(Sub)Fall Coloring and B-Coloring Parameterized by Treewidth

Davi de Andrade[1], **Ana Silva**[1]

[1]ParGO Group - Parallelism, Graphs and Optimization
Centro de Ciências - Departamento de Matemática
Universidade Federal do Ceará (UFC)

daviandradeiacono@alu.ufc.br, anasilva@mat.ufc.br

Abstract. *Given a proper coloring f of G, a vertex u is a b-vertex if it is adjacent to every color class distinct from its own. A proper coloring is said to be a b-coloring if each color class contains at least one b-vertex, and a fall coloring if all vertices are b-vertices. Also, if f is a fall coloring of an induced subgraph H of G, then we say that f is a subfall coloring of G. In this paper, we provide algorithms for each of the decision problems related to these colorings whose running times are* FPT *when parameterized by the number of colors plus the treewidth of the input graph.*

1. Introduction

Given a simple graph G, a *k-coloring* of G is a function $f : V(G) \to \{1, 2, \ldots, k\}$; it is *proper* if $f(u) \neq f(v)$ for every $uv \in E(G)$. The problem of deciding whether a graph has a proper k-coloring is one of the most investigated problems in Graph Theory, being NP-complete to decide if an arbitrary graph has a proper k-coloring even with $k = 3$, as proved in [Garey et al. 1974]. As alternative, colorings obtained by certain heuristics are applied, giving rise to many variations of the coloring problem, such as Grundy coloring [Christen and Selkow 1979], b-coloring [Irving and Manlove 1999] and a-coloring [Harary and Hedetniemi 1970]. Here, we concentrate on b-colorings [Irving and Manlove 1999], fall colorings and subfall colorings [Dunbar et al. 2000].

Given a proper k-coloring f of G, a vertex $v \in V(G)$ is a *b-vertex* in f if $f(N[v]) = [k]$, i.e., v has a neighbor in each color class different from its own. We say that f is a *b-coloring* if every color class contains at least one b-vertex; and that it is a *fall coloring* if every vertex is a b-vertex. Also, given a fall coloring f of an induced subgraph H of G, we say that f is a *subfall coloring of G*. The *b-chromatic number*, $b(G)$, of G is the largest value k for which G has a b-coloring with k colors. The *fall chromatic number*, $\psi(G)$, and the *subfall chromatic number*, $\psi_s(G)$, are defined similarly. In [Irving and Manlove 1999], they proved that computing $b(G)$ is NP-hard, while in [Kratochvíl et al. 2002] they proved that deciding whether a graph has a b-coloring with k colors is NP-complete even if G is a connected bipartite graph and $k = \Delta(G) + 1$. Furthermore, in [Dunbar et al. 2000], they proved that deciding if $\psi(G) \geq 3$ is NP-complete. More recently, in [Andrade and Silva 2021], the authors proved that deciding $\psi_s(G) \geq k$ is: NP-complete for every $k \geq 4$; and polynomial for $k \leq 2$. Additionally, they characterized the graphs with $\psi_s(G) \geq 3$.

Given the difficulty of such problems, a usual approach is to investigate their parameterized complexity. In particular, many works parameterize problems by measures on graphs called *widths* (see e.g. [Vatshelle 2012] for a nice overview of more than 30 different width measures). Perhaps the most famous of such widths is the *treewidth*, which intuitively measures how close a graph is from being a tree. This is due to its high applicability for obtaining FPT algorithms, even when the graph does not have bounded treewidth as witnessed by the win-win strategy (see e.g. [Cygan et al. 2015]).

In [Panolan et al. 2017] it is proved that deciding if $b(G) \geq k$ is W[1]-hard when parameterized by k, and in fact an XP algorithm is not yet known to exist. Also, up to our knowledge, the only result addressing b-coloring parameterized by the treewidth is presented in [Jaffke et al. 2021], but via Courcelle's Theorem [Courcelle 1990], which is regarded simply as a classification tool (see e.g. [Cygan et al. 2015]). Here, we provide an $O^*((6^k \cdot k)^w)$ algorithm to decide whether $b(G) \geq k$ for graphs with treewidth w.

As for the other problems, we provide an $O^*((6^k \cdot k)^w)$ algorithm to decide whether $\psi(G) \geq k$ and an $O^*((6^k \cdot 2 \cdot k)^w)$ algorithm to decide whether $\psi_s(G) \geq k$. Even though our algorithm does not improve the $O^*((4^k \cdot k^2)^w)$ algorithm for fall k-coloring presented in [Telle and Proskurowski 1997], it has the advantage of being more clear (the cited algorithm is a generic meta-algorithm), and of naturally leading to the algorithms for subfall coloring and for b-coloring. Additionally, we observe that, since $\psi(G) \leq \delta(G) \leq tw(G)$, it follows that computing the exact value of $\psi(G)$ can be done in FPT time when parameterized by the treewidth $tw(G)$. Similarly, we have that $\psi_s(G) \leq \max_{H \subseteq G} \delta(H)$. The value on the right side of the latter equation is also known as the *degeneracy of* G, and since it is known that this value is also at most equal to the treewidth of G, it again follows that ψ_s can be computed in FPT time when parameterized by $tw(G)$. It is an open question whether the same holds for $b(G)$.

In the remainder of the text, we assume familiarity with basic graph definitions and FPT notions, as well as with tree decomposition techniques. A nice tree decomposition $\mathcal{T} = (T, \mathcal{X})$ is assumed to be given (see [Cygan et al. 2015]).

2. Parameterized complexity of fall k-coloring

In this section, we give a dynamic-programming algorithm that establishes the complexity of fall coloring when parameterized by treewidth. The general idea for our table is that, for every possible k-coloring of bag X_t for a node t of the decomposition tree $\mathcal{T}$, we compute whether it can be extended to a partial fall k-coloring of the graph G_t induced by all vertices in the bags of the subtree of $\mathcal{T}$ rooted at t. By partial we mean that every vertex of G_t is a b-vertex, with the exception of possibly some vertices of X_t. For this, we need to keep track of which colors are indeed being used, and of which colors are missing in the neighborhood of each vertex of X_t. Formally, given node $t \in V(T)$, where $X_t = \{v_1, v_2, \ldots, v_p\}$, we define the table related to t as follows. For each proper k-coloring f of X_t, and for each $\mathcal{M} = \{M_0, \ldots, M_p\}$, with $M_i \subseteq [k]$ for every i, we say that $c_t(f, \mathcal{M}) = 1$ if and only if there exists a coloring f' that extends f to G_t and satisfies:

1. Every $u \in V(G_t) \setminus X_t$ is a b-vertex in f';
2. For every $v_i \in X_t$, we have $f'(N_{G_t}[v_i]) = [k] \setminus M_i$; and
3. $f'(V(G_t)) = [k] \setminus M_0$.

Let r be the root of the nice tree decomposition. Recall that $X_r = \emptyset$ and let $\mathcal{M}_\emptyset = \{M_0\}$ where $M_0 = \emptyset$. Observe that G has a fall k-coloring if and only if $c_r(\emptyset, \mathcal{M}_\emptyset) = 1$. Because of space constraints, we present only the computation of forget nodes. A forget node is a node t with $X_t = \{v_1, \ldots, v_{p-1}\}$, and an only child t' where $X_{t'} = X_t \cup \{v_p\}$. Let $f : X_t \to [k]$ and $\mathcal{M} = \{M_0, \ldots, M_{p-1}\}$. Then,

$$c_t(f, \mathcal{M}) = 1 \quad \text{iff} \quad \begin{array}{l}\exists c \in [k] \text{ s.t. } c_{t'}(f_c, \mathcal{M}') = 1, \text{ where } M'_p = \emptyset,\ f_c(v_p) = c, \text{ and} \\ \forall i \in \{0, \ldots, p-1\},\ M'_i = M_i \text{ and } f_c(v_i) = f(v_i)\ .\end{array}$$

The complexity $O^*((6^k \cdot k)^w)$ follows from the size of the table of each node, which is equal to $O^*((2^k \cdot k)^w)$, the fact that the table of a node can be computed in time $O^*(3^{kw})$, and that $|V(T)| = O(|V(G)|)$ (see [Cygan et al. 2015]).

3. Subfall coloring

In this section, we apply ideas analogous to the ones applied in the previous section in order to decide whether a given graph has a subfall k-coloring. For this, given node $t \in V(T)$, for each subset $S = \{v_1, \ldots, v_p\} \subseteq X_t$, each proper k-coloring f of S, and each $\mathcal{M} = \{M_0, \ldots, M_p\}$, with $M_i \subseteq [k]$ for every i, we say that $c_t(S, f, \mathcal{M}) = 1$ if and only if there exist $H \subseteq G_t$ and k-coloring f' that extends f to H that satisfy:

1. Every $u \in V(H) \setminus X_t$ is a b-vertex in f';
2. For every $v_i \in S$, we have $f'(N_H[v_i]) = [k] \setminus M_i$;
3. $f'(V(H)) = [k] \setminus M_0$; and
4. $V(H) \cap X_t = S$.

Let r and $\mathcal{M}_\emptyset$ be as before, and observe that G has a subfall k-coloring if and only if $c_r(\emptyset, \emptyset, \mathcal{M}_\emptyset) = 1$. For space constraints, we omit the computation of the tables.

4. B-Coloring

In this section, we show how to decide whether a given graph has a b-coloring with k colors. The difference in the tables is that M_0 now represents the colors that do not have b-vertices in $G_t \setminus X_t$. So, given $t \in V(T)$, for each proper k-coloring f of X_t, and for each $\mathcal{M} = \{M_0, \ldots, M_p\}$, with $M_i \subseteq [k]$ for every i, we say that $c_t(f, \mathcal{M}) = 1$ if and only if there exists a k-coloring f' of G_t that extends f and satisfies:

1. For every $c \in [k] \setminus M_0$, there exists $u \in V(G_t) \setminus X_t$ such that u is a b-vertex of color c in f'; and
2. For every $v_i \in X_t$, we have $f'(N_{G_t}[v_i]) = [k] \setminus M_i$.

Let r and $\mathcal{M}_\emptyset$ be as before, and observe that G has a b-coloring with k colors if and only if $c_r(\emptyset, \mathcal{M}_\emptyset) = 1$. For space constraints, we comment only on how the tables of forget nodes are computed. We choose these also because they are the ones that most differ from the previous computations.

So, let t be a forget node with $X_t = \{v_1, \ldots, v_{p-1}\}$, and t' be its child, with $X_t = X_{t'} \setminus \{v_p\}$. Also, let $f : X_t \to [k]$ and $\mathcal{M} = \{M_0, \ldots, M_{p-1}\}$. We need to investigate the possibilities of v_p being the sole b-vertex of some color not in M_0, and of v_p not being a b-vertex at all. One can verify that all these possibilities are covered in the following case analysis. Below, f_c denotes the coloring obtained from f by coloring v_p with c, while $\mathcal{M}'$ denotes the set $\{M'_0, \ldots, M'_p\}$:

1. If there exist $c \in [k]$ and $M'_p \subseteq [k]$ such that $c_{t'}(f_c, \mathcal{M}') = 1$, where $M'_i = M_i$ for every $i \in \{0, \dots, p-1\}$, then $c_t(f, \mathcal{M}) = 1$. This case covers the possibility of v_p either not being a b-vertex or not being the only b-vertex of some color $c \in [k] \setminus M_0$;
2. If there exists $c \in [k] \setminus M_0$ such that $c_{t'}(f_c, \mathcal{M}') = 1$, where $M'_i = M_i$ for every $i \in [p-1]$, $M'_0 = M_0 \setminus \{c\}$ and $M'_p = \emptyset$, then $c_t(f, \mathcal{M}) = 1$. This case covers the possibility of v_p being the only b-vertex of some color $c \in [k] \setminus M_0$;
3. Finally, if neither of the above conditions hold, then we can conclude that $c_t(f, \mathcal{M}) = 0$.

References

Andrade, D. and Silva, A. (2021). On the Complexity of Subfall Coloring of Graphs. In *Anais do VI Encontro de Teoria da Computação*, pages 70–73, Brasil. SBC.

Christen, C. A. and Selkow, S. M. (1979). Some Perfect Coloring Properties of Graphs. *Journal of Combinatorial Theory, Series B*, 27(1):49–59.

Courcelle, B. (1990). The Monadic Second-Order Logic of Graphs. I. Recognizable Sets of Finite Graphs. *Information and computation*, 85(1):12–75.

Cygan, M., Fomin, F. V., Kowalik, L., Lokshtanov, D., Marx, D., Pilipczuk, M., Pilipczuk, M., and Saurabh, S. (2015). *Parameterized Algorithms*. Springer, 1st edition.

Dunbar, J., Hedetniemi, S., Hedetniemi, S., Jacobs, D., Knisely, J., Laskar, R., and Rall, D. (2000). Fall Colorings of Graphs. *J. of Combinatorial Mathematics and Combinatorial Computing*, 33:257–273.

Garey, M. R., Johnson, D. S., and Stockmeyer, L. (1974). Some Simplified NP-Complete Problems. In *STOC '74*, page 47–63. Association for Computing Machinery.

Harary, F. and Hedetniemi, S. (1970). The Achromatic Number of a Graph. *Journal of Combinatorial Theory*, 8(2):154–161.

Irving, R. W. and Manlove, D. F. (1999). The B-Chromatic Number of a Graph. *Discrete Applied Mathematics*, 91(1):127–141.

Jaffke, L., Lima, P. T., and Lokshtanov, D. (2021). B-Coloring Parameterized by Clique-Width. In *STACS 2021*, volume 187 of *LIPIcs*, pages 43:1–43:15, Germany.

Kratochvíl, J., Tuza, Z., and Voigt, M. (2002). On the B-Chromatic Number of Graphs. *Journal of Optimization Theory and Applications*, 109:310–320.

Panolan, F., Philip, G., and Saurabh, S. (2017). On the Parameterized Complexity of B-Chromatic Number. *Journal of Computer and System Sciences*, 84:120–131.

Telle, J. A. and Proskurowski, A. (1997). Algorithms for Vertex Partitioning Problems on Partial k-Trees. *SIAM J. Discret. Math.*, 10:529–550.

Vatshelle, M. (2012). *New Width Parameters of Graphs*. PhD thesis, The University of Bergen.