Discrete Optimization 6 (2009) 362–369

Contents lists available at ScienceDirect

Discrete Optimization

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

Graph coloring with cardinality constraints on the neighborhoods

M.-C. Costa [a], D. de Werra [b], C. Picouleau [c], B. Ries [b,*]

[a] CEDRIC, ENSTA, Paris, France
[b] EPFL, IMA, Lausanne, Switzerland
[c] CEDRIC, CNAM, Paris, France

ARTICLE INFO

Article history:
Received 19 March 2008
Received in revised form 16 April 2009
Accepted 22 April 2009
Available online 20 May 2009

Keywords:
Vertex coloring
Bipartite graph
Tree
Cardinality constrained colorings

ABSTRACT

Extensions and variations of the basic problem of graph coloring are introduced. The problem consists essentially in finding in a graph G a k-coloring, i.e., a partition $V^1, \ldots, V^k$ of the vertex set of G such that, for some specified neighborhood $\tilde{N}(v)$ of each vertex v, the number of vertices in $\tilde{N}(v) \cap V^i$ is (at most) a given integer h_v^i. The complexity of some variations is discussed according to $\tilde{N}(v)$, which may be the usual neighbors, or the vertices at distance at most 2, or the closed neighborhood of v (v and its neighbors). Polynomially solvable cases are exhibited (in particular when G is a special tree).

1. Introduction

Various extensions of the basic graph coloring model (see [1]) have been studied by many authors from a theoretical point of view and also with a motivation stemming from applications in communication systems, operations scheduling, course timetabling, tomography, etc.

Here we shall consider a few variations of the vertex coloring problem which consist essentially in restricting the number of occurrences of the different colors in a given collection $\mathcal{P}$ of subsets P_i of vertices.

In [2], a formulation extending the basic image reconstruction problem in discrete tomography was discussed where the subsets P_i were chains in the underlying graph G. It was motivated by a simple maintenance scheduling problem in a city metro network.

Here we shall essentially consider colorings, i.e., partitions of the vertex set of a graph, such that, in some generalized neighborhood of each vertex x, the number of occurrences of each color i is a given integer h_x^i.

More precisely, we are given an undirected connected graph $G = (V, E)$ with n vertices and m edges. Given two vertices x and y, we denote by $d(x, y)$ the distance between x and y (the length of a shortest x–y path). We denote by $N_d(x)$ the d-neighborhood of $x \in V$ that is the set of vertices y such that $d(x, y) = d$. In the case where $d = 1$ we simply write $N(x)$ for the 1-neighborhood (or neighborhood, as usual) of x, i.e., the set of vertices y such that $[x, y] \in E$. We also define $N_{\leq d}(x) = \cup_{0 \leq l \leq d} N_l(x)$ as the set of vertices at distance at most d from x (with $N_0(x) = \{x\}$).

We are also given a set of colors $1, 2, \ldots, k$ as well as a set $H = \{h(x) = (h_x^1, \ldots, h_x^k) \in \mathbb{N}^k | x \in V\}$.

In the first problem, we have to find a k-partition $V^1, V^2, \ldots, V^k$ of V such that

$$\left| N(x) \bigcap V^i \right| = h_x^i \quad \text{for all } x \in V \text{ and all } 1 \leq i \leq k. \tag{1}$$

* Corresponding author.
E-mail addresses: marie-christine.costa@ensta.fr (M.-C. Costa), dominique.dewerra@epfl.ch (D. de Werra), chp@cnam.fr (C. Picouleau), bernard.ries@a3.epfl.ch (B. Ries).

1572-5286/$ – see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.disopt.2009.04.005

We call this problem $\mathcal{P}(G, H, k)$. In addition, in case we want to obtain a proper coloring (two adjacent vertices must be in two distinct sets V^i and V^j) we let $\mathcal{P}^*(G, H, k)$ denote the corresponding problem.

We will also study the *bounded* version of these problems: we have to find a k-partition $V^1, V^2, \dots, V^k$ of V such that

$$\left|N(x) \bigcap V^i\right| \le h_x^i \quad \text{for all } x \in V \text{ and all } 1 \le i \le k. \tag{2}$$

We will call these problems $\mathcal{BP}(G, H, k)$ and $\mathcal{BP}^*(G, H, k)$, respectively.

Our second problem is to find a k-partition $V^1, V^2, \dots, V^k$ of V such that

$$\left|N_{\le 1}(x) \bigcap V^i\right| = h_x^i \quad \text{for all } x \in V \text{ and all } 1 \le i \le k. \tag{3}$$

We call this problem and its proper coloring version $\mathcal{P}_{\le 1}(G, H, k)$ and $\mathcal{P}^*_{\le 1}(G, H, k)$, respectively.

We will also be interested in $\mathcal{P}_2(G, H, k)$ and $\mathcal{P}_2^*(G, H, k)$, the problems of finding a k-partition, respectively a proper coloring, $V^1, V^2, \dots, V^k$ of V such that

$$\left|N_2(x) \bigcap V^i\right| = h_x^i \quad \text{for all } x \in V \text{ and all } 1 \le i \le k. \tag{4}$$

Notice that our formulation includes the so-called cardinality constrained coloring problem which consists in determining if a graph $G = (V, E)$ has a proper k-coloring $(V^1, \dots, V^k)$ with given cardinality s_i for each color class V^i (see [3–7] for results on this problem): it suffices to take any d larger than or equal to the diameter of G in the set $N_{\le d}(x)$ defined above (since then $\bigcup_{l=0}^d N_l(x) = V$ for each x) with $h_x^i = s_i$ for all x and all $1 \le i \le k$.

These problems are close to the well known $L(h, k)$-Labelling problems (see [8] for a survey). The problem consists in an assignment of nonnegative integers to the vertices of a graph such that adjacent vertices get colors which differ by at least h and vertices joined by a chain of length 2 receive colors differing by at least k (even if there is an edge joining these vertices). Applications to channel assignment or to multihop radio networks are mentioned in [8]. Under the assumption $h_x^i = 1$, for all i and for all x, the colorings of $\mathcal{BP}^*(G, H, k)$ and those of $L(1, 1)$-Labelling satisfy the same requirements: adjacent vertices have different colors and vertices linked by a chain of length 2 (i.e., common neighbors of a single vertex) have different colors. It is also close to the so-called star coloring problem studied in [9], and to the frugal coloring problem studied in [10]. Related work has been carried out recently by several authors (see [11–16]) including dramatic applications of coloring (see [17]).

One should also recall that nonproper coloring models have been used under the name of defective coloring in [18] in a frequency assignment context where interferences had to be minimized. Applications to scheduling are also discussed there.

For graph theoretical terms not defined here, the reader is referred to [1]. For complexity theory, the reader is referred to [19].

Let us denote by $s(z) = \{i : h_z^i > 0\}$, $z \in V$, the set of colors required to occur in $N(z)$. Then the set of possible colors for a vertex x is given by $L(x) = \bigcap_{z \in N(x)} s(z)$. We have the following facts which will be used implicitly in the algorithms of the following sections.

Fact 1.1. *If $\mathcal{P}(G, H, k)$ has a solution, then $L(x) \ne \emptyset$ for all $x \in V$.*

Fact 1.2. *If, for a given $x \in V$, $L(x) = \{i\}$, then in any solution of $\mathcal{P}(G, H, k)$ we have $x \in V^i$.*

Notice that these facts also hold for $\mathcal{P}_{\le 1}(G, H, k)$.

Fact 1.3. *If $\mathcal{P}^*_{\le 1}(G, H, k)$ has a solution, then for every vertex x there is a color i such that $h_x^i = 1$.*

Fact 1.4. *If $\mathcal{P}^*_{\le 1}(G, H, k)$ has a solution, then for each color i and each vertex x such that $h_x^i \ne 1$ we have $x \notin V^i$.*

2. *NP*-completeness results

We shall study here the complexity status of problems $\mathcal{P}(G, H, 2)$, $\mathcal{P}^*(G, H, 3)$, $\mathcal{BP}^*(G, H, 3)$, $\mathcal{BP}^*(G, H, 4)$, $\mathcal{P}_{\le 1}(G, H, 2)$ and $\mathcal{P}^*_{\le 1}(G, H, 3)$.

Theorem 2.1. *$\mathcal{P}(G, H, 2)$ is NP-complete even if G is 3-regular planar bipartite.*

Proof. We use a transformation from the *CUBIC PLANAR MONOTONE 1-in-3SAT* problem which is known to be *NP*-complete (see [20]). In this problem we are given a set X of variables and a set C of clauses of the form $(a \vee b \vee c)$ where a, b and c are distinct variables without negation such that the underlying bipartite graph $G = (X \cup C, E) = (X \cup C, \{[x_i, \hat{c}] | x_i \textit{ occurring in clause } \hat{c} \in C\})$ is 3-regular and planar. The question is to decide whether there exists a truth assignment such that exactly one variable in each clause is true.

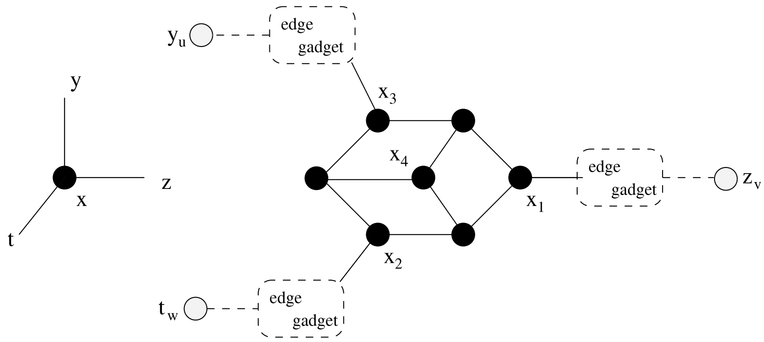

Fig. 1. The vertex gadget replacing a vertex x.

Consider an instance of *CUBIC PLANAR MONOTONE* 1*-in-3SAT* as well as its corresponding graph G. For each vertex $\hat{c}$, representing a clause, we set $h(\hat{c}) = (1, 2)$, and for each vertex x, representing a variable x, we set $h(x) = (3, 0)$.

Consider a positive instance of *CUBIC PLANAR MONOTONE* 1*-in-3SAT*. Then for each variable x, if x is true, we assign x to V^1 and if x is false, we assign x to V^2. All the vertices representing clauses are assigned to V^1. Thus we get a positive answer for the corresponding instance of $\mathcal{P}(G, H, 2)$. Conversely, if an instance of $\mathcal{P}(G, H, 2)$ is positive, then by setting x to true if x has color 1 and to false if x has color 2, the corresponding instance of *CUBIC PLANAR MONOTONE* 1*-in-3SAT* is true: all vertices corresponding to clauses $\hat{c}$ are in V^1 since $h(x) = (3, 0)$ for all vertices x. Every x will be in V^1 or V^2. Since $h(\hat{c}) = (1, 2)$, clause $\hat{c}$ will have exactly one variable x occurring in V^1, i.e., one variable which is true. □

Theorem 2.2. *$\mathcal{P}^*(G, H, 3)$ is NP-complete even if G is* 3*-regular planar bipartite.*

Proof. We use the same reduction as in the proof of Theorem 2.1 except that we take $h(x) = (0, 0, 3)$ for each vertex x representing a variable and $h(\hat{c}) = (1, 2, 0)$ for each vertex $\hat{c}$ representing a clause. Given a positive instance of *CUBIC PLANAR MONOTONE* 1*-in-3SAT*, each variable x which is true is assigned to V^1; it is assigned to V^2 if it is false. All clauses $\hat{c}$ are assigned to V^3. So we obtain a feasible solution of $\mathcal{P}^*(G, H, 3)$. Conversely, if an instance of $\mathcal{P}^*(G, H, 3)$ is positive, all vertices $\hat{c}$ corresponding to clauses are in V^3 since $h(x) = (0, 0, 3)$ for each x representing a variable. Since $h(\hat{c}) = (1, 2, 0)$, exactly one variable x occurring in $\hat{c}$ will be true (x will be in V^1) and two variables in $\hat{c}$ will be false. This will give a positive instance of *CUBIC PLANAR MONOTONE* 1*-in-3SAT*. □

Theorem 2.3. *$\mathcal{BP}^*(G, H, 4)$ is NP-complete even if G is bipartite with maximum degree* 3 *and $h_x^i = 1 \forall x \in V, i = 1, 2, 3, 4$.*

Proof. We use a reduction from the edge-3-coloring problem of a 3-regular graph. This problem is known to be *NP*-complete (see [21]).

Let G' be a 3-regular graph. For each vertex x of G' we introduce the vertex gadget including (among others) vertices x_1, x_2, x_3, x_4 shown in Fig. 1; each edge $[x, y]$ of G' corresponds to a unique edge $[x_u, y_v]$ in the new graph. We replace locally every edge $[x_u, y_v]$ by the edge gadget $J(x_u, y_v)$ given in Fig. 2. The resulting graph $G = (V, E)$ is bipartite and has maximum degree 3. Consider now a coloring κ of V satisfying the constraints of $\mathcal{BP}^*(G, H, 4)$ with $h_x^i = 1 \forall x \in V, i = 1, 2, 3, 4$. Then we clearly have the following two properties:

(i) in any vertex gadget replacing a vertex x, $\kappa(x_1)$, $\kappa(x_2)$, $\kappa(x_3)$, and $\kappa(x_4)$ are all different;
(ii) in any 4-cycle $\{[v_1, v_2], [v_2, v_3], [v_3, v_4], [v_4, v_1]\}$ with neighboring vertices w_1, w_2, w_3, w_4 such that $[v_i, w_i] \in E$, we must have $\kappa(w_1) = \kappa(v_3)$, $\kappa(w_2) = \kappa(v_4)$, $\kappa(w_3) = \kappa(v_1)$, and $\kappa(w_4) = \kappa(v_2)$.

Consider now an edge gadget $J(x_u, y_v)$. W.l.o.g. we may assume that $\kappa(x_4) = 4$ and $\kappa(x_u) = 1$ in the vertex gadget replacing vertex x. By property (ii), we immediately deduce that $\kappa(a) = \kappa(e) = 4$, $\kappa(d) = 1$, and $\kappa(b), \kappa(c) \in \{2, 3\}$. So we may assume w.l.o.g. that $\kappa(b) = 2$ and $\kappa(c) = 3$. Then by repeatedly using property (ii) we get the following: $\kappa(a_1) = \kappa(b') = \kappa(d_2) = 3$, $\kappa(a_2) = \kappa(c') = \kappa(d_1) = 2$. Thus $\kappa(a'), \kappa(d') \in \{1, 4\}$, $\kappa(a') \neq \kappa(d')$. If $\kappa(a') = 4$, then $\kappa(e') = 1$, but this will give us a contradiction, since $\kappa(e) = 4$. Hence $\kappa(a') = 1$ and $\kappa(y_v) = 1$. So we deduce that in any solution of $\mathcal{BP}^*(G, H, 4)$ with $h_x^i = 1 \forall x \in V, i = 1, 2, 3, 4$, and in any edge gadget $J(x_u, y_v)$, x_u and y_v get the same color.

Suppose that an instance of $\mathcal{BP}^*(G, H, 4)$ has a solution *true*. By coloring each edge $[x, y]$ in G' with the color of the corresponding vertices x_u, y_v in G (remember that these two vertices have necessarily the same color $c \in \{1, 2, 3\}$), we get a feasible 3-coloring of the edges of G'.

Now suppose that we have a 3-coloring of the edges of G'. If an edge $[x, y]$ has color $c \in \{1, 2, 3\}$, then color the corresponding vertices x_u, y_v in G with color c. Once we have done this for all the edges in G', we can complete the coloring, as explained above, using at most four colors and satisfying $|N(x) \cap V^i| \leq h_x^i = 1 \forall x \in V, i = 1, 2, 3, 4$. □

Corollary 2.1. *$L(1, 1)$ is NP-complete even in bipartite graphs with maximum degree* 3 *and four colors.*

This result was derived in the context of total colorings in [22].

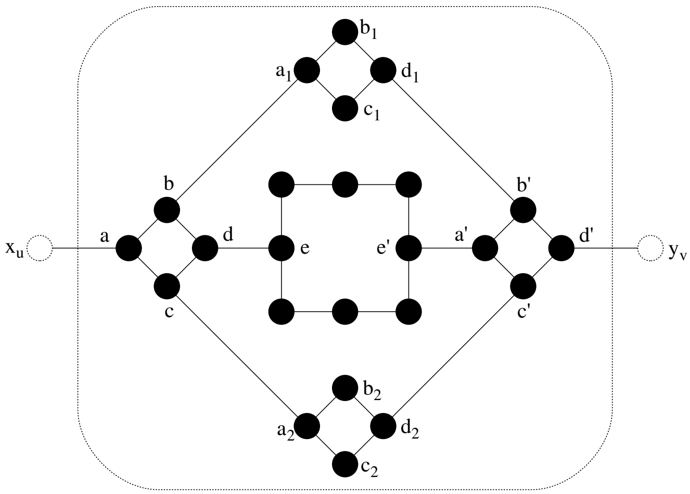

Fig. 2. The edge gadget $J(x_u, y_v)$ corresponding to an edge $[x_u, y_v]$.

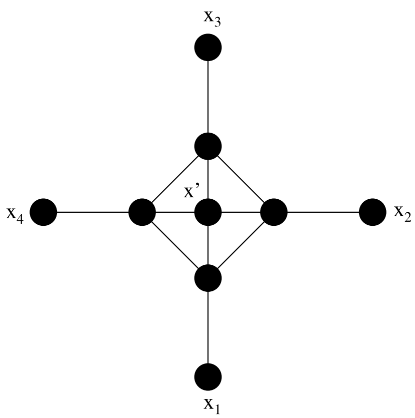

Fig. 3. The vertex gadget replacing a vertex x.

We will need the following Lemma in the proof of Theorem 2.4.

Lemma 2.1. *$\mathcal{BP}^*(G, H, 3)$ is NP-complete even if G is planar with maximum degree 4 and $h_x^i = 2\,\forall x \in V, i = 1, 2, 3$.*

Proof. We use a reduction from the problem of 3-coloring a planar graph with maximum degree 4. This problem is known to be *NP*-complete (see [23]). Let G' be a planar graph with maximum degree 4. We replace each vertex x by the vertex gadget shown in Fig. 3 and an edge $[x, y]$ in G' will be replaced by a suitable edge $[x_u, y_v]$, $u, v \in \{1, 2, 3, 4\}$. We obtain a planar graph G with maximum degree 4.

Now suppose that there is a 3-coloring of G such that $|N(x) \cap V^i| \le 2\,\forall x \in V, i = 1, 2, 3$. Necessarily x_1, x_2, x_3 and x_4 must be colored with the same color as x'. Coloring the corresponding vertex x in G' with this color will give us a 3-coloring of G'.

Conversely, suppose we have a 3-coloring of the vertices of G'. If x has color c, then color the corresponding vertices x', x_1, x_2, x_3, x_4 with this same color c in G. Then the remaining vertices can be colored using three colors in such a way that $|N(x) \cap V^i| \le 2\,\forall x \in V, i = 1, 2, 3$. So we get a positive solution for the instance of $\mathcal{BP}^*(G, H, 3)$. □

Theorem 2.4. *$\mathcal{BP}^*(G, H, 3)$ is NP-complete even if G is planar bipartite with maximum degree 4 and $h_x^i = 2\,\forall x \in V$, $i = 1, 2, 3$.*

Proof. We use a transformation from $\mathcal{BP}^*(G', H, 3)$ which is *NP*-complete when G' is planar with maximum degree 4 and $h_x^i = 2\,\forall x \in V, i = 1, 2, 3$, as shown in Lemma 2.1. Let G' be a planar graph with maximum degree 4. We replace each edge $[x, y]$ by the edge gadget shown in Fig. 4. We obtain a planar bipartite graph G with maximum degree 4. Now suppose that there is a 3-coloring of G such that $|N(x) \cap V^i| \le h_x^i = 2\,\forall x \in V, i = 1, 2, 3$. Denote by c this coloring. We must have $c(a) = c(b)$, since otherwise all vertices in $N(a) \cap N(b)$ should have the same color, which would violate the requirements on $h_a^i = h_b^i = 2$; similarly $c(e) = c(f)$. So let $c(a) = c(b) = 1$ and $c(e) = c(f) = 2$. We must have $c(g) = c(x) = 3$; then

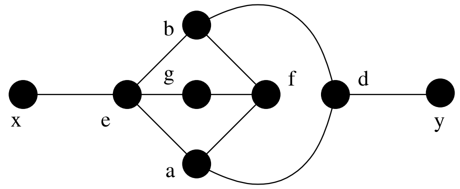

Fig. 4. The edge gadget replacing an edge $[x, y]$.

$c(d) \neq c(a) = 1$ since $d \in N(a)$ and $c(d) \neq c(f) = 2$ since $h_a^2 = 2$, so $c(d) = 3 = c(x) = c(g)$. Finally, $c(y) \neq c(d) = 3$ ($y \in N(d)$), $c(y) \neq 1$ (since $h_d^1 = 2$), so $c(y) = 2 = c(e) = c(f)$. Thus x and y get different colors. Coloring the vertices x, y in G' with the color they get in G, we obtain a 3-coloring of G'. In fact, since $c(e) = c(y)$ and $|N(x) \cap V^i| \leq 2$, $i = 1, 2, 3$, in G, we will obtain a solution in G' satisfying the constraints $|N(x) \cap V^i| \leq 2\, \forall x \in V$, $i = 1, 2, 3$.

Conversely, suppose that there is a 3-coloring of G' with $|N(x) \cap V^i| \leq 2\, \forall x \in V$, $i = 1, 2, 3$. Then by coloring the corresponding vertices in G with the same colors and by applying the rules mentioned above for the remaining vertices, we get a feasible 3-coloring of G. □

Theorem 2.5. *$\mathcal{P}_{\leq 1}(G, H, 2)$ is NP-complete even if G is planar bipartite of maximum degree* 4.

Proof. We use a transformation from $\mathcal{P}(G', H, 2)$ for a 3-regular planar bipartite graph G' (see Theorem 2.1). From G' we build a graph G as follows: for each vertex x' of G', we introduce a new vertex x; x and x' are linked by the edge $[x, x']$; every edge $[x', y']$ of G' is also an edge of G. Thus G is planar bipartite with maximum degree 4. Now, for each new vertex x we set $h(x) = (1, 1)$, and if we have $h(x') = (a, b)$ in the instance of $\mathcal{P}(G', H, 2)$ we set $h(x') = (a + 1, b + 1)$ for its corresponding instance $\mathcal{P}_{\leq 1}(G, H, 2)$. Let V^1, V^2 be a 2-coloring of G'; then we obtain a 2-coloring for G as follows: the twin x of x' is introduced into V^2 if $x' \in V^1$, and vice versa. Conversely, if we have a 2-coloring of G, then by deleting the new vertices we obtain a 2-coloring of G'. □

Theorem 2.6. *$\mathcal{P}^*_{\leq 1}(G, H, 3)$ is NP-complete even if G is planar bipartite of maximum degree* 4.

Proof. We use a reduction from *CUBIC PLANAR MONOTONE* 1-*in-3SAT*. Let G be the 3-regular planar bipartite graph associated with this problem. For each vertex x in G representing a variable, we introduce a new vertex x' and an edge $[x, x']$. We obtain a planar bipartite graph with maximum degree 4. We set $h(x) = (1, 1, 3)$, $h(x') = (1, 1, 0)$, and for the vertices $\hat{c}$ representing the clauses we set $h(\hat{c}) = (1, 2, 1)$.

Suppose that an instance of *CUBIC PLANAR MONOTONE* 1-*in-3SAT* has a solution *true*. Then for each variable x which is true, we assign x to V^1 and x' to V^2, and for each variable x which is false, we assign x to V^2 and x' to V^1. All the vertices $\hat{c}$ representing a clause are assigned to V^3. Thus we get a positive answer to the corresponding instance of $\mathcal{P}^*_{\leq 1}(G, H, 3)$.

Conversely, assume that an instance of $\mathcal{P}^*_{\leq 1}(G, H, 3)$ has a value *true*; then, since $h(x') = (1, 1, 0)$, vertices x, x' cannot be in V^3; one will be in V^1, and the other in V^2. Since every x must have exactly three neighbors in V^3, all vertices $\hat{c}$ representing clauses are necessarily in V^3. Setting x to true if x has color 1 and to false if x has color 2, we get a positive answer to the instance of *CUBIC PLANAR MONOTONE* 1-*in-3SAT*. □

3. The special case of trees

We shall now give a general dynamic programming algorithm which will show that $\mathcal{P}(G, H, k)$, $\mathcal{P}^*(G, H, k)$, $\mathcal{P}_{\leq 1}(G, H, k)$, $\mathcal{P}^*_{\leq 1}(G, H, k)$, $\mathcal{BP}(G, H, k)$ and $\mathcal{BP}^*(G, H, k)$ can be solved in polynomial time when G is a tree. A version adapted to $\mathcal{P}(G, H, k)$ will be described and we will show later how it can be modified to handle the other problems.

We consider a tree $T = (V, E)$ on n vertices. We root T at an arbitrary leaf r, i.e., a vertex of degree 1. For any vertex x of T we denote by $T(x)$ the subtree of T rooted at vertex x. By extension $T(x)$ will also be the set of vertices in $T(x)$. Let $f(x)$ denote the father of x, $x \neq r$, and let $S(x)$ denote the set of sons of x in T. Also, let $T'(x)$, $x \neq r$, be the subtree of T with vertex set $T(x) \cup \{f(x)\}$. Now we define for each vertex $x \neq r$ a set $F(x) = \{(b, c) : \exists$ a coloring κ of $T'(x)$ such that $\kappa(x) = b, \kappa(f(x)) = c\}$. If $F(x) = \emptyset$ for some vertex x, then clearly there is no solution to $\mathcal{P}(G, H, k)$.

If x is a leaf in the rooted tree, then $F(x) = \{(b, c) : b \in s(f(x)), h_x^c = 1\}$; note that the set $F(x)$ can be determined in constant time. In order to determine $F(x)$ for any vertex x which is neither a leaf nor the root r, we shall use an auxiliary graph. Given such a vertex x, we define for each $b \in L(x)$, and each $c \in L(f(x))$ a bipartite graph $B(x, b, c)$ as follows: $B(x, b, c) = (V_1, V_2, E)$ with $V_1 = S(x)$, $V_2 = W_1 \cup W_2 \cup \cdots \cup W_k$, where $W_i = \{i_j : j = 1, 2 \ldots, h_x^i\}$ for $i \neq c$, and $W_c = \{c_l : l = 1, \ldots, h_x^c - 1\}$. We introduce an edge $[z, w]$, $z \in V_1$, $w \in V_2$, if and only if $(a, b) \in F(z)$ and $w \in W_a$. Then clearly a coloring κ of $T'(x)$ with $\kappa(x) = b$ and $\kappa(f(x)) = c$ corresponds to a perfect matching in $B(x, b, c)$.

Thus $F(x)$, $x \neq r$, can be characterized recursively as follows:

(i) if x is a leaf, then $F(x) = \{(b, c) : b \in s(f(x)), h_x^c = 1\}$;
(ii) otherwise $F(x) = \{(b, c) : \exists$ perfect matching in $B(x, b, c)\}$.

Then we get the following algorithm:

Algorithm. 1. Number the vertices in reverse order of Breadth First Search (the leaves come first, the root is at the end). Let $x_1, \ldots, x_n$ be the vertices.
2. For $i = 1$ to $n-1$ compute $F(x_i)$. If $F(x_i) = \emptyset$ for some vertex x_i, there is no solution to $\mathcal{P}(T, H, k)$.
3. If there exists c such that for each $x \in S(r)$ $(c', c) \in F(x)$, then there exists a coloring κ such that $\kappa(r) = c$; else there is no solution to $\mathcal{P}(G, H, k)$
4. Construct the feasible coloring of $\mathcal{P}(T, H, k)$ starting from the root r and recalling the pairs $(c, c') \in F(x_i)$ for $i = 1, \ldots, n-1$.

Theorem 3.1. *The above algorithm solves problem $\mathcal{P}(T, H, k)$ in $O(k^2 n^{2.5})$ time.*

Proof. When $(c, c') \in F(x)$ it means that there is a feasible solution for the problem associated with the subtree $T(x)$ where x has color c and its father $y = f(x)$ has color c'. Since, for each x, all pairs (c, c') are examined we will obtain a solution whenever one exists. If there exists c such that for each $x \in S(r)$ $(c', c) \in F(x)$, assign color c to r; then for each arc (y, x) where y is colored with color c (x is not yet colored) and $(c', c) \in F(x)$, assign color c' to x; x is then colored.

Let us now analyse the complexity of this dynamic programming approach. For each vertex x in T we have $O(k^2)$ pairs of colors (c, c') for which we have to check whether they belong to $F(x)$. A perfect matching can be determined in $O(n^{2.5})$ in a bipartite graph with n vertices (see [24]). In our case the auxiliary bipartite graph $B(x, b, c)$ which we construct for a vertex x of T contains $2(d(x)-1)$ vertices, where $d(x) = |N(x)|$, and hence a perfect matching can be computed in $O(d(x)^{2.5})$ time. Thus the values of F for each vertex and each pair of colors can be obtained in $O(k^2 \sum_{x \in T} d(x)^{2.5})$ time, i.e., our algorithm has a complexity of $O(k^2 n^{2.5})$. □

We will now explain how the previous algorithm can be adapted to the problems $\mathcal{P}^*(G, H, k)$, $\mathcal{P}_{\leq 1}(G, H, k)$, $\mathcal{P}^*_{\leq 1}(G, H, k)$, $\mathcal{BP}(G, H, k)$ and $\mathcal{BP}^*(G, H, k)$:

- $\mathcal{P}^*(G, H, k)$
  We just have to add the constraint that $b \neq c$ in the definition of F; in this way we avoid having two adjacent vertices which will be colored with the same color.
- $\mathcal{P}_{\leq 1}(G, H, k)$
  First we have to adapt the definition of $L(x)$, i.e., $L(x) = \bigcap_{z \in N_{\leq 1}(x)} s(z)$. Then we must modify the computation of F in the following way:
  1. if x is a leaf, $(c, c') \in F(x)$ iff
     (a) $h_x^c = h_x^{c'} = 1$, with $c \neq c'$
     or
     (b) $h_x^c = 2$, with $c = c'$
  2. if x is not a leaf, $(c, c') \in F(x)$ iff
     $\forall z \in S(x)$ there exists a color c'' such that $(c'', c) \in F(z)$ and there exists a partition $U_1, U_2, \ldots, U_k$ of $S(x)$ such that
     (a) $|U_i| = h_x^i$ if $i \neq c, c'$
     (b) $|U_c| = h_x^c - 1$, and $|U_{c'}| = h_x^{c'} - 1$, if $c \neq c'$
     (c) $|U_c| = h_x^c - 2$, if $c = c'$.

  In the auxiliary graph $B(x, b, c)$ constructed as before we introduce $h_x^c - 1$ vertices for color c (instead of h_x^c as used in $\mathcal{P}(G, H, k)$).
- $\mathcal{P}^*_{\leq 1}(G, H, k)$
  We use the version for $\mathcal{P}_{\leq 1}(G, H, k)$ and add the constraint that $b \neq c$ in the definition of F.
- For all bounded problems $\mathcal{BP}$, we adapt the above procedure as follows: instead of constructing a perfect matching in $B(x, b, c)$, we simply determine a matching saturating all vertices in V_1. It need not be a perfect matching since we must have at most h_x^i vertices of color i in the neighborhood of x but not necessarily exactly h_x^i.

4. The case of $\mathcal{P}_2(G, H, k)$ and $\mathcal{P}^*_2(G, H, k)$

Here we will consider a special case of trees for which $\mathcal{P}_2(G, H, k)$ and $\mathcal{P}^*_2(G, H, k)$ can be solved in linear time. We will first give conditions of a solution for a star. We recall that a star $S(y; x_1, \ldots, x_n)$ is a tree with $n \geq 2$ such that $E = \{[y, x_i] : 1 \leq i \leq n\}$. y is the center of the star and the x_i's are the external vertices.

Proposition 4.1. *Given a star $S(y; x_1, \ldots, x_n)$ with a collection H of nonnegative integral vectors $h(x) = (h_x^1, h_x^2, h_x^3, \ldots, h_x^k)$ for each external vertex x, the following statements are equivalent:*

(a) *$\{x_1, \ldots, x_n\}$ has a unique coloring with h_i vertices of color i;*
(b) (1) *for each external vertex x, $h_x^1 + h_x^2 + h_x^3 + \cdots + h_x^k = n - 1$;*
  (2) *for each color i,*
    $n - h_i$ external vertices x have $h_x^i = h_i$ and
    h_i vertices x have $h_x^i = h_i - 1$;
  (3) *for each color i let $V(i) = \{x | h_x^i = h_i - 1\}$; then $V(i) \cap V(j) = \emptyset$ for all i, j with $i \neq j$.*

Proof. (a) $\Rightarrow$ (b): $\sum_{i=1}^{k} h_x^i$ is the number of colors (with their multiplicities) which have to occur at distance 2 from x. Since $|N_2(x)| = n-1$ for each external vertex x, (1) holds. An external vertex of color i (resp. color $j \neq i$) will have $h_i - 1$ (resp. h_i) vertices at distance 2 with color i, so (2) will hold. The set of external vertices with color i will be $V(i)$, and (3) holds.

(b) $\Rightarrow$ (a): For each i we color the h_i vertices x of $V(i)$ with color i and this will give us the required coloring which is uniquely defined. □

Remark 4.1. If G is a star, then the treatments of $\mathcal{P}_2(G, H, k)$ and $\mathcal{P}_2^*(G, H, k)$ are similar. We just have to assign any color $c \in \{1, \dots, k\}$ to the central vertex y for $\mathcal{P}_2(G, H, k)$ and any color $c \in \{1, \dots, k\}$ not used in $N(y)$ (if there is one) for $\mathcal{P}_2^*(G, H, k)$.

Remark 4.2. $\mathcal{P}_2(G, H, k)$ when G is a star with $n \geq 2$ external vertices is the same problem as $\mathcal{P}(G', H, k)$ when G' is a complete graph of order n; if we consider the pairs of external vertices x_p, x_q $(1 \leq p, q \leq n)$ in a star, they are all at distance 2. In a complete graph G' all pairs of vertices are at distance 1. Hence the announced equivalence.

For a special case of trees, we give a complete description of a simple algorithm which will determine in linear time whether a solution exists or not for $\mathcal{P}_2(G, H, k)$.

We define a *quaternary tree* (or shortly *quatree*) as a tree where all internal vertices (i.e., non leaves) have degree at least 4. Let (B, W) be the bipartition of the vertex set V (B is the set of black vertices and W of white vertices). The reader will find more about special trees in [25].

A *pendent star* $S_h(y; x_0, x_1, \dots, x_n)$ in a quatree Q is the subgraph induced by the vertex set $\{y\} \cup N(y)$ where $N(y) = \{x_0, x_1, \dots, x_n\}$ and $x_1, \dots, x_n$ are leaves of Q. Q being a quatree, we have $n \geq 3$. So S_h is a star for which at least three external vertices are leaves of Q. Notice that x_0 is generally not a leaf (except when Q itself is a star).

Proposition 4.2. *Let $S_h(y; x_0, x_1, \dots, x_n)$ be a pendent star. A necessary condition for a coloring of $N(y)$ to exist is that for any two external vertices x_p, x_q either $h(x_p) = h(x_q)$ or $|h_{x_p}^c - h_{x_q}^c| \leq 1$ for each color c and there are exactly two colors, say c and c', such that $h_{x_p}^c \neq h_{x_q}^c$ and $h_{x_p}^{c'} \neq h_{x_q}^{c'}$.*

Proof. As for the case of a star (see proof of Proposition 4.1) in any coloring there is no pair of external vertices x_p, x_q with $|h_{x_p}^c - h_{x_q}^c| \geq 2$ for some color c. We have necessarily $\sum_{i=1}^{k} h_x^i = n$, so we cannot have exactly one color c such that $h_{x_p}^c \neq h_{x_q}^c$. Now suppose that there are at least three colors c_1, c_2, c_3 with $h_{x_p}^{c_i} \neq h_{x_q}^{c_i}$, $i \in \{1, 2, 3\}$. As for the case of a star (see the proof of Proposition 4.1), if $h_{x_p}^{c_i} = h_{x_q}^{c_i} - 1$, x_p must have color c_i. It follows that x_p or x_q has at least two distinct colors, which is a contradiction. □

Proposition 4.3. *Let $S_h(y; x_0, x_1, \dots, x_n)$ be a pendent star. If there is a coloring of S_h, it is unique.*

Proof. Suppose that the condition of Proposition 4.2 is satisfied.

In the case where $h(x_p) = h(x_q)$ for each $1 \leq p, q \leq n$, each external vertex x has the same color c. Then for each x, $h_x^c = n - 1$ or $h_x^c = n$. In the first case, there is a color $c' \neq c$ such that, for each x, $h_x^{c'} = 1$ and thus x_0 must get color c'. In the second case, all external vertices $x_0, x_1, \dots, x_n$ necessarily have color c.

In the case where there exist two vertices x_p, x_q with $h(x_p) \neq h(x_q)$, there is a color c such that $h_{x_p}^c = h_{x_q}^c - 1$. Thus x_p has necessarily color c. So there is another color c' with $h_{x_p}^{c'} = h_{x_q}^{c'} + 1$ and x_q must have color c'. For each external vertex x_f, $f \neq p, q$, since $h(x_p) \neq h(x_q)$ we have $h(x_f) \neq h(x_p)$ or $h(x_f) \neq h(x_q)$. So as above we obtain the color of vertex x_f. In this way we can assign a color to each external vertex x. If an external vertex x receives two distinct colors, clearly there is no solution. Now, from each vector $h(x)$, we determine a unique color of x_0. If there are distinct colors assigned to x_0, there is no solution; otherwise we obtain a coloring for $x_0, x_1, \dots, x_n$ and this coloring is unique. □

Theorem 4.1. *$\mathcal{P}_2(Q, H, k)$ can be solved in linear time when Q is a quatree. Moreover, if there is a coloring, it is unique.*

Proof. In the following algorithm, we will start by coloring the vertices of W and a similar second run will color the vertices of B. W.l.o.g. we may remove all black leaves for the first run of the algorithm.

Algorithm. 1. $G \leftarrow Q$
2. while $G \neq \emptyset$ or G is not a star
   for each pendent star $S_h(y; x_0, x_1, \dots, x_n)$ do
   2.1 if the condition of Proposition 4.2 is not satisfied then there is no solution
   2.2 color $x_0, x_1, \dots, x_n$ according to $h(x_1), \dots, h(x_n)$
   2.3 if the coloring fails, there is no solution
   2.4 update $h(x_0)$ according to the (unique) coloring constructed
   $G \leftarrow G \setminus \{y, x_1, \dots, x_n\}$
3. if G is a star, then color $x_0, x_1, \dots, x_n$
   if the coloring fails, then there is no solution.

In step 2.2 the unique coloring is obtained as described in the proof of Proposition 4.3.

Applying the algorithm to B, we finally obtain a unique coloring of Q if such a coloring exists.

For each pendent star $S_h(y; x_0, x_1, \ldots, x_d)$, the condition of Proposition 4.2 can be checked in time $O(d(y))$ and its coloring (Proposition 4.3) can be obtained in time $O(d(y))$. It follows that the whole complexity is $O(\sum_y d(y)) = O(n)$ since Q is a quatree. □

From the previous result we conclude the following.

Corollary 4.1. *$\mathcal{P}_2^*(Q, H, k)$ can be solved in linear time when Q is a quatree. Moreover, if there is a coloring, it is unique.*

A (unique) coloring exists if there exist a coloring of the white vertices and a coloring of the black vertices and if both colorings are compatible (no two adjacent vertices get the same color).

We have restricted ourselves to the case of quatrees; this has allowed us to obtain a simple linear algorithm. Notice first that if all internal black vertices in a tree have degree 2, then the problem of coloring the white vertices is equivalent to $\mathcal{P}_1(G', H', k)$, where G' is the tree obtained by removing each black vertex linked to two white vertices w_1, w_2 and introducing an edge $[w_1, w_2]$.

In addition (i.e., besides having all internal black vertices with degree 2), if we have a degree at least 4 for each internal white vertex, then one can solve the coloring problem by using the algorithm of $\mathcal{P}_1(G, H, k)$ for the white vertices and the first run of the algorithm of $\mathcal{P}_2(G, H, k)$ in quatrees for the black vertices.

For the general case where G is a tree, the algorithms proposed here do not seem easy to be adapted to handle this case even if a single color class (B or W) has at the same time internal vertices of degree 2 and internal vertices with degree at least 4.

5. Conclusion

We have studied some problems which could be solved in polynomial time for trees or sometimes for a subclass of trees: the quatrees. These are generally *NP*-complete for more general graphs. It would be interesting to examine some extensions of these problems in the case of general trees; in particular, considering generalized neighborhoods like $N_{\leq d}(v)$ (with $d \geq 2$) could lead to further results.

Acknowledgements

This research was carried out when M.-C. Costa and Ch. Picouleau were visiting EPFL, and when D. de Werra and B. Ries were visiting CNAM in 2006 and 2007. The support of both institutions is gratefully acknowledged. Furthermore, the authors express their gratitude to the reviewers, whose comments have improved the presentation of some proofs in the paper.

References

[1] C. Berge, Graphes, Gauthier-Villars, Paris, 1983.
[2] C. Bentz, M.C. Costa, D. de Werra, C. Picouleau, B. Ries, On a graph coloring problem arising from discrete tomography, Networks 51 (4) (2008) 256–267.
[3] B.S. Baker, E.G. Coffman, Mutual exclusion scheduling, Theoretical Computer Science 162 (1996) 225–243.
[4] B.-L. Chen, K.-W. Lih, Equitable coloring of trees, Journal of Combinatorial Theory, Series B 61 (1) (1994) 83–87.
[5] D. de Werra, A. Hertz, D. Kobler, N.V.R. Mahadev, Feasible edge colorings of trees with cardinality constraints, Discrete Mathematics 222 (2000) 61–72.
[6] P. Hansen, A. Hertz, J. Kuplinsky, Bounded vertex colorings of graphs, Discrete Mathematics 111 (1993) 305–312.
[7] M. Jarvis, B. Zhou, Bounded vertex coloring of trees, Discrete Mathematics 232 (2001) 145–151.
[8] T. Calamoneri, The $L(h, k)$-labelling problem: A survey and annotated bibliography, The Computer Journal 49 (5) (2006) 585–608.
[9] G. Fertin, A. Raspaud, B. Reed, Star coloring of graphs, Journal of Graph Theory 47 (3) (2004) 163–182.
[10] H. Hind, M. Molloy, B. Reed, Colouring a graph frugally, Combinatorica 17 (4) (1997) 469–482.
[11] E. Győri, M. Horňák, C. Palmer, M. Woźniak, General neighbour-distinguishing index of a graph, Discrete Mathematics 308 (2008) 827–831.
[12] F. Havet, M.-L. Yu, $(p, 1)$-Total Labelling of graphs, Discrete Mathematics 308 (2008) 496–513.
[13] F. Kramer, H. Kramer, A survey on the distance-colouring of graphs, Discrete Mathematics 308 (2008) 422–426.
[14] B. Omoomi, A. Pourmiri, Local colourings of Kneser graphs, Discrete Mathematics 308 (24) (2008) 5922–5927.
[15] J. Williams, F. Holroyd, Semi-total graph colourings, the beta parameter, and total chromatic number, Discrete Mathematics 308 (2008) 940–954.
[16] S. Zhou, A distance-labelling problem for hypercubes, Discrete Applied Mathematics 156 (15) (2008) 2846–2854.
[17] G. May, Castigat ridendo mores. Revue in Luxemburg 1896–1940, Ein Kapitel nationaler Theatergeschichte, Nos Cahiers, Lëtzebuerger Zäitschrëft fir Kultur, 6. Joër, 1985, pp. 38–54.
[18] L. Cowen, W. Goddard, C.E. Jesurum, Defective coloring revisited, Journal of Graph Theory 23 (3) (1997) 205–219.
[19] M.R. Garey, D.S. Johnson, Computers and Intractability, a Guide to the Theory of NP-Completeness, Freeman, New York, 1979.
[20] C. Moore, J.M. Robson, Hard tiling problems with simple tiles, Discrete & Computational Geometry 26 (2001) 573–590.
[21] I. Holyer, *NP*-completeness of edge-coloring, SIAM Journal on Computing 10 (1981) 718–720.
[22] C.J. McDiarmid, A. Sanchez-Arroyo, Total colouring regular bipartite graphs is $\mathcal{NP}$-hard, Discrete Mathematics 124 (1994) 155–162.
[23] D.P. Dailey, Uniqueness of colorability and colorability of planar 4-regular graphs are NP-complete, Discrete Mathematics 30 (1980) 289–293.
[24] R.K. Ahuja, T.L. Magnanti, J.B. Orlin, Network Flows, Prentice-Hall, Englewood Cliffs, 1993.
[25] E. van Win, Famille Vandievoet à Schaerbeek au XVIII ème siècle, L'intermédiaire des généalogistes 195 (1978) 255–256.