ARTICLE IN PRESS

Discrete Applied Mathematics ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

A column generation based algorithm for the robust graph coloring problem

Birol Yüceoğlu [a,1], Güvenç Şahin [b,*], Stan P.M. van Hoesel [c]

[a] R&D Center, Information Technologies Department, Migros T.A.Ş, Ataşehir, 34758, İstanbul, Turkey

[b] Industrial Engineering, Faculty of Engineering and Natural Sciences, Sabanci University, Orhanli, Tuzla, 34956, Istanbul, Turkey

[c] Department of Quantitative Economics, Maastricht University, P.O. Box 616, 6200 MD, Maastricht, The Netherlands

ARTICLE INFO

Article history:
Received 17 June 2015
Received in revised form 27 June 2016
Accepted 4 September 2016
Available online xxxx

Keywords:
Robust graph coloring
Representatives formulation
Set-covering formulation
Column generation
Reduced cost fixing

ABSTRACT

Given an undirected simple graph G, an integer k, and a cost c_{ij} for each pair of non-adjacent vertices in G, a robust coloring of G is the assignment of k colors to vertices such that adjacent vertices get different colors and the total penalty of the pairs of vertices having the same color is minimum. The problem has applications in fields such as timetabling and scheduling. We present a new formulation for the problem, which extends an existing formulation for the graph coloring problem. We also discuss a column generation based solution method. We report computational study on the performance of alternative formulations and the column generation method.

1. Introduction

For a given undirected graph $G = (V, E)$, with V the set of vertices and E the set of edges, a (*vertex*) *coloring* of a graph is an assignment of colors to vertices such that no two vertices of an edge get the same color. A *k-coloring* of a graph uses k colors. The minimum number of colors necessary to color a graph is called the *chromatic number* of G and is denoted by $\chi(G)$.

The graph coloring problem has its origins from the need to color maps with a minimum number of colors. The problem dates back to 1852, when Francis Guthrie conjectured that a map could be colored using four colors, Kubale [13]. The problem of coloring a map (which can be transformed into a planar graph) is polynomially solvable, however the problem is *NP*-hard in general graphs, Karp [11]. The problem has applications in register allocation problem for compiler optimization [8], exam scheduling [15], frequency assignment in telecommunications [1], and course timetabling [5].

In scheduling, activities (such as duties to be assigned to operators) are represented by the vertices of a graph. Two activities that cannot be assigned to the same operator because of a time slot or equipment clash, are connected by an edge. Operators are assigned sets of tasks they can carry out without a clash. In timetabling, activities (such as courses or conference sessions to be assigned to time slots) are represented by the vertices of a graph. If two courses are taken by the same students or two conference sessions are similar in content the corresponding vertices are connected by an edge and those activities cannot be assigned to the same time slot.

Finding an optimal coloring in the context of scheduling and timetabling requires the problem data to be known with certainty apriori. In scheduling, activities are subject to delays. Therefore it may be undesirable to assign two activities to

* Corresponding author.
E-mail address: guvencs@sabanciuniv.edu (G. Şahin).

1 The research was carried while the author was a post-doctoral researcher at Sabancı University.

http://dx.doi.org/10.1016/j.dam.2016.09.006

the same operator if probability of a delay is high or a possible delay is costly. Similarly, in a course timetabling problem, it may not always be possible to create a timetable without any clashes. Furthermore, it is difficult to predict the choices of students beforehand. Therefore, it is desirable to create a timetable where number of clashes are minimized. In conference timetabling, two sessions that can be assigned to the same time slot can still attract the same set of people. Even though eliminating every possible clash may be difficult, it is still possible to minimize clashes.

The robust graph coloring problem is a generalization of the original graph coloring problem. As in the original problem, adjacent vertices are not allowed to take the same color while having the same color may only be undesirable and penalized for other vertices in the robust version. The major difference is that the objective function is not to minimize the number of colors but to minimize the sum of penalties due to pairs of vertices with the same color. The problem is introduced by Yáñez and Ramírez [21]. They show that the original graph coloring problem can be too restrictive if one also considers secondary objectives. They present an assignment type of integer linear programming (ILP) formulation and use a genetic algorithm to solve the problem. Lim and Wang [14] use the robust graph coloring problem to model the robust aircraft assignment problem; they employ heuristics to solve the problem. Guo et al. [9] and Kong et al. [12] also present heuristics. Wang and Xu [20] model the problem as an unconstrained quadratic programming problem and develop new heuristics.

To the best of our knowledge, the only exact approach for the robust graph coloring problem is the column generation approach in [2]. Archetti et al. [2] use the ILP formulation of Yáñez and Ramírez [21]. After establishing that the formulation is only suitable for small instances, they present a branch-and-price algorithm. They use both heuristics and exact methods to generate new columns.

We present a new ILP formulation for the robust graph coloring problem based on the asymmetric representatives formulation of Campêlo et al. [6], which is originally used for the graph coloring problem. The formulation, introduced by Campêlo et al. [7], uses the idea that vertices with the same color can represent each other. Furthermore, Campêlo et al. [6] use an ordering of the vertices to create the asymmetrical representative formulation. This enhanced formulation reduces the number of variables and different representation of color classes compared to the original representatives formulation. Furthermore, the asymmetric representatives formulation eliminates symmetries that result from the interchangeability of colors compared to the original formulation of Méndez-Díaz and Zabala [18]. We compare asymmetric representatives formulation with the original formulation in [21]; we show that it performs better than the original as it yields a much improved lower bound for the problem but is still heavily restricted by the size of the instances solved to optimality.

Even though the asymmetric representatives formulation performs better than the original formulation, it does not have a noticeable impact on the size of the instances solved to optimality. For this reason, we use the set-covering formulation in [2] and develop a column generation-based algorithm to solve it. Unlike Archetti et al. [2], we do not use branch-and-price; we employ a method in [19]. Even though this method enumerates all columns in the worst case, it performs well empirically.

In Section 2, we present our notation and the new ILP formulation based on the asymmetric representatives formulation. In Section 3, we develop a set-covering based formulation and our column generation based solution method. Computational experiments are presented in these sections. We discuss the results and conclude in Section 4.

2. Asymmetric representatives formulation

Given a simple, undirected, and connected graph $G = (V, E)$, where $n = |V|$ is the number of vertices and $m = |E|$ is the number of edges, two vertices i and j are *adjacent* if $\{i, j\} \in E$. $N(i) = \{j \in V \mid \{i, j\} \in E\}$ is called the *neighborhood* of i. An *ordering* of G is a mapping $\sigma : V \to \{1, \dots, n\}$, where $\sigma(i)$ denotes the position of i in the ordering; we use an ordering of the vertices to eliminate the symmetries in the problem. We identify each vertex with its position in the ordering, i.e., the vertices are numbered $1, 2, \dots, n$. For a given ordering of G, we call $N^-(i) = \{1, 2, \dots, i-1\} \cap N(i)$ the *in-neighborhood* of i and $N^+(i) = \{i+1, i+2, \dots, n\} \cap N(i)$ the *out-neighborhood* of i. $\bar{G} = (V, \bar{E})$ denotes the complement of G, where $\bar{E}$ consists of $\{i, j\} \notin E$; $\bar{N}(i) = \{j \in V \mid \{i, j\} \in \bar{E}\} \setminus \{i\}$ is called the *antineighborhood* of i. The in- and out-antineighborhoods of i in $\bar{G}$ are defined similarly as $\bar{N}^+(i)$ and $\bar{N}^-(i)$. The closed (anti)neighborhoods, where i is included, are denoted by $N[i]$, $N^-[i]$, $N^+[i]$, $\bar{N}[i]$, $\bar{N}^-[i]$, $\bar{N}^+[i]$. $\bar{N}^+[i]$ corresponds to the vertices that can be represented by i (including itself). $\bar{N}^-[i]$ corresponds to the vertices that can represent i (including itself).

We call $H = (V_H, E_H)$ an *induced subgraph* of G if $V_H \subseteq V$ and $\{i, j\} \in E_H$ if and only if $i \in V_H, j \in V_H$ and $\{i, j\} \in E$. H is called a *clique* if all vertices in H are pairwise adjacent. Each vertex of a clique has to have a different color. An *independent set* is a set of vertices, in which no two vertices are adjacent. In other words, H is an independent set if $E_H = \emptyset$. In any coloring, vertices having the same color form an independent set.

2.1. Mathematical model

We modify the asymmetric representatives formulation introduced by Campêlo et al. [6] which selects a subset of the vertices to represent the colors. Representative vertices can represent other vertices in their out-antineighborhood. The vertices represented by a representative vertex must form an independent set. Vertices that do not represent a color are identified by the color of their representatives, i.e., a vertex in the in-antineighborhood of i. c_{ij} denotes a non-negative cost associated with two vertices i and j such that $\{i, j\} \notin E$, it can be considered as the penalty of coloring two vertices with the same color.