a clique. A maximum clique is a clique with maximum cardinality.

The following theorem posed by Gilmore and Hoffman establishes the position of the interval graphs in the world of perfect graphs.

THEOREM 2.4. Let A_1 and A_2 he maximal cliques of G.

(a) These exists an edge in F with one endpoint in A_1 and the other endpoint in A_2.

(b) (b) All such edges of £ connecting A_1 with A_2 have the same orientation in F.

PROOF:

a) If no such edge exists in F, then $A_1 \cup A_2$ is a clique of G, contradicting maximality.

b) Suppose ab ∈ F and dc € F with a,c ∈ A_1 and b,d ∈A_2. We must show a contradiction. If either a = c or b = d, then transitivity of F immediately gives a contradiction; otherwise, these four vertices are distinct (Figure **), and ad or be is in F, since F may not have a chordless 4-cycle. Assume, without loss of generality, that ad ∈ $\bar{E}$; which way is it oriented? Using the transitivity of F, ad ∈ F (resp. da ∈ F) would imply ac ∈ F (resp db ∈ F), which is impossible, and Lemma is proved.

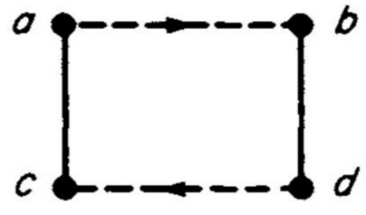


FIGURE **. Solid edges are in E; broken arrows denote the orientation F .

All maximal cliques of an interval graph can be computed in O(n) time. The maximal cliques versus vertices incidence matrix of a graph G is called clique matrix.

The necessary and sufficient condition that a graph is an interval graph is stated below:

THEOREM 2.5. A graph is an interval graph if and only if it contains none of the graphs shown in Figure 3 as an induced subgraph.

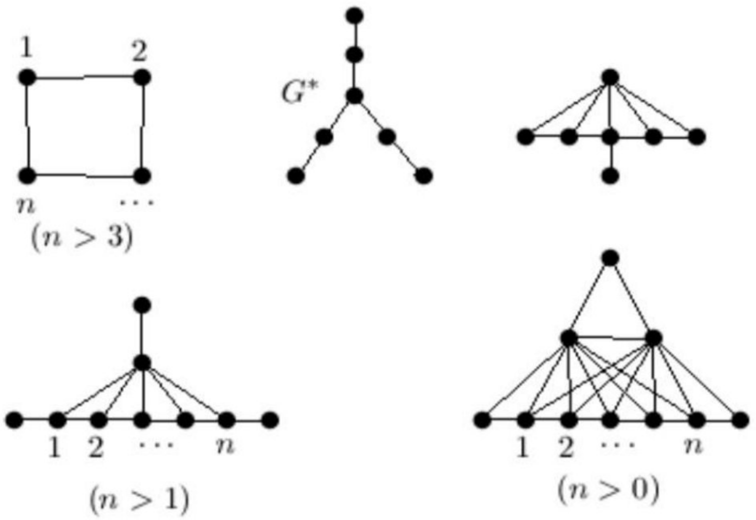


FIGURE 3: Forbidden structure for interval graphs.

COROLLARY 2.6. A tree is an interval graph if and only if it does not contain G* (Figure 3) as an induced subgraph.

Let G=(V,E) , where $|V| = n$, $|E| = m$ be a simple connected interval graph, where vertices are numbered as 1, 2, n ,… . Let $\mathrm{I} = \{\mathrm{I}_1, \mathrm{I}_2, \dots, \mathrm{I}_n\}$ be the interval representation of an interval graph G , where a_r is the left end point and the b_r is the right endpoint of the interval I_r , i.e. $\mathrm{I} = [a_r, b_r]$ for all r=1,2,…n .

Without any loss of generality we assume the following:

- The intervals in I are indexed by increasing right endpoints, i.e. $b_1 < b_2 < \cdots < b_n$

- The intervals are closed, i.e. contains both of its endpoints and that no two intervals share a common endpoint.
- Vertices of the interval graph and the intervals on the real line are one and the same thing.
- The interval graph G is connected and the list of sorted endpoints is given.

An interval graph and its interval representation are illustrated in Figure 4.
A lot of algorithms have been designed for interval graphs using different techniques.

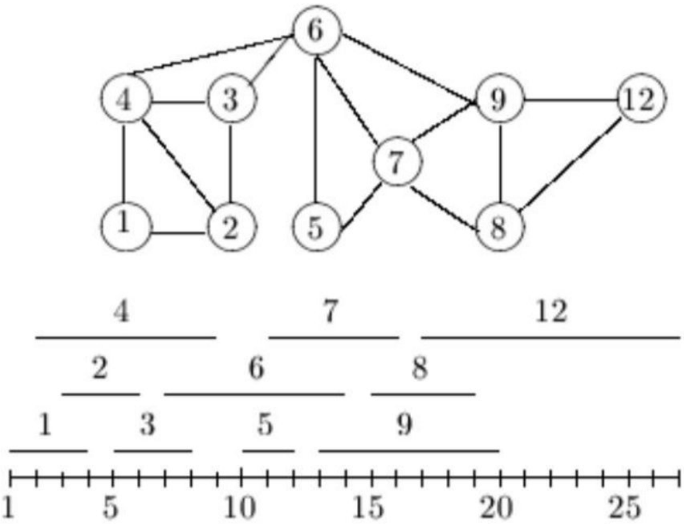


FIGURE 4 : An interval graph and its interval representation.

2.1. COLORING INTERVAL GRAPHS

To find a coloring of the vertices of the interval graph . Each color could be thought of as being a different room, and each course needs to have a room: if two classes conflict, they have to get two different rooms, say, the brown one and the red one. We may be interested in a feasible coloring or a minimum coloring – a coloring that gives the fewest number of possible classrooms.

Those who are familiar with algorithms know that some problems are hard and some of them are not so hard , and that the graph coloring problem " in general " happens to be one of those hard problems. If I am given a graph with a thousand vertices with the task of finding a minimum feasible coloring, i.e., a coloring with the smallest possible number of colors, I will have to spend a lot of computing time to find an optimal solution. It could take several weeks or months .The coloring problem is an NP- complete problem, which means that, in general, it is a difficult, computationally hard problem, potentially needing an exponentially long period of time to solve optimally.

However, there is good news in that we are not talking about any kind of graph. We are talking about interval graphs, and interval graphs have special properties. We can take advantage of these properties in order to color them efficiently. I am going to show you how to do this on an example.

Suppose we have a set of intervals, as in Figure 5. You might be given the intervals as pairs of endpoints, [1,6],[2,4],[3,11] and so forth, or in some other format like a sorted list of the endpoints shown in Figure 5.1. Figure 5 also shows the interval graph. Now we can go ahead and try to color it. The coloring algorithm uses the nice diagram of the intervals in Figure 5.1, where the intervals are sorted by their left endpoints, and this is the order in which they are processed. The coloring algorithm sweeps across from left to right assigning colors in what we call a "greedy manner". Interval a is the first to start – we will give it a color, solid "black". We come to b and give it the color "dashes", and now we come to c and give it the color "dots". Continuing across the diagram, notice "dashes" has finished. Now we have a little bit of time and d starts. I can give it "dashes" again. Next "black" becomes free so I give the next interval, e, the color "black". Now I am at a trouble spot because "dots", "dashes" and "black" are all busy. So I have to open up a new color called "brown" and assign that color to interval f. I continue coloring from left to right and finally finish at the end. This greedy method gives us a coloring using 4 colors. Can we show that the greedy method gives the smallest possible number of colors? The answer to these questions is "yes".

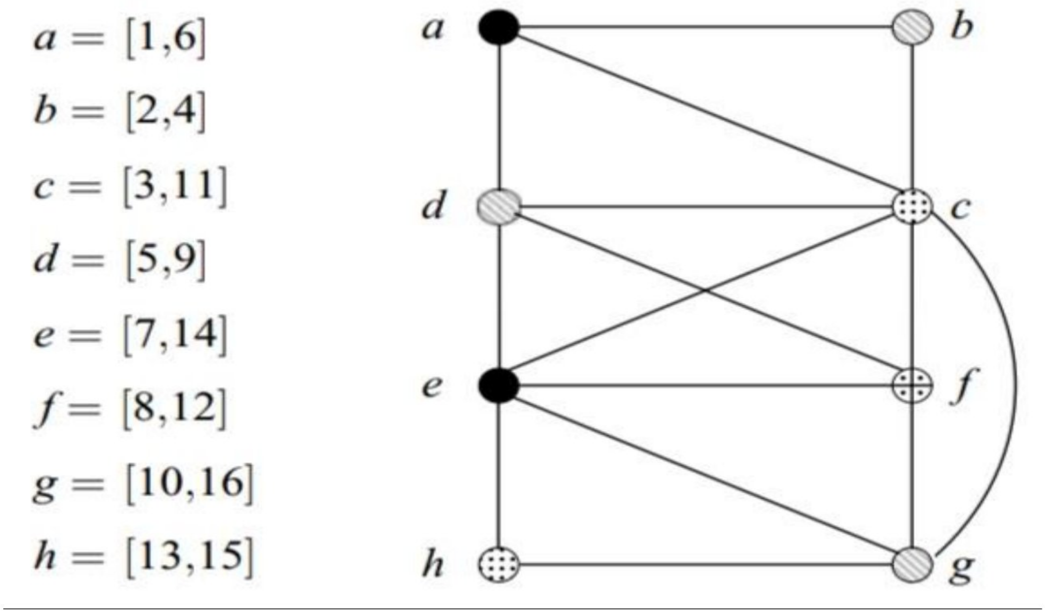


Figure5. A set of intervals and the corresponding (colored) interval graph

Since this is a mathematics lecture, we must have a proof. Indeed, the greedy method of coloring is optimal, and here is a very simple proof. Let k be the number of colors that the algorithm used. Now let's look at the point P, as we sweep across the intervals, when color k was used for the first time. In our example, k =4 and P =8 (the point when we had to open up the color "brown".) When we look at the point P, we observe that all the colors 1 through k−1 were busy, which is why we had to open up the last color k. How many intervals (lectures) are alive and running at that point P? The answer is k. I am forced to use k colors, and in the interval graph, they form a clique of size k. Formally, (1) the intervals crossing point P demonstrate that there is a k-clique in the interval graph – which means that at least k colors are required in any possible coloring, and (2) the greedy algorithm succeeded in coloring the graph using k colors. Therefore, the solution is optimal.

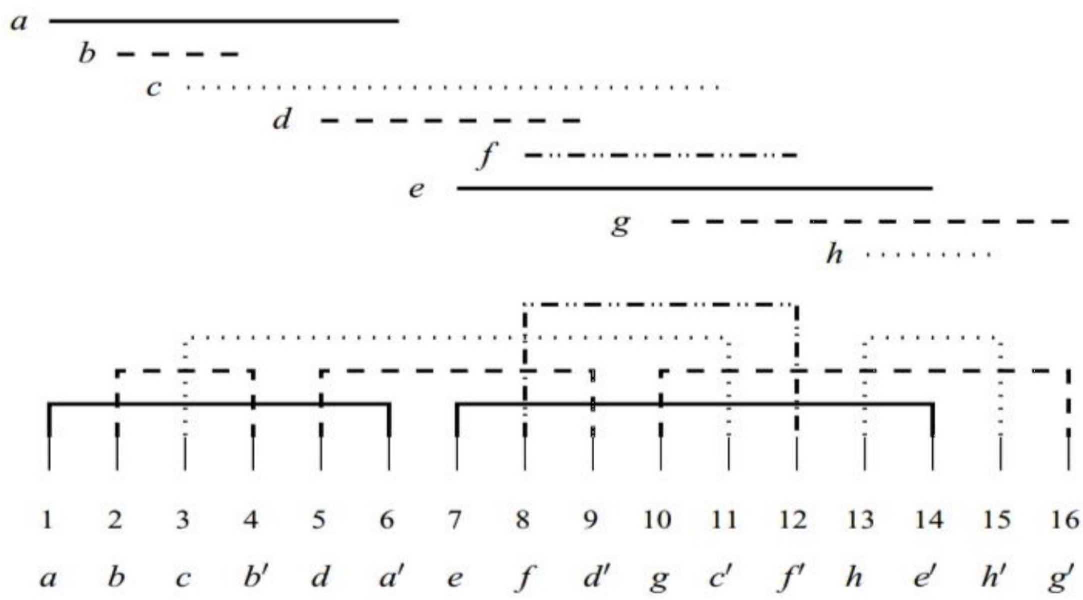


Figure 5.1: A sorted list of end points of the interval in Figure 5

A very useful data structure called interval tree (IT) is defined below

2.2. INTERVAL TREE

For each vertex V ∈ v let H(v) and L (v) represent respectively the highest and the lowest numbered adjacent vertices of v. It is assumed that (v,v) ∈ V is always true. So, if no adjacent vertex of v exist with higher (or lower) IG order than v then H(v) (or L(v)) is assumed to be v. In other words,

H(v)=$max\{u: (u, v) \in E, u \geq v\}$, and

L(v)=$min\{u: (u, v) \in E, u \leq v\}$.

It may be observed that the array H is monotonic non-decreasing, i.e. if u,v ∈ V, and u < v then H(u) ≤ H(v).

For a given interval graph G let a spanning subgraph $G' = (V, E)$ be defined as $E' = \{(u, v): u \in V \; and \; v = H(u), u \neq n\}$.

The subgraph G′ of a connected interval graph G is a tree. Since the subgraph G′ is built from the vertex set V and the edge set E′, where $E' \subseteq \mathrm{E}$, G′ is a spanning tree of G . In what follows the subgraph G′ is referred to as interval tree and it is denoted by $T_{\mathrm{I}}(G)$.
The interval tree $T_{\mathrm{I}}(G)$ of the interval graph of Figure 4 is shown in Figure 5. The level of a vertex u in the interval tree is denoted by $level_{\mathrm{I}}(u)$. Let N_i be the set of vertices which are at a distance i from the vertex n, i.e.N_i is the set of vertices at level i. Thus $N_i = \{u: \delta_G(u,n) = i\}$ and N_0 is the singleton set {n}. It may be noted that if $u \in N_i$then $level_{\mathrm{I}} = i$. Let k be the maximum length of a shortest path from the vertex n to any other vertex in G . It is easy to see that N_k is non-empty while N_{k+1} is empty.

2.3. K-GAP INTERVEL GRAPH

A multiple interval representation f of a graph G=(V,E) is a mapping that assigns to each vertex of G a non-empty collection of intervals on the real line, such that two distinct vertices u and v are adjacent if and only if there are intervals $\mathrm{I} \in f(u)$ and $J \in f(v)$ with $\mathrm{I} \cap J \neq \emptyset$. $|f(v)|$ denotes the number of intervals associated to v . The interval number of G is defined as
$i(G) = min\{max_{v\in V}\{|f(v)|\}: f is\ a\ multiple\ interval\ representation\ of\ G\}$.

The total interval number of a graph G= (V,E) is defined as
$\mathrm{I}(G) = min\{\sum_{v\in V}\{|f(v)|\}: f\ is\ a\ multiple\ interval\ representation\ of\ G\}$

A new generalization of interval graphs called t-interval graphs is introduced by Trotter and Harary , and Griggs and West. A graph G is called t-interval graphs if $i(G) \leq t$. The t -interval graphs are applicable in scheduling and resource allocation, communication protocols, computational biology, monitoring , etc.

The class of t-interval graph is richer than interval graphs, for example, the class of 2-interval graphs include circular-arc graphs, outerplanar graphs, cubic graphs, line graphs and 3-interval graphs include all planar graphs . The class of graphs with maximum degree $\Delta\ are\ \lfloor(\Delta + 1)/2\rfloor$ -interval graphs, while the complete bipartite graph $K_{m,n}$ *is a* $\lfloor (mn+1)/(m+n) \rfloor$ -interval graph. Every graph with n vertices is a $\lfloor (n+1)/4 \rfloor$ -interval graph.

Definition. The k -gap interval graph is a graph that have a multiple interval representation whose total number of intervals exceed the number of vertices by at most k , i.e. a graph G on n vertices is a k -gap interval graph if $\mathrm{I}(G) \leq n+k$.

In a k -gap interval graph with multiple interval representation f , a vertex v ∈ V has a gap if $|f(v)| \geq 2$.

A k -gap interval graph can be obtained from an interval graph by a sequence of at most k operations of identifying pairs of vertices. A multiple interval representation of G has k gaps if $\sum_{v \in V} |f(v)| = |V| + k$.

A graph G=(V,E) is an interval+kv graph if there is a vertex set $X \subseteq V with\ |X| \leq k$, such that G\X ia an imterval graph . The vertex set X is called as interval deletion set of G.

LEMMA 2.3.1 . An interval+kv graph of n vertices has $O(n2^k)$ maximal cliques.

2.4. DOTTED INTERVAL GRAPHS.

A dotted interval $\mathrm{I}(s,t,d)$ is an arithmetic progression $\{s, s+d, s+2d, \dots, t_g\}$, where s ,t and d are positive integers and d is called the jumb . When d=1 , the dotted interval $\mathrm{I}(s,t,d)$ is simply the ordinary interval [s ,t] over the positive integer line . A dotted interval graph is an intersection graph of dotted intervals. Like interval graph, for each dotted interval I_v there is a vertex v if $\mathrm{I}_u \cap \mathrm{I}_v \neq \phi$ the there is an edge between the vertices u and v .If the jumps of all intervals are at most d, then the graph is called the d -dotted-interval (d -DIG).

An example of 2-dotted interval graph is shown in Figure 6. In this figure we consider five dotted intervals $\mathrm{I}_a = \mathrm{I}(1,5,2\ ) = \{1,3,5\ \}$, $\mathrm{I}_b = \mathrm{I}(2,3,1\ ) = \{2,3\ \}$, $\mathrm{I}_c = \mathrm{I}(1,7,2\ ) = \{1,3,5,7\ \}$, $\mathrm{I}_d = \mathrm{I}(4,6,2\ ) = \{4,6\ \}$, $\mathrm{I}_e = \mathrm{I}(6,8,2\ ) = \{6,8\}$. Note that each dotted interval is a set of integers and each set of integers forms an arithmetic progression.

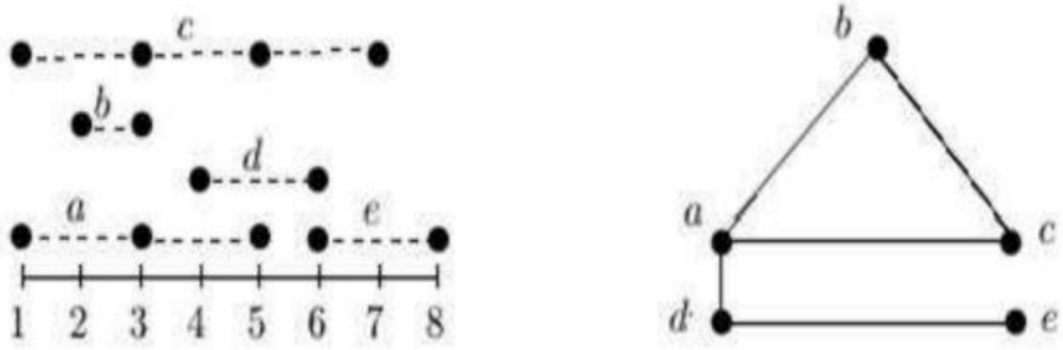


Figure 6: A set of dotted intervals and corresponding 2-dotted interval graph

THEOREM 2.4.1. Every graph with a countable number of nodes is a dotted interval graph.

THEOREM 2.4.2. For all $d \geq 1, d\text{-}DIG \subseteq (d+1)\text{-}DIG$.

2.5. RELATED FAMILIES OF INTERVAL GRAPH

Based on the fact that a graph is an interval graph if and only if it is chordal and its complement is a comparability graph, we have: A graph and its complement are interval graphs if and only if it is both a split graph and a permutation graph.

The interval graphs that have an interval representation in which every two intervals are either disjoint or nested are the trivially perfect graphs.

A graph has boxicity at most one if and only if it is an interval graph; the boxicity of an arbitrary graph G is the minimum number of interval graphs on the same set of vertices such that the intersection of the edges sets of the interval graphs is G.

I. PROPER INTERVAL GRAPHS

Proper interval graphs are interval graphs that have an interval representation in which no interval properly contains any other interval; unit interval graphs are the interval graphs that have an interval representation in which each interval has unit length. A unit interval representation without repeated intervals is necessarily a proper interval representation. Not every proper interval representation is a unit interval representation, but every proper interval graph is a unit interval graph, and vice versa. Every proper interval graph is a claw-free graph; conversely, the proper interval graphs are exactly the claw-free interval graphs. However, there exist claw- free graphs that are not interval graphs. An interval graph is called q-proper if there is a representation in which no interval is contained by more than q others. This notion extends the idea of proper interval graphs such that a 0-proper interval graph is a proper interval graph.

II. IMPROPER INTERVAL GRAPHS

An interval graph is called p-improper if there is a representation in which no interval contains more than p others. This notion extends the idea of proper interval graphs such that a 0-improper interval graph is a proper interval graph.

III. K – NESTED INTERVAL GRAPHS

An interval graph is k-nested if there is no chain of length k+1 of intervals nested in each other. This is a generalization of proper interval graphs as 1-nested interval graphs are exactly proper interval graphs.

IV. CIRCULAR ARC GRAPH

The intersection graphs obtained from collections of arcs on a circle are called circular-arc graphs. A circular-arc representation of an undirected graph G which fails to cover some point p on the circle will be topologically the same as an interval representation of G. Specifically, we can cut the circle at p and straighten it out to a line, the arcs becoming intervals. It is easy to see, therefore, that every interval graph is a circular-arc graph. The converse, however, is false. In fact, circular-arc graphs are, in general, not perfect graphs.

CHAPTER 3

3. TRAPEZOID GRAPHS

A trapezoid T_i is defined by four corner points $[a_i, b_i, c_i, d_i]$, where $a_i < b_i$ and $c_i < d_i$ *with* a_i, b_i lying on top line and c_i, d_i lying on bottom line of a rectangular channel. An undirected graph G= (V,E) with vertex set $V = \{v_1, v_2, \ldots, v_n\}$ and edge set $E = \{e_1, e_2, \ldots, e_n\}$ is called a trapezoid graph if a trapezoid representation can be obtained such that each vertex v_i in V corresponds to a trapezoid T_i and $(v_i, v_j) \in E$ if and only if the trapezoids T_i and T_j corresponding to the vertices v_i and v_j intersect. For simplicity the vertices $v_1, v_2, \ldots, v_n$are represented respectively by 1, 2, … , n . Thus the edge $(i, j) \in E$ if and only if T_i and T_j intersect in the trapezoid representation. Figure 15 illustrates a trapezoid graph and its trapezoid representation consisting of seven trapezoids $T_1, T_2, \ldots, T_7$. It is interesting to note that if $a_i = b_i$ and $c_i = d_i$ then the corresponding trapezoid T_i reduces to a straight line. So, in this way, if all the trapezoids reduce to straight lines the corresponding trapezoid graph reduces to nothing but a permutation graph. For simplicity, we assume that the corner points on the trapezoid representation are all distinct and so they can be given consecutive positions 1,2,…,n from left to right on both channels. In addition to this we may label these n trapezoids in increasing order of their right corner points on top channel, i.e. for two trapezoids T_i and $T_j, i < j$ if and only if b_i lies on the left of b_j.

Trapezoid graphs can be recognized in$O(n^2)$ time.Trapezoid graphs can be used for modelling a channel routing problem in VLSI, in a single-layer-per-net model. A channel consists of a pair of horizontal lines with points or terminals on each line numbered from 1 to n. All the terminals with the same label constitute a net. A routing is a connection of every net by wires inside the channel such that no two wires from different nets overlap (see Figure 16). A routing is allowed to use more than one layer. This channel routing problem is equivalent to the minimum colouring problem of a trapezoid graph, where each net is represented by a trapezoid. The single module k -planer (i.e. k layers) subset problem in VLSI is to assign maximum possible nets in k layers inside a channel in such a way that no two nets, assigned in any of the k layers overlap each other where the routing region is either a channel or bounded by a straight line and a solid module. A trapezoid graph with n vertices can be represented geometrically either by,

1. A set of n trapezoids drawn inside a rectangular channel or by,
2. A set of n segments drawn on a two dimensional plane or by,
3. A set of n boxes drawn on a two dimensional plane or by,
4. A permutation diagram π of n 2 lines drawn inside a channel.

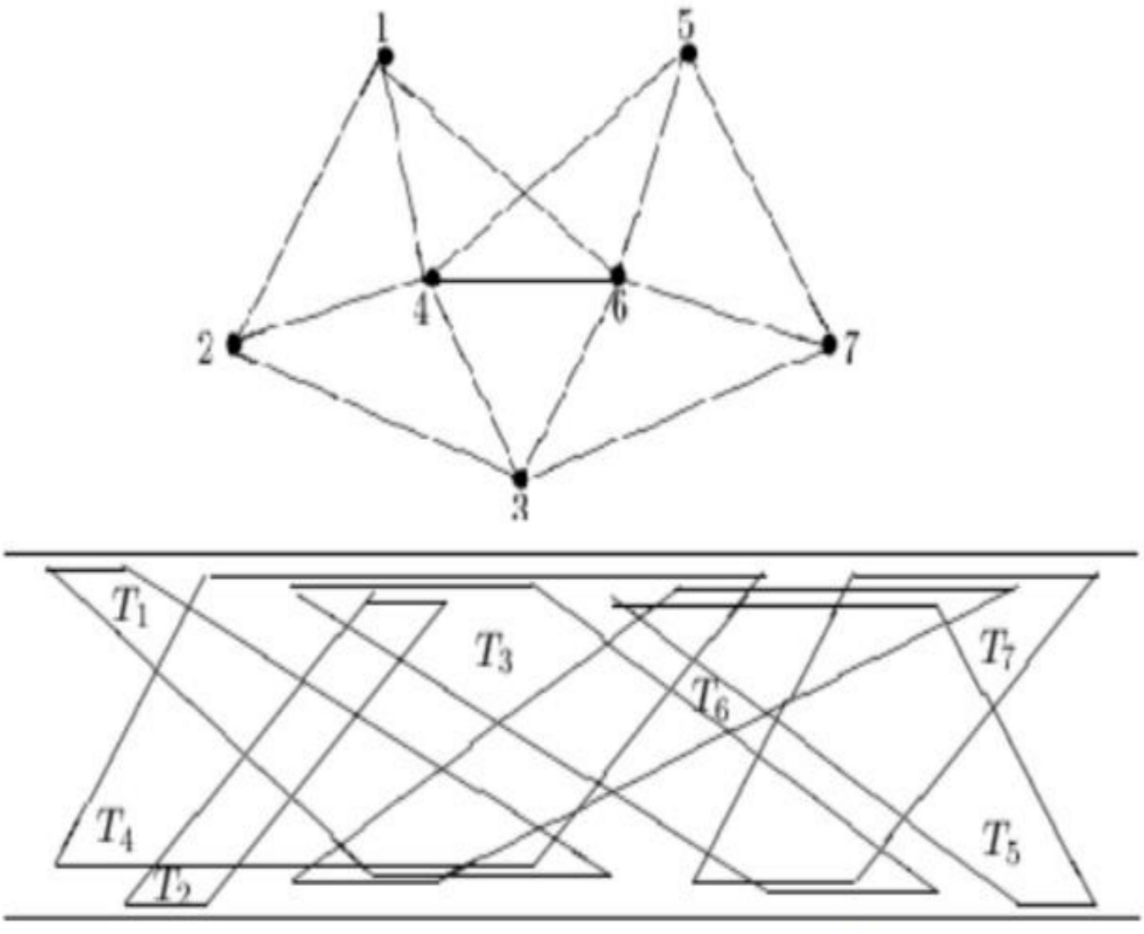


Figure 15: A trapezoid graph G and its trapezoid representation

3.1. TRAPEZOID REPRESENTATION

Let $T = \{T_1, T_2, \ldots, T_n\}$ be the set of n trapezoids where trapezoid T_i is represented in the trapezoid representation by four corner points $[a_i, b_i; c_i, d_i]$, $a_i, b_i (a_i < b_i)$ lying on the bottom line of a rectangular channel (see Figure 16). Without any loss of generality we assume the following:

1. A trapezoid contains four distinct corner points and that no two trapezoids share a common end point,

2. Trapezoids in the trapezoid representation and vertices in the trapezoid graph are one and same thing,

3. The trapezoids in the trapezoid representation T are indexed by increasing right end points on the top line, i.e. for any two trapezoids T_i and T_j in the trapezoid representation i <j if and only if $b_i < b_j$.

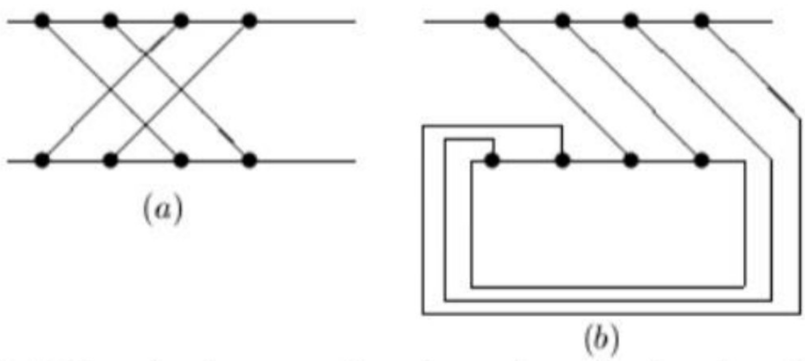


Figure 16: (a) A routing instance, where the routing region is a channel. (b) A routing instance, where the routing region is bounded by a straight line and a module.

This kind of ordering gives the following result which is quite useful in designing efficient algorithms.

In a trapezoid graph G if any three vertices i,j,k are such that i < j < k and (i,k) ∈ E , then either (i,j) ∈ E , or (j,k) ∈ E . This ordering is sometimes called as cocomparability ordering. In a cocomparability graph this ordering can be implemented in an $O(n^2)$ time. But for a trapezoid graph, this ordering can be implemented in only O(n) time with the help of its trapezoid representation .

The adjacency relation between any two vertices can be tested using the following result:

Let i and j be two vertices of a trapezoid graph G . Then two vertices i and j are not adjacent if and only if either

(i) $b_i < a_j$ and $d_i < c_j$ or (ii) $b_j < a_i$ and $d_j < c_i$. Otherwise the vertices i and j are adjacent.
Therefore, instead of storing a trapezoid graph, using adjacency matrix or adjacency list, one can store the trapezoid representation of the trapezoid graph using only n4 units of memory. The adjacency relation can be tested in O(1) time.

3.2. PERMUTATION REPRESENTATION

From trapezoid representation of a trapezoid graph a permutation diagram can be obtained with the use of the concept of vertex splitting. In a trapezoid graph, the vertex splitting process replaces each vertex v by two new vertices v_1 and v_2 where in the trapezoid representation the trapezoid representing v is replaced by two lines representing v_1 and v_2 respectively. Thus it may be seen that the trapezoid representation is evolving into a permutation graph representation.

The concept of vertex splitting was first introduced by Cheah and Corneil . They also proved that a graph is a trapezoid graph if and only if after an appropriate sequence of vertex splitting a permutation graph is obtained with a specific condition. Thus with the use of the concept of vertex splitting a trapezoid representation of n trapezoids will be transformed to a permutation diagram π of 2n lines.Trapezoid graphs are weakly chordal. All cut vertices of a trapezoid graph can be computed correctly in o(n) time.The very common problem in graph theory is all pairs shortest path problem. This problem has been solve for trapezoid graph . The time complexity needed to find all pairs shortest distances is stated below.The time complexity to find all pairs shortest distances on trapezoid graphs is O(n^2) .The time complexity to find the next-to-shortest path between any two vertices u and v in trapezoid graph is O(n^2) .All maximal cliques of a trapezoid graph can be generated in $O(n^2 + \gamma n)$ time, where n is the number of vertices of the graph and γ is the output size.

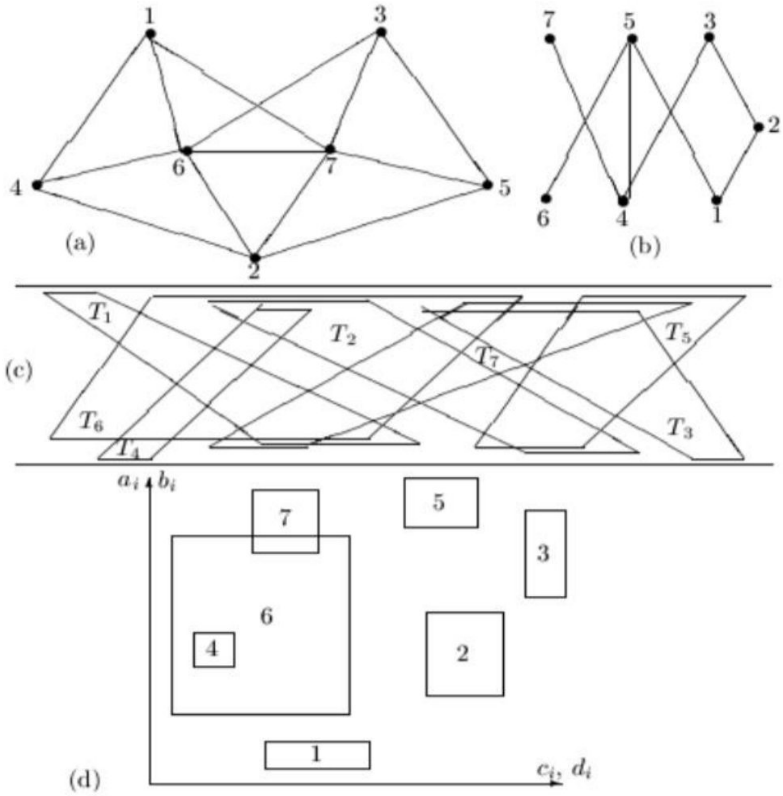


Figure 18: (a)The trapezoid graph of Figure 15 when vertices are given the dominance order P_G (b) The dominance order P_G (c) Its trapezoid representation and (d) Its box representation.

All maximal independent sets of a trapezoid graph G with n vertices can be computed in $O(\overline{nm} + \alpha n)$ time, where $\bar{m}$ and α denote respectively the total number of edges in G , the complement graph of the graph G , and the number of maximal independent sets in G .The maximum weight k -independent set problem can be solved for trapezoid graph in O(kn^2) time. In particular, maximum weight 2-independent set problem can be solved in O(n^2) time.

A clique of an undirected graph G=(V,E) is a complete subgraph of G , and a clique cover of G is a partition of V such that each set in the partition is a clique. A clique cover with the minimum cardinality $k(k \leq n)$ is known as a minimum clique cover (MCC). This number k is called the clique cover number.

The minimum clique cover is a well known NP-complete problem on general graphs . However, it can be solved in polynomial time for some special classes of graphs, like chordal graphs, interval graphs, circular-arc-graphs, circular permutation graph, etc.

A minimum clique cover of the trapezoid graph G can be computed in $O(n^2)$ time. The diameter and center of a trapezoid graph G=(V,E) can be computed in O(dn) time where d is the degree of vertex 1.

It is shown that every connected AT-free graph contains a dominating pair and the vertices which achieve the diameter are said to form a dominating pair. As trapezoid graphs belong to a subclass of AT-free graphs so the following results.

Every trapezoid graph has at least one dominating pair.The vertices u,v of a trapezoid graph form a dominating pair if $\delta(u, v) = diam(G)$.

A diameter path in a graph is a shortest path whose length is equal to the diameter of the graph. If P(u,v) is a diameter path in a trapezoid graph G , then P(u,v) is minimum connected dominating path.Dominating pairs and minimum connected dominating paths of a trapezoid graph can be computed in O(dn) time.

The set of all hinge vertices of a trapezoid graph with n vertices can be computed in O(n logn) time using O(n) space.

A spanning tree of a trapezoid graph can be computed in O(n) time.

The time complexity to find tree 4-spanner on trapezoid graphs is O(n) .The time complexity to find tree 3-spanner on trapezoid graphs is $O(n^2)$.

The minimum cardinality conditional covering set on trapezoid graphs can be determined in $O(n^2)$ time.

3.3. BOX REPRESENTATION

If $x = (x_1, x_2) and\ y = (y_1, y_2)$ are points in R^2, then x is said to be dominated by y , denoted as x< y , if x_i is less than y_i for i=1,2. . The order thus given between points in R^2 is called dominance order. This order can be extended to boxes, i.e. sets of the form $\{(x_1, x_2) \in R^2 : l_1 \le x_1 \le u_1, l_2 \le x_2 \le u_2\}$ where (l_1, l_2) is the lower corner and (u_1, u_2) is the upper corner of the box.

A box b dominates another box b′ if the lower corner of b dominates the upper corner of b′. Note that points may be understood as boxes where the lower and upper corner coincide. If one of the two boxes dominates the other we say that they are comparable. Otherwise they are incomparable. Now the vertices of a trapezoid graph may be represented by boxes with two boxes incomparable if and only if the corresponding vertices are joined by an edge.

In Figure 18, the trapezoid graph of Figure 15 is illustrated with a dominance order GP and its trapezoid representation, box representation are also given in the same figure. It is easy to observe that with the use of dominance order the vertices 6,7 4,5,2,3,1, in Figure 15 are renamed as vertices 7,5 6,3,4,2,1, respectively in Figure 18.

What makes the box representation useful is the additional dominance order on boxes that may be exploited by sweep line algorithms, where all computations are done in a single sweep.

3.4. BIOTOLERANCE REPRESENTATION

Bitolerance graphs are incomparability graphs of a bitolerance order. An order is a bitolerance order if and only if there are intervals I_x and real numbers $t_1(x)$ and $t_r(x)$ assigned to each vertex x in such a way that x < y if and only if the overlap of I_x and I_y is less than both $t_r(x)$ and $t_1(y)$ and the center of I_x is less than the center of I_y. In 1993, Langley showed that the bounded bitolerance graphs are equivalent to the class of trapezoid graphs.

3.5. RELATION TO OTHER FAMILIES OF GRAPHS

The class of trapezoid graphs properly contains the union of interval and permutation graphs and is equivalent to the incomparability graphs of partially ordered sets having interval order dimension at most two. Permutation graphs can be seen as the special case of trapezoid graphs when every trapezoid has zero area. This occurs when both of the trapezoid's points on the upper channel are in the same position and both points on the lower channel are in the same position.

Like all incomparability graphs, trapezoid graphs are perfect.

I. CIRCLE TRAPEZOID GRAPHS

Circle trapezoid graphs are a class of graphs proposed by Felsner et al. in 1993. They are a superclass of the trapezoid graph class, and also contain circle graphs and circular-arc graphs. A circle trapezoid is the region in a circle that lies between two non-crossing chords and a circle trapezoid graph is the intersection graph of families of circle trapezoids on a common circle. There is an$O(n^2)$ algorithm for maximum weighted independent set problem and an $O(n^2 logn)$ algorithm for the maximum weighted clique problem.

II. K-TRAPEZOID GRAPHS

k-Trapezoid graphs are an extension of trapezoid graphs to higher dimension orders. They were first proposed by Felsner,and they rely on the definition of dominating boxes carrying over to higher dimensions in which a point x is represented by a vector $(x_1, \dots, x_k)$. Using (k − 1)-dimensional range trees to store and query coordinates, Felsner's algorithms for chromatic number, maximum clique, and maximum independent set can be applied to k- trapezoid graphs in $O(nlog^{k-1}n)$ time.

CHAPTER 4

4. CIRCULAR – ARC GRAPHS.

A Circular-arc graph (see Figure 19) is an intersection graph of arcs on the circle. That is, every vertex is represented by an arc, such that two vertices are adjacent if and only if the corresponding arcs intersect. The arcs constitute a circular-arc model of the graph. Circular-arc graphs generalize interval graphs which are the intersection graphs of intervals on the line. A graph G is a circular-arc graph if it has a circular-arc model. Note that a circular-arc graph may have more than one model.

Circular-arc graphs can be used to model objects of a circular or a repetitive nature. Recent applications of circular-arc graphs are in modeling ring networks and item graphs of combinatorial auctions.

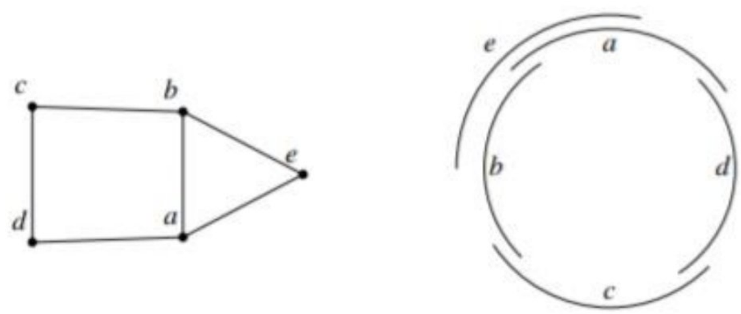


Figure 19: A circular-arc graph and a circular -arc model of it.The model and the graph are also proper circular-arc and unit circular-arc.

A circular-arc model in which no arc covers another arc is a proper circular-arc model. A graph G is a proper circular-arc graph if it has a proper circular-arc model. A circular-arc model in which all arcs are of the same length is a unit circular-arc model. A graph G is a unit circular-arc graph if it has a unit circular-arc model. Every unit circular-arc graph is a proper circular-arc graph.

There are four possible types of intersections between two arcs x and y.

• Cross: Arc x contains a single endpoint of arc y (see Figure 20 (a)).

• Cover the circle: Arcs x and y jointly cover the circle and each contains both endpoints of the other (see Figure 20 (b)).

• Arc x is contained in arc y.

• Arc x contains arc y (see Figure 20 (c)).

In addition, if x and y do not intersect then they are disjoint (see Figure 20 (d)).

If x and y either cross or cover the circle, we say that x and y overlap. In a proper circular-arc model, every pair of arcs that intersect, overlap each other. The relations between the different intersection types is illustrated in Figure 21.

Let u and v be a pair of adjacent vertices in G. The arc representing u can contain the arc representing v in a circular-arc model of G if and only if $N[u] \subseteq N[v]$. Such a relation between u and v is called a neighborhood containment relation. The arc representing u can cover the circle with the arc representing v in a circular-arc model of G if and only if for every $w \notin N[u]$ the arc representing w can be contained in the arc representing u.

If G is a circular-arc graph without a universal vertex, and without pair of vertices with the identical neighborhoods, then G has a circular-arc model ϱ such that for every pair of arcs x and y, if x can contain y then it does so in ϱ, and if x can cover there circle with y then it does so in ϱ. Such a circular-arc model is called a normalized model.

For convenience, we refer to the vertices of G as arcs even before we decide if G is a circular-arc graph and find a model for it. We say that two adjacent vertices intersect even before we have a model, because the arcs of adjacent vertices must intersect in every model. If G has a normalized model we say that v contains u when $N[u] \subset N[v]$, even before we have found a model. Additionally, if the arcs representing u and v cover the circle in a normalized model of G, we say that u and v cover the circle. We also say that two vertices overlap when they intersect but do not contain each other. If u and v overlap but do not cover the circle we would say that u and v cross.

To simplify we refer to the clockwise direction as right and to the counterclockwise direction as left, as we view them if we stand at the center of the circle.

In a circular-arc graph G with a circular-arc model ϱ, every vertex v ∈ V (G) has an arc in ϱ with two endpoints. We denote the left endpoint of v by $l(v)$ and the right endpoint of v by $r(v)$. If the arc x crosses the arc y and covers $r(y)$ then x overlaps the right side of y. Analogously, if the arc x crosses the arc y and covers $l(y)$ then x overlaps the left side of y.

The arcs in circular-arc models and proper circular-arc models may be either open or closed. For unit circular-arc models it makes a difference if an arc is open or closed. We assume that all arcs are closed. Our results also hold when all arcs are open. If arcs may be either open or closed then every proper circular-arc graphs is a unit circular-arc graph.

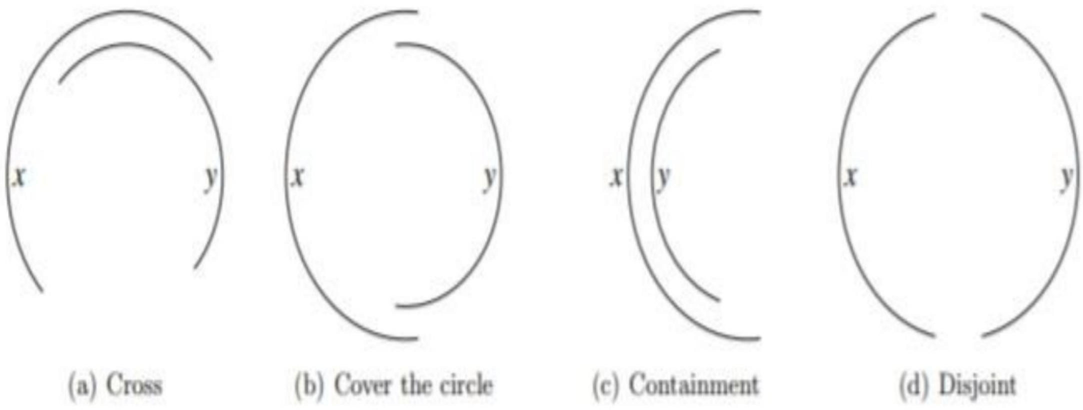


Figure 20: Intersection types of two arcs in circular- arc model.

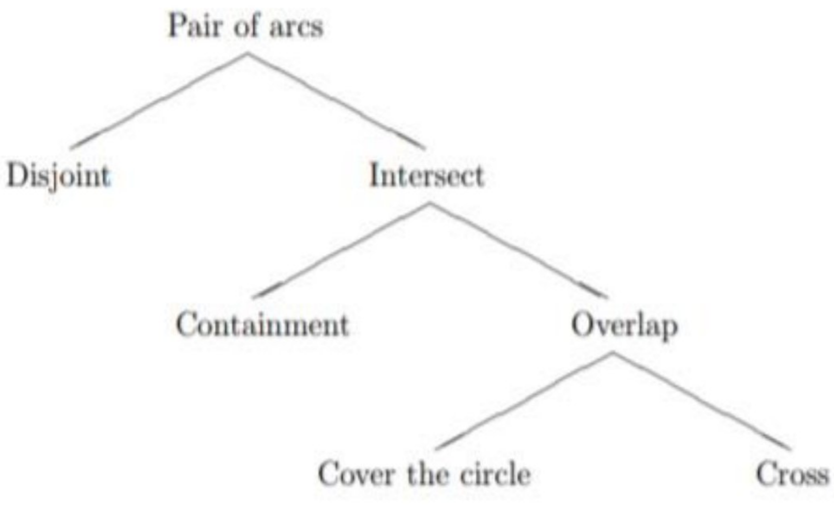


Figure 21: The relation between intersection types of arcs in a circular- arc model.

4.1. PROPER CIRCULAR – ARC GRAPHS

A circular-arc model in which no arc contains another arc is called a proper circular-arc model. A circular-arc graph that has a proper circular-arc model is a proper circular-arc graph.

Recognizing these graphs and constructing a proper arc model can both be performed in linear $\left(O(n+m)\right)$ time.

4.2. UNIT CIRCULAR – ARC GRAPHS

A circular-arc model in which all arcs are of the same length is called a unit circular-arc model. A circular-arc graph that has a unit circular-arc model is a unit circular-arc graph

The number of labeled unit circular –arc graphs on n vertices is given by $(n+1)\binom{2n-1}{n-1} - 2^{2n-1}$.

By definition, every unit circular-arc graph is a proper circular-arc graph.

4.3. HELLY CIRCULAR – ARC GRAPHS

G is a Helly circular-arc graph if there exists a corresponding arc model such that the arcs constitute a Helly family. Gavril gives a characterization of this class that implies an $O(n^3)$ recognition algorithm.

Joeris et al. gives other characterization of this class,which imply a recognition algorithm that runs in $O(n+m)$ time when the input is a graph. If the input graph is not a helly circular –arc graph ,then the algorithm returns a certificate of this fact in the form of a forbidden induced subgraph.

CONCLUSION

The goal of this talk has been to give you a feeling for the area of intersection graph , how it is relevant to applied mathematics and computer science, what applications it can solve, and why people do research in this area.

In the world of mathematics, sometimes I feel like a dweller, a permanent resident; at other times as a visitor or a tourist. As a mathematical resident, I am familiar with my surroundings. I do not get lost in proofs. I know how to get around .Yet, sometimes as a dweller, you can become jaded, lose track of what things are important as things become too routine. This is why I like different applications that stimulate different kinds of problems. The mathematical tourist, on the other hand, may get lost and may not know the formal language, but for him everything is new and exciting and interesting. I hope that my lecture today has given both the mathematical resident and the mathematical tourist some insight into the excitement and enjoyment of doing applied research in graph theory and algorithms.

REFERENCES

1. **Martin Charles Golumbic. -**“Algorithmic Graph Theory and Perfect Graphs”. (second edition)

2. **Martin Charles Golumbic , Irith Ben-Arroyo Hartman. -**“Graph Theory, Combinatorics and Algorithms – Interdisciplinary Applications.”

3. **P.C.Fishburn -“**Interval Graphs And Interval Order” (1985)

4. **Madhumangal Pal –** Intersection Graphs: An Introduction – Journal of Annals of Pure And Applied Mathematics.