A New Reduction from 3-SAT to Graph *K*-Colorability for Frequency Assignment Problem

Prakash C. Sharma
Indian Institute of Technology
Survey No. 113/2-B, Opposite to Veterinary College, A.B.Road, Village Harnia Khedi, MHOW, Indore (M.P.), India, Pin 453441

Narendra S. Chaudhari
Indian Institute of Technology
Survey No. 113/2-B, Opposite to Veterinary College, A.B.Road, Village Harnia Khedi, MHOW, Indore (M.P.), India, Pin 453441,

ABSTRACT
The satisfiability problem (SAT) is one of the most prominent problems in theoretical computer science for understanding of the fundamentals of computation. It is first known NP-Complete problem. Graph *k*-Colorability (for $k \geq 3$) Problem (GCP) is also an well known NP-Complete problem. We can reduce any NP-Complete problem to/from SAT. Reduction from satisfiability problem to graph *k*-colorability problem or vice versa is an important concept to solve one of the hard scheduling problem, frequency assignment in cellular network. The frequency assignment problem is very similar to the graph *k*-colorability problem. In this paper, we are presenting a polynomial reduction from any instance of 3-CNF-SAT formula to *k*-colorable graphs. Moret [2] gave an reduction approach from 3-SAT to 3-colorable graph. According to Moret, reduced 3-colorable graph having $(2n + 3m + 1)$ vertices and $(3n + 6m)$ edges, where n is the number of variables and m is number of clauses contained by 3-SAT formula. Here, we generalized the reduction approach to reduce any instance of 3-CNF-SAT formula to a *k*-colorable graph in polynomial time with mathematical proof. Our reduction approach generate a *k*-colorable graph with $|V| = (2n + 3m + (k\text{-}2))$ vertices and $|E| = (3n + 6m)$ edges for $k = 3$ and $|E| = (|E|$ of $(k\text{-}1)$-colorable graph + $(|V|\text{-}1))$ edges for $k > 3$ corresponding to any instance of 3-CNF-SAT. Further, we give the formulation of graph *k*-colorability to frequency assignment problem in cellular network.

Keywords
3-SAT, CNF, graph *k*-colorability, NP-Complete, chromatic number, frequency assignment

1. INTRODUCTION
The satisfiability problem (SAT) is one of the most prominent problems in theoretical computer science, which has become increasingly popular and important insights into our understanding of the fundamentals of computation. SAT is the first known NP-Complete problem. It is used as a starting point for proving that other problems are also NP-hard. This is done by polynomial-time reduction from 3-SAT to the other problem. We can reduce any NP-Complete problem to/from 3SAT. Reduction from satisfiability problem to graph *k*-colorability problem or vice versa is an important concept to solve one of the hard scheduling problem, frequency assignment in cellular network. The frequency assignment problem is very similar to the graph *k*-colorability problem.

Determining the *k*-colorability of any graph is also an NP-Complete problem [3][7]. In a proper graph coloring, if two vertices u and v of a graph share an edge (u, v), then they must be colored with different colors. The minimum number of colors needed to properly color the vertices of G is called the chromatic number of G, denoted $\chi(G)$. A *k*-coloring of graph G is an assignment of colors $\{1,2,\ldots,k\}$ to the vertices of G in such a way that neighbors receive different colors. Graph Coloring Problem is very important because it has many applications; some of them are planning and scheduling problems such as timetabling, channel assignment in cellular network [4][5][6][8] and many others.

The frequency band has become an important resource for communication service. There has been large increase in demand for using the frequency bands caused by the fast growth in mobile communication, satellite communication and mass communication service areas. To maximize utilization of frequency band, the limited band of available frequency is divided into a number of channels. A channel can be reused many times for different transmitters if the transmitters are far enough from one another so that the co-channel interference between them is low enough. If there are two close transmitters using the same channel simultaneously, they will suffer from severe co-channel interference and the quality of communication service will be unsatisfactory.

Since the available frequency band is limited, we are interested in using as small band of frequency as possible while satisfying all the frequency demand and the co-channel constraints. This can be efficiently done by the proper frequency assignment. It is shown that the frequency assignment problem is equivalent to an extended version of graph coloring problem [8]. A variation of the graph coloring problem is the graph *k*-colorability problem [7].

In this paper, we introduce a new framework to represent SAT problems, for this, we proceed for the reduction of the instance of 3-CNF-SAT formula to *k*-colorable graph in polynomial time. Our reduction formula generate a *k*-colorable graph with $|V| = (2n + 3m + (k\text{-}2))$ vertices and $|E| = (3n + 6m)$ edges for $k = 3$ and $|E| = (|E|$ of $(k\text{-}1)$-colorable graph + $(|V|\text{-}1))$ edges for $k > 3$ corresponding to any instance of 3-CNF-SAT. Previously, in standard reduction approach from 3-SAT to 3-Colorable graph [1], the generated graph having $(2n+5m+3)$ vertices and $(3n+10m+3)$ edges. Further, Moret [2] gave an improved reduction approach from 3-SAT to 3-colorable graph. According to Moret, reduced 3-colorable graph will have $(2n + 3m + 1)$ vertices and $(3n + 6m)$ edges. Here, we generalized the reduction approach to reduce any instance of 3-CNF-SAT formula to a *k*-colorable graph in polynomial time with mathematical proof.

In next section of this paper, we have explored basic detail of 3-SAT, *k*-colorable graph and frequency assignment problem. Section 3 describes our polynomial reduction approach from 3-SAT to *k*-colorable graph. Section 4 explored the formulation of graph *k*-colorability to frequency assignment problem.

2. BACKGROUND

2.1 3-Satisfiability (3-SAT)

The satisfiability problem in conjunctive normal form (CNF) consists of the conjunction of a number of clauses, where a clause is a disjunction of a number of propositions or their negations. Let F be a 3-CNF formula which is conjunctions of m clauses say $C_1 \wedge C_2 \wedge_{\dots} \wedge C_m$, where each clauses having at most n literals i.e. a clause of length-n (in case of 3-CNF, n=3) with disjunctive form say $l_1 \vee l_2 \vee ... \vee l_n$. Each literal may be variables or negation of these variables say $x_1, x_2, \dots, x_n$ or $\neg x_1, \neg x_2 \dots \neg x_n$ where $\neg$ indicates negation. Every literal can take a truth value (0 or false, 1 or true). In Satisfiability problem, a set of values for the literals should be found, in such a way that the evaluation of formula should be true; if it is true then formula is called satisfiable, and otherwise formula is unsatisfiable. Following expression E is an example of 3-SAT:

$$E = (x_1 \vee \neg x_2 \vee \neg x_3) \wedge (\neg x_1 \vee x_2 \vee x_3) \wedge (x_1 \vee x_2 \vee x_4)$$

Here, E has 3 clauses (denoted by parentheses), four variables (x_1, x_2, x_3, x_4), and clause length n=3 (three literals per clause). Here, x_1 is a positive literal and $\neg x_2$ is a negative literal. One of the truth assignments for satisfiability of above expression is $x_1 = x_3$ = true, & x_2 = false or $x_1 = x_2$ = true & x_3 = false. 3-SAT is an NP-complete problem and it is used as a starting point for proving that other problems are also NP-hard. This is done by polynomial-time reduction from 3-SAT to the other problem.

2.2 Graph *k*-Colorability

Coloring a Graph with k colors or k-Coloring Problem is as follows: Is it possible to assign one of k colors to each vertex of a graph $G = (V, E)$, such that no two adjacent nodes be assigned the same color? If the answer is positive (or YES), we say that the graph is k-colorable and k is the chromatic number of graph G. It is possible to transform any SAT formula F into k-colorable graph, generate a graph $G = (V,E)$ with number of colors k, such that G is k-colorable only in the case of F is satisfiable.The graph k-colorability problem has several important real-world applications, including register allocation, scheduling, frequency assignment, and many other problems. The problem also underlies various popular games, including Sudoku and Minesweeper.

2.3 Frequency Assignment Problem

The frequency assignment problem is described as follows [8]: Assume that there is a collection of possible channels for frequencies to be assigned to a set of transmitters. Two transmitters which are close to each other must have two different channels to maintain a requested grade of communication quality. The minimum difference between two assigned frequencies, to be assigned to two close transmitters, depends on the interference level between them. The objective of the frequency assignment problem is to find an efficient assignment using the smallest span of the frequency band while satisfying the communication quality constraints.

3. POLYNOMIAL REDUCTION FROM 3-CNF-SAT TO *K*-COLORABLE GRAPH

The method of showing that a problem is NP-Complete by polynomial reduction is one of the most elegant and productive in computational complexity [9]. To prove that problem A is NP-hard, reduce a known NP-hard problem to A. Cook [10] defines the following:

Definition 1: Suppose that L_i is a language over $\sum i$, i =1,2.

Then $L_1 \leq_p L_2$ (L_1 is polynomially reducible to L_2) iff there is a polynomial-time computable function f: $\sum_1 \to \sum_2$ such that $x \in L_1 \leftrightarrow f(x) \in L_2$, for all $x \in \sum_1$.

Definition 2: A language L is NP-Complete iff L is in NP, and $L' \leq_p L$ for every language L' in NP

Proposition 1: Given any two languages, L_1 and L_2:

1) If $L_1 \leq_p L_2$ and $L_2 \in P$ then $L_1 \in P$.

2) If L_1 is NP-Complete, $L_2 \in NP$ and $L_1 \leq_p L_2$ then L_2 is NP-Complete.

3.1 3-SAT $\leq_p$ 3-Color

Theorem 1: Graph 3-Colorability is NP-Complete [2].

Proof: First of all we have to proof it as NP then try to proof NP-Hard. If it is both then it will be NP-Complete.

1. First we show that 3-Color $\in$ NP. Given a graph G, and a coloring assignment of the vertices, simply walk the graph and make certain that all adjacent vertices have a different color, and make certain that only 3 colors are used. This is clearly by $O(|V| + |E|)$, where $|V|$ is the number of vertices and $|E|$ is the number of edges of graph G.

2. Now show that 3-Color $\in$ NP-Hard. To do this, we reduce from 3-SAT to 3-Color, or show that 3-SAT $\leq_p$ 3-Color. Start with an instance of 3-SAT formula F with n variables x_1, $x_2, \dots, x_n$ and m clauses $c_1, c_2, \dots, c_m$. Create a graph G such that G is 3-colorable iff F is satisfiable. Reduced graph G has vertices corresponds to variables and coloring of vertices corresponding to truth assignment to variables from instance of 3-SAT formula.

Graph Construction: Given a 3CNF formula, we produce a graph as follows. The graph consists of a *triangle* for each variable and one *triangle* for each clause in the formula. All triangles for variables have a common vertex B (we can say base vertex) which preempts one color, so that the other two vertices of each such triangle corresponding to the variable and it's negation (or complement) must be assigned two different colors i.e. truth assignment either TRUE or FALSE. Then, we connect each vertex of a clause triangle to the corresponding literal vertex. Each such edge forces its two endpoints to use different colors.

Correctness: A clause triangle can be colored if and only if not all three of its corresponding literal vertices have been given the same color, that is, a clause triangle will be proper 3-colored if and only if not all three literals in the clause have been assigned the same truth value. Thus, the transformed instance admits a solution if and only if the original 3-SAT instance does.

=> If the 3-SAT formula has a satisfying assignment then the graph has 3-coloring.

<= If the graph has a 3-coloring, then the SAT formula has a satisfying assignment.

Bound: The transformation takes an instance of 3-SAT with n variables and m clauses and generated a 3-colorable graph that will have the number of vertices and edges as follows:

$|V| = (2n + 3m + 1)$ vertices and

$|E| = (3n + 6m)$ edges

It is easily done in polynomial time.

Example 1: Transform following 3-CNF-SAT formula into 3-colorable graph:

$(x \vee y \vee z') \wedge (x \vee y' \vee z')$
(1)

Here, number of variable n = 3 and number of clauses m = 2; corresponding to this instance of 3-CNF, following fig 1 generated 3-colorable graph having $|V|$ = 13 vertices and $|E|$ = 21 edges.

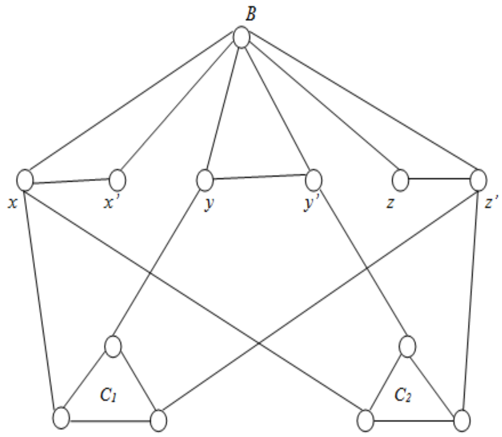


Fig 1: 3-Colorable Graph

3.2 3-SAT $\leq_p$ 4-Color

Theorem 2: Graph 4-Colorability is NP-Complete

Proof: First of all we have to proof it as NP then NP-Hard. If it is both then it will be NP-Complete.

1. First we show that 4-Color ∈ NP. Given a graph G, and a coloring assignment of the vertices, simply walk the graph and make certain that all adjacent vertices have a different color, and make certain that only 4 colors are used. This is clearly by $O(|V| + |E|)$.

2. Now show that 4-Color ∈ NP-Hard. To do this, we reduce from 3-Color to 4-Color, or show that 3-Color $\leq_p$ 4-Color. Let G^3 be an instance of 3-Color. Construct a new graph G^4 as follows: Add a single extra vertex B_1 and connect it to every other vertex in the graph. This is clearly polynomial in the size of the graph.

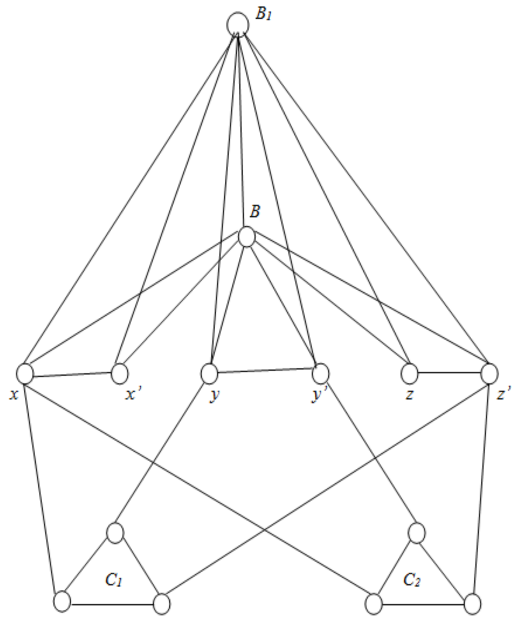
Correctness: Now we must show that G^4 is a *yes*-instance of 4-Color if and only if G^3 is a *yes*-instance of 3-Color. Consider the following proof.

=> Assume G^3 is 3-colorable. Therefore, G^4 is 4-colorable because the added vertex B_1, which is connected to all the other vertices in the graph, can be colored with a 4th color, and it will always be connected to vertices that are 1 of 3 other colors.

<= Assume G^4 is 4-colorable. Because B_1 is connected to every vertex in the graph, B_1 must be the only vertex in G^4 that has a certain color. Therefore, all other vertices in the graph are colored 1 of 3 colors. Therefore, G^3 is 3-colorable.

Since we have shown that 4-Color ∈ NP and 3-Color $\leq_p$ 4-Color, we have shown that 4-Color ∈ NP-Hard. Therefore, 4-Color ∈ NP-Complete.

Fig 2: 4-Colorable Graph



Bound: The transformation takes an instance of 3-SAT with n variables and m clauses and generated a 4-colorable graph that will have the number of vertices and edges as follows:

$|V| = (2n+3m+2)$ vertices and

$|E| = ((3n + 6m) + (2n+3m+1)) = (|E|$ of 3-colorable graph + $(|V|-1))$ edges

It is easily done in polynomial time.

Example 2: Transform (1) into 4-colorable graph:

$(x \vee y \vee z') \wedge (x \vee y' \vee z')$

Here, number of variable n = 3 and number of clauses m = 2; corresponding to this instance of 3-CNF, following fig 2 generated 4-colorable graph having $|V|$ = 14 vertices and $|E|$ = 21 + 13 = 34 edges.

3.3 3-SAT $\leq_p$ 5-Color

Theorem 3: Graph 5-Colorability is NP-Complete

Proof: First of all we have to proof it as NP then NP-Hard. If it is both then it will be NP-Complete.

1. First we show that 5-Color ∈ NP. Given a graph G, and a coloring assignment of the vertices, simply walk the graph and make certain that all adjacent vertices have a different color, and make certain that only 4 colors are used. This is clearly by $O(|V| + |E|)$.

2. Now show that 5-Color ∈ NP-Hard. To do this, we reduce from 4-Color to 5-Color, or show that 4-Color $\leq_p$ 5-Color. Let G^4 be an instance of 4-Color. Construct a new graph G^5 as follows: Add a single extra vertex B_2 and connect it to every other vertex in the graph. This is clearly polynomial in the size of the graph.

Correctness: Now we must show that G^5 is a *yes*-instance of 5-Color if and only if G^4 is a *yes*-instance of 4-Color. Consider the following proof.

=> Assume G^4 is 4-colorable. Therefore, G^5 is 5-colorable because the added vertex B_2, which is connected to all the other vertices in the graph, can be colored with a 5th color, and it will always be connected to vertices that are 1 of 4 other colors.

<= Assume G^5 is 5-colorable. Because B_2 is connected to every vertex in the graph, B_2 must be the only vertex in G^5 that has a certain color. Therefore, all other vertices in the graph are colored 1 of 4 colors. Therefore, G^4 is 4-colorable.

Since we have shown that 5-Color $\in$ NP and 4-Color $\leq_p$ 5-Color, we have shown that 5-Color $\in$ NP-Hard. Therefore, 5-Color $\in$ NP-Complete.

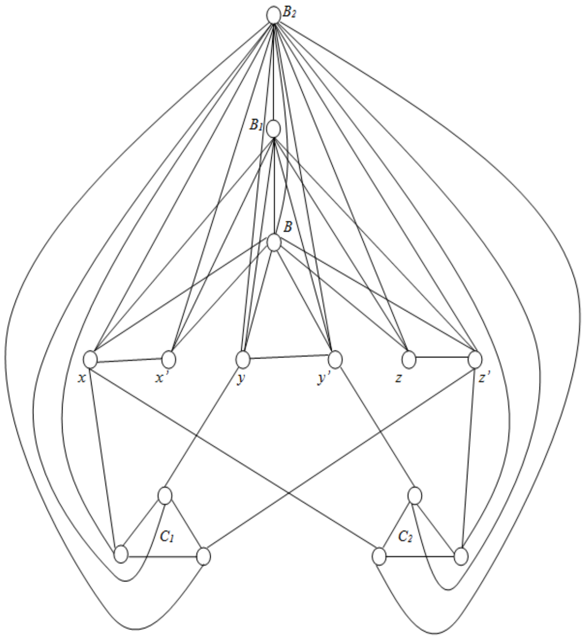


Fig 3: 5-Colorable Graph

Bound: The transformation takes an instance of 3-SAT with *n* variables and *m* clauses and generated a 5-colorable graph that will have the number of vertices and edges as follows:

$|V| = (2n+3m+3)$ vertices and

$|E| = (((3n + 6m) + (2n + 3m + 1)) + (2n + 3m + 2)) = (|E|$ of 4-colorable graph + $(|V|$-1)) edges

It is easily done in polynomial time.

Example 3: Transform (1) into 5-colorable graph:

$(x \vee y \vee z') \wedge (x \vee y' \vee z')$

Here, number of variable $n = 3$ and number of clauses $m = 2$; corresponding to this instance of 3-CNF, above fig 3 generated 5-colorable graph having $|V|$=14+1=15 vertices and $|E|$ = 34+14=48 edges.

3.4 3-SAT ≤p k-Color

Theorem 3: Graph *k*-Colorability is NP-Complete

Proof: First of all we have to proof it as NP then NP-Hard. If it is both then it will be NP-Complete.

1. First we show that *k*-Color $\in$ NP. Given a graph *G*, and a coloring assignment of the vertices, simply walk the graph and make certain that all adjacent vertices have a different color, and make certain that only *k* colors are used. This is clearly by O $(|V| + |E|)$.

2. Now show that *k*-Color $\in$ NP-Hard. To do this, we reduce from (*k*-1)-Color to *k*-Color, or show that (*k*-1)-Color $\leq_p$ *k*-Color. Let G^{k-1} be an instance of (*k*-1)-Color. Construct a new graph G^k as follows: Add a single extra vertex B_{k-3} and connect it to every other vertex in the graph. This is clearly polynomial in the size of the graph.

Correctness: Now we must show that G^k is a *yes*-instance of *k*-Color if and only if G^{k-1} is a *yes*-instance of (*k*-1)-Color. Consider the following proof.

=> Assume G^{k-1} is (*k*-1)-colorable. Therefore, G^k is *k*-colorable because the added vertex B_{k-3}, which is connected to all the other vertices in the graph, can be colored with a *k*th color, and it will always be connected to vertices that are 1 of (*k*-1) other colors.

<= Assume G^k is *k*-colorable. Because B_{k-3} is connected to every vertex in the graph, B_{k-3} must be the only vertex in G^k that has a certain color. Therefore, all other vertices in the graph are colored 1 of (*k*-1) colors. Therefore, G^{k-1} is (*k*-1)-colorable.

Since we have shown that *k*-Color $\in$ NP and (*k*-1)-Color $\leq_p$ *k*-Color, we have shown that *k*-Color $\in$ NP-Hard. Therefore, *k*-Color $\in$ NP-Complete.

Bound: The transformation takes an instance of 3-SAT with *n* variables and *m* clauses and generated a *k*-colorable graph that will have the number of vertices and edges as follows:

$|V| = (2n + 3m + (k\text{-}2))$ vertices and

$|E| = (3n + 6m)$ edges for k=3

= ($|E|$ of (*k*-1)-colorable graph + ($|V|$-1)) edges for k>3

So, it is easily done in polynomial time.

4. GRAPH K-COLORABILITY TO FREQUENCY ASSIGNMENT PROBLEM

Formulate the frequency assignment problem as a graph *k*-colorability problem. Let the vertices correspond to transmitters and edges correspond to interference between transmitters. Every vertex is labeled with a frequency range *Fi*. The question is whether one can allocate to each vertex a frequency from its frequency range so that no vertices are connected with an edge having the same frequency.

For doing this, first of all we have to show that the frequency assignment problem is in NP. Guess (non-deterministic) a frequency assignment. Go through each vertex and verify that its frequency is in the frequency set. Go also through each edge and verify that the endpoints of the frequencies are different. This takes linear time in the size of the graph.

In the second step we have to show that the frequency assignment problem is NP-hard. For this, reduce graph *k*-colorability problem to frequency assignment:

Graph *k*-coloring(*G*, *k*) =

for each vertex v_i in the graph *G*

$Fi \leftarrow \{1,\ldots,k\}$

return Frequency Assignment (G,{F_i})

Finally, check correctness of above as there is a *k*-coloring of graph *G* iff there is a correct assignment of frequencies to *G*, where every vertex has frequency set {1,…,*k*}. Suppose we have a *k*-coloring of *G*. Number the colors from 1 to *k*. If a vertex has color *i*, we assign to the corresponding vertex (transmitter) in the frequency allocation problem the frequency *i*. This is a correct frequency assignment because we have been based on a correct *k*-coloring. In the other direction: assume that we have a correct frequency assignment. we get a *k*- coloring by allowing a vertex to have color *i* if the corresponding transmitter have been assigned frequency *i*.

5. CONCLUSION

The primary focus of this paper is to introduce a generalized reduction approach from 3SAT to *k*-colorable graph. Our polynomial reduction approach generate a *k*-colorable graph with $|V| = (2n + 3m + (k\text{-}2))$ vertices and $|E| = (3n + 6m)$ edges for $k = 3$ and $|E|$ = ($|E|$ of (k-1)-colorable graph + ($|V|$-1)) edges for $k > 3$ corresponding to any instance of 3-CNF-SAT. Then, we give the formulation of graph *k*-colorability to frequency assignment problem in cellular network.

6. REFERENCES

[1] Alexander Tsiatas, "Phase Transitions in Boolean Satisfiability and Graph Coloring", May 2008, Department of Computer Science, Cornell University, (www.cseweb.ucsd.edu/users/atsiatas/phase.pdf).

[2] Bernard M. Moret "The Theory of Computation" Pearson Education, 1998, chapter 7, Proving Problem Hard, pp 226-252

[3] Garey, M. R. and Johnson, D. S., Computers and Interactability: A Guide to the Theory of NP-Completeness, Freeman, San Francisco, 1979.

[4] Daniel Marx, "Graph Colouring Problems and their applications in Scheduling", Periodica Polytechnica Ser El. Eng Vol.48, No.1, pp. 11-16 (2004)

[5] Maaly A. Hassan and Andrew Chickadel, "A Review Of Interference Reduction In Wireless Networks Using Graph Coloring Methods" , International journal on applications of graph theory in wireless ad hoc networks and sensor networks (GRAPH-HOC) Vol.3, No.1, March 2011, pp 58-67.

[6] Mohammad Malkawi, Mohammad Al-Haj Hassan and Osama Al-Haj Hassan, "New Exam Scheduling Algorithm using Graph Coloring", The International Arab Journal of Information Technology, Vol. 5, No. 1, January 2008, pp 80-87

[7] Taehoon P. and Lee, C.Y., (1994) "On the *k*-coloring problem", Journal of Korean OR/MS Society, 19, pp. 219-233.

[8] De Werra, D. and Gay, Y., (1994), "Chromatic scheduling and frequency assignment", Discrete Applied Mathematics 49, pp. 165-174.

[9] L. Adleman and K. Manders, "Reducibility, randomness and intractability (abstract)", in STOC 77: Proceedings of the ninth annual ACM symposium on Theory of computing. New York NY, USA: ACM Press, 1977, pp. 151-163.

[10] S. A. Cook, "The P versus NP problem", April 2000, Computer Science Department, University of Toronto. http://www.claymath.org/millennium/P_vs_NP/Official_Problem_Description.pdf.