Graph coloring Problem and its Applications

Raunak Tiwari - 180123038

Satyadev Badireddi - 180123041

Aim:

To study the state of the art of the problem, its algorithms and its applications.

Introduction to Graph Coloring :

- **The Graph Coloring Problem is defined as** : Given an undirected Graph G(V, E) ,each vertex is to be assigned a color(a label) in such a way that the colors on any pair of adjacent vertices are different.
- **Chromatic Number :** The smallest number of colors needed to color the graph G is called its chromatic number denoted as χ(G).
- **K-colorable :** A graph is said to be k-colorable if it can be properly colored using k colors.
- Vertex Covering Problem is one of the classic NP-complete problems. It is NP-Complete to determine if the given graph is k- colorable for all values of k excep for 0,1 and 2. For k > 3, k colorings exist on planar graphs by the four color theorem.
- Brooks theorem states that in connected, simple graphs which are not complete or odd cycles, the chromatic number is less than the maximum vertex degree of the graph.

Background :

The problem originated in map coloring when Frances Guthrie stated the four color conjecture i.e., that four colors were sufficient to wholly color a map without any pair of regions sharing a border having the same color. The four color theorem was proven in1976 by Kenneth Appel and Wolfgang Haken.

Study of Algorithms for graph Coloring

Unfortunately, there is no efficient algorithm available for coloring a graph with minimum number of colors as the problem is a known NP complete problem. There are approximate algorithms to solve the problem though.

We shall refer to the research paper named "*A Performance comparison of graph coloring algorithms*" mentioned in reference no. 2. In this paper, we compared diffe rent graph coloring algorithms such as First Fit Algorithm, Welsh – Powell Algorith m, Largest Degree Ordering, Recursive Largest First Algorithm, Degrees of Satura tion Algorithm. Of all these algorithms, Welsh Powell algorithm finds out the best s olution in the shortest time.

Welsh - Powell Algorithm

It is a heuristic algorithm that uses an iterative greedy approach to color the graph.

Pseudo Code:

We have a graph G(V, E) with n vertices from $v_1, v_2,$ to v_n. Colors are ordered in a list C.

1. Compute the degree(number of adjacent vertices) of each vertex represented by d(v_i).
2. List the vertices in the descending order of the degree d(v_i).
3. The non colored vertex with the highest degree is colored with the first color in C
4. The list is traversed and the vertices which are not adjacent to the colored vertex above are allocated the same color and all these non adjacent vertices are delet ed from the list.
5. Steps 3 and 4 are iterated until all the vertices have been colored.

Analysis:

The greedy coloring takes at most $max_i(\min(d(x_i) + 1, i))$ colors to fill the vertices where $d(x_i)$ is the degree of the vertex v_i in the graph.

Time Complexity of the algorithm is O(n^2) where n is the no. of vertices in the graph.

Applications of graph coloring

1. Sudoku solving :

Sudoku can be considered as a variation of a graph coloring problem. In this reduced problem, the graph has 81 vertices to be filled and every cell in the sudoku represen ts a vertex. Two vertices are adjacent if they are in the same row or column or block.

2. Exam timetabling problem

Problem Statement :

We want to make an exam schedule for a university, we have list of different subjects and students enrolled in every subject. Many subjects would have common students (of same batch, some backlog students, etc).

How do we schedule the exam so that no two exams with a common student are sched uled at same time? How many minimum time slots are needed to schedule all exams?

Relationship to a Graph :

This problem can be represented as a graph where every vertex is a subject and an edge between two vertices mean there is a common student. So this is a graph colorin g problem where minimum number of time slots is equal to the chromatic number of th e graph.

3. Existence of solution to class-teacher timetabling problem (CTTP)

(Studied with the help of research paper mentioned in reference no. 3)

Problem Statement :

The (class-teacher) timetable problem is defined as follows: Given

a set of teachers T = $\{t_i\}$ i = 1 , . . . , α,

a set of classes C = $\{c_j\}$ j = 1 , . . . , β,

a set of hours H = $\{h_k\}$ k = 1 , . . . , σ,

and an $\alpha \times \beta$ requirements matrix R = $[r_{ij}]$ where $r_{ij} \geq 0$, and r_{ij} equals the number of hours teacher t_i is to meet class c_j.

Unavailability Constraints :

The unavailability constraints are described by matrices D and E. D = $[d_{ik}]$ is an α × σ matrix with d_{ik} = 1, if teacher t_i is unavailable at hour h_k; and d_{ik} = 0 otherwise. Similarly, E = $[e_{jk}]$ is an β × σ matrix with e_{jk} = 1, if class c_j is unavailable at hour h_k; and e_{jk}= 0 otherwise.

Preassigned Meetings :

The preassigned meetings are described by sets P_{ij}, i = 1 , . . . , α; j = 1, . . ., β. The sets P_{ij} are defined by P_{ij} = {h_k | h_k ∈ H and h_k such that teacher t_i is to meet clas s c_j at hour h_k}. P_{ij} = ∅ if r_{ij} = 0, or if there are no preassigned meetings involving b oth teacher t_i and class c_j.

Is it possible to form a timetable without violating any constraints?

Relationship to a Graph :

Now we will try to transform CTTP into a graph coloring problem such that each vertex represents a lesson between a teacher and a class, an edge joining two vertices indicates that the respective associated lessons cannot be scheduled at the same hour and each color represents one hour of the corresponding timetable. So, solution should exist if and only if the graph is σ -colorable.

Let G be a graph with a set of vertices V and a set of undirected edges E.

A meeting of teacher t_i and class c_j will be denoted by V_{ij}. If r_{ij} > 0, then teacher t_i and class c_j must meet r_{ij} times. Let the p^{th} meeting, 1 ≤ p ≤ r_{ij}, of teacher t_i and class c_j be denoted by v_{ij}^p. Let V_{ij} = { v_{ij}^1, ... , $v_{ij}^{r_{ij}}$} represent these r_{ij} meetings of teacher t_i and class c_j; V_{ij} = ∅ if r_{ij} = 0.

Let $V_j = \cup_{i=1}^{\alpha} V_{ij}$ is the set of all meetings of class c_j and | V_j | =$\sum_{i=1}^{\alpha} r_{ij}$. Clearly, | V_j | must be less than or equal to σ in order for a solution to exist.

Let $V = \cup_{j=1}^{\beta} V_j$ is the set of all meetings of classes c_j ∈ C and all teachers t_i ∈ T. V is the set of all vertices of graph G, which contain vertex corresponding every meetings between any class and teacher.

Now define the set of edges E of the graph G as E = $E^1 \cup E^2$ where,

$E^1 = \cup_{j=1}^{\beta} E_j, \; where \; E_j = \{(v_1, v_2) \; v_1, v_2 \in V_j, v_1 \neq v_2\}$ and

$E^2 = \cup_{i=1}^{\alpha} E_{ij_1j_2}, \; where \; E_{ij_1j_2} = \{(v_1, v_2) \; v_1 \in V_{ij_1}, v_2 \in V_{ij_2}\}$ for all $j_1 \neq j_2$ with $r_{ij_1}, r_{ij_2} > 0$ and $E_{ij_1j_2}$ = ∅ otherwise.

The edges in set E^1 correspond to the following requirements in the timetable problem :

1. Class c_j must not meet more than one teacher during any given hour.
2. Class c_j must meet every teacher the required number of hours and teacher t_i must meet every class the required number of hours.

Similarly, the edges in the set E^2 correspond to the following requirements in the timetable problem :

1. Teacher t_i must not meet more than one class during any given hour.

- **Solution to the timetable problem having no unavailability constraints nor preassigned meetings :**

There exists a solution to the timetable problem having no unavailability constraints nor preassigned meetings if, and only if, there exists a σ -coloration for the graph G.

- **Solution to the timetable problem with unavailability constraints and no preassigned meetings :**

There exists a solution to the timetable problem with unavailability constraints described by matrix D and no preassigned meetings if, and only if, there exists a σ -coloration for the graph G such that for every $d_{ik} = 1$, no vertex in the set V_{ij} , j = 1 , . . ., β is assigned to color h_k.

There exists a solution to the timetable problem with unavailability constraints described by matrix E and no preassigned meetings if, and only if, there exists a σ -coloration for the graph G such that for every $e_{jk} = 1$, no vertex in the set V_{ij} , i = 1 , . . .,α is assigned to color h_k.

- **Solution to the timetable problem with no unavailability constraints and with preassigned meetings :**

There exists a solution to the timetable problem with no unavailability constraints and with preassigned meetings described by set P_{ij} if and only if, there exists a σ -coloration for the graph G such that corresponding to every nonempty set P_{ij} $=\{h_{ij}^1, . . . , h_{ij}^k\}$, vertices $v_{ij}^1, . . . , v_{ij}^k$ are assigned to color $h_{ij}^1, . . . , h_{ij}^k$, respectively.

References :

1. Link : https://en.wikipedia.org/wiki/Graph_coloring
2. Link : https://www.researchgate.net/publication/309585874_A_Performance_Comparison_of_Graph_Coloring_Algorithms
3. Link : https://dl.acm.org/doi/pdf/10.1145/361082.361092