The robust coloring problem

Article *in* European Journal of Operational Research · February 2003

DOI: 10.1016/S0377-2217(02)00362-4 · Source: RePEc

2 authors:

Javier Yáñez
Complutense University of Madrid

Javier Ramirez
Metropolitan Autonomous University

Available online at www.sciencedirect.com
SCIENCE DIRECT®
European Journal of Operational Research 148 (2003) 546–558
EUROPEAN JOURNAL OF OPERATIONAL RESEARCH
www.elsevier.com/locate/dsw

Discrete Optimization

The robust coloring problem

Javier Yáñez [a,*], Javier Ramírez [b]

[a] Department of Statistics and O.R., Facultad de Matematicas, Universidad Complutense de Madrid, 28040 Madrid, Spain
[b] Department of Systems, Universidad Autónoma Metropolitana-Azcapotzalco, 02200 México D.F., Mexico

Received 28 March 2001; accepted 28 March 2002

Abstract

Some problems can be modeled as graph coloring ones for which the criterion of minimizing the number of used colors is replaced by another criterion maintaining the number of colors as a constraint. Some examples of these problem types are introduced; it would be the case, for instance, of the problem of scheduling the courses at a university with a fixed number of time slots—the colors—and with the objective of minimizing the probability to include an edge to the graph with its endpoints equally colored. Based on this example, the new coloring problem introduced in this paper will be denoted as the Robust coloring problem, RCP for short. It is proved that this optimization problem is NP-hard and, consequently, only small-size problems could be solved with exact algorithms based on mathematical programming models; otherwise, for large size problems, some heuristics are needed in order to obtain appropriate solutions. A genetic algorithm which solves the RCP is outlined.

Keywords: Graph theory; Graph coloring; Timetabling

1. Introduction

Some scheduling problems can be stated as minimal coloring problems. In these problems, the objective is to minimize the number of colors in such a way that any pair of items to be scheduled and that cannot share the same resource must have a different color. This problem is stated, for instance, when we want to schedule the courses at a university in such a way that two courses taught by the same individual cannot be scheduled at the same time; the courses are represented by the nodes of a graph and every pair of incompatible courses are connected by an edge; the coloring of this graph provides a feasible schedule of the courses and a minimal coloring computes the minimal number of time slots needed as the chromatic number of the graph.

In some circumstances, however, this scheme seems to be very restrictive in the sense that it does not include any other scheduling problems which can be modeled as coloring problems.

* Corresponding author. Tel.: +34-91-394-4522; fax: +34-91-394-4606.
E-mail addresses: jayage@mat.ucm.es (J. Yáñez), jararo@correo.azc.uam.mx (J. Ramírez).

0377-2217/03/$ - see front matter
doi:10.1016/S0377-2217(02)00362-4

This is the case if the resource size to be shared is fixed. In the courses scheduling of a university problem, for instance, the number of available time slots could appear as a constraint instead of the objective function to minimize. Any c-coloring function, where c denotes the fixed resource size, gives a feasible schedule and the problem would be to choose the best one following some other criterion.

In this way, the minimal coloring problem can be a very restrictive model for some scheduling problems and a new graph coloring problem, denoted as the Robust coloring problem (RCP) in Ramírez [7], will be analyzed in this paper.

The major innovation of this problem is that the coloring function takes into account the overall topology of the graph: the endpoints of the edges cannot be equally colored—as in the classical coloring problem—and, also, the complementary edges are valued in such a way that their equally colored endpoints increase the objective function. Taking into account the classification of Pardalos et al. [6], this new problem cannot be considered as a internal or external generalization of the graph coloring problem.

Some examples of the new graph coloring problem are introduced in Section 2. In Section 3 the RCP is formalized and its computational complexity is analyzed; it will be concluded that the RCP can be classified as an NP—hard problem. The solution methods are analyzed in Section 4: an exact algorithm can be based on the binary programming model introduced in Section 4.1; an approximated genetic algorithm is introduced in Section 4.2. Computational experiences are included in Section 5.

2. Some introductory examples

Given three classical minimal coloring problems, the courses scheduling problem, the cluster problem and the map coloring problem, their objective function—number of time slots, number of cluster and number of colors—will be introduced as a constraint and three new coloring problems will be defined.

2.1. The examination timetabling problem

The following instance of the examination timetabling problem previously introduced in Section 1 is outlined.

Example 2.1. The examinations of six courses of a particular program at a university must be scheduled. Two courses sharing at least one student cannot be scheduled the same day. Taking into account the course incompatibilities, the following 6-nodes graph $G = (V, E)$ is constructed from V, the courses set, and the couple $\{i, j\}$ will be included in the edge set E when the courses i and j share at least one student. The adjacency matrix is the following:

$$B_G = \begin{pmatrix} 0 & & & & & \\ 1 & 0 & & & & \\ 1 & 1 & 0 & & & \\ 1 & 0 & 0 & 0 & & \\ 0 & 0 & 1 & 1 & 0 & \\ 0 & 1 & 0 & 1 & 1 & 0 \end{pmatrix}$$

The graph is depicted in Fig. 1.

In order to minimize the duration of examinations, a minimal coloring problem can be stated and the chromatic number of the incompatibility-graph is computed as $\chi(G) = 3$. The following coloring function C is in this sense optimal:

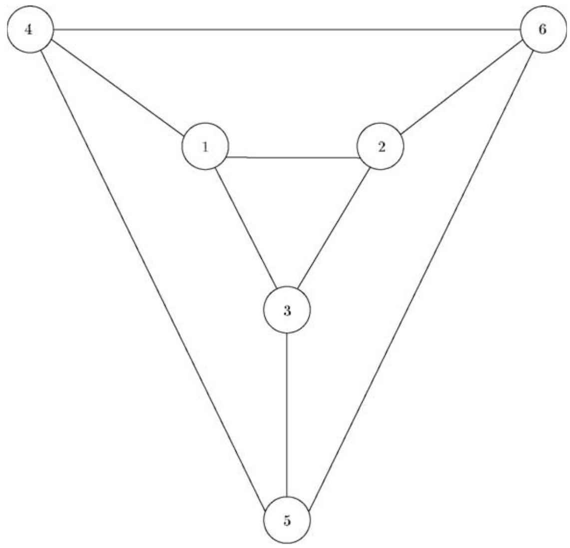

Fig. 1. Incompatibility graph of Example 2.1.

$$C(1) = 1, \quad C(2) = 2, \quad C(3) = 3,$$
$$C(4) = 2, \quad C(5) = 1, \quad C(6) = 3$$

Moreover, the stability of the scheduled solution deduced from the coloring function C would be desirable in the sense that it remains valid under some changes in the associated incompatibility graph. In Example 2.1, for instance, after the examination schedule has been published, one student could choose a new course in such a way that a new edge is included in the set E, the initial coloring becoming invalid.

Let $\overline{E}$ denote the complementary edge set:

$$\{i,j\} \in \overline{E} \iff \{i,j\} \notin E$$

and $\overline{G} = (V, \overline{E})$ will be the complementary graph of G.

The validity of a coloring function C under such changes of graph G, which can be defined as the *robustness* of the coloring, could be an important criterion which must be considered.

For Example 2.1, if the probabilities that any complementary edge $\bar{e} \in \overline{E}$ will be added to E are known, the robustness of the coloring can be measured as the probability of such coloring remaining valid after one random complementary edge has been added to the edge set. Let us suppose that the probability of the complementary edge $\{i,j\}$ will be proportional to the number of students registered in courses i and j. Let $n = 50$ be the total number of students which must choose two out of the six courses and let also n_i be the number of students registered in course i:

$$n_1 = 5, \quad n_2 = 30, \quad n_3 = 10,$$
$$n_4 = 30, \quad n_5 = 20, \quad n_6 = 5$$

The normalized probabilities are depicted in Table 1.

Table 1
Normalized probabilities of Example 2.1

$\{i,j\} \in \overline{E}$	$n_i \times n_j$	pr_{ij}
{1,5}	100	0.0506
{1,6}	25	0.0127
{2,4}	900	0.4557
{2,5}	600	0.3038
{3,4}	300	0.1519
{3,6}	50	0.0253

Given the above coloring function C, the probability that remain valid, assuming statistical independence, can be computed as:

$$(1 - \mathrm{pr}_{15})(1 - \mathrm{pr}_{24})(1 - \mathrm{pr}_{36})$$
$$= (1 - 0.0506)(1 - 0.4557)(1 - 0.0253)$$
$$= 0.5037$$

However, if the coloring function were C', defined by:

$$C'(1) = 1, \quad C'(2) = 2, \quad C'(3) = 3,$$
$$C'(4) = 3, \quad C'(5) = 2, \quad C'(6) = 1$$

the probability would be:

$$(1 - \mathrm{pr}_{16})(1 - \mathrm{pr}_{25})(1 - \mathrm{pr}_{34})$$
$$= (1 - 0.0127)(1 - 0.3038)(1 - 0.1519)$$
$$= 0.5829$$

Moreover, if the number of days to schedule the exams were not critical, say for instance 4, this probability will increase choosing the coloring function:

$$C''(1) = 2, \quad C''(2) = 4, \quad C''(3) = 1,$$
$$C''(4) = 3, \quad C''(5) = 2, \quad C''(6) = 1$$

which has a probability of remaining valid:

$$(1 - \mathrm{pr}_{15})(1 - \mathrm{pr}_{36}) = (1 - 0.0506)(1 - 0.0253)$$
$$= 0.9254$$

2.2. Cluster analysis

A basic problem in cluster analysis is how to partition the entities of a given set into a preassigned number of homogeneous subsets (clusters). The homogeneity of the clusters can be expressed as a function of a dissimilarity measure between entities. Let V be the set of entities and let E be the set of edges defined by every pair of entities $i, j \in V$ whose dissimilarity $d(i,j)$ is less than $\alpha > 0$, a given threshold. A c-coloring of this graph defines the c color classes as clusters. Hansen and Delattre (see Hansen et al. [3]) have shown that any of those c-coloring define a minimum-diameter partition of the entities set into c clusters. A numerical example of this problem is introduced.

Example 2.2. Let $V = \{1, 2, 3, 4, 5\}$ be the set of entities and let d be the dissimilarity distance defined by the matrix:

$$D = \begin{pmatrix} * & 0.01 & 0.02 & 0.05 & 0.04 \\ & * & 0.04 & 0.03 & 0.04 \\ & & * & 0.06 & 0.07 \\ & & & * & 0.03 \\ & & & & * \end{pmatrix}$$

For different values of the threshold α, different graphs G_α and coloring functions are obtained: For $\alpha = 0.05$, the graph G_α has two edges, i.e. $E = \{\{3, 4\}; \{3, 5\}\}$. This graph is depicted in Fig. 2.

Taking into account that $\chi(G_{0.05}) = 2$, the 2-coloring function C_1^2 defined by

$$C_1^2(1) = 1, \quad C_1^2(2) = 1, \quad C_1^2(3) = 2,$$
$$C_1^2(4) = 1, \quad C_1^2(5) = 1$$

defines two clusters:

$$V_{C_1^2}(1) = \{1, 2, 4, 5\}, \quad V_{C_1^2}(2) = \{3\}$$

Given another valid 2-coloring function C_2^2, defined by

$$C_2^2(1) = 1, \quad C_2^2(2) = 1, \quad C_2^2(3) = 1,$$
$$C_2^2(4) = 2, \quad C_2^2(5) = 2$$

defines another partition in clusters:

$$V_{C_2^2}(1) = \{1, 2, 3\}, \quad V_{C_2^2}(2) = \{4, 5\}$$

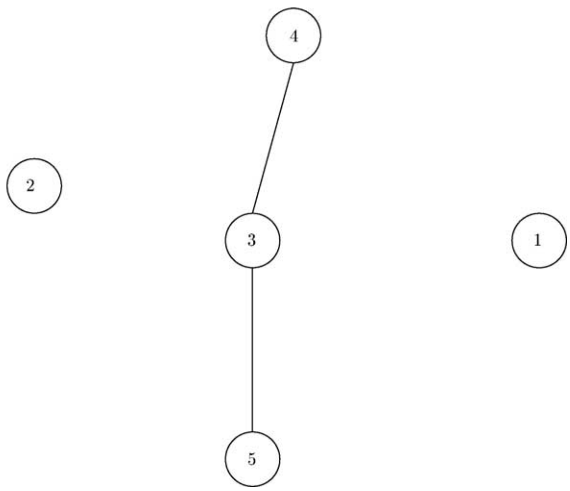

Fig. 2. Graph $G_{0.05}$ of Example 2.2.

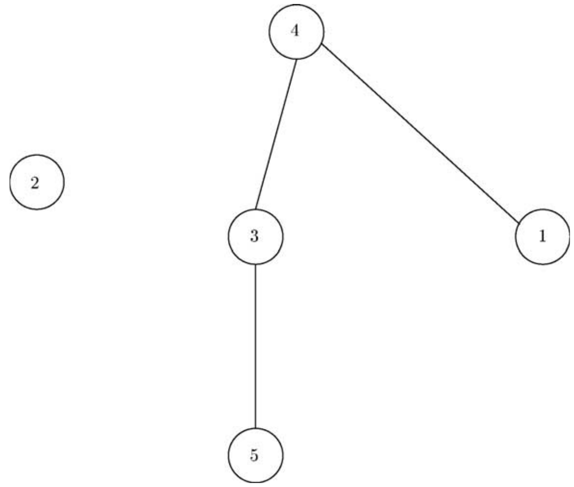

Fig. 3. Graph $G_{0.04}$ of Example 2.2.

Decreasing the threshold from $\alpha = 0.05$ to 0.04, one new edge is included, $\{1, 4\}$. The new graph $G_{0.04}$ is depicted in Fig. 3.

The chromatic number is also $\chi(G_{0.04}) = 2$. The coloring function C_1^2 is not valid but C_2^2 remains valid. This coloring function is in this sense preferable to C_1^2.

Decreasing the threshold again to the value $\alpha = 0.03$, neither of the two coloring functions remains valid because the chromatic number $\chi(G_{0.03}) = 3$.

Allowing three colors, the 3-coloring C_1^3 defined by:

$$C_1^3(1) = 1, \quad C_1^3(2) = 2, \quad C_1^3(3) = 1,$$
$$C_1^3(4) = 2, \quad C_1^3(5) = 3$$

whose color classes are:

$$V_{C_1^3}(1) = \{1, 3\}, \quad V_{C_1^3}(2) = \{2, 4\}, \quad V_{C_1^3}(3) = \{3\}$$

defines three clusters which remain valid until the threshold value $\alpha = 0.03$. The graph $G_{0.03}$ is depicted in Fig. 4.

In the cluster analysis, the minimal coloring seeks the minimum number of clusters. However, in some real situations, the number of cluster can be fixed, let c be this number, and the problem is how these c clusters can be filled in such a way that they remain valid under lower values for the dissimilarity measure threshold. Moreover, the limit

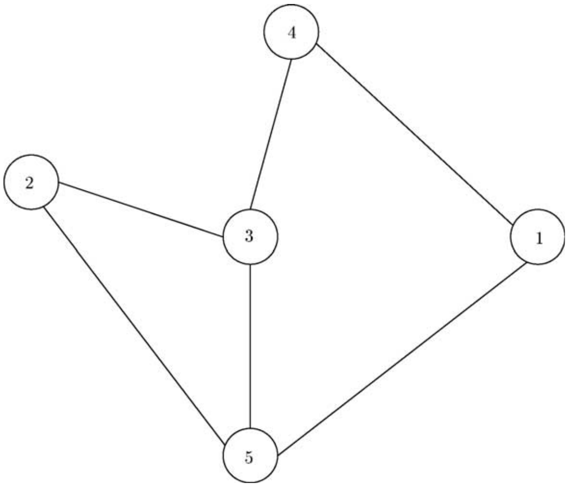

Fig. 4. Graph $G_{0.03}$ of Example 2.2.

value $\alpha(c)$ also can be computed, verifying that a c-coloring function C_*^c exists, and such that it remains valid for graph G_α for all $\alpha \geqslant \alpha(c)$. It is not difficult to see that $\alpha(c)$ is a decreasing function of c. In the limit case $c = n$, $\alpha(n) = 0$.

In Example 2.2, $\alpha(2) = 0.04, C_*^2 = C_2^2$ and $\alpha(3) = 0.03, C_*^3 = C_2^3$, where

$$C_2^3(1) = 1, \quad C_2^3(2) = 1, \quad C_2^3(3) = 2,$$
$$C_2^3(4) = 3, \quad C_2^3(5) = 3$$

2.3. Coloring geographical maps

The classical four-color problem faced with the minimal coloring of a planar map.

Example 2.3. Let G be the planar graph associated to the geographical map of eight regions depicted in Fig. 5.

Let C_1^4 and C_2^4 be two 4-coloring functions of G:

	1	2	3	4	5	6	7	8
C_1^4	3	2	2	1	4	2	2	3
C_2^4	3	2	4	1	3	2	1	4

and the corresponding color classes:

$$V_{C_1^4}(1) = \{4\}, \quad V_{C_1^4}(2) = \{2,3,6,7\},$$
$$V_{C_1^4}(3) = \{1,8\}, \quad V_{C_1^4}(4) = \{5\}$$

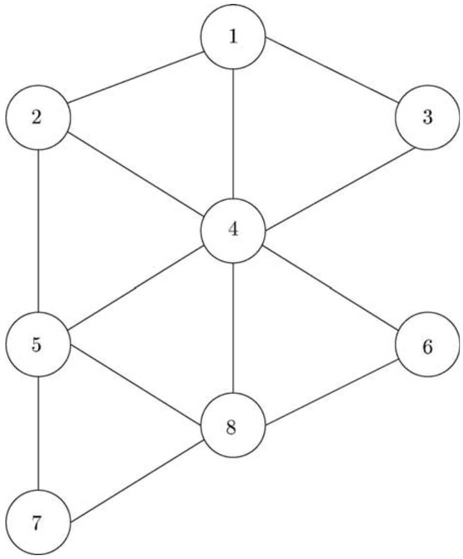

Fig. 5. Geographical map of Example 2.3.

and

$$V_{C_2^4}(1) = \{4,7\}, \quad V_{C_2^4}(2) = \{2,6\},$$
$$V_{C_2^4}(3) = \{1,5\}, \quad V_{C_2^4}(4) = \{3,8\}$$

We can notice that the color classes of C_2^4 are more homogeneous than the color classes of C_1^4. All of them have two regions. It is in this sense that we prefer coloring function C_2^4 to coloring function C_1^4.

All of these three problems have two types of constraints: a hard one which defines the valid coloring and a soft type constraint, which takes into account other criteria, for instance, the existence of additional edges of the graph. In these problems, the number of units of some used resource is considered as a constraint of the problem, instead of the objective function which must be minimized in the classical coloring problem.

3. The problem

In the previously proposed examples, the coloring functions of graph G does not search the minimum number of colors, which is fixed as c.

Obviously, this parameter must verify that $c \geqslant \chi(G)$. As was pointed out previously, and differing to the minimal coloring graph problem, the complementary edges are also considered.

In Example 2.1 the objective is to obtain a valid coloring function with no more than c colors in such a way that the probability of an added edge to the incompatibility graph with the two endpoints equally colored is minimized.

In Example 2.2 a coloring function is searched with the property of remaining valid when some edges were added after a decreasing of the threshold dissimilarity value.

In Example 2.3, the coloring function with the property of maintaining as closely as possible the cardinal of the color classes is preferred.

There is, however, a unified coloring graph problem which generalizes all of these examples. Taking into account that this graph coloring problem looks like a valid coloring function under possible changes of the graph, this problem will be denoted as the RCP for short, which can be stated formally in the following.

Given the graph $G = (V, E)$ with $|V| = n$ and $|E| = m$, let $c > 0$ be an integer number, then, a c-coloring is a mapping

$$C^c : V \to \{1, 2, \ldots, c\}$$

verifying

$$C^c(i) \neq C^c(j) \quad \forall \{i, j\} \in E$$

A c-coloring exists if and only if the number of colors allowed is equal or greater than the chromatic number of the graph G, i.e., $c \geqslant \chi(G)$.

Remark 3.1. From now on, any c-coloring will be presumed strict in the sense that all color classes are not empty:

$$\{i \in \{1, \ldots, n\} / C^c(i) = r\} \neq \emptyset \quad \forall r \in \{1, \ldots, c\}$$

The kernel of this new graph problem is the valuation of any valid c-coloring. In order to accomplish this objective, the following concept is introduced.

Definition 3.1. Given graph G and the penalty matrix defined on the complementary edge set

$$\{p_{ij} \geqslant 0, \ \{i, j\} \in \overline{E}\}$$

the *rigidity level* of the c-coloring C^c, denoted by $R(C^c)$, is defined as the sum of the penalties of complementary edges whose extremes are equally colored:

$$R(C^c) \equiv \sum_{\{i,j\} \in \overline{E}, C^c(i) = C^c(j)} p_{ij}$$

In Example 2.1, if any complementary edge

$$p_{ij} = -\ln(1 - \mathrm{pr}_{ij}) \quad \forall \{i, j\} \in \overline{E}$$

then, taking into account these penalties, the rigidity level of the c-coloring C is

$$R(C^c) \equiv \sum_{\{i,j\} \in \overline{E}, C^c(i) = C^c(j)} -\ln(1 - \mathrm{pr}_{ij})$$

$$= -\ln \left(\prod_{\{i,j\} \in \overline{E}, C^c(i) = C^c(j)} (1 - \mathrm{pr}_{ij}) \right)$$

In this way, the rigidity level is equal to minus the logarithm of the probability that the coloring remains valid. The most robust coloring minimizes the probability that an added edge to the incompatibility graph makes such coloring invalid.

In Example 2.2, however, the rigidity level has a more complex interpretation. It depends on the finite and ordered dissimilarity values of the objects.

Let $\{d^1, d^2, \ldots, d^k\}$ be the dissimilarity values set defined by $\overline{E} \times \overline{E}$. Consequently,

$$d_{ij} \in \{d^1, d^2, \ldots, d^k\} \quad \forall \{i, j\} \in \overline{E}_\alpha$$

verifying

$$d^1 < d^2 < \cdots < d^k$$

Given the threshold level α, the penalty of any complementary edge $\{i, j\} \in \overline{E}_\alpha$ whose endpoints are equally colored must be greater than the sum of all complementary edge penalties with dissimilarity lower than d_{ij}, i.e.

$$p_{ij} > \sum_{\{i',j'\} \in \overline{E}_\alpha / d_{i'j'} < d_{ij}} p_{i'j'}$$

This property assures that any c-coloring which remains valid for lower values of α will have a rigidity level lower than another c-coloring which does not remain valid for the same lower values of α. Let $\overline{m} = |\overline{E}|$ be the cardinal of the complementary edges set, the following penalties verify the above property:

$$p_{ij} = (\overline{m})^{s-1} \quad \text{if } d_{ij} = d^s \quad \forall \{i,j\} \in \overline{E}$$
$$s \in \{1, \dots, k\}$$

In Example 2.2, given the threshold $\alpha = 0.05$, the dissimilarity set is

$$\{0.01, 0.02, 0.03, 0.04, 0.05\}$$

in such a way that $k = 5$ and

$$d^1 = 0.01, \quad d^2 = 0.02, \quad d^3 = 0.03,$$
$$d^4 = 0.04, \quad d^5 = 0.05$$

Taking into account that $\overline{m} = 8$, one penalty set for this example is depicted in Table 2.

In Example 2.3, setting

$$p_{ij} = 1 \quad \forall \{i,j\} \in \overline{E}$$

the rigidity level of a c-coloring will be decreased as the number of regions equally colored tends to be uniform with respect to all colors.

The rigidity level $R(C^c)$ of a c-coloring C^c measures its *robustness* in the sense that the complementary edges whose endpoints are equally colored are penalized and, consequently, this c-coloring would not be valid if they were added to the graph.

Given a number of colors c, for a lower rigidity level the c-coloring will be more robust. Of course, as c increases, a more robust c-coloring can be obtained; in the limit case $c = n$, the most robust n-coloring will be

$$C^n(i) = i \quad \forall i \in V$$

which has a rigidity level $R(C^n) = 0$.

Table 2
Complementary edge penalties of Example 2.2

$\{i,j\} \in \overline{E}$	d_{ij}	d^s	p_{ij}
{1,2}	0.01	d^1	1
{1,3}	0.02	d^2	8
{1,4}	0.05	d^5	4096
{1,5}	0.04	d^4	512
{2,3}	0.04	d^4	512
{2,4}	0.03	d^3	64
{2,5}	0.04	d^4	512
{4,5}	0.03	d^3	64

The RCP, can be stated in the following way: Given a graph $G = (V, E)$, an integer number c and the penalty set $\{p_{ij}, \{i,j\} \in \overline{E}\}$. The RCP looks like those c-coloring C^c_{R} with the least rigidity level:

$$R(C^c_{\mathrm{R}}) = \min_{C^c} R(C^c)$$

Taking into account that the adjacency matrix B of a graph is symmetrical, any instance of the RCP can be characterized by (n, c, H), where n is the number of nodes of the graph, c is the number of valid colors, and H is the $n \times n$ matrix which stores the adjacency matrix of the graph in the lower triangle matrix and the penalties of its complementary edges in the upper triangle matrix:

$$h_{ij} = \begin{cases} p_{ij} & \text{if } i < j \\ b_{ji} & \text{if } i > j \end{cases}$$

where

$$b_{ij} = \begin{cases} 1 & \text{if } \{i,j\} \in E \\ 0 & \text{if } \{i,j\} \notin E \end{cases} \quad i < j$$

and p_{ij} is the penalty of complementary edge $\{i,j\} \notin E$ or 0 if $\{i,j\} \in E$.

The parameters (n, c, H) of Example 2.1 are:

$$n = 6 \quad c = 4$$
$$H = \begin{pmatrix} 0 & 0 & 0 & 0 & 0.0519 & 0.0128 \\ 1 & 0 & 0 & 0.6083 & 0.3621 & 0 \\ 1 & 1 & 0 & 0.1648 & 0 & 0.0256 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 \end{pmatrix}$$

where penalties h_{ij} have been obtained from the Table 3:

Table 3
Complementary edge penalties of Example 2.1

$\{i,j\} \in \overline{E}$	pr_{ij}	p_{ij}
{1,5}	0.0506	0.0519
{1,6}	0.0127	0.0128
{2,4}	0.4557	0.6083
{2,5}	0.3038	0.3621
{3,4}	0.1519	0.1648
{3,6}	0.0253	0.0256

The most robust 4-coloring is:

$$C^4_R(1) = 2, C^4_R(2) = 4, C^4_R(3) = 1, C^4_R(4) = 3,$$
$$C^4_R(5) = 2, C^4_R(6) = 1$$

Its rigidity level is

$$R(C^4_R) = 0.0519$$

in such a way that the probability that it will remain valid under a random change in the edge set is:

$$\frac{1}{\exp(0.0519)} = 0.9494$$

Given the threshold level $\alpha = 0.05$, Example 2.2 is characterized by:

$$n = 5 \quad c = 3 \quad H = \begin{pmatrix} 0 & 1 & 8 & 4096 & 512 \\ 0 & 0 & 512 & 64 & 512 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 64 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

The most robust 3-coloring is:

$$C^3_R(1) = 1, \quad C^3_R(2) = 1,$$
$$C^3_R(3) = 2, \quad C^3_R(4) = 3, \quad C^3_R(5) = 3$$

Its rigidity level is $R(C^3_R) = p_{12} + p_{45} = 65$ which can be interpreted in the following way: The 3-coloring C^3_R will remain valid for any threshold value $\alpha > d^3 = 0.03$, where the exponent 3 is chosen from the inequality chain

$$64 = 8^2 \leqslant 65 = R(C^3_R) < 512 = 8^3$$

The critical threshold $\alpha(3) = \alpha(C^3_R) = 0.03$

Example 2.3 is characterized by:

$$n = 8 \quad c = 4 \quad H = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \end{pmatrix}$$

The most robust 4-coloring is C^4_2. Its rigidity level is

$$R(C^4_2) = p_{47} + p_{26} + p_{15} + p_{38} = 4$$

In general, when all penalties are equal to a constant $p > 0$, then the most robust c-coloring computes the c-coloring with the most homogeneous cardinal of color classes.

Proposition 3.1. *Given graph $G = (V, E)$ with a c-coloring C^c_1, if the penalties of the complementary edges verify*

$$p_{ij} = p > 0 \quad \forall \{i, j\} \in \overline{E}$$

and two different color classes verifying

$$n_1 = |V_{C_1}(k)| < |V_{C_1}(l)| = n_2$$

exist with $n_2 > n_1 + 1$, then the resulting c-coloring C^c_2 after changing, if it is possible, one node from the color class $V_{C_1}(l)$ to the color class $V_{C_1}(k)$ is more robust than C^c_1.

Proof. The cardinals of color classes $V_{C_2}(k)$ and $V_{C_2}(l)$ are $n_1 + 1$ and $n_2 - 1$ respectively. The rigidity level added by these classes is lower than that added by color classes $V_{C_1}(k)$ and $V_{C_1}(l)$:

$$\binom{n_1}{2} + \binom{n_2}{2} > \binom{n_1 + 1}{2} + \binom{n_2 - 1}{2}$$

developing the binomial coefficients:

$$\frac{1}{2(n_1^2 - n_1 + n_2^2 - n_2)} > \frac{1}{2(n_1^2 + n_1 + n_2^2 - 3n_2 + 2)}$$

which is equivalent to

$$-n_1 - n_2 > n_1 - 3n_2 + 2$$

and, it is also equivalent to

$$n_2 > n_1 + 1$$

and the proposition is thus proven.

Consequently, the *equitable coloring problem* (see de Werra [9] and Lih [5]) which seeks a c-coloring whose color classes are nearly as equal in size as possible, can be viewed as a particular case of the RCP.

3.1. Computational complexity analysis

When we are confronted with a new problem, in our case the RCP, a natural question to ask is: Can it be solved with a polynomial time algorithm? If the answer is positive, any instance of this problem can be solved in a reasonable amount of time; otherwise, the computational complexity theory can help us to decide if this problem is computationally difficult in the sense that only small size instances can be solved. See Garey and Johnson [2] for further details. The complexity analysis of the RCP begins with its associated decision problem:

DRCP:
INSTANCE: A graph $G = (V, E)$, one positive integer $c \leqslant |V|$, a family of non-negative penalties defined over $\overline{E}$ and an upper bound $\bar{r}$,
QUESTION: Does a c-coloring C^c of G exist such that $R(C^c) = \sum_{i<j, C^c(i)=C^c(j)} h_{ij} \leqslant \bar{r}$?

Proposition 3.2. *The DRCP is NP—complete.*

Proof. The DRCP $\in$ NP, because a non-deterministic algorithm only needs to check in polynomial time that a c-coloring verifies all constraints. Let DMCP be the decision problem associated with the minimal coloring problem:

DMCP:
INSTANCE: A graph $G = (V, E)$ and one positive integer $k \leqslant |V|$,
QUESTION: Is G k-colorable, i.e., does a k-coloring of G exist?

We transform the DMCP to DRCP. Given an instance $I \in D_{\mathrm{DMCP}}$, let $h_{ij} = 0\ \forall \{i,j\} \in \overline{E}$, be the trivial penalty matrix and let also $c = k$ be the number of valid colors; then, considering $\bar{r} = 0$, an associated instance $f(I) \in D_{\mathrm{DRCP}}$ has been defined in such a way that

$$I \in Y_{\mathrm{DMCP}} \iff f(I) \in Y_{\mathrm{DRCP}}$$

This mapping f can be constructed in polynomial time. □

4. The solution methods

As a consequence of the above computational complexity analysis, the optimization problem RCP is NP-hard and only small size instances can be solved exactly; otherwise, some heuristics must be used. In Section 4.1 a binary programming model for the RCP will be introduced. In Section 4.2 a genetic algorithm to solve the RCP will be outlined.

4.1. Mathematical programming model

Let $G = (V, E)$ be the graph, with $n = |V|$ and $m = |E|$. Let c be the number of valid colors. The penalties of complementary edges are defined by $\{p_{ij}/\{i,j\} \in \overline{E}\}$.

In order to solve the RCP exactly the following binary programming model is introduced.

Let $x_{ik} \in \{0, 1\}$ be the decision variable defined by

$$x_{ik} = \begin{cases} 1 & C^c(i) = k \\ 0 & \text{otherwise} \end{cases} \quad i \in \{1, \ldots, n\} \quad k \in \{1, \ldots, c\}$$

identifying the function

$$C^c : V \to \{1, \ldots, c\}$$

as the c-coloring of the graph G.

Introducing the auxiliary variables:

$$y_{ij} = \begin{cases} 1 & \text{if there exists } k \in \{1, \ldots, c\} \\ & \text{such that } x_{ik} = x_{jk} \\ 0 & \text{otherwise} \end{cases} \qquad \forall \{i,j\} \in \overline{E}$$

the RCP can be stated as:

$$\text{Min} \sum_{\{i,j\} \in \overline{E}} p_{ij} y_{ij}$$

subject to

$$\sum_{k=1}^{c} x_{ik} = 1 \quad \forall i \in \{1, \ldots, n\}$$

$$x_{ik} + x_{jk} \leqslant 1 \quad \forall \{i,j\} \in E \quad \text{and} \quad \forall k \in \{1, \ldots, c\}$$

$$x_{ik} + x_{jk} - 1 \leqslant y_{ij} \quad \forall \{i,j\} \in \overline{E} \quad \text{and} \quad \forall k \in \{1, \ldots, c\}$$

The first and second constraint groups assure that all nodes have been colored with an unique color and that two adjacent nodes cannot be equally colored. The third constraint group assigns the values $y_{ij} = 1$ for any endpoints of a complementary edge equally colored.

Taking into account that $\bar{m} = |\overline{E}| = (n(n-1)/2) - m$ the number of binary variables is $nc + \overline{m}$ and the number of constraints is $n + cm + c\overline{m} = n + c(n(n-1)/2)$.

The binary programming model of Example 2.1, with $c = 4$, is the following:

$$\begin{aligned}\mathrm{Min}\{&0.0519y_{15} + 0.0128y_{16} + 0.6083y_{24} + 0.3621y_{25}\\ &+ 0.1648y_{34} + 0.0256y_{36}\}\end{aligned}$$

subject to

$$x_{11} + x_{12} + x_{13} + x_{14} = 1$$

$$\cdots\cdots\cdots$$

$$x_{61} + x_{62} + x_{63} + x_{64} = 1$$

$$x_{11} + x_{21} \leqslant 1 \cdots x_{14} + x_{24} \leqslant 1$$

$$x_{11} + x_{31} \leqslant 1 \cdots x_{14} + x_{34} \leqslant 1$$

$$\cdots\cdots\cdots$$

$$x_{51} + x_{61} \leqslant 1 \cdots x_{54} + x_{64} \leqslant 1$$

$$x_{11} + x_{51} - 1 \leqslant y_{15} \cdots x_{14} + x_{54} - 1 \leqslant y_{15}$$

$$\cdots\cdots\cdots$$

$$x_{31} + x_{61} - 1 \leqslant y_{36} \cdots x_{34} + x_{64} - 1 \leqslant y_{36}$$

The non-null binary variables identifying the most robust 4-coloring C^4_{R} are:

$$x_{12} = 1, \quad x_{24} = 1, \quad x_{31} = 1, \quad x_{43} = 1,$$

$$x_{52} = 1, \quad x_{61} = 1$$

The non-null auxiliary variables identifying the equally colored complementary edges are:

$$y_{15} = 1, \quad y_{36} = 1$$

and the non-null terms of the objective function are:

$$\begin{aligned}p_{15}y_{15} + p_{36}y_{36} &= 0.0519 + 0.0256 = 0.0775\\ &= R(C^4_{\mathrm{R}})\end{aligned}$$

4.2. A genetic algorithm for the RCP

Genetic algorithm (GA), originally developed by Holland [4], proved efficient in solving several combinatorial optimization problems. The GAs (see Chelouah and Siarry [1]) use natural processes, such as selection, crossover and mutation to manage a population or set of valid solutions for the problem. The selection operator determines the individuals to be chosen for mating; crossover between two selected individuals produces two new ones, the sons, which will replace their parents; the mutation alters some characteristics of some (very few) individuals from the population.

The representation of any individual of the population will be a vector of arranged nodes σ. From this vector σ, a c-coloring function will be greedily constructed: for any node, the least feasible color is computed taking into account the colors of the previous adjacent nodes. This process does not guarantee the feasibility of the coloring, eventually more than c colors would be needed, in this case, this invalid coloring must be penalized, let $P_{\mathrm{I}} > 0$ be such a constant which must be high enough for that any invalid coloring is avoided.

Let m be the population size. This parameter is chosen based on the trade off between a small value—a low computational time consuming—and a high value—a broad coverage of the solution space. Empirical results from many authors suggest that population sizes as small as 30 are quite adequate (see Reeves [8]).

Another parameter of the GA is the maximum number of iterations $\mathrm{it_M}$. This parameter can be adjusted from empirical results and in order to decrease the time consumption.

At any iteration $\mathrm{it} \in \{0, 1, \ldots, \mathrm{it}_M\}$, the population is identified by $(\sigma^{\mathrm{it}}_1, \sigma^{\mathrm{it}}_2, \ldots, \sigma^{\mathrm{it}}_m)$.

Each individual σ^{it}_i of the population at iteration *it* induces—through the greedy procedure already outlined—a coloring function C^{it}_i valued as

$$\widehat{R}^{\mathrm{it}}_i = R(C^{\mathrm{it}}_i) + y_{\mathrm{I}}(C^{\mathrm{it}}_i)P_{\mathrm{I}}$$

where

$$R(C) = \sum_{\{i,j\}\in\overline{E}, C(i)=C(j)} p_{ij}$$

$$y_{\mathrm{I}}(C) = \begin{cases} 1 & \text{if } C \text{ is invalid} \\ 0 & \text{otherwise} \end{cases}$$

Since the RCP is stated as a minimization problem, to each individual will be assigned the inverse value:

$$F^{\mathrm{it}}_i = \frac{1}{\widehat{R}^{\mathrm{it}}_i} \quad \forall i \in \{1, \ldots, m\}$$

The overall population at iteration *it* is also valued with

$$TF^{\mathrm{it}} = \sum_{i=1}^{m} F_i^{\mathrm{it}}$$

As is proved by the fundamental *schema theorem* (see Reeves [8]) as *it* increases the expected total value $E(TF^{\mathrm{it}})$ also increases. Moreover, the optimization solution proposed by the algorithm will be the best obtained individual of any iteration σ^* and this solution is updated through the evolution of the population.

In order to improve the quality of the populations at generation *it*, those individuals σ_i^{it} better adapted will be selected, with $i \in \{1, \ldots, m\}$; i.e., with a higher value F_i^{it} and, consequently, a lower value $\widehat{R}_i^{\mathrm{it}}$.

One way to accomplish this objective is through a Monte-Carlo method. The probability of selecting the individual σ_i^{it} will be proportional to its relative value with respect to the total value of its generation:

$$\mathrm{Prob}\{\text{Select individual } \sigma_i^{\mathrm{it}}\} = \frac{F_i^{\mathrm{it}}}{TF^{\mathrm{it}}} \quad \forall i \in \{1, \ldots, m\}$$

Generating m random numbers, with the above probabilities m individuals will be selected; repetitions are allowed.

Given a randomly generated couple from the population, with probability p_{c} these two individuals will be crossed and they will be replaced by their offspring.

Given the two individuals σ and σ' of a selected couple, the crossover operator works in the following way:

1. Let u be an integer uniform random number in the set $\{1, 2, \ldots, n\}$.
2. The first u elements of first (second) offspring are $(\sigma(1), \ldots, \sigma(u))$ $((\sigma'(1), \ldots, \sigma'(u)))$.
3. The last $n-u$ elements of first (second) offspring are $(\sigma'(u+1), \ldots, \sigma'(n))$ $((\sigma(u+1), \ldots, \sigma(n)))$ avoiding the repetitions and maintaining the relative order of σ' (σ).

For instance, let $n = 10$ be the number of groups and let σ and σ' be two parents:

$$\sigma = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10)$$

$$\sigma' = (7, 3, 9, 1, 4, 6, 2, 8, 10, 5)$$

If $u = 3$, then the two offsprings are:

$$\sigma = (1, 2, 3, 4, 6, 8, 10, 5, 7, 9)$$

$$\sigma' = (7, 3, 9, 4, 5, 6, 8, 10, 1, 2)$$

Applying the operator selection and crossover to the population $(\sigma_1^{\mathrm{it}}, \sigma_2^{\mathrm{it}}, \ldots, \sigma_m^{\mathrm{it}})$ a new population $(\sigma_1^{\mathrm{it}+1}, \sigma_2^{\mathrm{it}+1}, \ldots, \sigma_m^{\mathrm{it}+1})$ is obtained.

With probability p_m, each of the individuals of this new population is selected so that two elements $\sigma(k)$ and $\sigma(k')$ of it are permuted. The elements k and k' are integer uniform random numbers in the set $\{1, 2, \ldots, n\}$.

The family of genetic algorithms is characterized by the parameter vector $(m, \mathrm{it}_M, p_c, p_m)$.

5. Computational experiences

A set of random graphs $G_{n,p}$ was generated, where n is the number of nodes and p is the graph density (the probability that an edge exists between two arbitrary nodes). The parameter n varies from 10 to 15. The graph density was fixed as $p = 0.5$. The parameter c varies from 4 to 6. Each complementary edge penalty p_{ij}, with $\{i, j\} \in \overline{E}$, has also been randomly generated with the uniform distribution in the interval [0,1].

The associated RCP have been solved exactly modeling them as binary programming models and using CPLEX 6.5. They have also been solved approximately with a genetic algorithm with the following parameter setting $(m = 20, \mathrm{it}_M = 20, p_c = 0.6, p_m = 0.1)$ and setting also the penalty constant $P_{\mathrm{I}} = 10^4$. A personal computer Pentium III was used.

The computational results are shown in Table 4. For any random graph (row), is identified the input file, the number of nodes n, the number of valid colors c, and the minimum rigidity obtained level $R(C_{\mathrm{R}}^{c})$ and the CPU time in seconds obtained by the exact and approximated algorithms are identified. The symbol $(*)$ indicates those cases where the optimum solution was attained.

Table 4
Comparative computational results

Input file	n	c	Binary programming		GA	
			$R(C_R^c)$	CPU time	$R(C_R^c)$	CPU time
aleato10	10	4	3.548	1.01	(*) 3.548	0.06
aleato10	10	5	2.177	4.04	2.903	0.06
aleato11	11	4	3.445	1.50	(*) 3.445	0.06
aleato11	11	5	2.023	8.80	2.962	0.06
aleato12	12	4	(#)	(#)	(#)	(#)
aleato12	12	5	2.904	5.92	2.915	0.06
aleato13	13	5	3.587	21.69	(*) 3.587	0.06
aleato14	14	5	5.269	27.25	5.490	0.06
aleato15	15	5	5.637	130.01	(*) 5.637	0.06
aleato15	15	6	3.559	1053.43	4.671	0.06

Note: the symbol # indicates that this problem has no feasible coloring function.

Table 5
High size computational experiences with GAs

Input file	n	c	it_M	CPU time	$R(C_R^c)$
al0050	50	18	50	1.54	46
al0100	100	35	50	5.50	97
al0250	250	70	50	54.32	334
al0250	250	80	50	46.47	280
al0250	250	90	50	39.38	238
al0500	500	200	50	166.14	411
al1000	1000	300	30	656.42	1235

Given the explosive growth of the number of variables and constraints when the dimension of the graph increases, the binary programming model can only solve small or medium size problems. As a consequence, for high size instances for the RCP only heuristics should be applied.

In order to check the validity of the proposed GA, some other graphs have been randomly generated. In these graphs, and without any loss of a generality, the penalty of complementary edges was fixed as 1. Table 5 shows the obtained results for the parameter setting:

$$(m = 20, \text{it}_M = 50(30), p_c = 0.6, p_m = 0.1)$$

In all cases, except for instance of file aleato12, for which there is no feasible 4-coloring, the obtained coloring function is feasible. We can see that the results obtained by genetic algorithms are very good when compared to the exact method (Table 4). Moreover, very high size instances for the RCP can be solved with a very low CPU time consumption (Table 5).

6. Conclusions

The RCP takes into account the complementary edges penalizing them if their extremes were equally colored and they were added to the graph. In this way, other criteria can be considered for classical coloring problems.

For the examination problem, we can look for a *robust* coloring function in the sense that it remains valid under some changes of the graph topology. In this way, the name of the new proposed problem is justified.

For the cluster problem, the most robust c-coloring will compute c clusters, the color classes, in such a way that they remain valid for lower values for the dissimilarity measure threshold.

Also, with a constant penalty for every complementary edges, the most robust c-coloring function will compute the most homogeneous color classes in the sense that their cardinals are very similar.

The RCP can be considered as a coloring graph extension problem and, consequently, it is an NP-hard. This fact has been shown in the computational experiences section. As the graph size increases, only approximate solutions can be obtained. However, all approximate algorithms solving the minimal coloring problem can be easily adapted and very good solutions can be obtained in a few seconds of CPU time.

As a consequence, some other interesting problems, as in the case of scheduling problems, can be stated as RCPs so that the graph-based heuristics could be applied to solve them.

Acknowledgements

Research supported by DGICYT (National grant number PB95-0407) and UCM (grant number PR52/00-8920).

References

[1] R. Chelouah, P. Siarry, A continuous genetic algorithm designed for the global optimization of multimodal functions, Journal of Heuristics 6 (2000) 191–213.
[2] M.R. Garey, D.S. Johnson, Computers and intractability: a guide to the theory of NP-completeness, W.H. Freeman and Company, New York, 1979.
[3] P. Hansen, M. Delattre, Complete-link cluster analysis by graph coloring, Journal of the American Statistical Association 73 (362) (1978) 397–403.
[4] J.H. Holland, Adaptation in Natural and Artificial Systems, University of Michigan Press, Ann Arbor, MI, Internal Report, 1975.
[5] K.-W. Lih, The equitable coloring of graphs, in: D.-Z. Du, P.M. Pardalos (Eds.), Handbook of Combinatorial Optimization (Vol. 3), Kluwer Academic Publishers, Boston, 1998, pp. 543–566.
[6] P.M. Pardalos, T. Mavridou, J. Xue, The graph coloring problem: A bibliographic survey, in: D.-Z. Du, P.M. Pardalos (Eds.), Handbook of Combinatorial Optimization, Kluwer Academic Publishers, Boston, 1998, pp. 331–395.
[7] J. Ramírez Rodríguez, Extensiones del Problema de Coloración de Grafos, Doctoral dissertation, Universidad Complutense de Madrid, Madrid, Spain, 2001.
[8] C.R. Reeves (Ed.), Modern Heuristic Techniques for Combinatorial Problems, Blackwell Scientific Publications, Oxford, 1993.
[9] D. de Werra, Extensions of coloring models for scheduling purposes, European Journal of Operational Research 92 (1996) 474–492.