Vertex coloring is the most common graph coloring problem. The problem is, given m colors, find a way of coloring the vertices of a graph such that no two adjacent vertices are colored using same color. The other graph coloring problems like **Edge Coloring** (No vertex is incident to two edges of same color) and **Face Coloring** (Geographical Map Coloring) can be transformed into vertex coloring (Preeti Gupta, 2014).
Chromatic Number: The smallest number of colors needed to color a graph G is called its chromatic number. For example, the following can be colored minimum 3colors (http://www.geeksforgeeks.org/graph-coloring).

2. Basic Results on Graph Coloring

2.1 Vertex Coloring

Let $G = (V, E)$ be a graph of order n without loops and multiple edges, V is a set of n vertices and E is a set of m edges.

Definition I (Kubale, 2004):

- A disjoint collection of independent sets that cover all the vertices in the graph.
- A partition $V = I_1 \cup I_2 \cup \ldots \cup I_\chi$ such that Ij is an independent set for all $1 \leq j \leq \chi$.

Definition II (Kuhn, 2009):

- An assignment of colors to the vertices such that two adjacent vertices are assigned different colors.
- A function $c : V \rightarrow \{1, \ldots, \chi\}$ such that if $(u, v) \in E$ then $c(u) \neq c(v)$.

Observation: Both definitions are equivalent.

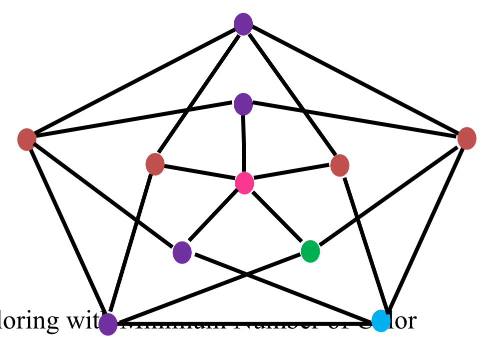


Fig.1 Coloring with Minimum Number of Color

2.2 The Vertex Coloring Problem

The optimization problem: Find a vertex coloring with minimum number of colors.
Notation: The chromatic number of G, denoted by $\chi(G)$, is the minimum number of colors required to color all the vertices of G.
Hardness: A very hard problem (an NP-Complete problem).
Hardnes of vertex coloring:

- It is NP-Hard to color a 3-colorable graph with 3 colors.
- It is NP-Hard to construct an algorithms that colors a graph with at most $n^{\varepsilon}\,\chi(G)$ colors for any constant $0 < \varepsilon < 1$(Kuhn, 2009).

3. Known Algorithms for Vertex Coloring

- There exists an optimal algorithm for coloring whose running time is $O\,(mn\,(1 + 3^{1/3})^n) \approx \text{mn}1.442^n$.
- There exists a polynomial time algorithm that colors any graph with at most $O(n/\log n)\chi(G)$ colors.
- There exists an algorithm that colors a 3-colorable graph with $O(n^{1/3})$ colors(Kuhn, 2009).

3.1 Properties of Vertex Coloring

Observation: $\omega(G) \leq \chi(G)$.

Because in any vertex coloring, each member of a clique must be colored by a different color.

Observation: $\chi(G) \geq \left\lceil \frac{n}{I(G)} \right\rceil$. A pigeon hole argument: the size of each color-set is at most $I(G)$.

Example for : $\omega(G) = \chi(G)$.

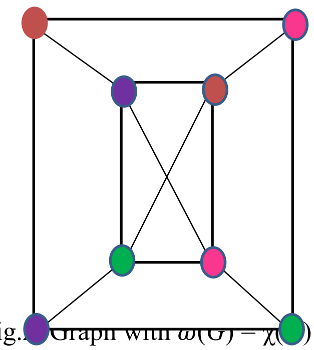


Fig.2. Graph with $\omega(G) = \chi(G) = 4$

Where $\omega(G) = 4$ and $\chi(G) = 4$. Every member of the only clique of size 4 must be colored with a different color.

Example for $\chi(G) > \omega(G)$

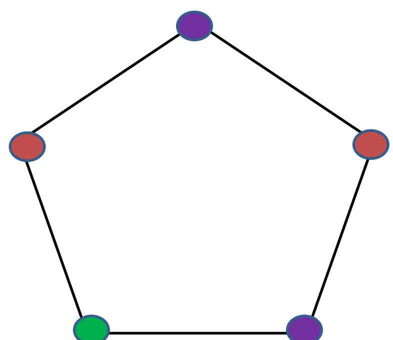


Fig.3. Graph with $\omega(G) = 2$ and $\chi(G) = 3$

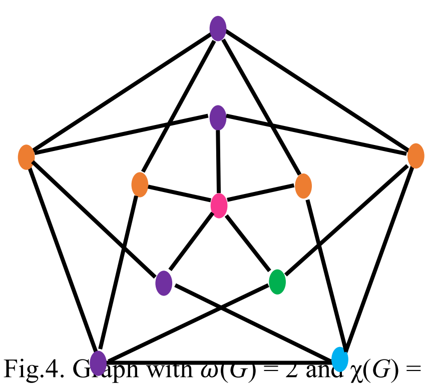


Fig.4. Graph with $\omega(G) = 2$ and $\chi(G) = 4$

3.2 Triangle-Free Graph Construction

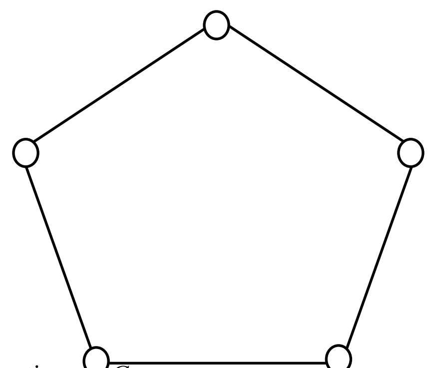
For $\chi(G) >> \omega(G)$

Theorem: For any $k \geq 3$, there exists a triangle-free graph G_k ($\omega(G_k) = 2$) for which $\chi(G_k) = k$.

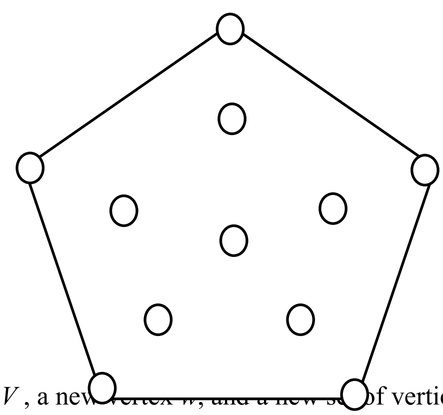
A construction: G_3 and G_4 are the examples above.
Construct G_{k+1} from G_k.

- Let $V = \{v_1, \ldots, v_n\}$ be the vertices of G_k.
- The vertices of G_{k+1} include V , a new vertex w, and a new set of vertices $U = \{u_1, \ldots, u_n\}$ for a total of $2n+1$ vertices.
- The edges of G_{k+1} include all the edges of G_k, w is connected to all the vertices in U, and $u_i \in U$ is connected to all the neighbors of v_i in G_k.

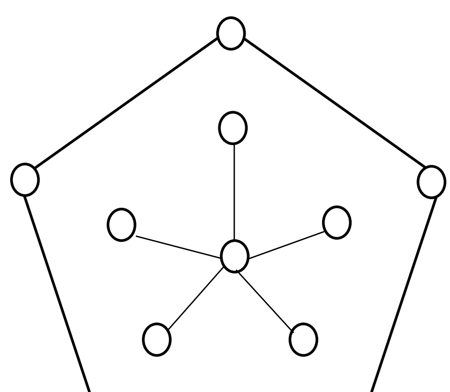
Constructing G_4 from G_3 (Construct G_{k+1} from G_k.)



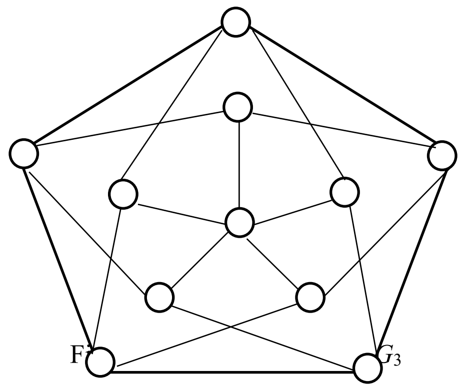
Steps 1: Let $V = \{v_1, \ldots, v_n\}$ be the vertices of G_k.



Steps 2: The vertices of G_{k+1} include V , a new vertex w, and a new set of vertices $U = \{u_1, \ldots, u_n\}$ for a total of $2n+1$ vertices.



Steps 3: The edges of G_{k+1} include all the edges of G_k, w is connected to all the vertices in U, and $u_i \in U$ is connected to all the neighbors of v_i in G_k.



4. Triangle-Free Graph Coloring

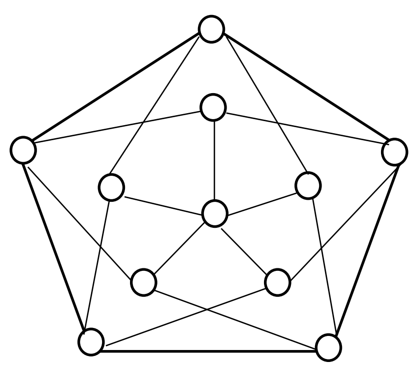
G_{k+1} is a Triangle-Free Graph

Let U is an independent set in G_{k+1} and therefore there is no triangle with at least 2 vertices from U, w is not adjacent to V and is adjacent to the independent set U. Therefore w cannot be a member in a triangle. V contains no triangles because G_k is a triangle-free graph. The remaining case is a triangle with 1 vertex $u_i \in U$ and 2 vertices v, $v' \in V$. This is impossible since u_i is connected to the neighbors of v_i and therefore the triangle u_ivv' would imply the triangle v_ivv' in the triangle-free graph G_k.

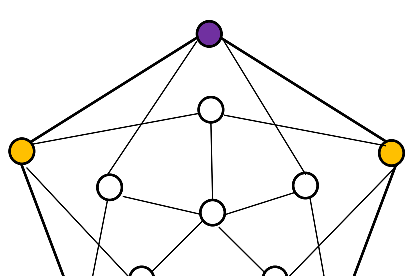
$\chi\,(G_{k+1}) \leq k + 1$

- Color the vertices in V with k colors as in G_k.
- Color u_i with the color of v_i. This is a legal coloring since u_i is connected to the neighbors of v_i.
- Color w with a new color.

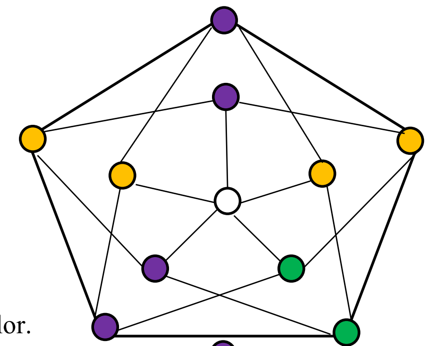
Coloring G_4 :



Steps 1: Color the vertices in V with k colors as in G_k.



Steps 2: Color u_i with the color of v_i. This is a legal coloring since u_i is connected to the neighbors of v_i.



Steps 3: Color w (red) with a new color.

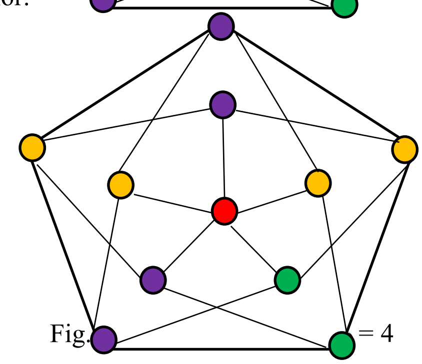


Fig. = 4

$\chi(G_{k+1}) > k$

Assume that G_{k+1} is colored with the colors $1, \ldots, k$. Let the color of w be k, Since w is adjacent to all the vertices in U it follows that the vertices in U are colored with the colors $1, \ldots, k-1$. Color each v_i that is colored by k with the color of u_i. This produces a legal coloring of the G_k subgraph of the G_{k+1} graph because u_i is adjacent to all the neighbors of v_i and the set of all the k-colored v_i is an independent set, a contradiction since $\chi(G_k) = k$.

Perfect Graphs

- In a perfect graph $\chi(G) = \omega(G)$ for any "induced" subgraph of G.
- Coloring is not Hard for perfect graphs.
- The complement of a perfect graph is a perfect graph.
- Interval graphs are perfect graphs.

The Trivial Cases

Observation: A graph with $n \geq 1$ vertices needs at least 1 color and at most n colors. $1 \leq \chi(G) \leq n$.
Null Graphs: No edges ⇒ 1 color is enough. $\chi(N_n) = 1$.
Complete Graphs: All edges ⇒ n colors are required. $\chi(K_n) = n$.

The Easy Case

Theorem: The following three statements are equivalent for a simple undirected graph G:

1. G is a bipartite graph.
2. There are no odd length cycles in G.
3. G can be colored with 2 colors.

Proof:(1 ⇒ 2)
- The vertices of G can be partitioned into 2 sets A and B such that each edge connects a vertex from A with a vertex from B.
- The vertices of any cycle alternate between A and B.
- Therefore, any cycle must have an even length.

Proof: 2 ⇒ 3
- Run BFS on G starting with an arbitrary vertex.
- Color odd-levels vertices 1 and even-level vertices 2.
- Tree edges connect vertices with different colors.
- In a BFS there are no forward and backward edges and a cross edge connects level ℓ with level ℓ' only if $|\ell - \ell'| \leq 1$.
- If $\ell = \ell' + -1$ then the cross edge connects vertices with different colors.
- If $\ell = \ell'$ then the cross edge closes an odd-length cycle contradicting the assumption.
- Thus, all the edges connect vertices with different colors.

Proof: 3 ⇒ 1
- Let A be all the vertices with color 1 and let B be all the vertices with color 2.
- By the definition of coloring, any edge connects a vertex from A with a vertex from B.
- Therefore, the graph is bipartite.

5. Conclusions

The main motive of this paper is to present the new Graph Coloring Algorithm with its space and time complexity, this algorithm can be applied to so many applications based on Graph coloring.

Refereces

http://en.wikipedia.org/wiki/graph coloring.

http://www.geeksforgeeks.org/graph-coloring

Barnier, N. Brisset, P.: Graph Coloring for Air Traffic Flow Management. In: Proceedings of the Fourth International Workshop on Integration of AI and OR Techniques, Le Croisic, France, 2002, pp. 133–147.

Chaitin, G. J. Auslander, M. J. Chandra, A. K. Cocke, J. Hopkins, M. E. Markstein, P.W.: Register Allocation Via Coloring. Computer Languages, Vol. 6, 1981, pp. 47–57.

De Werra, D.: An Introduction to Timetabling. European Journal of Operational Research, Vol. 19, 1985, pp. 151–162.

Giaro, K. Kubale, M. Obszarski, P.: A Graph Coloring Approach to Scheduling of Multiprocessor Tasks on Dedicated Machines With Availability Constraints. Discrete Applied Mathematics, Vol. 157, 2009, pp. 3625–3630.

Gamst, A.: Some Lower Bounds for a Class of Frequency Assignment Problems. IEEE Transactions of Vehicular Technology, Vol. 35, 1986, No. 1, pp. 8–14.

Jensen, T. R. Toft, B.: Graph Coloring Problems. John Wiley & Sons, USA 1994.

Kubale, M. (2004), G.raph Colorings, American Mathematical Society, ISBN 0-8218-3458-4

Kuhn, F. (2009), "Weak graph colorings: Distributed algorithms and applications", Proceedings of the 21st Symposium on Parallelism in Algorithms and Architectures, pp. 138–144,

Suyudi, M., Sukono, Mamat, M., and Bon, A.T. Branch and Bound Algorithm for Finding the Maximum Clique Problem. *Proceedings of the International Conference on Industrial Engineering and Operations Management*, Volume 2018-March, 2018, Pages 2734-2742

Suyudi, M.,Mamat, M.,Sukono,and Supian, S. Branch and Bound for the Cutwidth Minimization Problem. *Journal of Engineering and Applied Sciences*, Volume 12, Issue Special Issue1, 2017, Pages 5684-5689.

Suyudi, M.,Mamat, M., and Sukono. An Efficient Approach for Traveling Salesman Problem Solution with Branch-and-Bound. *Proceedings of the International Conference on Industrial Engineering and Operations Management*, 2016, Pages 543-546.

Preeti Gupta.(2014) , A study of Vertex - Edge Coloring Techniques with Application, Journal Of Core Engineering & Management Volume - 1, issue 2,MAY, 2014 27

Biographies

Mochamad Suyudi, is a lecturer at the Department of Mathematics, Faculty of Mathematics and Natural Sciences, Universitas Padjadjaran. Bachelor in Mathematics at the Faculty of Mathematics and Natural Sciences, Universitas Padjadjaran, and Master in Mathematics at the Faculty of Mathematics and Natural Sciences, Universitas Gajah Mada. Currently pursuing Ph.D. program in the field of Graphs at Universiti Sultan Zainal Abidin(UNISZA) Malaysia Terengganu.

Mustafa Mamat is a lecturerin the Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin, Malaysia. Currently serves as Dean of Graduate School Universiti Sultan Zainal Abidin, Terengganu, Malaysia. The field of applied mathematics, with a field of concentration of optimization.

Sukono is a lecturer in the Department of Mathematics, Faculty of Mathematics and Natural Sciences, Universitas Padjadjaran. Currently serves as Head of Master's Program in Mathematics, the field of applied mathematics, with a field of concentration of financial mathematics and actuarial sciences.

Abdul Talib Bon is a professor of Production and Operations Management in the Faculty of Technology Management and Business at the Universiti Tun Hussein Onn Malaysia since 1999. He has a PhD in Computer Science, which he obtained from the Universite de La Rochelle, France in the year 2008. His doctoral thesis was on topic Process Quality Improvement on Beltline Moulding Manufacturing. He studied Business Administration in the Universiti Kebangsaan Malaysia for which he was awarded the MBA in the year 1998. He’s bachelor degree and diploma in Mechanical Engineering which his obtained from the Universiti Teknologi Malaysia. He received his postgraduate certificate in Mechatronics and Robotics from Carlisle, United Kingdom in 1997. He had published more 150 International Proceedings and International Journals and 8 books. He is a member of MSORSM, IIF, IEOM, IIE, INFORMS, TAM and MIM.