STAR COLORING

Dakshraj Sadashiv Kashid (12040500)
Suresh Bairwa (12041530)

Subject: : **Advanced Graph Algoithms**

Introduction

A star vertex coloring of a graph G is a proper vertex coloring in which every path on four vertices uses at least three distinct colors. In other words, it is a proper coloring such that any path on four vertices in G is not 2-colored. In a star coloring, the induced subgraphs formed by the vertices of any two colors has connected components that are star graphs. The star chromatic number $\chi_s(G)$ of G is the least number of colors needed to star color G.

Complexity

It is *NP*-complete to determine whether $\chi_s(G) \leq 3$, even when G is a graph that is both planar and bipartite.

1. Distance to Co-Cluster:

We show that the star coloring is fixed-parameter tractable when parameterized by distance to co-cluster.
Decision version: $\chi_s(G) \leq k_s$

So we for a given graph G, we have a set $S \subseteq V(G)$, such that $V(G)/S$ induces a co-cluster graph. Let the $|S| \leq k$, the parameter k is called as distance to co-cluster. To show that the star coloring is FPT we have to find the star coloring of G in $f(k)n^{O(1)}$ time.Now the graph $G = S \bigcup \{I_1, I_2, .., I_p\}$, where each $I_i\ \forall i \in [1, p]$ is an independent set. And all edges are of the form:$uv : u \in I_i; v \in I_j : i \neq j$.

Lets say we have the star coloring of the set S, *(using brute force method in time* $k^k * \binom{k}{4}$). Let the $\chi_s(S) = l$ where $l \leq k$. Now to minimize the number of colors, we have to check if we can color the independent sets using same l colors or we need more. Note the following observations:

1. Case 1: We can color an independent set using l colors. WLOG lets say we can color I_i from previously used colors.
   a) If no color is repeated in I_i then, we check the same for another I_i. We will discard the colors used in coloring of I_i as every vertex in I_i is adjacent to all the other vertices in $I_j : i \neq j$.
   b) If there is repetition of color in I_i, then we have to color all the vertices of remaining $I_j : j \neq i$ with distinct color otherwise we will have bicolored P_4.
2. Case 2: In case we cannot color the independent set using previously used colors, then we can color all the vertices in I_i using a single color and the follow-up coloring will be the same case as Case1.b.

According to the above observation it is suffice to find the star coloring of $G[S \bigcup I_i]$. In case of no-repetition we will follow the same procedure for the rest the independent sets, else we will distinctly color all the remaining vertices.

Theorem 1: The q-variable Integer Linear Programming Feasibility problem can be solved using $O(q^{2,5q+o(q)}n)$ arithmetic operations and pace polynomial in n, where n is the number of bits of the input.

Now we know the coloring of S, we will formulate ILP for the coloring of the I_i. We will partition the the vertices into m sets $V_1, V_2, ..V_m$, such that $u, v \in V_i; N_S(u) = N_S(v)$. We can see that $m \leq 2^k$ and color all the vertices in each partition set with single color as they form an independent set and every P_4 will contain only one vertex from one partition set.

1.1. Formulation of ILP:

We will have the variable $x(v, c)$ for each partition set v and every available colors c. $x(v, c) = 1$ if the partition set v is colored with color c else it is 0. And n_c which defines the use of color c, if any vertex is colored c, the $n_c = 1$ else it is 0. Notation $col(v)$ is used for the color used for a pre-colored vertex $v \in S$.
Optimization Goal: minimize $\sum_c n_c$ i.e minimize the number of colors used.
Constraints:

1. Every set should be singly colored: $\forall v$, $\sum_c x(v, c) = 1$. And the color used must set n_c to 1, so $x(v, c) \leq n_c$.
2. Constraint for proper coloring. Discard the colors of the neighbors.

$$\forall u \in \bigcup_{a \in v} N_S(a) \rightarrow x(v, col(u)) = 0$$

.

3. Constraint for star coloring:

$$\forall ab \in E(S \cup I_i); a \in V_i; b \in V_j : i \neq j \rightarrow x(V_i, col(b)) + x(V_j, col(a)) \leq 1$$

.

Now the total number of variables are $k * 2^k + k$. So using Theorem 1 we can find the star coloring of I_i in time $O(q^{2,5q+o(q)}n)$ where $q = k * 2^k + k$. Now if there is no repetition then we will chose another independent set I_j and find the coloring using the same approach as above only with the additional constraint. We cannot use the colors used for I_i, thus additional constraint in the formulation will be:

$$\forall color \in STAR\_COLORING(I_i); \forall v \in I_j \rightarrow x(v, color) = 0$$

. Now to check P_4 involving previously colored independent set (I_i),

$$\forall ab \in E(S); \forall u \in N_{I_i}(a) : col(u) = col(b) \rightarrow x(v, col(a)) = 0$$

. Thus with these additional constraints we can achieve the star coloring of the independent set I_j. We can do this iteratively until a repetition appears. In case of repetition add the constraint $\sum_v x(v,c) = 1$. The remaining case is if the ILP is an NO instance then we introduce a new color and use it to color the all the vertices of I_j. We keep doing this until we find the star coloring of the G or there is NO instance and we can no longer add new colors i.e we have $\chi_s(G) > k_s$.

2. Star Coloring of Interval Graph

Interval graph: A graph whose nodes can be represented as an interval on real line and endpoints of each edge have an non-empty intersection in their corresponding intervals. Star coloring on interval graph is P-class problem. We can calculate the star coloring for the interval graphs using a polynomial time algorithm. The algorithms is as follow:

1. Initialize two list one AC (available colors) and NA(not available). Both the lists are empty initially. And a list L(list of colored vertices).
2. Mark the endpoints of the interval graph of the interval graph and sort them in the ascending order, in case starting and endpoint are same then keep starting point first in order.
3. Iterate through the list of sorted endpoints and start coloring based on AC:
   a) If we are at starting point of vertex v, in case if AC is empty introduce new color and color the corresponding vertex. Push this color to NA.
   b) Else if AC is non-empty iterate through all the ordered triplets. For any triplet (a, b, c) if $ab, bc, cv \in E(G)$ and $color(c) = color(a)$, then add color(b) to a list P. Choose smallest color from $AC - P$ to color the vertex v, if $AC - P = \phi$ then introduce a new color to color v and push this color to NA.
   c) In case you are at an endpoint of v, delete the $color(v)$ from NA. If there are no more $color(v)$ element in NA, push this color to AC.

Here the color is being made available for the coloring of new node only when there is no adjacent vertices have same color. The list AC and NA maintain this. As after the endpoint of a vertex that vertex will not be adjacent to any other vertex. And the color is made available only when we have covered all the endpoints of that particular color. Thus it is the proper coloring.
Now consider 3.b in this it check every possible P_4 incident to the vertex to be colored and discard the color in such way to avoid bi-colored P_4. Now from remaining available color we color the vertex. Thus it is the star coloring of G.
We claim that this assignment is the optimal star coloring for G. Because we introduce minimum number of colors to avoid bi-colored P_4 as after the assignment of color to a vertex there will be no further P_4 such that the vertex is an right endpoint and it will be intermediate vertex for uncolored vertices.

The algorithm is of order $n^{O(1)}$ and we obtain the optimal star coloring of the interval graph.

3. Split graph

A split graph is of the form $G = (K, I)$ where K is clique and I is an independent set. $\chi_s(G) \leq k+1$ as we have to use k distinct colors to color the clique and we can color I with one additional color else we will need at least k colors to color the clique and reuse them to color I.

3.1. Problem:

Therefore $|K| \leq \chi_s(G) \leq |K|+1$. Now we want to determine the case when its $|K|$ or $|K|+1$. Or determining this is NP complete.

3.2. Observation

For any two pair of vertices $u, v \in I$ if $N(u) \cup N(v) = K$, then we need $k+1$ colors. Because consider the path uk_1k_2v here if we want to use k colors $col(u) \in \bigcup_{ver \in N(v)} col(ver)$ and $col(v) \in \bigcup_{ver \in N(u)} col(ver)$. Thus there exist a bicolored P_4.
Every P_4 involving at least 3 vertices from K will be at least 3 colored as vertices in clique are differently colored. Every P_4 involving independent set will be of form $i_1k_1i_2k_2$ or $i_1k_1k_2i_2$.

1. *Case1 ($i_1k_1i_2k_2$):* This type of P_4 is will be at-least 3-colored when we just take care of the adjacency. Because $k_1 \neq k_2$, $i_2 \neq k_1$ and $i_2 \neq k_2$ thus k_1, k_2, i_2 are all distinctly colored.
2. *Case2 ($i_1k_1k_2i_2$):* In this once we assign color c_1 to i_1 where $col(k_1) = c_1$, we need to make sure that for all uncolored neighbors of k_1 in independent set they should not be assigned any colors from $\bigcup_{ver \in N(i_1)} col(ver)$. Otherwise we will have bicolored P_4 of form *abab*.

Based on above observation we formed a polynomial algorithm to decide whether G is k colorable or not. And we claim it will determine the case when $\chi_s(G) = |K|$ or $\chi_s(G) = |K|+1$. The algorithm is as follow:

1. We will color the clique k distinct colors and then we will iterate through all possible colors and starting vertex. Initialize a NA(not available colors) lists for all vertices.
2. Let v_s be the starting vertex. Now we will color the v_s based on its adjacent colored vertices. We will choose the color c_i whose corresponding vertex k_i in K have minimum neighbor except v_s(in case) in I.
3. Upon coloring a vertex with c_i. We will add $col \in \bigcup_{ver \in N(v_s)} col(ver)$ to NA list of all the neighbors of k_i. (Infer Case2 above for the reason).

4. For every other vertex we will discard all the adjacent colors and colors from NA list. And from the remaining colors we will choose according to Step2 criterion.
5. If for any vertex we exhaust all the available colors then we stop this iteration here and move to next iteration.

If for a single iteration gets complete then we are able to star color the G and if not then we claim that it is $k+1$ colorable.

4. Classes of Bipartite Graphs.

The problem is NP-complete on bipartite graphs. Thus we will find the complexity class for star coloring of special bipartite graphs. Any bipartite graph which is $P_4 - free$ have $\chi_s = 2$. For the following sections we will consider graphs which has no isolated vertices, and is not $P_4 - free$.

4.1. Convex Bipartite Graph:

In convex bipartite graph $G(A, B)$ there is ordering of vertices in B let it be $b_1, b_2, ..b_m$. Then for every vertex $a \in A$, if $ab_i, ab_j \in E(G)$, then $ab_i i+1, ab_{i+2}..ab_{j-1} \in E(G)$.

Reduction If for any vertex $a \in A$, such that $\exists b \in B; b \in N(a)$ and $deg(b) > 1$. Then remove all 1-degree nodes from $N(a)$ as you can color them with $col(b)$.
Otherwise $\nexists b \in B; b \in N(a)$ and $deg(b) > 1$, then we have all degree one neighbors of a. Keep only single vertex from $N(a)$ remove the rest of vertices from $N(a)$ as they all can be colored with same color.

After above reduction let the new graph be $G = (A, B)$. *We keep the same notation as original graph because all the removed vertices have definite coloring.*

Claim 1: $\chi_s(G) \leq min(min(|A|, |B|) + 1, \delta_A + 1)$ where δ_A is the maximum degree of any vertex in A.
Proof:
Case1: $min(|A|, |B|) + 1 \leq \delta_A + 1$, WLOG $|A| > |B|$ we will color A with distinct colors and B with a single color. This is valid assignment as it is clearly a proper coloring and every path is 3-colored : $abac$.
Case2: Otherwise we have to distinctly color the neighborhood of $v \in A$. As maximum size for any $a \in A; |N(a)| = \delta_A$. And we will color the all the vertices of A with one additional color. As for every edge $ab \in E(G); col(a) \neq col(b)$ thus it is a proper coloring. This is valid star coloring as every P_4 will be of form $a_1b_1a_2b_2$, as $b_1, b_2 \in N(a_2)$ and we colored every neighborhood with distinct colors, implying $col(b_1) \neq col(b_2)$ and a_2 is given an additional color $col(a_2) \neq col(b_1), col(b_2)$ thus every P_4 is atleast 3-colored. Hence it is valid star coloring.
Now we prove that our bound is tight. For the first case take complete bipartite graph, as it is also a convex bipartite graph and $\chi_s(G) = (min(|A|, |B|) + 1$. For the second case consider a convex bipartite graph where in star coloring we have $a \in A; b_1, b_2 \in N(a)$:

$col(b_1) = col(b_2)$, then consider any path $a_1 b_1 a_2 b_2$ (we can construct such a path without disturbing convex nature of the graph) we have to distinctly color a_1 and a_2, thus you will need atleast $\delta_A + 1$ colors this graph, else we can construct a bicolored P_4.

4.2. Biconvex Bipartite Graph:

A biconvex graph $G = (A, B)$ is convex on both A and B. As every biconvex bipartite graph is also a convex bipartite graph we will have the same results from previous section. But as it is convex on both A and B we will apply reduction on both the sets and modify the bound to:

$$\chi_s(G) \leq min(min(|A|, |B|) + 1, \delta_A + 1, \delta_B + 1)$$

. *Proof:* Follows from Claim1.