Weighted Coloring on P_4-sparse Graphs

Julio Araujo[1,2] Claudia Linhares Sales[1] Ignasi Sau[3]

***Abstract*—Given an undirected graph $G = (V, E)$ and a weight function $w : V \to \mathbb{R}^+$, a vertex coloring of G is a partition of V into independent sets, or color classes. The *weight* of a vertex coloring of G is defined as the sum of the weights of its color classes, where the weight of a color class is the weight of a heaviest vertex belonging to it. In the Weighted Coloring problem, we want to determine the minimum weight among all vertex colorings of G [1]. This problem is NP-hard on general graphs, as it reduces to determining the chromatic number when all the weights are equal. In this article we study the Weighted Coloring problem on P_4-*sparse* graphs, which are defined as graphs in which every subset of five vertices induces at most one path on four vertices [2]. This class of graphs has been extensively studied in the literature during the last decade, and many hard optimization problems are known to be in P when restricted to this class. Note that cographs (that is, P_4-free graphs) are P_4-sparse, and that P_4-sparse graphs are P_5-free. The Weighted Coloring problem is in P on cographs [3] and NP-hard on P_5-free graphs [4]. We show that Weighted Coloring can be solved in polynomial time on a subclass of P_4-sparse graphs that strictly contains cographs, and we present a 2-approximation algorithm on general P_4-sparse graphs. The complexity of Weighted Coloring on P_4-sparse graphs remains open.**

I. I

In this paper, we adopt the graph terminology defined in [5]. Additionally, the definition and classical results about the modular decomposition of graphs can be found in [6].

The classical V C problem is one of the most studied problems in graph theory, due to its many applications in both theoretical and practical domains. Given a graph $G = (V, E)$, a *(vertex) k-coloring* of G is a function $c : V \to \{1, \ldots, k\}$ that associates to each vertex $v \in V$ a color $c(v)$ such that if $(u, v) \in E$, then $c(u) \neq c(v)$. The minimum integer k such that a graph G admits a k-coloring is the *chromatic number* of G, denoted by $\chi(G)$. A k-coloring can also be seen as a partition $\mathcal{S} = (S_1, \ldots, S_k)$ of the vertex set into color classes.

Given a vertex weighted graph $G = (V, E, w)$, the weight of a vertex coloring of G is given by the sum of the weights of its color classes, where the weight of a color class is the weight of a heaviest vertex belonging to it. In the W C problem, we want to determine the minimum weight among all the colorings of G. The *weighted chromatic number*, denoted by $\chi_w(G)$, of a graph G is the value of the minimum weight of a coloring of G. The definition of this problem was motivated by the Distributed Dual Bus Network Media Access Control Protocol, which is a standard IEEE802.6 for metropolitan networks [1].

The W C problem is a generalization of the classical V C problem and hence it is also NP-hard. In fact, it is hard even for classes where V C can be easily solved [3], [4], [7].

In this paper, we consider vertex-weighted P_4-sparse graphs. A graph G is called P_4-sparse if every 5 vertices of $V(G)$ induce at most one P_4 [2]. This class of graphs was first studied by Jamison and Olariu [8]–[11].

The class of P_4-sparse graphs is strictly contained in the class of P_5-free graphs (for which W C is NP-hard [3]) and strictly contains P_4-free graphs, or cographs (for which W C is polynomial [4]).

There are a number of optimization problems that can be solved in polynomial time on P_4-sparse graphs [10], [12]. The algorithms that solve these problems usually calculate the desired parameter in a simple post-order traversal in the modular decomposition tree of the graph, which can be found in linear time [13] for any graph. We use the same approach to determine the weighted chromatic number for a subclass of P_4-sparse graphs. Recall that in a modular decomposition tree of any graph, each node either is *series*, which means that there is a *complete join* between the modules defined by its children, or is *parallel*, which means that there is a *disjoint union* between the modules defined by its children, or is *neighborhood*, which means that the quotient graph of the modules defined by its children and its complement are connected.

P_4-sparse graphs can be characterized by their modular decomposition. In order to present this result, we need to define a *spider* graph.

Definition 1.1: A *spider* is a graph whose vertex set can be partitioned into disjoint sets S, K, and R such that:

1) $|S| = |K| \geq 2$, S is a stable set, K is a clique;
2) Every vertex in R is adjacent to all the vertices in K and to no vertex in S;
3) There exists a bijection $f : S \longrightarrow K$ such that either the spider is of Type 1 (called *thick* spider), i.e.:

$$N_G(s) \cap K = K - \{f(s)\}, \text{ for all vertices } s \in S;$$

or it is of Type 2 (called *thin* spider), i.e.:

$$N_G(s) \cap K = \{f(s)\}, \text{ for all vertices } s \in S.$$

Observe that the unique non-trivial maximal strong sub-module of a spider is exactly the set R.

[1]Research Group ParGO. Federal University of Ceará - Fortaleza - BRAZIL. {juliocesar, linhares}@lia.ufc.br
[2]Mascotte joint Project - INRIA/CNRS-I3S/UNSA - Sophia-Antipolis, FRANCE. julio-cesar.silva_araujo@sophia.inria.fr
[3]Technion, Department of Computer Science, Haifa, ISRAEL. ignasi@cs.technion.ac.il

Theorem 1.1 ([14]): G is a P_4-sparse graph if, and only if, the quotient graph of each neighborhood node of its modular decomposition tree $T(G)$ is isomorphic to a spider $H = (S \cup K \cup R, E)$.

In Section II, we present the main results of this paper. In Section III, we show that there exists a 2-approximation algorithm for W C on P_4-sparse graphs. Finally, we propose a conjecture in Section IV.

II. A P -T A

It is not difficult to see that:

Remark 2.1: Given the weighted chromatic numbers of two graphs G_1 and G_2, the weighted chromatic number of the graph G obtained by the complete join of G_1 and G_2 is equal to $\chi_w(G) = \chi_w(G_1) + \chi_w(G_2)$.

Our algorithm will traverse the modular decomposition tree of the graph in a post-order way, in order to calculate its weighted chromatic number. Remark 2.1 implies that it is easy to deal with the *series* nodes. The rest of this section is dedicated to the *neighborhood* nodes (spiders) and *parallel* nodes (disjoint union) of P_4-sparse graphs.

A. Spiders

From now on we suppose, unless said otherwise, that $G = (V = S \cup K \cup R, E, w)$ is a spider. We prove that an optimal weighted coloring of G can be obtained in polynomial time, provided that we have an optimal weighted coloring of R. We start by making some remarks.

Remark 2.2: We can assume that $w(v) > 0$, for all $v \in V(G)$, since given any coloring c of G, we can put each vertex v with weight zero in a color class consisting only of v, without increasing the weight of c.

Remark 2.3: Without loss of generality, we can suppose that all the non-neighbors of each vertex s_i of S, for all $i = 1, \ldots, |S|$, have weight strictly smaller than $w(s_i)$. Otherwise, given a coloring c of G such that there exists a vertex $s_i \in S$ that does not belong to a color class of one of its heavier non-neighbors, then we can find a coloring c' of G such that $w(c') \leq w(c)$ by recoloring s_i with a color from one of its heavier non-neighbors.

By the definition of a spider, all the edges between the vertices of K and R exist. By consequence, for any l-coloring $\mathcal{S} = \{S_1, \ldots, S_l\}$ of G, there is no class S_i containing vertices from both K and R. We can then define CK (resp. CR), the set of *colors of* K (resp. *colors of* R), as the set whose elements are the color classes that contain at least one vertex of K (resp. one vertex of R). Observe that the sets CK and CR are disjoint.

Lemma 2.1: Given an optimal weighted coloring $\mathcal{S} = \{S_1, \ldots, S_l\}$ of a spider G, the following holds: if $R = \emptyset$, then there exists at most one color class S_i of $\mathcal{S}$, such that $S_i \notin CK$. Otherwise, there is no color class S_i in the set $\mathcal{S} \backslash (CK \cup CR)$.

Proof: If $R = \emptyset$, for otherwise, one could obtain a coloring $\mathcal{S}'$, with weight strictly smaller than $\mathcal{S}$, by merging the color classes that have only vertices from S. In the case $R \neq \emptyset$, again by contradiction, one could merge S_i, for some color $S_i \in \mathcal{S} \backslash (CK \cup CR)$, with some color of CR an obtain a coloring with weight strictly smaller than $\mathcal{S}$. ■

We will denote by the *color of* S, or simply cS, the unique possible color class which does not belong to $CK \cup CR$.

Lemma 2.2: There exists at most one color class S_j from every optimal weighted coloring $\mathcal{S} = \{S_1, \ldots, S_l\}$ of G such that S_j intersects both S and R.

Proof: Suppose, by contradiction, that there are two colors from CR, S_j and S'_j, $j \neq j'$, such that S_j and S'_j contain vertices of S. Moreover, without loss of generality, suppose that $w(S_j) \geq w(S'_j)$. Again, By Remark 2.3, the vertices with the greatest weight in each color class S_j and S'_j belong to S. Thus, the coloring $\mathcal{S}$' obtained from $\mathcal{S}$ by moving the vertices of $S \cap S_j$ to S'_j would have weight strictly smaller than $w(\mathcal{S})$, a contradiction. ■

Now we prove the following lemma to be used in the sequel:

Lemma 2.3: If $R \neq \emptyset$, then given an optimal weighted coloring $\mathcal{S}_R$ of the subgraph of G induced by R, there exists an optimal weighted coloring $\mathcal{S}$ of G that is an extension of $\mathcal{S}_R$.

Proof: Let $\mathcal{S}' = \{S'_1, \ldots, S'_k\}$ be an optimal weighted coloring of G and let S'_i, by Lemma 2.2, be the unique possible color of $\mathcal{S}'$ that contains vertices from both S and R, for some $i \in \{1, \ldots, k\}$.

Observe that S'_i contains a vertex r^* with the maximum weight of a vertex in R (for otherwise, by recoloring all the vertices of $S \cap S'_i$ with the color of a vertex with maximum weight in R, we would obtain a coloring with weight strictly smaller than $w(\mathcal{S}')$).

Let $\mathcal{S}$ be a coloring of G such that the partition of the vertices of R agrees with the partition given by $\mathcal{S}_R$, the vertices of $S \cap S'_i$ are assigned to the same element of the partition of r^*, and the vertices of $\{S \cup K\} - \{S \cap S'_i\}$ maintain the partition given by $\mathcal{S}'$.

Since $\mathcal{S}_R$ is an optimal weighted coloring to $G[R]$, observe that $\mathcal{S}$ is an optimal weighted coloring to G, because in both colorings $\mathcal{S}'$ and $\mathcal{S}$, the color classes of CK and cS, if the latter one exists, are the same. ■

Suppose now that the vertices of S are labeled $S = \{s_1, \ldots, s_m\}$ satisfying $w(s_1) \leq \ldots \leq w(s_m)$. We are ready to prove that:

Lemma 2.4: There exists an optimal weighted coloring c' of G such that exactly one of the following statements holds:

1) There exists an integer j, such that the vertices $s_1, \ldots, s_{j-1}$ are either assigned to color cS or to the color of a heaviest vertex of R, while the vertices $s_j, \ldots, s_m$ are assigned, each one individually, to colors of their non-neighbors in K;
2) The vertices of S are all assigned to cS, or to the color of a heaviest vertex of R, or, each one individually, to a color of one of its non-neighbors in K.

Proof: Observe that, for any vertex $s_i \in S$, either it belongs to cS or it has a color of one of its non-neighbors in K or R.

Consider now an optimal weighted coloring c of G. Let $j \in \{1, \ldots, m, m+1\}$ be the highest index of a heaviest vertex of S that is colored either with a color of R or with the color of S (consider that if $j-1=0$ then there is no vertex with these colors, and if $j-1=m$ then all the vertices of S are colored by colors of R or S). Observe that we can obtain a coloring c' such that $w(c') \leq w(c)$ by assigning to all $s_1, \ldots, s_{j-1}$ the color of s_j. Moreover, if $s_j \in CR$, using similar arguments to those used in Lemma 2.3, we may recolor $s_1, \ldots, s_{j-1}$ with the color of a heaviest vertex of R. ∎

Now, denote by k^* a heaviest vertex of K, by k^{**} a second heaviest vertex of K and by s^* the neighbor of k^*, if the spider G is of type 2.

Lemma 2.5: Let $G=(S \cup K \cup R, E, w)$ be a spider of type 2 and let c' be an optimal weighted coloring of G as described in Lemma 2.4. We can construct a coloring c'' satisfying $w(c'') \leq w(c')$ and such that either:

- the color of the vertices $s_j, \ldots, s_m$ is equal to the color of a heaviest vertex k^* of K, except possibly s^* that would have a color of a second heaviest vertex k^{**}; or
- the color of the vertices $s_j, \ldots, s_m$ is equal to the color of $k_i \neq k^*$, for some vertex $k_i \in K$, except possibly the vertex s_i, the only neighbor of k_i in S, that would have the color of k^*.

Proof: We need to show that we can obtain from an optimal weighted coloring c', a coloring c'' such that $w(c'') \leq w(c')$ and c'' satisfies the lemma conditions. If in the coloring c' no vertex of S has colors of K, then the lemma is trivially true. Otherwise, let $j-1$ be the highest index of a heaviest vertex of S that is colored either with a color of R or with the color of S. To prove the lemma we distinguish the following cases:

1) $c'(s_m) = c'(k^*)$
   a) $s^* \notin \{s_j, \ldots, s_m\}$
      Observe that in this case all the vertices with colors of K in S are not adjacent to k^* and, consequently, they could all receive the color of k^*. Let c'' be the coloring obtained from c' by assigning to all the vertices in the set $\{s_j, \ldots, s_m\}$ the color $c'(k^*)$ of k^*. At first, observe that $w(c'(k^*)) = w(c''(k^*))$, because s_m is a heaviest vertex of S and by hypothesis $c'(s_m) = c'(k^*)$. Moreover, all the other color classes have not increased their weight, because they have just lost some vertices. Then, $w(c'') \leq w(c')$ and c'' satisfies the conditions of the lemma.
   b) $s^* \in \{s_j, \ldots, s_m\}$
      Observe that $c'(s_m) = c'(k^*)$, so $s^* \neq s_m$. Let us construct the coloring c'' in two steps. At first, observe that if we put all the vertices of $\{s_j, \ldots, s_m\} \backslash \{s^*\}$ in the color class $c'(k^*)$ we will not increase the weight of the coloring, because by hypothesis s_m already belongs to $c'(k^*)$.
      If, after this first change, the color of s^* is equal to the color of k^{**}, we have already obtained a coloring satisfying the conditions of the lemma, otherwise assume $c'(s^*) = c'(k_i)$, for some vertex $k_i \neq k^{**}$, $k_i \in K$.
      In this case, we claim that if we recolor s^* with the color $c'(k^{**})$ we will create a coloring c'' such that $w(c'') \leq w(c')$. To show this fact, observe that the color classes that may change their weight by recoloring s^* with $c'(k^{**})$ are $c'(k_i)$ and $c'(k^{**})$. However, by Remark 2.3, $w(c'(k_i)) = w(c''(k^{**}))$, and observe that $w(c'(k^{**})) = w(k^{**}) \geq w(k_i) = w(c''(k_i))$. Finally, c'' satisfies the conditions of the lemma.
2) $c'(s_m) \neq c'(k^*)$
   a) $s^* \notin \{s_j, \ldots, s_m\}$
      Suppose $c'(s_m) = c'(k_i)$. We claim that if we put all the vertices of S with color $c'(k_i)$ in the color class $c'(k^*)$ we will create a coloring c such that $w(c) \leq w(c')$. Again this verification is simple because only the color classes $c'(k_i)$ and $c'(k^*)$ may have their weights modified. Observe that $w(c'(k_i)) = w(s_m) = w(c(k^*))$ and $w(c'(k^*)) \geq w(k^*) \geq w(k_i) = w(c(k_i))$.
      At last, observe that in the coloring c we have $c(s_m) = c(k^*)$ and we are again in the case 1a.
   b) $s^* \in \{s_j, \ldots, s_m\}$
      i) $c'(s^*) \neq c'(s_m)$
         We can repeat the steps of case 2a to find a coloring c from c' such that $w(c) \leq w(c')$ by recoloring all the vertices in S colored $c'(s_m)$ with the color $c'(k^*)$. Then, we obtain a coloring as in the case 1b.
      ii) $c'(s^*) = c'(s_m)$
         Suppose that $c'(s^*) = c'(s_m) = c'(k_i)$, for some $k_i \neq k^*$.
         In this case, observe that we cannot modify the color of s_m to the color $c'(k^*)$, because s^* and s_m have the same color and s^* and k^* are neighbors. We cannot use Remark 2.3 to compare the weights of these vertices and, consequently, to be sure that the weight of the coloring will not increase after moving the vertices in $S \backslash \{s^*\}$ with color $c'(k_i)$ to the color $c'(k^*)$.
         However, as in the case 1, if the only neighbor of k_i in S, say s_i, does not belong to the set $\{s_j, \ldots, s_m\}$, then we can put all the vertices from s_j to s_m in the color $c'(k_i)$ obtaining a coloring c'' satisfying the condition of the lemma.
         If $s_i \in \{s_j, \ldots, s_m\}$, we can use the arguments of the case 1b to conclude that we can assign to all the vertices $\{s_j, \ldots, s_m\} \backslash \{s_i\}$ the color $c'(k_i) = c'(s_m)$ without increasing the weight of the coloring. Moreover, observe that we can assign to s_i the color $c'(k^*)$ generating a coloring c'' satisfying the condition of the lemma, because $w(c'(k_i)) = w(s_i) = w(c''(k^*)$ and $w(c'(k^*)) = w(k^*) \geq w(k_i) = w(c''(k_i))$.

■

We know, by Lemma 2.3, that there is an optimal weighted coloring of G that is an extension of an optimal weighted coloring of R. By Lemmas 2.4 and 2.5 we know that there is an optimal weighted coloring of G satisfying the conditions of both lemmas. Finally, we proved the following:

Lemma 2.6: Let G be a spider and $\mathcal{S}$ be an optimal weighted coloring of G satisfying Lemmas 2.4 and 2.5. Then, the coloring $\mathcal{S}$ when restricted to R is an optimal weighted coloring of $G[R]$.

Proof: If $R = \emptyset$, then the lemma is trivially true. Suppose then, by contradiction that $\mathcal{S}$ does not satisfy the lemma and let $\mathcal{S}_R$ be an optimal weighted coloring to R.

By Lemma 2.2, there is at most one color S_i of $\mathcal{S}$ containing vertices from both S and R. If there is no such color, then a coloring $\mathcal{S}'$ obtained from $\mathcal{S}$ by recoloring all the vertices of R like in the coloring $\mathcal{S}_R$ would have weight strictly smaller than the weight of $\mathcal{S}$. This would be a contradiction to the optimality of $\mathcal{S}$.

Suppose than that there is a color S_i containing vertices from S and R. Thus, by the same arguments of Lemma 2.3, this color contains a vertex r^* with the greatest weight of a vertex of S. Using the same ideas of Lemma 2.3, we may recolor the vertices of R like in $\mathcal{S}_R$ generating a coloring $\mathcal{S}'$ in such a way that $w(\mathcal{S}') < w(\mathcal{S})$. It is just necessary to set the color of r^* to be the same of S_i. Observe that the colors of CK and cS do not change their weights and the sum of the weights of the colors in CR decreases. This is a contradiction to the optimality of $\mathcal{S}$. ■

Proposition 2.1: Given a spider $G = (S \cup K \cup R, E, w)$ and an optimal weighted coloring c_R of $G[R]$, then an optimal weighted coloring of G can be found in $\mathcal{O}(n^3)$ time.

Proof: The algorithm that calculates such a coloring is Algorithm 1. Its correctness follows from Lemmas 2.3, 2.4, 2.5 and 2.6. The vertices of G can be ordered by their weights in $\mathcal{O}(n \log n)$ and the vertices of K and R can be colored in linear time, provided we are given an optimal weighted coloring of $G[R]$. However, to color the vertices of S, we have to try all the colorings satisfying Lemmas 2.4 and 2.5 and this can take $\mathcal{O}(n^3)$ in the case we have a spider of type 2. The proposition follows. ■

Corollary 2.1: Let G be a weighted P_4-sparse graph whose modular decomposition tree $T(G)$ satisfies the following statement: if $T(G)$ contains a parallel node v, then v represents a module that is a cograph. Then an optimal weighted coloring of G can be found in $\mathcal{O}(n^3)$ time.

Proof: At first, the modular decomposition tree of G, $T(G)$, can be found in linear time. Then, we do a pre-order traverse in $T(G)$ by calculating $\chi_w(G[M])$ at each node parallel node m, where M is the module defined by m. Since $G[M]$ is a cograph, this can be done by using the already known algorithm for cographs. Finally, we have to visit $T(G)$ in a post-order way and use Remark 2.1 and Proposition 2.1 to determine $\chi_w(G[M])$ at each series or neighborhood node m of $T(G)$. ■

Algorithm 1: W C of spiders

```
Input: Spider G = (S ∪ K ∪ R, E) and an optimal weighted coloring c_R of G[R]
Output: Optimal weighted coloring of G
m ← |S|;
Create artificial vertices s_0 and s_{m+1} in S and order them such that
  w(s_0) ≤ ... ≤ w(s_{m+1});
Choose k*, k** and r* and define c, c' ← ∅;
foreach r ∈ R do
    c'(r) := c_R(r);
foreach k ∈ K do
    c'(k) := a color among the |K| colors of K;
for j = 1, ..., m + 1 do
    for i = 0, ..., j − 1 do
        if R ≠ ∅ then
            c'(s_i) ← c'(r*);
        else
            c'(s_i) ← cS;
    if Spider G is of type 1 then
        for i = j, ..., m do
            c'(s_i) ← the color of its non-neighbor in K (c'(f(s_i)));
        if w(c') < w(c) then
            c ← c';
    else
        for i = j, ..., m do
            if (s_i, k*) ∉ E(G) then
                c'(s_i) ← c'(k*);
            else
                c'(s_i) ← c'(k**);
        if w(c') < w(c) then
            c ← c';
        foreach k_i ∈ K\{k*} do
            for i = j, ..., m do
                if (s_i, k_i) ∉ E(G) then
                    c'(s_i) ← c'(k_i);
                else
                    c'(s_i) ← c'(k*);
            if w(c') < w(c) then
                c ← c';
Result: c
```

Observe that in Corollary 2.1, we present an algorithm to solve the W C problem for a subclass of P_4-sparse graphs which strictly contains cographs, since its modular decomposition tree may have modules whose the quotient graphs are isomorphic to spiders.

B. Disjoint Union

To illustrate the problem tackled in this section, consider the P_4-sparse graph $G = A \cup B$ of Fig. 1. An optimal coloring c_A of A with weight 5 is given by $S_1 = \{k_1\}$, $S_2 = \{k_2\}$, $S_3 = \{k_3\}$, and $S_4 = \{s_1, s_2, s_3\}$. An obvious optimal coloring c_B of B with weight 6 is given by $S'_1 = \{u_1\}$, $S'_2 = \{u_2\}$, and $S'_3 = \{u_3\}$. If we combine both colorings by merging the color classes of c_A and c_B we obtain a coloring of G with weight 7. But there exists a better coloring c_G of G with weight 6 given by $S''_1 = \{s_1, k_1, u_1\}$, $S''_2 = \{s_2, k_2, u_2\}$, and $S''_3 = \{s_3, k_3, u_3\}$. This optimal coloring c_G restricted to A has weight 6, which is strictly greater than the weight of c_A.

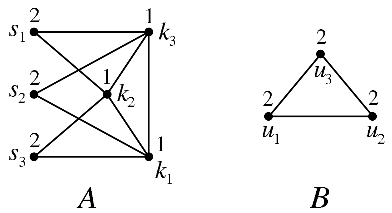

Fig. 1. An optimal weighted coloring of a disjoint union is not given by merging an optimal weighted coloring of each component.

The previous example shows that to compute an optimal weighted coloring of a disjoint union of two graphs, it is not enough to compute an optimal coloring of each component, and then merge the color classes appropriately (as happens for the classical vertex coloring problem). However, we could prove the following:

Proposition 2.2: Given a k-coloring $S = (S_1, \ldots, S_k)$ of a disconnected weighted graph $G = G_1 \cup G_2 \cup \ldots \cup G_m$, such that each G_i is a connected component of G, for all $i \in \{1, \ldots, m\}$, and $w(S_1) \geq \ldots \geq w(S_k)$, we can construct a coloring $S' = (S'_1, \ldots, S'_{k'})$ such that $w(S') \leq w(S)$ and the color class S'_i, restricted to the component G_j, is the i-th heaviest color class of G_j. Moreover, $k' \leq k$.

Proof: We need to introduce some extra notation. For $i = 1 \ldots, m$, let $S^i_1, \ldots, S^i_l$, be the stable sets induced by S on G^i, with $w(S^i_1) \geq \ldots \geq w(S^i_l)$. If $l < k$, for $j = l+1, \ldots, k$ we also consider, with slight abuse of notation, the empty sets S^i_j with $w(S^i_j) = 0$, for all $i \in \{1, \ldots, m\}$.

Given this notation, we claim that:

Claim 1:

$$w(S_i) \geq \max_{j \in \{1, \ldots, m\}} \{w(S^j_i)\}$$

For $j = 1$ the claim is true, since the weight of S_1 is given by the weight of a heaviest vertex in G, which equals $\max\{w(S^1_1), \ldots, w(S^m_1)\}$. Suppose that the claim is not true for some $j > 2$, i.e., $w(S_j) < \max\{w(S^1_j), \ldots, w(S^m_j)\}$. Suppose, without loss of generality, that $\max\{w(S^1_j), \ldots, w(S^m_j)\} = w(S^1_j)$. Then, by hypothesis:

$$\begin{aligned} w(S_j) < \max\{w(S^1_j), \ldots, w(S^k_j)\} = \\ w(S^1_j) \leq w(S^1_{j-1}) \leq \ldots \leq w(S^1_r) \leq \ldots \leq w(S^1_1). \end{aligned} \tag{1}$$

For $r = 1, \ldots, j$, let $S_{q_r} \in \{S_1, \ldots, S_k\}$ be the stable set of $\mathcal{S}$ containing S^1_r. Observe that, by definition, all these sets S_{q_r} are distinct. Then,

$$w(S^1_r) \leq w(S_{q_r}), \ r = 1, \ldots, j. \tag{2}$$

Combining Equations (1) and (2) we deduce that $w(S_j) < w(S_{q_r})$, for each $r = 1, \ldots, j$. In other words, there exist j chromatic classes with weight strictly greater than $w(S_j)$, a contradiction to the hypothesis that $w(S_1) \geq \ldots \geq w(S_k)$. Thus, claim follows.

Define then a coloring S' with color classes $S'_1, \ldots, S'_{k'}$ as follows:

$$S'_j := S^1_j \cup \ldots \cup S^m_j, \ j = 1, \ldots, k'.$$

By the claim, it is not difficult to conclude that $\mathcal{S}'$ satisfies the Proposition. ■

Consider that $\omega = \omega(G)$ is the size of a biggest clique of a graph G. As a consequence of the previous proposition, we can conclude the following corollary:

Corollary 2.2: Let $G = G_1 \cup G_2 \cup \ldots \cup G_m$ be a disconnected weighted graph, such that each connected component $G_i = (S_i, K_i, R_i)$ is a spider with $R_i = \emptyset$, for all $i \in \{1, \ldots, m\}$. Then, there exists an optimal weighted coloring of G with either ω or $\omega + 1$ colors.

Proof: At first, observe that $\omega(G) = \max_{i \in \{1, \ldots, m\}}\{\omega(G_i)\}$. Suppose that $\mathcal{S}' = (S'_1, \ldots, S'_k)$ is an optimal weighted coloring of G. Moreover, observe that each component G_i has at least ω_i colors, for all $i \in \{1, \ldots, m\}$. Let S^*_i be the subset of vertices of S_i colored with colors not belonging to the set of colors used by K_i. Then we may create a coloring $\mathcal{S}''$ by recoloring all the vertices of S^*_i with the color of a vertex with greatest weight in S^*_i, without increasing $w(\mathcal{S}')$, i.e., $w(\mathcal{S}'') \leq (\mathcal{S}')$.

Now, using the previous proposition over the coloring $\mathcal{S}''$, as we have at most $\omega(G) + 1$ colors for each component of G, we may obtain an optimal weighted coloring for G using at most $\omega(G) + 1$ colors.

■

III. A A

To show our approximation algorithm, let us first consider the special partition given by Jamison and Olariu [10], [11]:

Definition 3.1: A graph G has a *special partition* if there exists a family $\Sigma = \{S_1, \ldots, S_q\}$ of disjoint stable sets of G with $q \geq 1$ and $|S_i| \geq 2$, for all $i \in \{1, \ldots, q\}$, and there exists an injection $f : \bigcup_{i=1}^q S_i \longrightarrow V - \bigcup_{i=1}^q S_i$ such that the following occurs:

1) $K_i = \{z \mid z = f(s) \ for \ some \ s \in S_i\}$ is a clique, for all $i \in \{1, \ldots, q\}$;
2) A set of vertices A induces a P_4 in G if, and only if, there exists a subscript $i \in \{1, \ldots, q\}$ and distinct vertices $x, y \in S_i$ such that $A = \{x, y, f(x), f(y)\}$.

Let us define $S = \bigcup_{i=1}^q S_i$ and $K = V - \bigcup_{i=1}^q S_i$. Observe that the graphs induced by S and K are cographs and their weighted chromatic number can be determined in polynomial time [4].

Theorem 3.1 ([8]): A graph is a P_4-sparse graph if, and only if, it is a cograph or it has a special partition.

Then, we can state the following:

Proposition 3.1: There exists a linear time approximation algorithm for W C on P_4-sparse graphs with approximation ratio bounded above by 2.

Proof: We claim that if G and H are weighted graphs such that $H \subseteq G$, then, $\chi_w(H) \leq \chi_w(G)$. For otherwise, if $H \subseteq G$ is a counterexample and c is an optimal weighted coloring of G, by restricting c to the vertices of H, we would obtain a proper coloring c' of H such that $w(c') < \chi_w(H)$. As a consequence of this claim, our approximation algorithm will just color the cographs S and K in linear time with disjoint sets of colors. Once $\chi_w(S) \leq \chi_w(G)$ and $\chi_w(K) \leq \chi_w(G)$, the proof is completed. ■

IV. F R

We finish the paper with the following conjecture:

Conjecture 4.1: There is a polynomial-time algorithm to solve the W C problem on P_4-sparse graphs.

A

This work has been partially supported by National Council for the Scientific and Technological Development - CNPq, Brazil, by the European project IST FET AEOLUS, PACA region of France, by Ministerio de Educación y Ciencia of Spain, European Regional Development Fund under project TEC2005-03575, by the Catalan Research Council under project 2005SGR00256, and by the COST action 293 GRAAL. It also has been done in the context of the C with France Telecom.

R

[1] D. Guan and X. Zhu, "A coloring problem for weighted graphs," *Information Processing Letters*, vol. 61, pp. 77–81, 1997.

[2] C. Hoàng, "Perfect graphs," Ph.D. dissertation, McGill University, 1985.

[3] J. Monnot, V. T. Paschos, D. de Werra, M. Demange, and B. Escoffier, "Weighted Coloring on Planar, Bipartite and Split Graphs: Complexity and Improved Approximation," in *Proceedings of the 15th International Symposium on Algorithms and Computation (ISAAC)*, ser. Lecture Notes in Computer Science, vol. 3341, 2004, pp. 896–907.

[4] D. de Werra, M. Demange, J. Monnot, and V. T. Paschos, "Weighted node coloring: when stable sets are expensive," in *Proceedings of the 28th International Workshop on Graph-Theoretic Concepts in Computer Science (WG)*, ser. Lecture Notes in Computer Science, vol. 2573, 2002, pp. 114–125.

[5] J. A. Bondy and U. S. R. Murty, *Graph Theory*, ser. Graduate Texts in Mathematics. Springer, 2008.

[6] R. M. McConnell and J. P. Spinrad, "Modular decomposition and transitive orientation," *Discrete Mathematics*, vol. 201, pp. 189–241, 1999.

[7] B. Escoffier, J. Monnot, and V. T. Paschos, "Weighted Coloring: further complexity and approximability results," *Information Processing Letters*, vol. 97, pp. 98–103, 2006.

[8] B. Jamison and S. Olariu, "Linear-time recognition algorithm for P_4-sparse graphs," *SIAM Journal on Computing*, vol. 21, pp. 381–406, 1992.

[9] ——, "A tree representation for P_4-sparse graphs." *Discrete Applied Mathematics*, vol. 35, no. 2, pp. 115–129, 1992.

[10] ——, "Linear-time optimization algorithms for P_4-sparse graphs," *Discrete Applied Mathematics*, vol. 61, pp. 155–175, 1995.

[11] R. Lin and S. Olariu, "A fast parallel algorithm to recognize P_4-sparse graphs," *Discrete Applied Mathematics*, vol. 81, no. 1-3, pp. 191–215, 1998.

[12] J. Araujo and C. Linhares Sales, "Grundy Number on P_4-classes," in *Proceedings of the 5th Latin-american Algorithms, Graphs and Optimization Symposium (LAGOS)*, ser. Electronic Notes in Discrete Mathematics. Springer, 2009.

[13] M. Tedder, D. G. Corneil, M. Habib, and C. Paul, "Simpler Linear-Time Modular Decomposition Via Recursive Factorizing Permutations," in *Proceedings of the 35th International Colloquium on Automata, Languages and Programming (ICALP)*, ser. Lecture Notes in Computer Science, vol. 5125, 2008, pp. 634–645.

[14] V. Giakoumakis and J. Vanherpe, "On extended P_4-reducible and extended P_4-sparse graphs," *Theoretical Computer Science*, vol. 180, pp. 269–286, 1997.