b-Coloring is NP-Hard on Co-Bipartite Graphs and Polytime Solvable on Tree-Cographs

Flavia Bonomo[1]([envelope]), Oliver Schaudt[2], Maya Stein[3], and Mario Valencia-Pabon[4]

[1] CONICET and Dep. de Computación, Facultad de Ciencias Exactas y Naturales, Universidad de Buenos Aires, Buenos Aires, Argentina
fbonomo@dc.uba.ar
[2] Institut de Mathématiques de Jussieu, CNRS UMR7586, Université Pierre et Marie Curie (Paris 6), Paris, France
schaudt@math.jussieu.fr
[3] Centro de Mod. Mat., Universidad de Chile, Santiago, Chile
mstein@dim.uchile.cl
[4] Université Paris 13, Sorbonne Paris Cité, LIPN, CNRS UMR7030, Villetaneuse, France
valencia@lipn.univ-paris13.fr

Abstract. A *b-coloring* of a graph is a proper coloring such that every color class contains a vertex that is adjacent to all other color classes. The *b-chromatic number* of a graph G, denoted by $\chi_b(G)$, is the maximum number t such that G admits a b-coloring with t colors. A graph G is called *b-continuous* if it admits a b-coloring with t colors, for every $t = \chi(G), \ldots, \chi_b(G)$, and *b-monotonic* if $\chi_b(H_1) \geq \chi_b(H_2)$ for every induced subgraph H_1 of G, and every induced subgraph H_2 of H_1.
We investigate the b-chromatic number of graphs with stability number two. These are exactly the complements of triangle-free graphs, thus including all complements of bipartite graphs. The main results of this work are the following:

1. We characterize the b-colorings of a graph with stability number two in terms of matchings with no augmenting paths of length one or three. We derive that graphs with stability number two are b-continuous and b-monotonic.
2. We prove that it is NP-complete to decide whether the b-chromatic number of a co-bipartite graph is at most a given threshold.
3. We describe a polynomial time dynamic programming algorithm to compute the b-chromatic number of co-trees.
4. Extending several previous results, we show that there is a polynomial time dynamic programming algorithm for computing the b-chromatic number of tree-cographs. Moreover, we show that tree-cographs are b-continuous and b-monotonic.

1   Introduction

A *b-coloring* of a graph G by k colors is a proper coloring of the vertices of G such that every color class contains a vertex that is adjacent to all the other $k-1$ color classes. Such a vertex will be called a *dominant vertex*.

The *b-chromatic number* of a graph G, denoted by $\chi_b(G)$, is the maximum number k such that G admits a b-coloring with k colors. Clearly, $\chi_b(G) \leq \Delta(G) + 1$ where $\Delta(G)$ denotes the maximum degree of G. The b-chromatic number was introduced in [11]. The motivation, similarly as the well known achromatic number (cf. e.g., [2,7] and ref. therein), comes from algorithmic graph theory. Suppose one colors a given graph properly, but in an arbitrary way. After all vertices are colored, one would wish to perform some simple operations to reduce the number of colors. A simple operation consists in recoloring all the vertices in one color class with a possible different color. Such recoloring is impossible if each color class contains a dominant vertex. Hence, the b-chromatic number of the graph serves as the tight upper bound for the number of colors used by this coloring heuristic. From this point of view, both complexity results and polynomial time algorithms for particular graph families are interesting. And, in particular, this arguments show that any coloring of a graph G with $\chi(G)$ many colors is a b-coloring (as usual, we denote by $\chi(G)$ the minimum number of colors needed for a proper coloring of the vertices of a graph).

Assume that the vertices $v_1, v_2, \ldots, v_n$ of a graph G are ordered such that $d(v_1) \geq d(v_2) \geq \ldots \geq d(v_n)$, where $d(x)$ denotes the degree of vertex x in G. Let

$$m(G) := \max\{i : d(v_i) \geq i - 1\}$$

be the maximum number i such that G contains at least i vertices of degree $\geq i - 1$. It is clear that $m(G) \leq \Delta(G) + 1$. Irving and Manlove [11] show that this parameter bounds the b-chromatic number:

Proposition 1. *For every graph G, $\chi(G) \leq \chi_b(G) \leq m(G)$.*

Irving and Manlove [11] also show that determining $\chi_b(G)$ is NP-complete for general graphs, but polynomial-time solvable for trees. Kratochvíl, Tuza and Voigt [13] prove that the problem of determining if $\chi_b(G) = m(G)$ is NP-complete even for connected bipartite graphs G with $m(G) = \Delta(G) + 1$. A graph G is *tight* if it has exactly $m(G)$ *dense* vertices (a vertex v of a graph G is dense if $d(v) \geq m(G) - 1$), each of which has degree exactly $m(G) - 1$. Havet, Linhares-Sales and Sampaio [8] recently investigated the problem on tight graphs. They proved that the problem of determining if a tight graph G has $\chi_b(G) = m(G)$ is NP-complete for bipartite graphs and ptolemaic graphs, but polynomial-time solvable for complements of bipartite graphs, split graphs and block graphs.

In last years, several related concepts concerning b-colorings of graphs have been studied in [6,9,10,12]. A graph G is defined to be *b-continuous* [6] if it admits a b-coloring with t colors, for every $t = \chi(G), \ldots, \chi_b(G)$. In [12] (see also [6]) it is proved that chordal graphs and some planar graphs are b-continuous. A graph G is defined to be *b-monotonic* [3] if $\chi_b(H_1) \geq \chi_b(H_2)$ for

every induced subgraph H_1 of G, and every induced subgraph H_2 of H_1. They prove that P_4-sparse graphs (and, in particular, cographs) are b-continuous and b-monotonic. Besides, they give a dynamic programming algorithm to compute the b-chromatic number in polynomial time within these graph classes.

Our paper is organized as follows. In the next section, we characterize b-colorings of graphs with stability number two in terms of matchings with no augmenting paths of length one or three. In Sect. 3, we prove that graphs with stability at most two are both b-continuous and b-monotonic. In Sect. 4, we prove that computing the b-chromatic number of co-bipartite graphs is an NP-complete problem. Finally, in Sect. 5, first we describe a polynomial-time dynamic programming algorithm to compute the b-chromatic number of co-trees. Next, we extend our results to the family of tree-cographs by showing that there is a polynomial time dynamic programming algorithm for computing the b-chromatic number of graphs in this family and that these are also b-continuous and b-monotonic.

Most of our results are given without proof due to lack of space.

2   b-Colorings and Matchings

The *stability* of a graph G is defined as the maximum cardinality of a subset of pairwise non-adjacent vertices in G (i.e. a *stable set* or *independent set*). Given a graph G, we denote by $\overline{G}$ the complement graph of G, which is the graph on the same set of vertices as G that has an edge between two different vertices u and v if and only if u and v are non-adjacent in G. It is not difficult to see that G is a graph with stability one if and only if it is complete, and G is a graph with stability at most two if and only if $\overline{G}$ is a triangle-free graph. In this section, we will see that matchings in triangle-free graphs are very important when we deal with b-colorings of graphs with stability at most two.

Let M be a matching of a graph G. Denote by $V(M)$ the set of all vertices covered by M. An *augmenting path* for M is a path starting and ending outside $V(M)$ whose edges alternate between $E(G) - M$ and M. Usually, M is called *maximal* if no further edge can be included in M. In other words, G does not contain an augmenting path of length one with respect to M. Following this terminology we call M *strongly maximal* if G does not contain augmenting paths of length one or three with respect to M. By definition, maximum matchings are strongly maximal, and strongly maximal matchings are maximal. Our next lemma shows why strongly maximal matchings are important in our setting.

Lemma 1. *Let G be a graph of stability at most two and let c be a proper coloring of G. Then c is a b-coloring if and only if the set*

$$M = \{uv : u, v \in V, u \neq v \text{ and } c(u) = c(v)\}$$

is a strongly maximal matching in $\overline{G}$. Moreover, the number of colors c uses is $|V(G)| - |M|$.

Proof. First, observe that M is a (possibly empty) matching of $\overline{G}$ because G has stability at most two. Now, suppose that $\overline{G}$ contains an augmenting path P of length 1 or 3 for M. If P consists of only one edge uv, then in G, the vertices u and v are non-adjacent, and each makes up a singleton colour class. Thus c is not a b-coloring. If P has three edges, then for each of the endvertices of its middle edge uv there is a singleton color class which it does not see in G. So the color class $\{u, v\}$ witnesses the fact that c is not a b-coloring.

Next, suppose that c is not a b-coloring. Note that, as G has stability at most two, every vertex of G is adjacent (in G) to at least one vertex of any given color class of size 2. So, the witness for c not being a b-coloring is one of the following two: either it is a singleton color class whose vertex is non-adjacent to another singleton color class, or it is a color class $\{u, v\}$ of size two, such that u is non-adjacent to some singleton color class, and v is non-adjacent to a different singleton color class. The first situation corresponds to an augmenting path of M on one edge, and the second situation corresponds to an augmenting path of M on three edges. □

Observe that coloring c from Lemma 1 is a maximum (minimum) b-coloring of G if and only if M is a minimum (maximum) strongly maximal matching of $\overline{G}$.

3   b-Continuity and b-Monotonicity of Graphs with Stability at most Two

In order to prove the b-continuity of graphs with stability at most two, we need the following result.

Lemma 2. *Let M be a strongly maximal matching of a graph G and let P be a minimum length augmenting path in G with respect to M. Then, the matching $M' = (M \backslash E(P)) \cup (E(P) \backslash M)$ is a strongly maximal matching of G, and $|M'| = |M| + 1$.*

Proof. Let $P = (x_1, x_2, \ldots, x_k)$. By basic results from matching theory, the only thing we need to prove is that M' is again strongly maximal. Since the maximality of M' is clear, suppose for contradiction that there is an augmenting path of length 3, say $Q = (u, v, w, x)$. Necessarily vw is an edge of $M' \backslash M$, and thus w.l.o.g. there is some $i \in \{1, 2, \ldots, k-1\}$ with $v = x_i$ and $w = x_{i+1}$. Moreover, $u, x \notin V(M)$. Thus both paths $(x_1, x_2, \ldots, x_i, u)$ and $(x, x_{i+1}, x_{i+2}, \ldots, x_k)$ are augmenting paths for M and at least one of these paths is shorter than P. This is a contradiction to the choice of P. □

By Lemma 1, any b-coloring using $k > \chi(G)$ colors of a graph G of stability at most two corresponds to a strongly maximal matching M that is not maximum. By Berge's lemma [1], there is an augmenting path for M. Using Lemma 2 we obtain a strongly maximal matching M' of cardinality $|M| + 1$, which, again by Lemma 1, corresponds to a b-coloring with $k - 1$ colors. Repeatedly applying this argument gives the following result.

Theorem 1. *Graphs of stability at most two are b-continuous.*

Given a maximum b-coloring of a graph G of stability at most two, we can thus find b-colorings for all values between $\chi(G)$ and $\chi_b(G)$. Moreover, we can do this in polynomial time, provided we can find a minimum length augmenting path for a given matching in polynomial time. This is the aim of the following lemma that can be derived by a slight modification of Edmonds' blossom algorithm [5].

Lemma 3. *Let M be a matching in a graph G. Then, a minimum length augmenting path P in G with respect to M can be computed in polynomial time.*

Lemma 3 together with the proof of Theorem 1 implies that given a graph G of stability at most two, and a b-coloring of G using $k > \chi(G)$ colors, we can compute in polynomial time a b-coloring for G with $k-1$ colors. Notice that the converse is not necessarily true, i.e., if we have a b-coloring of G using $k < \chi_b(G)$ colors, we do not know how to compute in polynomial time a b-coloring for G with $k+1$ colors. Indeed, we will prove in the next section that the problem of computing the b-chromatic number of a graph with stability at most two is NP-complete, even restricted to the smaller class of co-bipartite graphs.

We now turn to the b-monotonicity of graphs of stability at most two.

Theorem 2. *Graphs of stability at most two are b-monotonic.*

Proof. (Sketch). The class of graphs of stability at most two is closed under taking induced subgraphs. Thus we only have to prove that χ_b is monotonously decreasing under the deletion of a vertex. In view of Lemma 1, it is sufficient to show that given a graph G of stability 2 and some vertex $v \in V(G)$ the following holds: If there is a strongly maximal matching of $\overline{G-v}$ of size k, then there is a strongly maximal matching of $\overline{G}$ of size at most $k+1$. This implies $\chi_b(G) \geq \chi_b(G-v)$.   □

4   NP-Hardness Result for Co-Bipartite Graphs

As mentioned in Sect. 1, Havet, Linhares-Sales and Sampaio [8] proved that the problem of determining if a tight co-bipartite graph G has $\chi_b(G) = m(G)$ is polynomial-time solvable. However, the computational complexity of χ_b in the class of co-bipartite graphs was left open. In the next theorem, we prove that b-coloring general co-bipartite graphs is a hard problem.

Theorem 3. *Given a co-bipartite graph G and a natural number t, it is NP-complete to decide whether G admits a b-coloring with at least t colors.*

Proof. By Lemma 1, it suffices to prove that it is NP-complete to decide whether a bipartite graph G admits a strongly maximal matching containing at most k edges, when G and k are given input.

Our reduction is from the minimum maximal matching problem which is to decide whether a given graph admits a maximal matching of at most k edges,

for given k. This problem is NP-complete even if the instances are restricted to bipartite graphs, as shown by Yannakakis and Gavril [15].

Given a bipartite graph G with m edges, we define a new graph H_G as follows. For each edge $uv \in E(G)$ we introduce a set of new vertices

$$X_{uv} = \{x_{uv}^1, x_{uv}^2, x_{uv}^3, x_{uv}^4, x_{vu}^1, x_{vu}^2, x_{vu}^3, x_{vu}^4\}$$

and edges

$$F_{uv} = \{ux_{uv}^1, x_{uv}^1 x_{uv}^2, x_{uv}^2 x_{uv}^3, x_{uv}^3 x_{uv}^4, x_{uv}^1 x_{vu}^1, x_{vu}^1 x_{vu}^2, x_{vu}^2 x_{vu}^3, x_{vu}^3 x_{vu}^4, vx_{vu}^1\}.$$

Note that $X_{uv} = X_{vu}$ and $F_{uv} = F_{vu}$. Then H_G is defined by

$$V(H_G) = V(G) \cup \bigcup_{uv \in E(G)} X_{uv},$$

$$E(H_G) = \bigcup_{uv \in E(G)} F_{uv}.$$

Clearly, H_G can be computed in polynomial time. Moreover, H_G is bipartite since G is. For each edge $uv \in E(G)$, we define the following auxiliary sets of edges in H_G:

$$F_{uv}^{\in} = \{ux_{uv}^1, x_{uv}^2 x_{uv}^3, x_{vu}^2 x_{vu}^3, vx_{vu}^1\} \text{ and } F_{uv}^{\notin} = \{x_{uv}^1 x_{vu}^1, x_{uv}^2 x_{uv}^3, x_{vu}^2 x_{vu}^3\}$$

We claim the following:

Claim. There exists a minimum strongly maximal matching M of H_G such that

$$x_{uv}^3 x_{uv}^4 \notin M \text{ for each edge } uv \in E(G).$$

Moreover, M can be obtained from any minimum strongly maximal matching of H_G in polynomial time.

In order to prove this claim, we proceed by contradiction. Assume that every minimum strongly maximal matching of H_G contains at least an edge $x_{uv}^3 x_{uv}^4$ for some edge $uv \in E(G)$, and let M be a minimum strongly maximal matching of H_G having a minimum number of edges of the form $x_{uv}^3 x_{uv}^4$. Note that the choice of M implies that for every edge $uv \in E(G)$ we have that

(i) $x_{uv}^3 x_{uv}^4 \in M$ if and only if $x_{uv}^1 x_{uv}^2 \in M$. If $x_{uv}^1 x_{uv}^2 \in M$ then $x_{uv}^3 x_{uv}^4 \in M$, otherwise, M is not maximal. If $x_{uv}^3 x_{uv}^4 \in M$ then $x_{uv}^1 x_{uv}^2 \in M$, otherwise, we could replace $x_{uv}^3 x_{uv}^4$ by $x_{uv}^2 x_{uv}^3$ in M (the resulting matching is strongly maximal as M is), contradicting the choice of M.

(ii) If the edges $x_{uv}^3 x_{uv}^4$ and $x_{uv}^1 x_{uv}^2$ are in M, then we have that vertices u and x_{vu}^1 are each matched by M. Otherwise, if u is unmatched, we can replace $x_{uv}^1 x_{uv}^2, x_{uv}^3 x_{uv}^4 \in M$ with the edges $x_{uv}^2 x_{uv}^3, ux_{uv}^1$. This again yields a strongly maximal matching (since u has no neighbors unmatched by M), contradicting the choice of M. We can use the same argument in the case x_{vu}^1 is unmatched.

These are also some of the steps in order to transform any minimum strongly maximal matching into the desired one.

Now, let uv be an edge in the graph G such that $x^3_{uv} x^4_{uv} \in M$. By (i) and (ii), we can deduce that $|M \cap F_{uv}| = 4$. Consider the matching

$$\tilde{M} := (M \backslash F_{uv}) \cup F^{\notin}_{uv}$$

We claim that $\tilde{M}$ is strongly maximal. As $\tilde{M}$ is smaller than M, we thus obtain the desired contradiction. So assume $\tilde{M}$ is not strongly maximal. Then, as u is matched, there is an augmenting path P of length 1 or 3 starting at v.

Now, observe that all neighbors of v are of the form x^1_{vw} (for some $w \in V(G)$), and thus, as neither $x^1_{vw} x^2_{vw}$ nor $x^1_{vw} x^2_{vw} x^3_{vw} x^4_{vw}$ is an augmenting path for the strongly maximal matching M, all neighbors of v are matched by M.

So, P has length 3, and it is easy to see that P has to end in some (unmatched) vertex $w \in V(G) \backslash \{u, v\}$ (by the maximality of M, every vertex x^3_{wz} is matched by M, and by the choice of M, every vertex x^2_{wz} is matched by M). By (i) and (ii), we know that $F_{vw} \cap M = F^{\notin}_{vw}$. Consider the matching

$$(\tilde{M} \backslash F^{\notin}_{vw}) \cup F^{\in}_{vw}.$$

It can be seen that this matching is strongly maximal, and has fewer edges of the form $x^3_{uv} x^4_{uv}$, contradicting the choice of M. (And this is the remaining step in order to transform any minimum strongly maximal matching into the desired one.) This ends the proof of this claim.

Therefore, by the previous claim, we have that there is a minimum strongly maximal M' in H_G that verifies either $F_{uv} \cap M' = F^{\in}_{uv}$ or $F_{uv} \cap M' = F^{\notin}_{uv}$ for each edge $uv \in E(G)$. Next we show that if M is a minimum maximal matching of G and M' is a minimum strongly maximal matching of H_G, $|M| = |M'| - 3\,\text{m}$. As explained above, this completes the proof.

Let M be a minimum maximal matching of G. Using the auxiliary sets $F^{\in}_{uv}$ and $F^{\notin}_{uv}$, we define a strongly maximal matching M' of H_G by

$$M' = \bigcup_{uv \in M} F^{\in}_{uv} \cup \bigcup_{uv \notin M} F^{\notin}_{uv}.$$

Note that $|M'| = |M| + 3|E(G)|$.

Now, let M' be a minimum strongly maximal matching of H_G that verifies either $F_{uv} \cap M' = F^{\in}_{uv}$ or $F_{uv} \cap M' = F^{\notin}_{uv}$ for each edge $uv \in E(G)$. We define a maximal matching M of G by setting

$$M = \{uv : uv \in E(G),\ F_{uv} \cap M' = F^{\in}_{uv}\}.$$

Clearly, $|M| = |M'| - 3|E(G)|$, which completes the proof. $\qquad\square$

5   b-Coloring Co-Trees and Tree-Cographs

5.1   Co-Trees

Theorem 4. *In the class of co-trees, χ_b can be computed in polynomial time.*

Proof. According to Lemma 1, the problem is equivalent to finding a minimum strongly maximal matching (MSMM) in a tree. We will find such a matching by dynamic programming. In order to do so, we will define five functions $F_i(r, s)$, $i = 1, \ldots, 5$, for a nontrivial tree T_{rs} rooted at a leaf r with neighbor s. As we will apply them to the subtrees of a tree, we will assume that r can have neighbors outside T_{rs}.

– $F_1(r, s)$: cardinality of an MSMM of T_{rs} such that r is unmatched, and ∞ if it does not exist.
– $F_2(r, s)$: cardinality of an MSMM of T_{rs} that uses the edge rs and such that s may or may not have an unmatched neighbor (this case will apply when r has no unmatched neighbor outside T_{rs}), and ∞ if it does not exist.
– $F_3(r, s)$: cardinality of an MSMM of T_{rs} that uses the edge rs and such that s does not have an unmatched neighbor (this case will apply when r has already an unmatched neighbor outside T_{rs}, so an unmatched neighbor of s will complete an augmenting path of length 3 in the whole tree), and ∞ if it does not exist.
– $F_4(r, s)$: cardinality of an MSMM of T_{rs} such that the vertex s is matched with some vertex different from r and the vertex r is considered as "already matched" (this case will apply when r is already matched with a vertex outside T_{rs}), and ∞ if it does not exist.
– $F_5(r, s)$: cardinality of an MSMM of T_{rs} such that the vertex s remains unmatched and the vertex r is considered as "already matched", and ∞ if it does not exist.

With these definitions, for the base case in which $V(T_{rs}) = \{r, s\}$, we have

– $F_1(r, s) = \infty$ (if r is unmatched and s has no further neighbors, the matching will never be maximal)
– $F_2(r, s) = 1$ (precisely, the edge rs)
– $F_3(r, s) = 1$ (precisely, the edge rs)
– $F_4(r, s) = \infty$ (it is not feasible because s has no further neighbors)
– $F_5(r, s) = 0$

For the case in which s has children $v_1, \ldots, v_k$, we have

– $F_1(r, s) = \min_{i=1,\ldots,k}\{F_3(s, v_i) + \sum_{j=1,\ldots,k;j\neq i} \min\{F_4(s, v_j), F_5(s, v_j)\}\}$. In order to obtain a maximal matching, we need to match s with one of its children, say v_i. Since r will be unmatched, v_i should not have an unmatched neighbor, in order to prevent an augmenting path of length 3. When considering the trees T_{sv_j} for $j \neq i$, the vertex s will have the status of "already matched". Furthermore, since we are already assuming that s has an unmatched neighbor, we do not need to care about the vertices v_j being matched or not.
– $F_2(r, s) = 1 + \sum_{i=1,\ldots,k} \min\{F_4(s, v_i), F_5(s, v_i)\}$. We will use the edge rs, and then when considering the trees T_{sv_i} for $i = 1, \ldots, k$, the vertex s will have the status of "already matched". Furthermore, since s may or may not have an unmatched neighbor, we can take the minimum over F_4 and F_5 for each of the trees T_{sv_i}.

- $F_3(r, s) = 1 + \sum_{i=1,\ldots,k} F_4(s, v_i)$. This case is similar to the previous one, but now the vertex s cannot have unmatched neighbors, so we will just consider F_4 for each of the trees T_{sv_i}.
- $F_4(r, s) = \min\{\min_{i=1,\ldots,k}\{F_2(s, v_i) + \sum_{j=1,\ldots,k;j\neq i} F_4(s, v_j)\}, \min_{i=1,\ldots,k}\{F_3(s, v_i) + \sum_{j=1,\ldots,k;j\neq i} \min\{F_4(s, v_j), F_5(s, v_j)\}\}\}$. As in the first case, we need to match s with one of its children, say v_i. But now, since r is assumed to be matched, s may or may not have an unmatched neighbor, depending on the matching status of the vertices v_j with $j \neq i$. So we will take the minimum among allowing v_i to have an unmatched neighbor and forcing v_j, $j \neq i$, to be matched, or forbidding v_i to have an unmatched neighbor and allowing v_j, $j \neq i$, to be either matched or not.
- $F_5(r, s) = \sum_{i=1,\ldots,k} F_1(s, v_i)$. This last case is quite clear.

In this way, in order to obtain the cardinality of a minimum strongly maximal matching of a nontrivial tree T, we can root it at a leaf r whose neighbor is s and compute $\min\{F_1(r, s), F_2(r, s)\}$. By keeping some extra information, we can also obtain in polynomial time the matching itself. $\qquad\square$

5.2 Tree-Cographs

A graph is a *tree-cograph* if it can be constructed from trees by disjoint union and complement operations. Tree-cographs have been introduced by Tinhofer [14] as a generalization of trees and cographs. Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be two graphs with $V_1 \cap V_2 = \emptyset$. The *union* of G_1 and G_2 is the graph $G_1 \cup G_2 = (V_1 \cup V_2, E_1 \cup E_2)$, and the *join* of G_1 and G_2 is the graph $G_1 \vee G_2 = (V_1 \cup V_2, E_1 \cup E_2 \cup V_1 \times V_2)$. Note that $\overline{G_1 \vee G_2} = \overline{G_1} \cup \overline{G_2}$. Tree-cographs can be recursively defined as follows: a graph G is a tree-cograph if and only if

- (i) G is a tree or a co-tree, or
- (ii) G is the union of two tree-cographs G_1 and G_2, or
- (iii) G is the join of two tree-cographs G_1 and G_2.

Notice that if (i) in the above definition is replaced by "G is a single vertex" then, the obtained graph is a cograph. The notion of *dominance sequence* has been introduced in [3] in order to compute the b-chromatic number of P_4-sparse graphs and, in particular, cographs. Formally, given a graph G, the dominance sequence $\mathrm{dom}_G \in \mathbb{Z}^{\mathbb{N}_{\geq \chi(G)}}$, is defined such that $\mathrm{dom}_G[t]$ is the maximum number of distinct color classes admitting dominant vertices in any coloring of G with t colors, for every $t \geq \chi(G)$. Note that it suffices to consider this sequence until $t = |V(G)|$, since $\mathrm{dom}_G[t] = 0$ for $t > |V(G)|$. Therefore, in the sequel we shall consider only the *dominance vector* $(\mathrm{dom}_G[\chi(G)], \ldots, \mathrm{dom}_G[|V(G)|])$. Notice that a graph G admits a b-coloring with t colors if and only if $\mathrm{dom}_G[t] = t$. Moreover, it is clear that $\mathrm{dom}_G[\chi(G)] = \chi(G)$. The following results given in [3] are very important in order to compute the b-chromatic number of graphs that can be decomposed recursively in modules via disjoint union or join operations.

Theorem 5 [3]. *Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be two graphs such that $V_1 \cap V_2 = \emptyset$. If $G = G_1 \cup G_2$ and $t \geq \chi(G)$, then*

$$\text{dom}_G[t] = \min\{t, \text{dom}_{G_1}[t] + \text{dom}_{G_2}[t]\}.$$

Theorem 6 [3]. *Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be two graphs such that $V_1 \cap V_2 = \emptyset$. Let $G = G_1 \vee G_2$ and $\chi(G) \leq t \leq |V(G)|$. Let $a = \max\{\chi(G_1), t - |V(G_2)|\}$ and $b = \min\{|V(G_1)|, t - \chi(G_2)\}$. Then $a \leq b$ and*

$$\text{dom}_G[t] = \max_{a \leq j \leq b}\{\text{dom}_{G_1}[j] + \text{dom}_{G_2}[t - j]\}.$$

In order to compute the dominance vector of a tree-cograph and its corresponding b-chromatic number, by Theorems 5 and 6, it is sufficient to compute the dominance vector for both trees and co-trees.

Dominance Vector for Trees. Irving and Manlove [11] shown that the b-chromatic number of any tree T is equal to $m(T) - 1$ or $m(T)$, depending on the existence of a unique vertex in T called a *pivot*. A vertex v of T is called *dense* if $d(v) \geq m(T) - 1$. Based on Irving and Manlove's results, we are able to prove the following result.

Theorem 7. *If G is tree, then dom_G can be computed in polynomial time.*

Dominance Vector for Co-Trees. Let G be a graph and M be a matching of it. Let $S_1(G, M)$ be the number of unmatched vertices that have at least one unmatched neighbor and let $S_2(G, M)$ be the number of edges of M that are the center of an augmenting path of length 3 for M. Let $F(\overline{G}, k)$ be the minimum value of $S_1(G, M) + S_2(G, M)$ over all matchings M of G with $|M| = k$.

Now, let G be a graph with stability at most two and consider a coloring of it. Let M be the matching of $\overline{G}$ corresponding to that coloring. The number of color classes without a dominant vertex are exactly $S_1(\overline{G}, M) + S_2(\overline{G}, M)$. So, for $\chi(G) \leq i \leq |V(G)|$, $\text{dom}_G[i] = i - F(\overline{G}, |V(G)| - i)$. We will show how to compute $F(T, k)$ for a tree T and a nonnegative integer k in polynomial time.

Theorem 8. *If G is a co-tree, then dom_G can be computed in polynomial time.*

Proof. (Sketch) As we noticed above, the problem is equivalent to computing $F(\overline{G}, k)$. We use dynamic programming. In order to do so, and in a similar fashion as in Theorem 4, we will define seven functions $F_i(r, s, k)$, $i = 1, \ldots, 7$, for a nontrivial tree T_{rs} rooted at a leaf r with neighbor s and a nonnegative integer k. As we will apply them to the subtrees of a tree, we will assume that r can have neighbors outside T_{rs}. Nevertheless, we will count for S_2 just the edges of $M \cap E(T_{rs})$ and for S_1 the vertices of $V(T_{rs})$, with the exception of r when it is unmatched but has already an unmatched neighbor outside T_{rs}, in order to avoid double counting.

For $i = 1, \ldots, 7$, $F_i(r, s, k)$ will be the minimum of $S_1(T_{rs}, M) + S_2(T_{rs}, M)$ over all the matchings M with $|M| = k$ such that:

- $F_1(r, s, k)$: r is unmatched and s is matched by M with some vertex different from r.
- $F_2(r, s, k)$: M uses the edge rs and r has no unmatched neighbor outside T_{rs}.
- $F_3(r, s, k)$: M uses the edge rs and r has an unmatched neighbor outside T_{rs}.
- $F_4(r, s, k)$: the vertex s is matched by M with some vertex different from r and the vertex r is already matched with a vertex outside T_{rs}.
- $F_5(r, s, k)$: the vertex s remains unmatched and the vertex r is already matched with a vertex outside T_{rs}.
- $F_6(r, s, k)$: r is unmatched, s remains unmatched, and r has no unmatched neighbor outside T_{rs}.
- $F_7(r, s, k)$: r is unmatched, s remains unmatched, and r has an unmatched neighbor outside T_{rs} (we will not count r for S_1 as we assume it is already counted).

In all cases, the value will be ∞ if no such M does exist.

Notice that as the values of the functions F_i are bounded by the number of vertices of the corresponding tree, and k is also bounded by that number, taking the minimum over $k_1 + \cdots + k_\ell = k$ of some combination of these F_i is equivalent to solving a polynomially bounded number of knapsack problems where both the weights and the utilities are polynomially bounded as well, so this can be done by dynamic programming in polynomial time [4].

In this way, in order to obtain $F(T, k)$ for a nontrivial tree T, we can root it at a leaf r whose neighbor is s and compute $\min\{F_1(r, s, k), F_2(r, s, k), F_6(r, s, k)\}$. By keeping some extra information, we can also obtain in polynomial time the matching itself.   □

b-Continuity and b-Monotonicity of Tree-Cographs. The following result was proved for union and join of graphs.

Lemma 4 [3]. *Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be two graphs such that $V_1 \cap V_2 = \emptyset$. If G_1 and G_2 are b-continuous, then $G_1 \cup G_2$ and $G_1 \vee G_2$ are b-continuous.*

As a corollary of the lemma, Theorem 1, and the b-continuity of chordal graphs [6,12], we have the following result.

Theorem 9. *Tree-cographs are b-continuous.*

Concerning the b-monotonicity, the following results are known for general graphs and for union and join of graphs.

Lemma 5 [3]. *Let G be a graph. The maximum value of $\mathrm{dom}_G[t]$ is attained in $t = \chi_b(G)$.*

Lemma 6 [3]. *Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be two b-continuous graphs such that $V_1 \cap V_2 = \emptyset$, and let $G = G_1 \cup G_2$ (resp., let $G = G_1 \vee G_2$). Assume that for every $t \geq \chi(G_i)$ and every induced subgraph H of G_i we have $\mathrm{dom}_H[t] \leq \mathrm{dom}_{G_i}[t]$, for $i = 1, 2$. Then, for every $t \geq \chi(G)$ and every induced subgraph H of G, $\mathrm{dom}_H[t] \leq \mathrm{dom}_G[t]$ holds.*

In order to prove the b-monotonicity of tree-cographs, we need the following two lemmas.

Lemma 7. *Let T be a tree and H an induced subgraph of T. Then for every $t \geq 2$, $\mathrm{dom}_H[t] \leq \mathrm{dom}_T[t]$.*

Lemma 8. *Let G be a graph with stability at most two and H an induced subgraph of G. Then for every $t \geq \chi(G)$, $\mathrm{dom}_H[t] \leq \mathrm{dom}_G[t]$.*

So, we can conclude the following.

Theorem 10. *Tree-cographs are b-monotonic.*

References

1. Berge, C.: Two theorems in graph theory. Proc. Natl. Acad. Sci. U.S.A. **43**, 842–844 (1957)
2. Bodlaender, H.L.: Achromatic number is NP-complete for cographs and interval graphs. Inf. Process. Lett. **31**, 135–138 (1989)
3. Bonomo, F., Durán, G., Maffray, F., Marenco, J., Valencia-Pabon, M.: On the b-coloring of cographs and P_4-sparse graphs. Graphs Comb. **25**(2), 153–167 (2009)
4. Dantzig, G.B.: Discrete-variable extremum problems. Oper. Res. **5**, 266–277 (1957)
5. Edmonds, J.: Paths, trees and flowers. Can. J. Math. **17**, 449–467 (1965)
6. Faik, T.: La b-continuité des b-colorations: complexité, propriétés structurelles et algorithmes. Ph.D. thesis, L.R.I., Université Paris-Sud, Orsay, France (2005)
7. Harary, F., Hedetniemi, S.: The achromatic number of a graph. J. Comb. Theor. **8**, 154–161 (1970)
8. Havet, F., Linhares-Sales, C., Sampaio, L.: b-coloring of tight graphs. Discrete Appl. Math. **160**(18), 2709–2715 (2012)
9. Hoàng, C.T., Kouider, M.: On the b-dominating coloring of graphs. Discrete Appl. Math. **152**, 176–186 (2005)
10. Hoàng, C.T., Linhares Sales, C., Maffray, F.: On minimally b-imperfect graphs. Discrete Appl. Math. **157**(17), 3519–3530 (2009)
11. Irving, R.W., Manlove, D.F.: The b-chromatic number of a graph. Discrete Appl. Math. **91**, 127–141 (1999)
12. Kára, J., Kratochvíl, J., Voigt, M.: b-continuity. Technical report M 14/04, Technical University Ilmenau, Faculty of Mathematics and Natural Sciences (2004)
13. Kratochvíl, J., Tuza, Z., Voigt, M.: On the b-Chromatic number of graphs. In: Kučera, L. (ed.) WG 2002. LNCS, vol. 2573, pp. 310–320. Springer, Heidelberg (2002)
14. Tinhofer, G.: Strong tree-cographs are Birkoff graphs. Discrete Appl. Math. **22**(3), 275–288 (1989)
15. Yannakakis, M., Gavril, F.: Edge dominating sets in graphs. SIAM J. Appl. Math. **38**(3), 364–372 (1980)