A Branch-and-Cut Algorithm for Equitable Coloring

Laura Bahiense[*] Yuri Frota [⋆] Nelson Maculan[⋄] Thiago F. Noronha[†] Celso C. Ribeiro[‡]

[*] *Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia de Produção*
Rio de Janeiro, RJ 21945-970, Brazil, laura@pep.ufrj.br

[⋆] *Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia de Sistemas e Computação*
Rio de Janeiro, RJ 21945-970, Brazil, abitbol@cos.ufrj.br

[⋄] *Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia de Sistemas e Computação*
Rio de Janeiro, RJ 21945-970, Brazil, maculan@cos.ufrj.br

[†] *Pontifícia Universidade Católica do Rio de Janeiro, Department of Computer Science*
Rio de Janeiro 22453-900, Brazil, tfn@inf.puc-rio.br

[‡] *Universidade Federal Fluminense, Department of Computer Science*
Niterói, RJ 22410-240, Brazil, celso@ic.uff.br

Abstract

We present two new integer programming formulations for the equitable coloring problem. We also propose a primal constructive heuristic, branching strategies, and branch-and-cut algorithms based on each of these two formulations. The computational experiments showed that the results obtained by the new algorithms are far superior to those obtained by previously existing approaches.

Keywords: *equitable coloring, branch-and-cut, formulation by representatives.*

1 Introduction and motivation

Let $G = (V, E)$ be an undirected graph, where V is the set of vertices and E is that of edges. An equitable k-coloring of G is a partition of V into k disjoint stable subsets such that the difference on the cardinalities of any two subsets is at most one. Each subset is associated with a color and called a *color set*. The Equitable Coloring Problem (ECP) consists of finding the minimum value of k such that there is an equitable k-coloring of G. This number is said to be the *equitable chromatic number* of G and it is denoted by $\chi_=(G)$.

The equitable coloring problem was first introduced in [16], motivated by a problem concerning municipal garbage collection [19]. The vertices of the graph represent garbage collection routes. A pair of vertices share an edge if the corresponding routes should not be run on the same day. It is desirable that the number of routes run on each day be approximately the same. Therefore, the problem of assigning one of the six weekly working days to each route reduces to an equitable 6-coloring problem. Other applications arise from scheduling in communication systems [11], parallel memory systems [7], and partitioning and load balancing [2].

Branch-and-cut algorithms for the classical graph coloring problem have been developed by Figueiredo [8] and Méndez-Díaz and Zabala [14, 15]. Campêlo et al. [6] proposed a new 0-1 integer formulation for the graph coloring problem based on the idea of representative vertices. An asymmetric formulation and valid inequalities for the same problem have been proposed in [3]. The formulations in [3, 6] have been extended by Frota et al. [9] to handle the partition coloring problem.

The equitable coloring problem is proved to be NP-hard [13]. Polynomial algorithms are known only for split graphs [4] and trees [5]. A branch-and-bound algorithm based on integer programming was proposed in [1] for arbitrary graphs, but no expressive results were obtained. This paper presents a new exact algorithm for the problem, based on a generalization of the 0-1 formulation for the vertex coloring problem in [3, 6]. The linear relaxation of the proposed formulation gives lower bounds. A tabu search heuristic provides feasible solutions and upper bounds. We also propose a cutting plane procedure to strength the lower bounds and a decomposition strategy for solving ECP. Section 2 summarizes the proposed branch-and-cut algorithm. Computational experiments are reported in Section 3. Concluding remarks are drawn in the last section.

2 Branch-and-cut algorithm

In this section, we describe the branch-and-cut algorithm for ECP. We first present an integer programming formulation based on a generalization of the 0-1 formulation proposed in [3, 6] for the graph coloring problem. Next, we describe the branching strategy followed by a brief description of the valid inequalities that are used in a cutting plane procedure developed for improving the linear relaxation bound. Finally, we show how to build good feasible solutions for the problem.

The representative formulation for ECP is based on the following idea. Instead of associating vertices to colors, we choose one vertex to be the representative of all vertices with the same color. Therefore, each vertex is in one of the following two states: either it represents its color or there exists another vertex that represents its color. The representative formulation has been successfully applied to other graph coloring problems [3, 6, 9].

Let $A(u) = \{w \in V : (u, w) \notin E, w \neq u\}$ be the *anti-neighborhood* of vertex u (i.e., the subset of vertices that are not adjacent to u). We also define $A'(u) = A(u) \cup \{u\}$. Given a subset of vertices $V' \subseteq V$, we denote by $E[V']$ the subset of edges induced in the graph $G = (V, E)$ by V'. A vertex $v \in A(u)$ is said to be *isolated* in $A(u)$ if $E[A(u)] = E[A(u) \setminus \{v\}]$ (i.e., vertex v has no adjacent vertex in $A(u)$). We define the binary variables x_{uv} for all $u \in V$ and for all $v \in A'(u)$, such that $x_{uv} = 1$ if and only if vertex u represents the color of vertex v; otherwise $x_{uv} = 0$. We also define the equilibrium variable $w \in \mathbb{R}$, indicating the cardinality of the maximum stable set of the equitable coloring (i.e., the cardinality of each stable set is w or $w - 1$). We also define L_w and U_w as integral lower and upper bounds for the value of w, respectively. ECP can be formulated as the following integer programming problem, in which the non-linear constraints (5) and (6) will be linearized later:

$$SF = \min \sum_{u \in V} x_{uu} \tag{1}$$

subject to:

$$\sum_{v \in A'(u)} x_{vu} = 1, \qquad \forall u \in V \tag{2}$$

$$x_{uv} + x_{uw} \leq x_{uu}, \qquad \forall u \in V, \quad \forall (v, w) \in E : v, w \in A(u) \tag{3}$$

$$x_{uv} \leq x_{uu} \qquad \forall u \in V, \quad \forall v \in A(u) : v \text{ is isolated in } A(u) \tag{4}$$

$$x_{uu} + \sum_{v \in A(u)} x_{uv} \leq w \cdot x_{uu}, \qquad \forall u \in V \tag{5}$$

$$x_{uu} + \sum_{v \in A(u)} x_{uv} \geq (w - 1) \cdot x_{uu}, \qquad \forall u \in V \tag{6}$$

$$x_{uv} \in \{0, 1\}, \qquad \forall u \in V, \quad \forall v \in A'(u) \tag{7}$$

$$w \in \mathbb{R}. \tag{8}$$

The above model is said to be the *formulation by representatives* of the equitable coloring problem. The objective function (1) counts the number of representative vertices, i.e., the number of colors. Constraints (2) enforce that each vertex $u \in V$ must be represented either by itself or by another vertex v in its anti-neighborhood. Inequalities (3) enforce that adjacent vertices have distinct representatives. Inequalities (3) together with constraints (4) ensure that a vertex can only be represented by a representative vertex. Inequalities (5) and (6) guarantee that the difference on the cardinalities of any two color sets is at most one.

Constraints (5) and (6) in the above formulation can be linearized as (9) and (10) in the resulting LF_1 formulation (equations (1)-(4) and (7)-(10)):

$$x_{uu} + \sum_{v \in A(u)} x_{uv} \leq w - L_w \cdot (1 - x_{uu}), \qquad \forall u \in V \tag{9}$$

$$x_{uu} + \sum_{v \in A(u)} x_{uv} \geq (w-1) - (U_w - 1)(1 - x_{uu}), \qquad \forall u \in V. \tag{10}$$

An alternative approach to linearize constraints (5) and (6) introduces new variables y_i, for $i \in [L_w, U_w]$. Then, $y_i = 1$ if the maximum cardinality of a stable set is i; $y_i = 0$ otherwise. Variable w can be replaced by $\sum_{i=L_w}^{U_w} i \cdot y_i$ by adding constraint (11) to formulation SF:

$$\sum_{i=L_w}^{U_w} y_i = 1. \tag{11}$$

We also introduce variables z_{ui} that assume the result of the multiplication of variable x_{uu} by variable y_i, for all $u \in V$ and $i \in [L_w, U_w]$, together with the following linear inequalities:

$$z_{ui} \leq y_i, \quad z_{ui} \leq x_{uu}, \quad z_{ui} \geq y_i + x_{uu} - 1, \quad \forall u \in V \text{ and } i \in [L_w, U_w]. \tag{12}$$

Now, equations (5) and (6) are rewritten as equations (13) and (14) in the resulting formulation LF_2 (equations (1)-(4), (7) and (11)-(14)):

$$x_{uu} + \sum_{v \in A(u)} x_{uv} \leq \sum_{i \in [L_w, U_w]} i.z_{ui}, \qquad \forall u \in V \tag{13}$$

$$2x_{uu} + \sum_{v \in A(u)} x_{uv} \geq \sum_{i \in [L_w, U_w]} i.z_{ui}, \qquad \forall u \in V. \tag{14}$$

The asymmetric formulation by representatives in [3] is generalized to break symmetries in the above formulations. We establish that a vertex u can only represent the color of a vertex v if $u < v$. Therefore, the representative vertex of a color is that with the smallest index among all those with this same color. The asymmetric formulations are omitted for the sake of conciseness.

The branching strategy plays a major role in the success of a branch-and-cut algorithm. Branching on the x_{uv} variables, with $u \in V$ and $v \in A'(u)$, is not efficient because most of them are null in integral solutions. Therefore, our branching strategy is based on the cardinality variables w in LF_1 and on the y_i variables in LF_2. Branching on the x_{uv} variables starts only after all the w or y_i variables are integral.

Let w^* be the optimal value of variable w in the linear relaxation of LF_1. Two branches are generated if w^* is fractional: constraint $w \leq \lfloor w^* \rfloor$ is added in the first branch, while constraint $w \geq \lceil w^* \rceil$ is added in the second branch. If any of the y variables is fractional in the linear relaxation of LF_2, we branch on the variable y_i whose value in the linear relaxation is closest to 0.5 and two branches are generated: constraint $\sum_{j=L_w}^{i-1} y_j = 0$ is added in the first branch, while constraint $\sum_{j=i}^{U_w} y_j = 0$ is added in the second branch.

In both cases, LF_1 and LF_2, if no cardinality variable is fractional, then we branch on the variable x_{uu} whose value in the corresponding linear relaxation is closest to 0.5, with $u \in V$. If none of these variables is fractional, we branch on variables x_{uv}, with $u \in V$ and $v \in A(u)$.

To improve the linear relaxation bound, we use the two families of valid inequalities proposed in [3, 6]. *External cuts* (15) limit the maximum number of vertices in a subset $K \subseteq A(u)$ with a particular structure that can be represented at the same solution by the same vertex $u \in V$:

$$\sum_{v \in K} x_{uv} \leq \alpha_K \cdot x_{uu}, \tag{15}$$

where K is a clique, a hole, or an anti-hole, and α_K is the size of a maximum independent set of K. *Internal cuts* (16) limit the minimum number of colors necessary to color any odd hole or anti-hole $H \subseteq V$:

$$\sum_{v \in H} (x_{vv} + \sum_{u \in A(v) \setminus H} x_{uv}) \geq \chi(H), \tag{16}$$

where $\chi(H)$ is the chromatic number of the subgraph induced by H in G.

The separation of violated external and internal cuts consists basically of finding cliques, holes, and anti-holes in G. A GRASP [17] heuristic was used for finding clique cuts and a modification of the Hoffman and Padberg heuristic [10] for finding odd holes and anti-holes cuts. A detailed description of both heuristics can be found in [9].

The adaptive tabu search heuristic of Touhami [18] that provides upper bounds for the frequency assignment problem was adapted to give initial feasible solutions for ECP at each node of the branch-and-cut tree. Due to the limitation of space, we omit here the details of this algorithm.

3 Computational experiments

Two branch-and-cut algorithms were experimented. The first (B&C-LF_1) is based on formulation LF_1 (equations (1)-(4) and (7)-(10)), while the second (B&C-LF_2) is based on formulation LF_2 (equations (1)-(4), (7) and (11)-(14)). The results obtained by the branch-and-cut algorithms are compared with those provided by the branch-and-bound in [1].

The three algorithms were implemented in C++ and compiled with version v3.41 of the Linux/GNU compiler. XPRESS version 2005-a was used as the linear programming solver. All experiments were performed on an AMD-Atlon machine with a 1.8 GHz clock and one Gbyte of RAM memory. Graph coloring instances from the DIMACS challenge [12] and instances already used in [1] were used in the numerical experiments.

Some preliminary results are presented in Table 1. The first three columns display the name of each instance, its number of vertices, and its number of edges. The next four columns give the lower bound, the upper bound, the number of evaluated nodes in the branch-and-cut tree and the CPU times (in seconds) for finding the optimal solution provided by the algorithm in [1]. The last columns give the same results for algorithms B&C-LF_1 and B&C-LF_2. A missing time entry in this table indicates that the problem could not be solved within two hours of processing time. In this case, we display the lower bound, the upper bound, and the number of evaluated nodes at the time the algorithm was stopped.

Table 1 shows that the lower bounds provided by the branch-and-cut algorithms based on formulations LF_1 and LF_2 are very close and always better than or equal to those provided by the branch-and-cut algorithm in [1]. The number of instances solved to optimality by algorithms B&C-LF_1 (22 instances) and B&C-LF_2 (24 instances) is greater than those solved by the branch-and-cut in [1] (14 instances), and this in significantly smaller computation times for most test instances.

4 Concluding remarks

We presented two new integer programming formulations for the equitable coloring problem. We also proposed a primal constructive heuristic, branching strategies, and branch-and-cut algorithms based on each of these two formulations. The computational experiments were carried out on classical DIMACS [12] instances and on instances used in [1]. Instances with up to 206 vertices and 10,396 edges were solved

Graph	$\mid V \mid$	$\mid E \mid$	B&C [1] LB	UB	nodes	time	B&C-LF_1 LB	UB	nodes	time	B&C-LF_2 LB	UB	nodes	time
miles750	128	2113	3	58	1	-	31	31	5	124	31	31	6	171
miles1000	128	3216	3	67	130	-	42	42	10	214	42	42	13	267
miles1500	128	5198	2	86	1	-	73	73	1	13	73	73	1	13
zeroin.i.1	211	4100	2	211	1	-	49	49	3	79	49	49	1	50
zeroin.i.2	211	3541	2	211	1	-	35	61	97	-	36	36	23	510
zeroin.i.3	206	3540	2	206	2	-	35	61	101	-	36	36	28	491
queen.6_6	36	290	7	7	39447	941	7	7	1	2	7	7	1	1
queen.7_7	49	476	7	7	778	149	7	7	1	3	7	7	1	0
queen.8_8	64	728	8	10	4424	-	9	9	525	572	9	9	297	441
myciel3	11	20	4	4	19	0	4	4	5	0	4	4	7	0
myciel4	23	71	5	5	8867	12	5	5	255	4	5	5	237	5
jean	80	254	4	10	3453	-	10	10	1	3	10	10	1	4
anna	138	493	5	11	5913	-	11	11	1	18	11	11	2	26
david	87	406	7	30	8546	-	30	30	1	11	30	30	1	13
games120	120	638	9	9	11	39	9	9	1	35	9	9	1	30
$K_{5,2}$	10	15	3	3	1	0	3	3	1	0	3	3	1	0
$K_{7,2}$	21	105	6	6	40435	87	6	6	407	4	6	6	357	6
$K_{7,3}$	35	70	3	3	1	0	3	3	4	1	3	3	4	2
$K_{9,4}$	126	315	3	3	183	22	3	3	7	461	3	3	4	809
1-FullIns-3	30	100	4	4	73	3	4	4	10	2	4	4	34	2
2-FullIns-3	52	201	5	5	79	20	5	5	5	4	5	5	84	25
3-FullIns-3	80	346	6	6	21011	243	6	6	121	146	6	6	38	85
4-FullIns-3	114	541	7	7	5085	720	7	7	8	98	7	7	3	72
5-FullIns-3	154	792	8	8	24033	3088	8	8	77	1649	8	8	5	268

Table 1: Computational results

to optimality. The results obtained by the new algorithms were far superior to those found by the previously existing approach. Possible extensions of this work involve the identification of new families of valid inequalities that strengthen the equitable constraints.

References

[1] L. Bahiense, S. Jurkiewicz, A. Lozano, M. Pimenta, C. Waga, and C. Valladares, *An integer programming approach to equitable coloring problems*, in Proceedings of the XXXIX Brazilian Symposium on Operations Research, vol. 1, Fortaleza, 2007, pp. 1795–1801.

[2] J. Blazewicz, K. Ecker, E. Pesch, G. Schmidt, and J. Weglarz, *Scheduling Computer and Manufacturing Processes*, Springer, Berlin, 2001.

[3] M. Campêlo, V. Campos, and R. Corrêa, *On the asymmetric representatives formulation for the vertex coloring problem*, Electronic Notes in Discrete Mathematics **19** (2005) pp. 337–343.

[4] B.L. Chen, M.T. Ko, and K.W. Lih, *Equitable and m-bounded coloring of split graphs*, Lecture Notes in Computer Science **1120** (1996) pp. 1–5.

[5] B.L. Chen and K.W. Lih, *Equitable coloring of trees*, Journal of Combinatorial Theory – Series B **61** (1994) pp. 83–87.

[6] M. Campêlo, R.C. Corrêa, and Y. Frota, *Cliques, holes and the vertex coloring polytope*, Information Processing Letters **89** (2004) pp. 159–164.

[7] S. K. Das, I. Finocchi, and R. Petreschi, *Conflict-free star-access in parallel memory systems*, Journal of Parallel and Distributed Computing **66** (2006) pp. 1431–1441.

[8] R. Figueiredo, V. Barbosa, N. Maculan, and C. Souza, *Acyclic Orientations with Path Constraints*, RAIRO Operations Research **42** (200 8) pp. 455-467.

[9] Y. Frota, N. Maculan, T.F. Noronha, and C.C. Ribeiro, *A branch-and-cut algorithm for partition coloring*, Networks, to appear.

[10] K.L. Hoffman and M. Padberg, *Solving airline crew scheduling problems*, Management Science **39** (1993) pp. 657–682.

[11] S. Irani and V. Leung, *Scheduling with conflicts and applications to traffic signal control*, in Proceedings of the Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, Atlanta, 1996, pp. 85–94.

[12] D. Johnson and M. Trick, *Introduction to the Second DIMACS Challenge: Cliques, Coloring, and Satisfiability*, in Cliques, Coloring, and Satisfiability, D. Johnson and M. Trick, eds, American Mathematical Society, 1996, pp. 1-10.

[13] M. Kubale, *Graph Colorings*, American Mathematical Society, Providence, 2004.

[14] I. Méndez-Díaz and P. Zabala, *A branch-and-cut algorithm for graph coloring*, Discrete Applied Mathematics **154** (2006) pp. 826–847.

[15] I. Méndez-Díaz and P. Zabala, *A cutting plane algorithm for graph coloring*, Discrete Applied Mathematics **156** (2008) pp. 159–179.

[16] W. Meyer, *Equitable coloring*, American Mathematical Monthly **80** (1973) pp. 143–149.

[17] M.G.C. Resende and C.C. Ribeiro, *Greedy randomized adaptive search procedures*, in Handbook of Metaheuristics, F. Glover and G. Kochenberger, eds, Kluwer, 2003, pp. 219–249.

[18] S. Touhami, *Optimization Problems in cellular Networks*, PhD thesis, John Molson School of Business, Concordia University, Montreal, 2004.

[19] A. Tucker, *Perfect graphs and an application to optimizing municipal services*, SIAM Review **15** (1973) pp. 585–590.