

Minimum Color-Degree Perfect b -Matchings

Mariia Anapolska¹ | Christina Büsing¹ | Martin Comis¹
| Tabea Krabs¹

¹Lehrstuhl II für Mathematik, RWTH Aachen University, , Pontdriesch 10-12, 52062 Aachen, Germany

Correspondence

Christina Büsing, Lehrstuhl II für Mathematik, RWTH Aachen University, Pontdriesch 10-12, 52062 Aachen, Germany
Email: buesing@math2.rwth-aachen.de

Funding information

This work was supported by the Freigeist-Fellowship of the Volkswagen Stiftung; and the German research council (DFG) Research Training Group 2236 UnRAVeL.

Abstract

The minimum color-degree perfect b -matching problem (Col-BM) is a new extension of the perfect b -matching problem to edge-colored graphs. The objective of Col-BM is to minimize the maximum number of differently colored edges in a perfect b -matching that are incident to the same node. We show that Col-BM is \mathcal{NP} -hard on bipartite graphs by a reduction from (3,B2)-Sat, and conclude that there exists no $(2 - \varepsilon)$ -approximation algorithm unless $\mathcal{P} = \mathcal{NP}$. However, we identify a class of two-colored complete bipartite graphs on which we can solve Col-BM in polynomial time. Furthermore, we use dynamic programming to devise polynomial-time algorithms solving Col-BM with a fixed number of colors on series-parallel graphs and simple graphs with bounded treewidth.

KEYWORDS

b -matching, bounded treewidth, complexity, dynamic programming, edge-colored graph, series-parallel graph

1 | INTRODUCTION

Assignment problems are among the most famous combinatorial optimization problems. In its most basic form, the assignment problem consists of a set of agents A , a set of jobs B , and a set of agent-job pairs $E \subseteq A \times B$ that define which agent can perform which job [1]. The objective is to find a one-to-one assignment of jobs to agents. Graph-theoretically the assignment problem corresponds to the maximum (weighted) matching problem in a bipartite graph which is known to be polynomial-time solvable by the Hungarian method [2]. However, for many applications this original version of the assignment problem fails to capture all relevant requirements. Therefore, various more complex forms of the assignment problem are studied, e.g., the (capacitated) b -matching problem [1] or the restricted matching problem [3]. In the maximum b -matching problem, there is a specified b -value for every node which determines

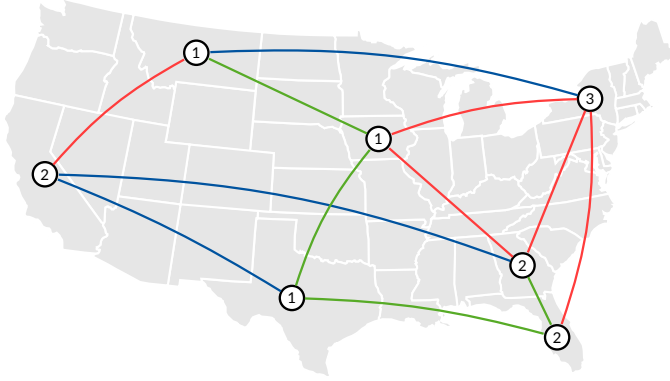


FIGURE 1 Exemplary Col-BM instance.

how many incident edges of this node can be chosen at most in a b -matching. The general matching problem thus corresponds to a b -matching problem where all b -values are equal to one.

In this paper we study a new extension, the so-called *minimum color-degree perfect b -matching problem* (Col-BM), which we introduce via the following application; see Figure 1. Imagine that an airline aims to establish new flight connections using different types of aircraft. The appropriate type of aircraft is given for every connection of interest, and the number of operable connections at each airport is dictated by the takeoff and landing slots owned by the airline. As unused slots have to be returned permanently by policy so that they can be reassigned to other airlines [4], all available slots at all airports have to be utilized. However, operating different types of aircraft at the same airport decreases flexibility in crew scheduling, and increases the necessary space for spare-part storage. Therefore, the maximum number of different types of aircraft operated at any airport should be minimized.

In the setting above, the selection of appropriate flight connections corresponds to a perfect simple b -matching problem, which consists in finding an edge subset of a graph such that the vertices in the resulting subgraph have certain prespecified degrees. However, a classical b -matching neglects the diversity induced by different types of aircraft. We model the different types of aircraft by adding colors to the edges of the underlying graph. This leads to Col-BM, a b -matching extension on an edge-colored graph with the objective of minimizing the maximum number of differently colored edges incident to the same node.

Before providing a formal definition of Col-BM, we introduce some notation that is used in the paper. Let $G = (V, E)$ be an undirected graph with an edge coloring $E_1 \dot{\cup} \dots \dot{\cup} E_q = E$ and $\bar{c} : E \rightarrow \{1, \dots, q\}$ be the corresponding color function with $\bar{c}(e) = j$ if and only if $e \in E_j$. For an edge subset $M \subseteq E$ and a node $v \in V$, $\delta_M(v)$ denotes the set of edges in M that are incident to v , i.e., $\delta_M(v) := \delta(v) \cap M$ with $\delta(v) := \{e \in E \mid v \in e\}$. Further, $\text{col}_M(v)$ denotes the set of colors in $\delta_M(v)$, i.e.,

$$\text{col}_M(v) := \bar{c}(\delta_M(v)) = \{j \in \{1, \dots, q\} \mid \delta_M(v) \cap E_j \neq \emptyset\}.$$

We call the number of different colors of edges in M which are incident to v , i.e. $|\text{col}_M(v)|$, the (M -)color degree of v , similar to [5]. For an edge subset $M \subseteq E$, the color degree of M , $f_G^{\max}(M)$, is defined as the maximum M -color degree of nodes in G , i.e., $f_G^{\max}(M) := \max_{v \in V} |\text{col}_M(v)|$. Finally, we call a subset $M \subseteq E$ a *perfect b -matching* for a mapping $b : V \rightarrow \mathbb{N}_0$ if $|\delta_M(v)| = b(v)$ for every $v \in V$.

Definition 1. Minimum Color-Degree Perfect b -Matching

Given an undirected graph $G = (V, E)$, an edge coloring $E_1 \dot{\cup} \dots \dot{\cup} E_q = E$, and a mapping $b: V \rightarrow \mathbb{N}_0$, the *minimum color-degree perfect b -matching problem* (Col-BM) asks for a perfect b -matching $M \subseteq E$ of minimum color degree $f_G^{\max}(M)$.

In this paper, we study the complexity of Col-BM on different graph classes. Our main contributions can be summarized as follows:

- Col-BM is strongly \mathcal{NP} -hard on two-colored bipartite graphs $G = (V_A \cup V_B, E)$ with $b(v) = 1$ for all $v \in V_A$ and $b(v) = 2$ for all $v \in V_B$.
- There exists no $(2 - \varepsilon)$ -approximation algorithm for Col-BM unless $\mathcal{P} = \mathcal{NP}$.
- Col-BM on two-colored complete bipartite graphs $G = (V_A \cup V_B, E)$, with $b(v) = 1$ for all $v \in V_A$ and $b(v) = 2$ for all $v \in V_B$ can be solved in $O(|V|^2)$ time.
- Col-BM with a fixed number of q colors on series-parallel graphs can be solved in $O(|E| \cdot \max_{v \in V} b(v)^4)$ time.
- Col-BM with a fixed number of q colors on simple graphs $G = (V, E)$ with treewidth $\text{tw}(G) < W$ can be solved in $O(|V| \cdot \max_{v \in V} b(v)^{2W})$ time.

With these results, we extend ongoing studies of matching problems on edge-colored graphs. Most contributions in this field incorporate restrictions depending on the edge coloring in order to reduce the space of feasible solutions. The probably first problem of this kind is the rainbow (or multiple-choice) matching problem [6]: Given an edge-colored graph, find a maximum matching such that all edges have distinct colors. The rainbow matching problem is known to be \mathcal{NP} -complete on bipartite graphs [7], and Le and Pfender [8] more recently proved that it is even \mathcal{APX} -complete on paths. Another problem of this kind is the blue-red matching problem (BRM): Given a blue-red-colored graph and $w \in \mathbb{N}_0$, find a maximum matching which consists of at most w blue and at most w red edges. Nomikos et al. [9] devised an \mathcal{RNC}^2 as well as an asymptotic $\frac{3}{4}$ -approximation algorithm for BRM. The exact complexity of BRM is still open.

One of the earliest weighted matching problems considered on edge-colored graphs is the bounded color matching problem (BCM): Given an edge-colored graph with edge weights, find a maximum weighted matching such that the number of edges in each color satisfies a color-specific upper bound. As a generalization of rainbow matching, all complexity results of the former directly translate to BCM. A straight forward, greedy strategy leads to a $\frac{1}{3}$ -approximation algorithm for BCM [10]. Moreover, several bi-criteria approximation algorithms for BCM, which are allowed to slightly violate the color constraints, are due to Mastrolilli and Stamoulis [10, 11]. Recently, an extension of BCM that additionally incorporates edge costs was studied under the name budgeted colored matching problem [12]. Büsing and Comis [12] present pseudo-polynomial dynamic programs for the budgeted colored matching problem with a fixed number of colors on series-parallel graphs and trees.

The concept of incorporating the edge-coloring into the objective function of a matching problem is, to our knowledge, relatively new and only few problems of this type have been studied yet. One such problem that is closely related to Col-BM is the labeled maximum matching problem (LMM): Given an edge-colored graph, LMM asks for a maximum matching that uses the minimum number of different colors. Monnot [13] showed that LMM is \mathcal{APX} -complete on bipartite complete graphs and 2-approximable on 2-regular bipartite graphs. Subsequently, Carrabs et al. [14] presented alternative mathematical formulations and an exact branch-and-bound scheme for LMM. Another family of related problems are so-called reload cost problems. In reload cost problems, the edge colors symbolize different types of transport, and costs arise for every change of color at a node. The task is to find a specific subgraph for which

the weighted sum of all color changes is minimal. As subgraphs have been considered, e.g., spanning trees [15], paths between two vertices [16], tour or flow problems [17]. For a detailed review of this kind of problems we refer to [18].

A weighted b -matching problem with an objective function incorporating the edge coloring is the diverse weighted b -matching problem (D-WBM). D-WBM can be considered as the counterpart of Col-BM: Given a weighted, edge-colored, bipartite graph, D-WBM asks for a b -matching satisfying upper and lower vertex degree bounds such that the weights of edges incident to the same node are evenly distributed among all colors. In [19], D-WBM is claimed to be \mathcal{NP} -hard and diversification is ensured by minimizing a quadratic function that penalizes unbalanced weight-color distributions rather than adopting a Max-Min approach analogous to our Min-Max approach. For a more extensive review on general matching theory we refer to [10].

This paper is organized as follows. In Section 2, we prove that Col-BM is \mathcal{NP} -hard in general. However, in Section 3, we identify a class of two-colored complete bipartite graphs for which Col-BM is solvable in polynomial time. Furthermore, we provide dynamic programs for solving Col-BM on series-parallel graphs (Section 4) and on simple graphs with bounded treewidth (Section 5) that run in polynomial time if the number of colors is fixed. We close this paper with a conclusion and an outline of future research (Section 6).

2 | COMPLEXITY

Concerning the complexity of Col-BM, we remark that if $b(v) = 1$ for all $v \in V(G)$, Col-BM reduces to a simple, polynomial-time solvable perfect matching problem. In the following we show that in general the decision version of Col-BM is strongly \mathcal{NP} -complete even if $b(v) \in \{1, 2\}$ for all $v \in V(G)$ and $q = 2$.

Theorem 2. *The decision version of Col-BM on two-colored bipartite graphs $G = (V_A \cup V_B, E)$ with $b(v) = 1$ for all $v \in V_A$ and $b(v) = 2$ for all $v \in V_B$ is strongly \mathcal{NP} -complete.*

Proof We reduce (3,B2)-Sat to the decision version of Col-BM. The problem (3,B2)-Sat is a strongly \mathcal{NP} -complete [20] special case of 3-Sat where every literal occurs exactly twice in the formula. Let \mathcal{I} be a (3,B2)-Sat instance with

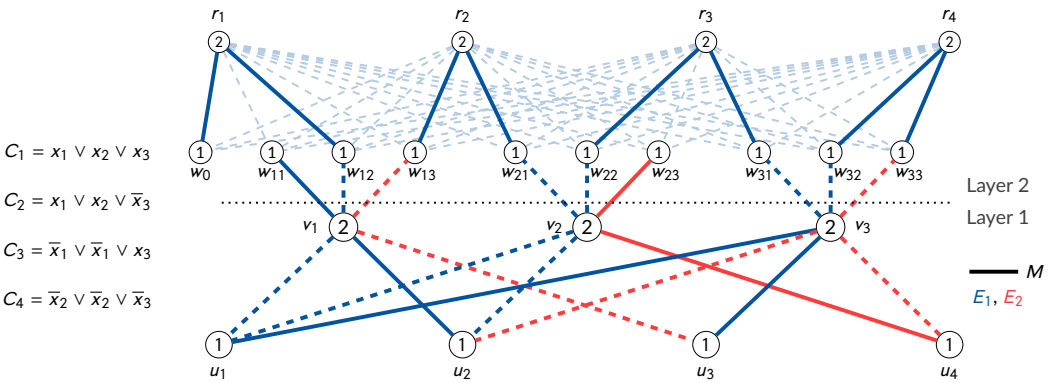


FIGURE 2 Construction of the perfect b -matching.

variables x_1, \dots, x_n and clauses C_1, \dots, C_m . We construct a corresponding Col-BM instance

$$\tilde{\mathcal{I}} := (G = ((U \cup W) \cup (V \cup R), E = E_1 \cup E_2), b),$$

where G is composed of two layers; see Figure 2. Layer 1 models the correspondence between a perfect b -matching with color degree one and a satisfying truth assignment for an instance of (3,B2)-Sat. Layer 2 is an auxiliary, complete bipartite graph ensuring the existence of a perfect b -matching. In the following we refer to edges in E_1 as blue edges and to edges in E_2 as red edges.

Layer 1 contains two sets of nodes $V := \{v_1, \dots, v_n\}$ and $U := \{u_1, \dots, u_m\}$, representing the variables and clauses of \mathcal{I} , respectively. We connect V and U via the following edges: blue edges $\{v_i, u_j\}$ for all positive literals $x_i \in C_j$ and red edges $\{v_i, u_j\}$ for all negative literals $\bar{x}_i \in C_j$. Finally, we set $b(v) = 2$ for all $v \in V$ and $b(u) = 1$ for all $u \in U$. As a result, Layer 1 is bipartite by construction and $\sum_{v \in V} b(v) > \sum_{u \in U} b(u)$ as for \mathcal{I} holds $3m = 4n$.

Layer 2 contains two sets of nodes $W := \{w_{i,k} \mid i \in \{1, \dots, n\}, k \in \{1, 2, 3\}\} \cup W'$ and $R := \{r_i \mid i \in \{1, \dots, \lceil \frac{7}{6}n \rceil\}\}$, ensuring the existence of a perfect b -matching in G . Note that $\frac{7}{3}n$ is integer as 3 divides n . If $\frac{7}{3}n$ is even, we define $W' := \emptyset$ and otherwise $W' := \{w_0\}$. We connect W with V and R via the following edges: a red-colored edge $\{v_i, w_{i,3}\}$ and blue-colored edges $\{v_i, w_{i,1}\}, \{v_i, w_{i,2}\}$ for each $i \in \{1, \dots, n\}$, as well as blue-colored edges $\{r, w\}$ for all $r \in R$ and $w \in W$. Finally, we set $b(w) = 1$ for all $w \in W$ and $b(r) = 2$ for all $r \in R$. As a result, G is bipartite by construction with node partitions $V \cup R$ and $U \cup W$, b -values $b(x) = 2$ for $x \in V \cup R$ and $b(y) = 1$ for $y \in U \cup W$, and

$$\sum_{v \in V} b(v) + \sum_{r \in R} b(r) = \sum_{u \in U} b(u) + \sum_{w \in W} b(w).$$

The Col-BM instance $\tilde{\mathcal{I}}$ can be constructed in polynomial time. Hence, it remains to be shown that \mathcal{I} is a Yes-instance if and only if $\tilde{\mathcal{I}}$ has a perfect b -matching M with color degree $f_G^{\max}(M) = 1$.

Let M be a perfect b -matching in G with $f_G^{\max}(M) = 1$. Then $|\text{col}_M(v_i)| = 1$ for all $i \in \{1, \dots, n\}$ and we set $x_i = \text{True}$ if both edges in $\delta_M(v_i)$ are blue and $x_i = \text{False}$ if both are red. It remains to be shown that x is a satisfying assignment for \mathcal{I} . By construction, for all $j \in \{1, \dots, m\}$ there exists exactly one $i \in \{1, \dots, n\}$ such that $\delta_M(u_j) = \{\{v_i, u_j\}\}$. If $\{v_i, u_j\}$ is blue, then $x_i \in C_j$ by construction. Hence, our choice $x_i = \text{True}$ verifies clause C_j . Analogously, if $\{v_i, u_j\}$ is red, then $\bar{x}_i \in C_j$. Hence, our choice $x_i = \text{False}$ verifies clause C_j . Consequently, x is a satisfying assignment for \mathcal{I} .

Conversely, let x be a satisfying truth assignment for \mathcal{I} and $M = \emptyset$. We choose a verifying literal x_i (\bar{x}_i) for each clause C_j and add the corresponding blue (red) edge $\{v_i, u_j\}$ to M . Thus, we select m edges in Layer 1 and $|\delta_M(u)| = b(u) = 1$ holds for all $u \in U$. As x_i and \bar{x}_i cannot simultaneously be satisfied by x , $\delta_M(v)$ contains only edges of the same color for all $v \in V$. Hence, $f_G^{\max}(M) = 1$.

To conclude our reduction, it suffices to extend M to Layer 2 without increasing $f_G^{\max}(M)$. Therefore, we proceed for every $v_i \in V$ with $|\delta_M(v_i)| < b(v_i)$ as follows: if $\delta_M(v_i) \cap E_1 \neq \emptyset$, add $\{v_i, w_{i,1}\}$ to M ; if $\delta_M(v_i) \cap E_2 \neq \emptyset$, add $\{v_i, w_{i,3}\}$ to M ; if $\delta_M(v_i) = \emptyset$, add both $\{v_i, w_{i,1}\}$ and $\{v_i, w_{i,2}\}$ to M . Thus, $|\delta_M(v)| = 2$ and $|\text{col}_M(v)| = 1$ for all $v \in V$. Finally, let M' be a perfect b -matching in $G' := G[R \cup \{w \in W \mid \delta_M(w) = \emptyset\}]$, which exists as G' is a complete bipartite graph and, by construction,

$$\sum_{r \in R} b(r) = \sum_{w \in W: \delta_M(w) = \emptyset} b(w).$$

Consequently, $M^* := M \cup M'$ is a perfect b -matching in G with $f_G^{\max}(M^*) = 1$.

As the decision version of Col-BM is obviously in \mathcal{NP} , as we can check the feasibility and color-degree of a given b -matching in $O(|V(G)| \cdot |E|)$ time, the problem's strong \mathcal{NP} -completeness follows. ■

Theorem 2 states that we can solve the strongly \mathcal{NP} -complete (3,B2)-Sat problem by deciding whether an optimal perfect b -matching in G has color degree one or two. This directly implies the inapproximability of Col-BM.

Corollary 3. *There exists no $(2 - \varepsilon)$ -approximation algorithm for Col-BM unless $\mathcal{P} = \mathcal{NP}$.*

Note that any b -matching in a two-colored graph has color degree at most two. Hence, every exact algorithm solving the perfect b -matching problem is a 2-approximation algorithm for Col-BM on two-colored graphs.

Remark that Col-BM on a two-colored bipartite graph $G = (V_A \cup V_B, E)$ with $b(v) = 1$ for all $v \in V_A$ and $b(v) = 2$ for all $v \in V_B$ corresponds to the task of partitioning G into monochromatic paths of length 3 whose end-nodes are exclusively in V_A (spanning P_3 Partition). It is known that partitioning an uncolored graph into paths of length 3 (P_3 Partition) is \mathcal{NP} -complete on bipartite graphs of maximum degree 3 [21]. However, to the best of our knowledge, no work has been published on monochromatic P_3 Partition problems in edge-colored graphs nor on spanning P_3 Partition problems in uncolored graphs. In the case that $b(v) = r$, $r \in \mathbb{N}$, for all $v \in V$, Col-BM is closely related to the partitioning of graphs into monochromatic r -factors. A survey on partitioning problems of edge-colored graphs into monochromatic subgraphs can be found in [22].

3 | COMPLETE BIPARTITE GRAPHS

In the previous section, we have proven that Col-BM is \mathcal{NP} -hard on two-colored bipartite graphs $G = (V_A \cup V_B, E = E_1 \cup E_2)$ with $b(v) = 1$ for all $v \in V_A$ and $b(v) = 2$ for all $v \in V_B$. In this section, we additionally assume G to be complete bipartite and prove that in this case Col-BM is solvable in polynomial time by providing a constructive algorithm. For better lucidity we abbreviate the edge notation $\{v, w\}$ as vw in this section.

Let $G = (V_A \cup V_B, E = E_1 \cup E_2)$ be a two-colored complete bipartite graph with color function \bar{c} and $b(v) = 1$ for all $v \in V_A$ and $b(v) = 2$ for all $v \in V_B$. We assume $|V_A| = 2|V_B|$ to ensure that G contains a perfect b -matching. As a result, Col-BM reduces to the question whether G contains a perfect b -matching M with $f_G^{\max}(M) = 1$.

We utilize two characteristics of such graphs to classify those for which a perfect b -matching M with $f_G^{\max}(M) = 1$ exists. We begin by identifying a subgraph which is sufficient for the existence of a perfect b -matching M with $f_G^{\max}(M) = 1$.

Lemma 4. Let $G = (V_A \cup V_B, E)$ be a two-colored complete bipartite graph with $b(v) = 1$ for all $v \in V_A$, $b(v) = 2$ for all $v \in V_B$. If G contains the gadget

$$G' := (\{b, c, d\} \cup \{r, s\}, \{br\} \cup \{bs, cr, cs, ds\}),$$

illustrated in Figure 3(a) as a subgraph, then there exists a perfect b -matching M in G with $f_G^{\max}(M) = 1$.

Proof Let G be a graph that contains the subgraph G' . We present an algorithm to construct a perfect b -matching M in G with $f_G^{\max}(M) = 1$. Therefore, let G' be the subgraph defined above and initialize $M = \emptyset$. For a given $M \subseteq E$, we call a node $v \in V(G)$ (M)-unsatisfied if $|\delta_M(v)| < b(v)$.

Repeat the following two steps until all $w \in V_B \setminus V(G')$ are satisfied. First, choose a node $w \in V_B \setminus V(G')$ and three distinct, unsatisfied nodes $v_1, v_2, v_3 \in V_A \setminus V(G')$. Second, add two arbitrary edges $e, f \in \{v_1w, v_2w, v_3w\}$ of identical color to M , which exist as G is two-colored.

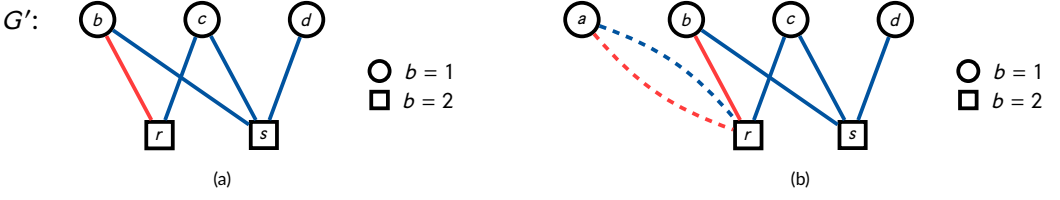


FIGURE 3 (a) Sufficient subgraph G' . (b) Options for edge ar .

By construction $f_G^{\max}(M) = 1$, exactly one node $a \in V_A \setminus V(G')$ remains unsatisfied, and M is a perfect b -matching in $G[V \setminus (\{a\} \cup V(G'))]$. Hence, it suffices to prove that there always exists a perfect b -matching M' in the induced subgraph $G'' := G[\{a\} \cup V(G')]$ with $f_{G''}^{\max}(M') = 1$, as then $M^* := M \cup M'$ is a perfect b -matching in G with $f_G^{\max}(M^*) = 1$. We distinguish two cases based on the color of the edge ar ; see Figure 3(b):

- 1) If $\bar{c}(ar) = \bar{c}(br)$, then $M' = \{ar, br, cs, ds\}$ is a perfect b -matching in G'' with $f_{G''}^{\max}(M') = 1$.
- 2) If $\bar{c}(ar) \neq \bar{c}(br)$, then $\bar{c}(ar) = \bar{c}(cr)$ and $M' = \{ar, cr, bs, ds\}$ is a perfect b -matching in G'' with $f_{G''}^{\max}(M') = 1$.

Hence, $M^* := M \cup M'$ is a perfect b -matching in G with $f_G^{\max}(M^*) = 1$. ■

As not all instances contain the gadget G' , we continue by exploiting the fact that in every perfect b -matching M in a q -colored complete bipartite graph $G = (V_A \cup V_B, E)$ with $f_G^{\max}(M) = 1$, the incident edges $\delta_M(v)$ of every node $v \in V_B$ are necessarily of the same color. We still assume $b(v) = 1$ for all $v \in V_A$ and $b(v) = 2$ for all $v \in V_B$. As a result, for every node $v \in V_B$ only node pairs that are connected to v by edges of the same color are potential matching partners.

Definition 5. Let G be a q -colored graph. For $v \in V(G)$ and a color $i \in \{1, \dots, q\}$, we define the i -colored neighborhood of v as

$$N^i(v) := \{w \in V(G) \mid \bar{c}(vw) = i\}.$$

Remark that in a complete bipartite graph $G = (V_A \cup V_B, E)$ every node $v \in V_B$ induces a partition $\{N^1(v), \dots, N^q(v)\}$ of V_A . If this partition of V_A is identical for all $v \in V_B$, i.e., $\{N^1(r), \dots, N^q(r)\} = \{N^1(s), \dots, N^q(s)\}$ for all $r, s \in V_B$, we call G *stable (color) partitioned*; see Figure 4. We use the notion of a stable partitioning to determine whether a perfect b -Matching M in G with $f_G^{\max}(M) = 1$ exists.

Lemma 6. Let $G = (V_A \cup V_B, E)$ be a q -colored, stable-partitioned, complete bipartite graph with $b(v) = 1$ for all

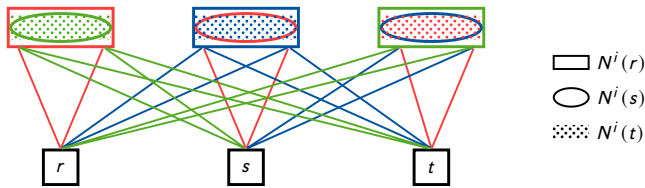


FIGURE 4 Stable-partitioned graph with the identical partitions of V_A induced by the i -colored neighborhoods of nodes $r, s, t \in V_B$.

$v \in V_A$, $b(v) = 2$ for all $v \in V_B$. Then there exists a perfect b -matching M in G with $f_G^{\max}(M) = 1$ if and only if $|N^i(w)|$ is even for all colors $i \in \{1, \dots, q\}$ and $w \in V_B$.

Proof Let $\{P_1, \dots, P_q\}$ denote the unique partition of V_A induced by the set of i -colored neighborhoods of $r \in V_B$. On the one hand, if $|P_i|$ is even for all $i \in \{1, \dots, q\}$, then we construct a perfect b -matching M with $f_G^{\max}(M) = 1$ by iteratively matching two unsatisfied nodes belonging to the same class P_i to an unsatisfied node of V_B .

On the other hand, if M is a perfect b -Matching with $f_G^{\max}(M) = 1$, then every P_i is canonically partitioned by M into disjoint node pairs. Thus, $|P_i|$ has to be even for all $i \in \{1, \dots, q\}$. ■

We proceed by proving that every two-colored complete bipartite graph $G = (V, E)$ with $|V| > 6$ either fulfills the conditions of Lemma 4 or the conditions of Lemma 6. This leads to a complete characterization of two-colored complete bipartite graphs with more than six nodes and will be used to derive an algorithm for this graph class.

Lemma 7. Let $G = (V_A \cup V_B, E)$ be a two-colored complete bipartite graph with $|V_A| = 2|V_B|$, and $|V(G)| > 6$. Then exactly one of the following is true.

- 1) G contains the gadget G' defined in Lemma 4.
- 2) G is stable partitioned.

Proof Assume 1) holds. Then 2) is violated as r and s induce different partitions of $\{b, c, d\}$.

Conversely, assume 2) is violated. Therefore, there exist $r, s \in V_B$ such that $\{N^1(r), N^2(r)\} \neq \{N^1(s), N^2(s)\}$. Hence, at least one of the following holds

- i) $N^1(s)$ intersects both $N^1(r)$ and $N^2(r)$, i.e., $N^1(r) \cap N^1(s) \neq \emptyset \wedge N^2(r) \cap N^1(s) \neq \emptyset$,
- ii) $N^2(s)$ intersects both $N^1(r)$ and $N^2(r)$, i.e., $N^1(r) \cap N^2(s) \neq \emptyset \wedge N^2(r) \cap N^2(s) \neq \emptyset$.

Remark that as $|V(G)| > 6$ and $|V_A| = 2|V_B|$, it directly follows that $|V_A| \geq 6$ and $|V_B| \geq 3$.

Without loss of generality assume that i) holds, as the argumentation is analogous in the case that ii) holds. The validity of i) directly implies $|N^1(s)| \geq 2$. If $|N^1(s)| \geq 3$, then we choose $b \in N^2(r) \cap N^1(s)$, $c \in N^1(r) \cap N^1(s)$ and $d \in N^1(s) \setminus \{b, c\}$. Therefore, bs , cr , cs and ds are of color one whereas br is of color two; see Figure 5. Consequently, $(\{b, c, d, r, s\}, \{br, bs, cr, cs, ds\})$ represents a gadget as defined in Lemma 4. If $|N^1(s)| = 2$ and ii) holds, then $|N^2(s)| \geq 3$ and the statement follows via symmetry.

Therefore, assume that $|N^1(s)| = 2$ and ii) is violated. Then either $|N^1(r)| = 1$ or $|N^2(r)| = 1$; see Figure 6. If $|N^1(r)| = 1$, then $|N^2(r)| \geq 5$ and $|N^2(r) \cap N^2(s)| \geq 4$, and we choose $b \in N^2(r) \cap N^1(s)$, $c \in N^2(r) \cap N^2(s)$, and $d \in N^2(r) \setminus \{b, c\}$. Therefore, br , cr , cs and dr are of color two whereas bs is of color one. Consequently, $(\{b, c, d, r, s\}, \{br, bs, cr, cs, dr\})$ represents a gadget as defined in Lemma 4; see Figure 6(a). If

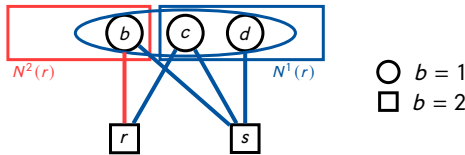


FIGURE 5 Setting if i) holds and $|N^1(s)| \geq 3$.

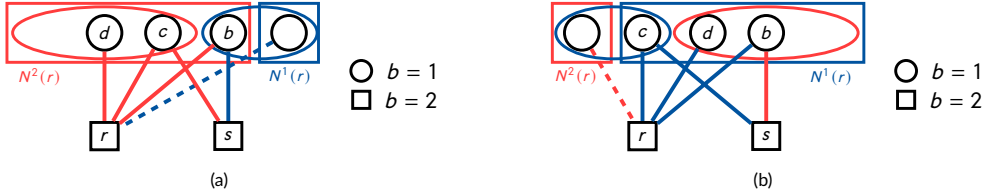


FIGURE 6 $|N^1(s)| = 2$ and ii) is violated and (a) $|N^1(r)| = 1$; (b) $|N^2(r)| = 1$.

$|N^2(r)| = 1$, then $|N^1(r)| \geq 5$ and $|N^1(r) \cap N^2(s)| \geq 4$, and we choose $b \in N^1(r) \cap N^2(s)$, $c \in N^1(r) \cap N^1(s)$ and $d \in N^1(r) \setminus \{b, c\}$. Therefore br, cr, cs and dr are of color one whereas bs is of color two. Consequently, $(\{b, c, d, r, s\}, \{br, bs, cr, cs, dr\})$ represents a gadget as defined in Lemma 4; see Figure 6(b).

We conclude, if 2) is violated, then 1) holds. ■

Remark that the condition imposed on the size of the graph in Lemma 7 is tight.

Proposition 8. *There exists a graph G with $|V(G)| = 6$ that is neither stable partitioned nor does it contain the gadget G' .*

Proof The graph

$$(\{a, b, c, d\} \cup \{r, s\}, \{ar, as, br, ds\} \cup \{bs, cr, cs, dr\})$$

neither is stable partitioned nor contains the gadget G' ; see Figure 7. ■

The results from Lemma 7 imply that, on a two-colored complete bipartite graph $G = (V_A \cup V_B, E)$ with $|V(G)| > 6$, Col-BM can be reduced to identifying the gadget G' as subgraph, or determining that none exists; see Algorithm 1: We first check whether G is stable partitioned. If this is the case, G does not contain the gadget G' and we can determine the minimum color degree of a perfect b -matching in G by checking the cardinalities of the elements of the unique partition of V_A : if all cardinalities are even, the minimum color degree of a perfect b -matching in G is one otherwise it is two. In the other case, G contains the gadget G' and, hence, the minimum color degree of a perfect b -matching in G is one.

Theorem 9. *Col-BM on two-colored complete bipartite graphs $G = (V_A \cup V_B, E)$ with $b(v) = 1$ for all $v \in V_A$ and $b(v) = 2$ for all $v \in V_B$ can be solved in $O(|V|^2)$ time using Algorithm 1.*

Proof The correctness of Algorithm 1 follows from Lemmata 4, 6, and 7. Regarding the runtime: P and S can be computed in $O(|V_A|) = O(|V|)$ time. The comparison of S and P can be performed in $O(|V_A|)$ time if they are represented using characteristic vectors. Thus, Algorithm 1 checks if every $w \in V_B$ induces the same partition of V_A

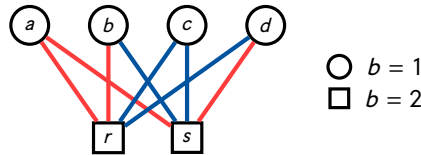


FIGURE 7 Graph G with $|V(G)| = 6$ that neither is stable partitioned nor contains the gadget G' .

Algorithm 1: Solve Col-BM

```

1 choose  $r \in V_B$ 
2  $P := \{N^1(r), N^2(r)\}$ 
3 for  $s \in V_B \setminus \{r\}$  do // check if  $G$  is stable partitioned
4    $S := \{N^1(s), N^2(s)\}$ 
5   if  $P \neq S$  then // gadget  $G'$  exists
6     return  $f^{\max} := 1$ 
7 for  $p \in P$  do // check if all  $|\rho|$  are even
8   if  $|\rho|$  is odd then
9     return  $f^{\max} := 2$ 
10 return  $f^{\max} := 1$ 

```

in $O(|V_A| + |V_B| |V_A|)$ time. The cardinalities of the two color classes and their parity can be checked in $O(|V_A|)$. Hence, Algorithm 1 solves Col-BM in $O(|V|^2)$ time. ■

Remark that if Algorithm 1 terminates in line 6 (line 10), an optimal perfect b -Matching can be determined using the construction from the proof of Lemma 4 (Lemma 6).

4 | SERIES-PARALLEL GRAPHS

In this section, we consider Col-BM on series-parallel (SP)-graphs. We show that, in case of a fixed number of colors, Col-BM can be solved in polynomial time on SP-graphs by dynamic programming. Subsequently, we extend our dynamic program to solve Col-BM on trees.

We start with a formal definition of SP-graphs based on the one given in [23].

Definition 10. A (2-terminal) SP-graph with two distinguished nodes σ and τ , called *source* and *sink*, is defined as follows.

- 1) An edge $\{\sigma, \tau\}$ is SP.
- 2) A graph, constructed by a finite number of the following operations, is SP.
 - i) Combine two SP-graphs G_1, G_2 with sources σ^1, σ^2 and sinks τ^1, τ^2 by identifying τ^1 with σ^2 , called *series composition* of G_1 and G_2 .
 - ii) Combine two SP-graphs G_1, G_2 with sources σ^1, σ^2 and sinks τ^1, τ^2 by identifying σ^1 with σ^2 and τ^1 with τ^2 , called *parallel composition* of G_1 and G_2 .

The series and parallel composition are visualized exemplarily in Figure 8(a).

Every SP-graph G can be associated with a *decomposition tree* $T = T(G)$, which is a rooted, binary tree whose nodes correspond to the subgraphs of G appearing in the recursive construction; see Figure 8(b). The leaves of the decomposition tree correspond to edges in G . The inner nodes of the decomposition tree are of two different types: an S -node corresponds to the series-composition of the graphs associated with its child nodes and, analogously, a P -node corresponds to the parallel composition of its child nodes. We denote the root of T with r and it corresponds

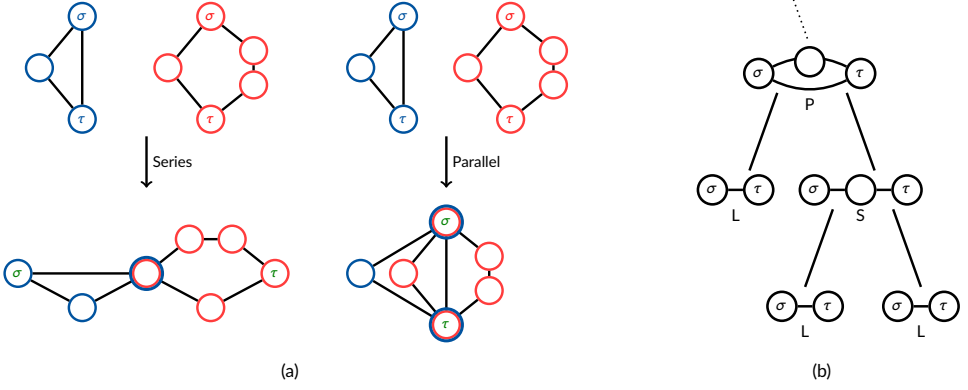


FIGURE 8 (a) Parallel and series composition. (b) Example of an SP-tree.

to G itself by construction.

Let $G = (V, E)$ be an SP-graph with edge coloring $E_1 \dot{\cup} \dots \dot{\cup} E_q = E$, and $b: V \rightarrow \mathbb{N}_0$ a mapping. It is known that a decomposition tree can be computed in linear time for an SP-graph [24]. Thus, let T be a decomposition tree for G . For $t \in V(T)$, let G_t denote the subgraph of G with source σ^t and sink τ^t corresponding to t . We propose a dynamic program to solve Col-BM on SP-graphs using the corresponding decomposition trees.

First, we introduce a set of labels

$$\mathcal{L}^t = \{ (\alpha, F_\sigma, \beta, F_\tau) \mid 0 \leq \alpha \leq b(\sigma^t), 0 \leq \beta \leq b(\tau^t), F_\sigma, F_\tau \subseteq \{1, \dots, q\} \}$$

for every $t \in V(T)$. The parameters α and β define new, smaller b -values for σ^t and τ^t , whereas the *color-subsets* F_σ, F_τ define the prespecified set of colors for edges incident to σ^t and τ^t .

Before we specify our dynamic program, we introduce some more notation. For a node $t \in V(T)$ and a label $x = (\alpha, F_\sigma, \beta, F_\tau) \in \mathcal{L}^t$, we call a subset $M \subseteq E(G_t)$ a (t, x) -restricted matching if $|\delta_M(\sigma^t)| = \alpha$, $|\delta_M(\tau^t)| = \beta$, $\text{col}_M(\sigma^t) = F_\sigma$, $\text{col}_M(\tau^t) = F_\tau$ and $|\delta_M(v)| = b(v)$ for all $v \in V(G_t) \setminus \{\sigma^t, \tau^t\}$. Consequently, we define the (t, x) -restricted Col-BM as

$$\min_{M \subseteq E(G_t)} \left\{ f_{G_t}^{\max}(M) \mid M \text{ is } (t, x)\text{-restricted matching in } G_t \right\}.$$

For a node $t \in V(T)$ and a label $x \in \mathcal{L}^t$, we call the optimal solution value of the (t, x) -restricted Col-BM the cost $c^t(x)$ of x at t . Thus, for all perfect b -matchings M^* in G with minimum color degree it holds that

$$f_G^{\max}(M^*) = \min_{F_\sigma, F_\tau \subseteq \{1, \dots, q\}} c^r((b(\sigma^r), F_\sigma, b(\tau^r), F_\tau)),$$

for the root r of T . Our dynamic program solving the Col-BM on SP-graphs exploits the structure of decomposition trees and recursively computes label costs bottom-up. To that end, we consider the three types of nodes in the decomposition tree of G starting with the initialization in leaves.

Lemma 11. Let $t \in V(T)$ be a leaf in T , and let e denote the only edge in the corresponding graph G_t . Then $c^t((0, \emptyset, 0, \emptyset)) = 0$, $c^t((1, \{\bar{e}(e)\}, 1, \{\bar{e}(e)\})) = 1$, and $c^t(x) = \infty$ for all other labels $x \in \mathcal{L}^t$.

Proof If $t \in V(T)$ is a leaf in T , the corresponding graph G_t consists of exactly one edge e by the definition of decomposition trees. Therefore, there exists exactly one $(t, (0, \emptyset, 0, \emptyset))$ -restricted matching: $M_0 = \emptyset$. Hence, $c^t((0, \emptyset, 0, \emptyset)) = f_{G_t}^{\max}(M_0) = 0$. There also exists exactly one $(t, (1, \{\bar{c}(e)\}, 1, \{\bar{c}(e)\}))$ -restricted matching: $M_1 = \{e\}$. Hence, $c^t((1, \{\bar{c}(e)\}, 1, \{\bar{c}(e)\})) = f_{G_t}^{\max}(M_1) = 1$. For all other labels $x \in \mathcal{L}^t$, the (t, x) -restricted Col-BM is infeasible and hence, $c^t(x) = \infty$. ■

For the two remaining types of tree nodes, label costs can be derived recursively from the label costs of child nodes. We begin by considering S -nodes, which correspond to the series composition of the graphs associated with its child nodes. As a result of this interrelation, every restricted matching at an S -nodes can be decomposed into two restricted matchings at its child nodes. By minimizing over all feasible combinations of restricted matchings at the child nodes, we get the following.

Lemma 12. Let $t \in V(T)$ be an S -node in T with child nodes ℓ and u . Then the cost of $x^t = (\alpha^t, F_\sigma^t, \beta^t, F_\tau^t) \in \mathcal{L}^t$ at t can be computed as

$$c^t(x^t) = \min_{\substack{0 \leq k \leq b(\tau^\ell), \\ F_\tau^\ell, F_\sigma^u \subseteq \{1, \dots, q\}}} \max \left\{ c^\ell \left(\left(\alpha^t, F_\sigma^t, k, F_\tau^\ell \right) \right), c^u \left(\left(b(\tau^\ell) - k, F_\sigma^u, \beta^t, F_\tau^t \right) \right), \left| F_\tau^\ell \cup F_\sigma^u \right| \right\}.$$

Proof If $t \in V(T)$ is an S -node with child nodes ℓ and u , by definition $\sigma^t = \sigma^\ell$, $\tau^t = \tau^u$, and $\tau^\ell = \sigma^u =: y$. Let $x^t = (\alpha^t, F_\sigma^t, \beta^t, F_\tau^t) \in \mathcal{L}^t$ and $M^t \subseteq E(G_t)$ be an optimal solution to the (t, x^t) -restricted Col-BM, i.e., $c^t(x^t) = f_{G_t}^{\max}(M^t)$. By defining $M^\ell := M^t \cap E(G_\ell)$ and $M^u := M^t \cap E(G_u)$, it follows that

$$f_{G_t}^{\max}(M^t) = \max \left\{ f_{G_\ell}^{\max}(M^\ell), f_{G_u}^{\max}(M^u), |\text{col}_{M^\ell}(y) \cup \text{col}_{M^u}(y)| \right\}. \quad (1)$$

Furthermore, for $\bar{k} := |\delta_{M^\ell}(y)|$, $\bar{F}_\tau^\ell := \text{col}_{M^\ell}(y)$, and $\bar{F}_\sigma^u := \text{col}_{M^u}(y)$ it holds that M^ℓ is an $(\ell, (\alpha^t, F_\sigma^t, \bar{k}, \bar{F}_\tau^\ell))$ -restricted matching in G_ℓ while M^u is a $(u, (b(y) - \bar{k}, \bar{F}_\sigma^u, \beta^t, F_\tau^t))$ -restricted matching in G_u . Thus by definition, $f_{G_\ell}^{\max}(M^\ell) \geq c^\ell \left((\alpha^t, F_\sigma^t, \bar{k}, \bar{F}_\tau^\ell) \right)$ and $f_{G_u}^{\max}(M^u) \geq c^u \left((b(y) - \bar{k}, \bar{F}_\sigma^u, \beta^t, F_\tau^t) \right)$ which yields in combination with (1) that

$$\begin{aligned} c^t(x^t) &= f_{G_t}^{\max}(M^t) = \max \left\{ f_{G_\ell}^{\max}(M^\ell), f_{G_u}^{\max}(M^u), |\text{col}_{M^\ell}(y) \cup \text{col}_{M^u}(y)| \right\} \\ &\geq \max \left\{ c^\ell \left((\alpha^t, F_\sigma^t, \bar{k}, \bar{F}_\tau^\ell) \right), c^u \left((b(y) - \bar{k}, \bar{F}_\sigma^u, \beta^t, F_\tau^t) \right), \left| \bar{F}_\tau^\ell \cup \bar{F}_\sigma^u \right| \right\} \\ &\geq \min_{\substack{0 \leq k \leq b(y), \\ F_\tau^\ell, F_\sigma^u \subseteq \{1, \dots, q\}}} \max \left\{ c^\ell \left((\alpha^t, F_\sigma^t, k, F_\tau^\ell) \right), c^u \left((b(y) - k, F_\sigma^u, \beta^t, F_\tau^t) \right), \left| F_\tau^\ell \cup F_\sigma^u \right| \right\}. \end{aligned}$$

Conversely, let

$$k^*, F_\tau^{\ell*}, F_\sigma^{u*} = \arg \min_{\substack{0 \leq k \leq b(y), \\ F_\tau^\ell, F_\sigma^u \subseteq \{1, \dots, q\}}} \max \left\{ c^\ell \left((\alpha^t, F_\sigma^t, k, F_\tau^\ell) \right), c^u \left((b(y) - k, F_\sigma^u, \beta^t, F_\tau^t) \right), \left| F_\tau^\ell \cup F_\sigma^u \right| \right\}.$$

Moreover, let $M^\ell \subseteq E(G_\ell)$ be an optimal solution to the $(\ell, (\alpha^t, F_\sigma^t, k^*, F_\tau^{\ell*}))$ -restricted Col-BM on G_ℓ and $M^u \subseteq E(G_u)$ be an optimal solution to the $(u, (b(y) - k^*, F_\sigma^{u*}, \beta^t, F_\tau^t))$ -restricted Col-BM on G_u . We define the matching $M^t := M^\ell \cup M^u$ in G_t . By construction, M^t is (t, x^t) -restricted, $\text{col}_{M^\ell}(\tau) = F_\tau^{\ell*}$, and $\text{col}_{M^u}(\sigma) = F_\sigma^{u*}$. Thus,

$$\begin{aligned} c^t(x^t) &\leq f_{G_t}^{\max}(M^t) = \max \left\{ f_{G_\ell}^{\max}(M^\ell), f_{G_u}^{\max}(M^u), |\text{col}_{M^\ell}(\tau) \cup \text{col}_{M^u}(\sigma)| \right\} \\ &= \max \left\{ c^\ell \left((\alpha^t, F_\sigma^t, k^*, F_\tau^{\ell*}) \right), c^u \left((b(y) - k^*, F_\sigma^{u*}, \beta^t, F_\tau^t) \right), \left| F_\tau^{\ell*} \cup F_\sigma^{u*} \right| \right\} \\ &= \min_{\substack{0 \leq k \leq b(y), \\ F_\tau^\ell, F_\sigma^u \subseteq \{1, \dots, q\}}} \max \left\{ c^\ell \left((\alpha^t, F_\sigma^t, k, F_\tau^\ell) \right), c^u \left((b(y) - k, F_\sigma^u, \beta^t, F_\tau^t) \right), \left| F_\tau^\ell \cup F_\sigma^u \right| \right\}. \end{aligned}$$

To conclude the computation of label costs, we consider P -nodes. Recall, that P -nodes correspond to the parallel composition of the graphs associated with its child nodes. Thus, we can again compute the cost of labels by minimizing over all feasible combinations of restricted matchings at the child nodes.

Lemma 13. Let $t \in V(T)$ be a P -node in T with child nodes ℓ and u . Then the cost of $x^t = (\alpha^t, F_\sigma^t, \beta^t, F_\tau^t) \in \mathcal{L}^t$ at t can be computed as

$$c^t(x^t) = \min_{\substack{0 \leq k \leq \alpha^t, F_\sigma^\ell \cup F_\sigma^u = F_\sigma^t \\ 0 \leq m \leq \beta^t, F_\tau^\ell \cup F_\tau^u = F_\tau^t}} \max \left\{ c^\ell \left((k, F_\sigma^\ell, m, F_\tau^\ell) \right), c^u \left((\alpha^t - k, F_\sigma^u, \beta^t - m, F_\tau^u) \right), |F_\sigma^t|, |F_\tau^t| \right\}.$$

Proof If $t \in V(T)$ is a P -node with child nodes ℓ and u , by definition $\sigma^\ell = \sigma^u = \sigma^t$ and $\tau^\ell = \tau^u = \tau^t$. Let $x^t = (\alpha^t, F_\sigma^t, \beta^t, F_\tau^t) \in \mathcal{L}^t$ and $M^t \subseteq E(G_t)$ be an optimal solution to the (t, x^t) -restricted Col-BM, i.e., $c^t(x^t) = f_{G_t}^{\max}(M^t)$. By defining $M^\ell := M^t \cap E(G_\ell)$ and $M^u := M^t \cap E(G_u)$, it follows that

$$\begin{aligned} f_{G_t}^{\max}(M^t) &= \max \left\{ f_{G_\ell}^{\max}(M^\ell), f_{G_u}^{\max}(M^u), |\text{col}_{M^\ell}(\sigma) \cup \text{col}_{M^u}(\sigma)|, |\text{col}_{M^\ell}(\tau) \cup \text{col}_{M^u}(\tau)| \right\} \\ &= \max \left\{ f_{G_\ell}^{\max}(M^\ell), f_{G_u}^{\max}(M^u), |F_\sigma^t|, |F_\tau^t| \right\}. \end{aligned} \quad (2)$$

For the choice of $\bar{k} := |\delta_{M^\ell}(\sigma)|$, $\bar{m} := |\delta_{M^\ell}(\tau)|$, $\bar{F}_\sigma^\ell := \text{col}_{M^\ell}(\sigma)$, and $\bar{F}_\tau^\ell := \text{col}_{M^\ell}(\tau)$, the matching M^ℓ is $(\ell, (\bar{k}, \bar{F}_\sigma^\ell, \bar{m}, \bar{F}_\tau^\ell))$ -restricted by construction. Moreover, for $\bar{F}_\sigma^u := \text{col}_{M^u}(\sigma)$ and $\bar{F}_\tau^u := \text{col}_{M^u}(\tau)$ the matching M^u is $(u, (\alpha^t - \bar{k}, \bar{F}_\sigma^u, \beta^t - \bar{m}, \bar{F}_\tau^u))$ -restricted. Thus, it follows by definition that $f_{G_\ell}^{\max}(M^\ell) \geq c^\ell \left((\bar{k}, \bar{F}_\sigma^\ell, \bar{m}, \bar{F}_\tau^\ell) \right)$ and $f_{G_u}^{\max}(M^u) \geq c^u \left((\alpha^t - \bar{k}, \bar{F}_\sigma^u, \beta^t - \bar{m}, \bar{F}_\tau^u) \right)$ which yields in combination with (2) that

$$\begin{aligned} c^t(x^t) &= f_{G_t}^{\max}(M^t) = \max \left\{ f_{G_\ell}^{\max}(M^\ell), f_{G_u}^{\max}(M^u), |F_\sigma^t|, |F_\tau^t| \right\} \\ &\geq \max \left\{ c^\ell \left((\bar{k}, \bar{F}_\sigma^\ell, \bar{m}, \bar{F}_\tau^\ell) \right), c^u \left((\alpha^t - \bar{k}, \bar{F}_\sigma^u, \beta^t - \bar{m}, \bar{F}_\tau^u) \right), |F_\sigma^t|, |F_\tau^t| \right\} \\ &\geq \min_{\substack{0 \leq k \leq \alpha^t, F_\sigma^\ell \cup F_\sigma^u = F_\sigma^t \\ 0 \leq m \leq \beta^t, F_\tau^\ell \cup F_\tau^u = F_\tau^t}} \max \left\{ c^\ell \left((k, F_\sigma^\ell, m, F_\tau^\ell) \right), c^u \left((\alpha^t - k, F_\sigma^u, \beta^t - m, F_\tau^u) \right), |F_\sigma^t|, |F_\tau^t| \right\}. \end{aligned}$$

Conversely, let

$$k^*, F_\sigma^{\ell*}, F_\sigma^{u*}, m^*, F_\tau^{\ell*}, F_\tau^{u*} = \arg \min_{\substack{0 \leq k \leq \alpha^t, F_\sigma^\ell \cup F_\sigma^u = F_\sigma^t \\ 0 \leq m \leq \beta^t, F_\tau^\ell \cup F_\tau^u = F_\tau^t}} \max \left\{ c^\ell \left((k, F_\sigma^\ell, m, F_\tau^\ell) \right), c^u \left((\alpha^t - k, F_\sigma^u, \beta^t - m, F_\tau^u) \right), |F_\sigma^t|, |F_\tau^t| \right\}.$$

Moreover, let $M^\ell \subseteq E(G_\ell)$ be an optimal solution to the $(\ell, (k^*, F_\sigma^{\ell*}, m^*, F_\tau^{\ell*}))$ -restricted Col-BM on G_ℓ and $M^u \subseteq E(G_u)$ be an optimal solution to the $(u, (\alpha^t - k^*, F_\sigma^{u*}, \beta^t - m^*, F_\tau^{u*}))$ -restricted Col-BM on G_u . We define the matching $M^t := M^\ell \cup M^u$ in G_t . By construction, M^t is (t, x^t) -restricted, $F_\sigma^{\ell*} \cup F_\sigma^{u*} = F_\sigma^t$, and $F_\tau^{\ell*} \cup F_\tau^{u*} = F_\tau^t$. Thus,

$$\begin{aligned} c^t(x^t) &\leq f_{G_t}^{\max}(M^t) = \max \left\{ f_{G_\ell}^{\max}(M^\ell), f_{G_u}^{\max}(M^u), |\text{col}_{M^\ell}(\sigma) \cup \text{col}_{M^u}(\sigma)|, |\text{col}_{M^\ell}(\tau) \cup \text{col}_{M^u}(\tau)| \right\} \\ &= \max \left\{ f_{G_\ell}^{\max}(M^\ell), f_{G_u}^{\max}(M^u), |F_\sigma^{\ell*} \cup F_\sigma^{u*}|, |F_\tau^{\ell*} \cup F_\tau^{u*}| \right\} = \max \left\{ f_{G_\ell}^{\max}(M^\ell), f_{G_u}^{\max}(M^u), |F_\sigma^t|, |F_\tau^t| \right\} \\ &= \max \left\{ c^\ell \left((k^*, F_\sigma^{\ell*}, m^*, F_\tau^{\ell*}) \right), c^u \left((\alpha^t - k^*, F_\sigma^{u*}, \beta^t - m^*, F_\tau^{u*}) \right), |F_\sigma^t|, |F_\tau^t| \right\} \\ &= \min_{\substack{0 \leq k \leq \alpha^t, F_\sigma^\ell \cup F_\sigma^u = F_\sigma^t \\ 0 \leq m \leq \beta^t, F_\tau^\ell \cup F_\tau^u = F_\tau^t}} \max \left\{ c^\ell \left((k, F_\sigma^\ell, m, F_\tau^\ell) \right), c^u \left((\alpha^t - k, F_\sigma^u, \beta^t - m, F_\tau^u) \right), |F_\sigma^t|, |F_\tau^t| \right\}. \end{aligned}$$



A perfect b -matching M^* in G of minimum color degree can be obtained by backtracking the chosen minima in the steps of the dynamic program.

Next, we consider the runtime of our dynamic program. For better lucidity, let $B = \max_{v \in V} b(v)$.

Theorem 14. *Col-BM parameterized by the number of colors q on SP-graphs is fixed-parameter tractable (FPT) and can be solved in $O(|E| \cdot 36^q \cdot B^4)$ time.*

Proof The correctness of the algorithm follows from Lemmata 11, 12, and 13. Regarding its runtime, observe that the costs of $O(B^2 \cdot 4^q)$ labels need to be computed for each node $t \in V(T)$. The computational complexity of computing costs of labels is dominated by the computation time of label costs for P -nodes. For P -nodes, we have to minimize over $O(B)$ choices for k and m , respectively. For each color in F_G^t , that color can be either in F_G^ℓ , in F_G^u , or in both which yields $O(3^q)$ possibilities. The same estimation holds for F_G^t and thus we compute the minimum of at most $O(9^q \cdot B^2)$ maxima and every maximum can be calculated in $O(1)$ time. As $|V(T)| = 2|E| - 1$, the total runtime of the algorithm is in $O(|E| \cdot 36^q \cdot B^4)$. ■

We note that in all Col-BM instances, $B \leq |E|$ and therefore our algorithm has polynomial runtime if q is constant.

Moreover, we can extend our algorithm to solve Col-BM on trees as follows: given a Col-BM instance \mathcal{I} on a tree $T = (V, E)$, we construct an auxiliary graph G by adding a new vertex y , connecting it to all leaves of T and setting $b(y) = 0$. By construction, G is SP [25] and contains at most $2(|V| - 1)$ edges. Furthermore, every perfect b -matching in G contains no edges from $\delta_G(y) = E(G) \setminus E$ and is therefore a perfect b -matching in T .

Corollary 15. *Col-BM parameterized by the number of colors q on trees is FPT and can be solved in $O(|V| \cdot 36^q \cdot B^4)$ time.*

5 | GRAPHS WITH BOUNDED TREewidth

We proceed by considering Col-BM on graphs with bounded treewidth, which is a more general graph class that includes SP-graphs. Using dynamic programming, we show that Col-BM on graphs with bounded treewidth is polynomial-time solvable for a fixed number of colors. Before we present the details of our algorithm, we introduce the concept of tree decompositions, followed by the definition of a graph's treewidth according to Robertson and Seymour [26].

Definition 16. For a graph $G = (V, E)$, a pair (T, \mathcal{X}) consisting of a tree $T = (V(T), E(T))$ and a collection of vertex subsets (called *bags*) $\mathcal{X} = \{X_t \subseteq V \mid t \in V(T)\}$ associated to the nodes of T , is called a *tree decomposition* of G if it satisfies the following properties:

- 1) Every vertex of G is contained in at least one bag, i.e., $\bigcup_{t \in V(T)} X_t = V$.
- 2) For each edge $\{v, w\} \in E$ there exists a node $t \in V(T)$ such that $v, w \in X_t$.
- 3) For all vertices $t, \ell, u \in V(T)$ such that ℓ lies on the unique path between t and u in T , it holds that $X_t \cap X_u \subseteq X_\ell$.

The *width* of a tree decomposition of graph G is defined as the cardinality of its largest bag minus one, i.e.,

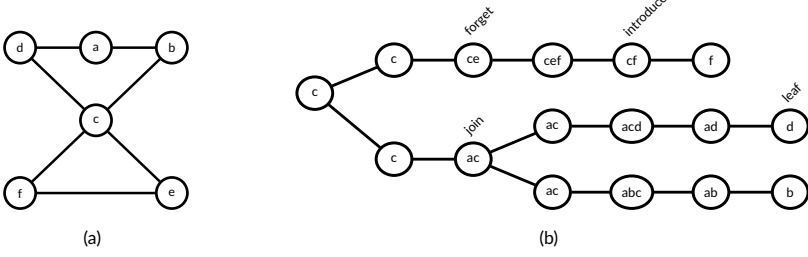


FIGURE 9 (a) Graph G with $\text{tw}(G) = 2$. (b) Nice tree decomposition (T, \mathcal{X}) of G .

$\text{tw}(G, (T, \mathcal{X})) := \max_{t \in V(T)} |X_t| - 1$. The *treewidth* of a graph G is now defined as the smallest width among all tree decompositions of G , i.e., $\text{tw}(G) := \min\{\text{tw}(G, (T, \mathcal{X})) \mid (T, \mathcal{X}) \text{ is a tree decomposition of } G\}$.

For better lucidity of dynamic programs, so-called *nice* tree decompositions were defined as a subclass of all tree decompositions [27].

Definition 17. A tree decomposition (T, \mathcal{X}) of a graph $G = (V, E)$ is called *nice* if T is a rooted tree and all nodes $t \in V(T)$ can be categorized into four groups:

- 1) *Leaves* $t \in V(T)$ have no child nodes and their bag contains exactly one vertex $v \in V$, i.e., $X_t = \{v\}$.
- 2) *Introduce nodes* $t \in V(T)$ have exactly one child node $\ell \in V(T)$ such that $X_\ell \subsetneq X_t$ and $X_t \setminus X_\ell = \{w\}$ for some $w \in V$.
- 3) *Forget nodes* $t \in V(T)$ have exactly one child node $\ell \in V(T)$ such that $X_t \subsetneq X_\ell$ and $X_\ell \setminus X_t = \{w\}$ for some $w \in V$.
- 4) *Join nodes* $t \in V(T)$ have exactly two child nodes $\ell, u \in V(T)$ such that $X_t = X_\ell = X_u$.

For any graph $G = (V, E)$ with bounded treewidth $\text{tw}(G) < W$, a nice tree decomposition (T, \mathcal{X}) with $O(|V|)$ nodes and $\text{tw}(G, (T, \mathcal{X})) < W$ can be computed in linear time [28, 29]. An exemplary graph with corresponding nice tree decomposition is visualized in Figure 9.

Our dynamic program for solving the Col-BM on graphs with bounded treewidth exploits the structure of nice tree decompositions and recursively computes label costs bottom-up. Let $G = (V, E)$ be a graph with bounded treewidth $\text{tw}(G) < W \in \mathbb{N}$, $E_1 \dot{\cup} \dots \dot{\cup} E_q = E$ be an edge coloring of G , and $\bar{c}: E \rightarrow \{1, \dots, q\}$ the corresponding color function. Further, let (T, \mathcal{X}) be a nice tree decomposition of G such that $\text{tw}(G, (T, \mathcal{X})) < W$. Without loss of generality, we assume that the bag X_r , corresponding to the root r of T , contains exactly one vertex. Should (T, \mathcal{X}) violate this assumption, we simply add a sequence of forget nodes to r and redefine T 's root.

For a tree node $t \in V(T)$ we denote the set of edges of G induced by its bag X_t with $E[X_t]$ and the subgraph of G induced by the vertices in the bags of the subtree of T rooted in t with G_t . As before, for a vertex $v \in V$ and a subset of edges $M \subseteq E$, we denote the set of colors in $\delta_M(v)$ by $\text{col}_M(v)$. Finally, for all mappings $f: A \rightarrow B$, we abbreviate $f_a := f(a)$ for $a \in A$ for ease of notation.

We use labels of the form

$$x = (m, F, \beta) \in \mathcal{L}^t := \{0, 1\}^{E[X_t]} \times \mathcal{P}(\{1, \dots, q\})^{X_t} \times \mathbb{N}_0^{X_t}$$

at the tree nodes $t \in V(T)$ to define an auxiliary variant of Col-BM on the subgraph G_t which we refer to as xCol-BM(t, x). To that end, the binary-valued mapping $m: E[X_t] \rightarrow \{0, 1\}$ prespecifies whether an edge $e \in E[X_t]$ is part of the b -matching in G_t or not. The mapping $F: X_t \rightarrow \mathcal{P}(\{1, \dots, q\})$ indicates for each vertex $v \in X_t$ the set of *unlocked* edge colors $F_v \subseteq \{1, \dots, q\}$. Only edges from $\delta_{G_t}(v)$ with unlocked colors may be chosen as part of a matching and all unlocked colors count towards the color degree of a vertex – even if they are unused. This gives rise to the definition of the x -(M -)color degree of $v \in V(G_t)$:

$$|\text{col}_M(x, v)| = \begin{cases} |F_v| & \text{if } v \in X_t, \\ |\text{col}_M(v)| & \text{else.} \end{cases}$$

Finally, the mapping $\beta: X_t \rightarrow \mathbb{N}_0$ defines the required degree of each vertex $v \in X_t$ with respect to matching edges in $E(G_t) \setminus E[X_t]$. We formalize the auxiliary problem xCol-BM(t, x) as follows:

$$\begin{aligned} \min_{M \subseteq E(G_t)} \quad & \max_{v \in V(G_t)} |\text{col}_M(x, v)| \\ \text{s.t.} \quad & |e \cap M| = m_e & \forall e \in E[X_t] \quad (3) \\ & |\delta_M(v)| = b(v) & \forall v \in V(G_t) \setminus X_t \quad (4) \\ & |\delta_M(v)| \leq b(v) & \forall v \in X_t \quad (5) \\ & |\delta_M(v) \setminus E[X_t]| = \beta_v & \forall v \in X_t \quad (6) \\ & \text{col}_M(v) \subseteq F_v & \forall v \in X_t \quad (7) \end{aligned}$$

Every b -matching in G_t satisfying the constraints (3) – (7) is called (t, x) -feasible. We define the cost $c^t(x)$ of label $x \in \mathcal{L}^t$ at tree node t as the optimal solution value to xCol-BM(t, x). If xCol-BM(t, x) is infeasible, we call x *invalid* and we set $c^t(x) = +\infty$. All remaining labels are called *valid* and we calculate their cost recursively. To that end, we consider the four types of nodes in the nice tree decomposition (T, X) of G starting with the initialization in leaves.

Lemma 18. Let $t \in V(T)$ be a leaf with $X_t = \{v\}$ for some $v \in V$. Then the cost of a valid label $x = (m, F, \beta) \in \mathcal{L}^t$ at t can be computed as

$$c^t(x) = |F_v|.$$

Proof As t is a leaf, $E[X_t] = \emptyset$ and G_t consists of the isolated vertex $v \in X_t$. All valid labels $x \in \mathcal{L}^t$ have the form $x = (m, F, \beta)$ with $F_v \subseteq \{1, \dots, q\}$, and $\beta_v = 0$. The only (t, x) -feasible matching in G_t is $M := \emptyset$ and thus, $c^t(x) = |\text{col}_M(x, v)| = |F_v|$. ■

For the three remaining types of tree nodes, label costs can be derived recursively from the label costs of child nodes. We begin by considering introduce nodes.

Lemma 19. Let $t \in V(T)$ be an introduce node with unique child node $\ell \in V(T)$, and let $w \in V$ be the introduced vertex, i.e., $X_t \setminus X_\ell = \{w\}$; see Figure 10(a). Given a valid label $x^\ell = (m^\ell, F^\ell, \beta^\ell) \in \mathcal{L}^\ell$, we define the label $x^t = (m^t, F^t, \beta^t) \in \mathcal{L}^t$ via $m_e^t := m_e^\ell$ for all $e \in E[X_\ell]$, $F_v^t := F_v^\ell$ for all $v \in X_\ell$, and $\beta_w^t := \beta_w^\ell$ for all $v \in X_\ell$. Then the cost of x^t at t can be computed as

$$c^t(x^t) = \max \left\{ c^\ell(x^\ell), |F_w^t| \right\}.$$

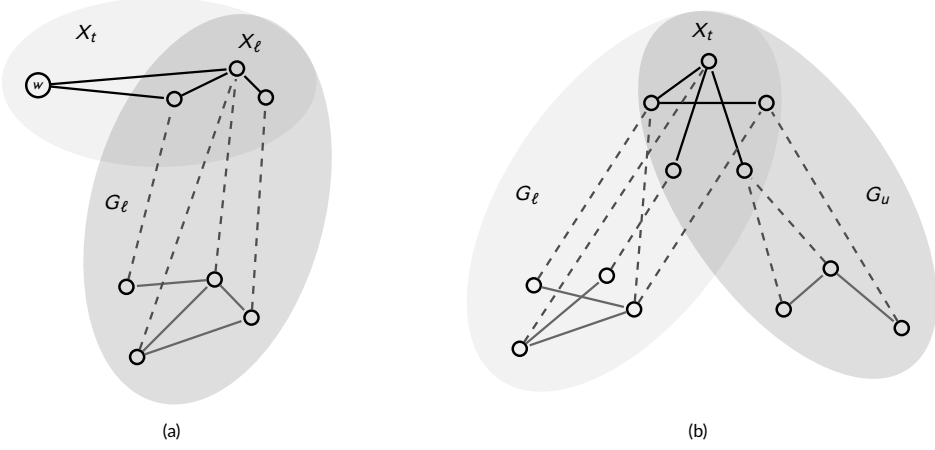


FIGURE 10 (a) Visualization of G_t for $t \in V(T)$ being an introduce node with child node ℓ . (b) Visualization of G_t for $t \in V(T)$ being a join node with child nodes ℓ, u .

Proof We begin by showing $c^t(x^t) \geq \max\{c^\ell(x^\ell), |F_w^t|\}$. Let M^t be an optimal solution to $\text{xCol-BM}(t, x^t)$. For the vertex $w \in V$ introduced by node $t \in V(T)$, it holds that $|\text{col}_{M^t}(x^t, w)| = |F_w^t|$ as $w \in X_t$. Hence,

$$c^t(x^t) = \max_{v \in V(G_t)} |\text{col}_{M^t}(x^t, v)| \geq |\text{col}_{M^t}(x^t, w)| = |F_w^t|. \quad (8)$$

Next, let $U := \delta_{G_t}(w) \subseteq E[X_t]$ be the set of edges introduced by $t \in V(T)$ and $M^\ell := M^t \setminus \{U\}$. We show that M^ℓ is an (ℓ, x^ℓ) -feasible matching in order to bound $c^\ell(x^\ell)$ from above.

By construction of M^ℓ , $e \cap M^\ell = e \cap M^t = m_e^t = m_e^\ell$ holds for all $e \in E[X_\ell]$ and thus equalities (3) are satisfied. Concerning equalities (4), it holds that $|\delta_{M^\ell}(v)| = |\delta_{M^t}(v)| = b(v)$ for $v \in V(G_\ell) \setminus X_\ell$. Finally, as $M^\ell \subseteq M^t$ and M^t is (t, x^t) -feasible, it follows that

$$\begin{aligned} |\delta_{M^\ell}(v)| &\leq |\delta_{M^t}(v)| \leq b(v) && \forall v \in X_\ell, \\ |\delta_{M^\ell}(v) \setminus E[X_\ell]| &= |\delta_{M^t}(v) \setminus E[X_t]| = \beta_v^t = \beta_v^\ell && \forall v \in X_\ell, \\ \text{col}_{M^\ell}(v) &\subseteq \text{col}_{M^t}(v) \subseteq F_v^t = F_v^\ell && \forall v \in X_\ell, \end{aligned}$$

and hence conditions (5) – (7) are satisfied. Therefore, M^ℓ is (ℓ, x^ℓ) -feasible.

Additionally, $|\text{col}_{M^\ell}(x^\ell, v)| = |\text{col}_{M^t}(x^t, v)|$ for all $v \in V(G_\ell)$ as $M^t \setminus E[X_t] = M^\ell \setminus E[X_\ell]$ and $F_v^t = F_v^\ell$ for all $v \in X_\ell$. As a result,

$$\begin{aligned} c^\ell(x^\ell) &\leq \max_{v \in V(G_\ell)} |\text{col}_{M^\ell}(x^\ell, v)| = \max_{v \in V(G_\ell)} |\text{col}_{M^t}(x^t, v)| \\ &\leq \max_{v \in V(G_t)} |\text{col}_{M^t}(x^t, v)| = c^t(x^t). \end{aligned} \quad (9)$$

By combining inequalities (8) and (9), we obtain $c^t(x^t) \geq \max\{c^\ell(x^\ell), |F_w^t|\}$.

Conversely, we show that $c^t(x^t) \leq \max\{c^\ell(x^\ell), |F_w^t|\}$. Let M^ℓ be an optimal solution to $\text{xCol-BM}(\ell, x^\ell)$. We define the matching $M^t := M^\ell \cup \{e \in U \mid m_e^\ell = 1\}$ and show that M^t is (t, x^t) -feasible in order to bound $c^t(x^t)$ from above.

For $e \in U$, equation (3) holds by definition. For $e \in E[X_t] \setminus U = E[X_\ell]$, equation (3) is satisfied as $e \cap M^t = e \cap M^\ell = m_e^\ell = m_e^t$ holds. Concerning equations (4), $|\delta_{M^t}(v)| = |\delta_{M^\ell}(v)| = b(v)$ holds for all $v \in V(G_t) \setminus X_t$. For the introduced vertex $w \in X_t$, constraints (6) and (7) hold by the validity of x^t . For $v \in X_t \setminus \{w\} = X_\ell$, equation (6) holds as

$$|\delta_{M^t}(v) \setminus E[X_t]| = |\delta_{M^\ell}(v) \setminus E[X_\ell]| = \beta_v^\ell = \beta_v^t.$$

By the validity of x^t , condition (7) holds for $v \in X_t \setminus \{w\}$ as

$$\text{col}_{M^t}(v) = \text{col}_{M^\ell}(v) \cup \text{col}_{M^t \setminus M^\ell}(v) \subseteq F_v^\ell \cup F_v^t = F_v^t.$$

Finally, equations (3) and (6) in combination with the validity of x^t imply that inequalities (5) hold for all $v \in V(G_t)$. Therefore, M^t is (t, x^t) -feasible.

The construction of M^t implies that $|\text{col}_{M^t}(x^t, v)| = |\text{col}_{M^\ell}(x^\ell, v)|$ for all $v \in V(G_\ell)$. Thus,

$$\begin{aligned} c^t(x^t) &\leq \max_{v \in V(G_t)} |\text{col}_{M^t}(x^t, v)| = \max \left\{ \max_{v \in V(G_\ell)} |\text{col}_{M^t}(x^t, v)|, |\text{col}_{M^t}(x^t, w)| \right\} \\ &= \max \left\{ \max_{v \in V(G_\ell)} |\text{col}_{M^\ell}(x^\ell, v)|, |F_w^t| \right\} = \max \left\{ c^\ell(x^\ell), |F_w^t| \right\}. \end{aligned}$$

We conclude $c^t(x^t) = \max \{c^\ell(x^\ell), |F_w^t|\}$. ■

Next, we consider the computation of label costs for forget nodes.

Lemma 20. Let $t \in V(T)$ be a forget node with unique child node $\ell \in V(T)$. Let $w \in V$ be the forgotten vertex, i.e., $\{w\} = X_\ell \setminus X_t$, and denote its incident edges with respect to $G[X_\ell]$ by $U := E[X_\ell] \setminus E[X_t] = \delta_{G_\ell}(w) \cap E[X_\ell]$. Given a valid label $x^t = (m^t, F^t, \beta^t) \in \mathcal{L}^t$, we define the set $\mathcal{L}^\ell(x^t) \subseteq \mathcal{L}^\ell$ of labels at ℓ via

$$\mathcal{L}^\ell(x^t) := \left\{ (m^\ell, F^\ell, \beta^\ell) \in \mathcal{L}^\ell \mid \begin{array}{l} m_e^\ell = m_e^t \quad \forall e \in E[X_t], \\ F_v^\ell = F_v^t \quad \forall v \in X_t, \\ \beta_v^\ell = \beta_v^t - \sum_{e \in \delta_{U}(v)} m_e^\ell \quad \forall v \in X_t, \\ \beta_w^\ell = b(w) - \sum_{e \in U} m_e^\ell \end{array} \right\}.$$

Then the cost of x^t at t can be computed as

$$c^t(x^t) = \min_{x^\ell \in \mathcal{L}^\ell(x^t)} c^\ell(x^\ell).$$

Proof We begin by showing $c^t(x^t) \geq \min_{x^\ell \in \mathcal{L}^\ell(x^t)} c^\ell(x^\ell)$. To that end, note that $G_t = G_\ell$ and let M^t be an optimal solution to $\text{xCol-BM}(t, x^t)$. We define a label $x^\ell = (m^\ell, F^\ell, \beta^\ell) \in \mathcal{L}^\ell$ as follows:

$$\begin{aligned} m_e^\ell &:= \begin{cases} m_e^t & \text{for } e \in E[X_t], \\ |e \cap M^t| & \text{for } e \in U, \end{cases} \\ F_v^\ell &:= \begin{cases} F_v^t & \text{for } v \in X_t, \\ \text{col}_{M^t}(v) & \text{for } v = w, \end{cases} \\ \beta_v^\ell &:= \begin{cases} \beta_v^t - \sum_{e \in \delta_{U}(v)} m_e^\ell & \text{for } v \in X_t, \\ b(v) - \sum_{e \in U} m_e^\ell & \text{for } v = w. \end{cases} \end{aligned}$$

By construction, $x^\ell \in \mathcal{L}^\ell(x^t)$. We show that M^t is (ℓ, x^ℓ) -feasible in order to bound $c^\ell(x^\ell)$ from above. By definition of x^ℓ , equations (3) are satisfied for all $e \in U$, whereas $|e \cap M^t| = m_e^t = m_e^\ell$ for $e \in E[X^t]$ since M^t is (t, x^t) -feasible. As $V(G_\ell) \setminus X_\ell \subseteq V(G_t) \setminus X_t$, constraints (4) and (5) hold by the x^t -feasibility of M^t . Concerning equalities (6), for any vertex $v \in X_t$

$$|\delta_{M^t}(v) \setminus E[X_\ell]| = |\delta_{M^t}(v) \setminus E[X_t]| - |\delta_{M^t}(v) \cap U| = \beta_v^t - \sum_{e \in \delta_U(v)} m_e^\ell = \beta_v^\ell,$$

whereas for the forgotten vertex w it holds that

$$|\delta_{M^t}(w) \setminus E[X_\ell]| = |\delta_{M^t}(w) \setminus (M^t \cap U)| = b(w) - \sum_{e \in U} m_e^\ell = \beta_w^\ell.$$

Finally, as M^t is (t, x^t) -feasible, constraints (7) are satisfied by our definition of x^ℓ and thus M^t is (ℓ, x^ℓ) -feasible.

For the forgotten vertex w , $F_w^\ell = \text{col}_{M^t}(w)$ by our definition of x^ℓ . Thus, we conclude that $|\text{col}_{M^t}(x^t, v)| = |\text{col}_{M^t}(x^\ell, v)|$ for all $v \in V(G_t)$ and it follows that

$$\begin{aligned} c^t(x^t) &= \max_{v \in V(G_t)} |\text{col}_{M^t}(x^t, v)| = \max_{v \in V(G_\ell)} |\text{col}_{M^t}(x^\ell, v)| \\ &\geq c^\ell(x^\ell) \geq \min_{\bar{x}^\ell \in \mathcal{L}^\ell(x^t)} c^\ell(\bar{x}^\ell). \end{aligned}$$

Conversely, for all labels $x^\ell \in \mathcal{L}^\ell(x^t)$, $\text{xCol-BM}(t, x^t)$ is a relaxation of $\text{xCol-BM}(\ell, x^\ell)$, and therefore $c^t(x^t) \leq \min_{x^\ell \in \mathcal{L}^\ell(x^t)} c^\ell(x^\ell)$.

We conclude $c^t(x^t) = \min_{x^\ell \in \mathcal{L}^\ell(x^t)} c^\ell(x^\ell)$. ■

To conclude the computation of label costs, we consider join nodes.

Lemma 21. Let $t \in V(T)$ be a join node with child nodes ℓ and u . Given a valid label $x^t = (m^t, F^t, \beta^t) \in \mathcal{L}^t$, we define the set $\mathcal{L}^{\ell, u}(x^t) \subseteq \mathcal{L}^\ell \times \mathcal{L}^u$ of pairs of labels at ℓ and u via

$$\begin{aligned} \mathcal{L}^{\ell, u}(x^t) := \{ & ((m^\ell, F^\ell, \beta^\ell), (m^u, F^u, \beta^u)) \in \mathcal{L}^\ell \times \mathcal{L}^u \mid m_e^\ell = m_e^u = m_e^t & \forall e \in E[X_t], \\ & F_v^\ell = F_v^u = F_v^t & \forall v \in X_t, \\ & \beta_v^\ell + \beta_v^u = \beta_v^t & \forall v \in X_t \}. \end{aligned}$$

Then the cost of x^t at t can be computed as

$$c^t(x^t) = \min_{(x^\ell, x^u) \in \mathcal{L}^{\ell, u}(x^t)} \max \{c^\ell(x^\ell), c^u(x^u)\}.$$

Proof Recall that for join nodes $X_t = X_\ell = X_u$ and $(V(G_\ell) \setminus X_\ell) \cap (V(G_u) \setminus X_u) = \emptyset$; see Figure 10(b).

We begin by showing $c^t(x^t) \geq \min_{(x^\ell, x^u) \in \mathcal{L}^{\ell, u}(x^t)} \max \{c^\ell(x^\ell), c^u(x^u)\}$. Let $M^t \subseteq E(G_t)$ be an optimal solution to $\text{xCol-BM}(t, x^t)$, and $M^\ell := M^t \cap E(G_\ell)$ and $M^u := M^t \cap E(G_u)$ the restrictions of M^t to the subgraphs G_ℓ and G_u , respectively. We define labels $x^\ell := (m^\ell, F^\ell, \beta^\ell) \in \mathcal{L}^\ell$ and $x^u := (m^u, F^u, \beta^u) \in \mathcal{L}^u$ such that $\beta_v^\ell = |\delta_{M^\ell}(v) \setminus E[X_t]|$ and $\beta_v^u = |\delta_{M^u}(v) \setminus E[X_t]|$ for all $v \in X_t$. The x^t -feasibility of M^t implies for all vertices $v \in X_t$ that

$$\beta_v^\ell + \beta_v^u = |\delta_{M^\ell}(v) \setminus E[X_t]| + |\delta_{M^u}(v) \setminus E[X_t]| = |\delta_{M^t}(v) \setminus E[X_t]| = \beta_v^t,$$

and consequently $(x^\ell, x^u) \in \mathcal{L}^{\ell, u}(x^t)$.

By construction, the matchings M^ℓ and M^u are feasible for $\text{xCol-BM}(\ell, x^\ell)$ and $\text{xCol-BM}(u, x^u)$, respectively. Moreover, as $F_v^\ell = F_v^u = F_v^t$ for all $v \in X_t$, it follows that $|\text{col}_{M^t}(x^t, v)| = |\text{col}_{M^\ell}(x^\ell, v)|$ for all $v \in V(G_\ell)$ and $|\text{col}_{M^t}(x^t, v)| = |\text{col}_{M^u}(x^u, v)|$ for all $v \in V(G_u)$. Hence,

$$\begin{aligned} c^t(x^t) &= \max_{v \in V(G_t)} |\text{col}_{M^t}(x^t, v)| \\ &= \max \left\{ \max_{v \in V(G_\ell)} |\text{col}_{M^t}(x^t, v)|, \max_{v \in V(G_u)} |\text{col}_{M^t}(x^t, v)| \right\} \\ &= \max \left\{ \max_{v \in V(G_\ell)} |\text{col}_{M^\ell}(x^\ell, v)|, \max_{v \in V(G_u)} |\text{col}_{M^u}(x^u, v)| \right\} \\ &\geq \max \left\{ c^\ell(x^\ell), c^u(x^u) \right\} \\ &\geq \min_{(\bar{x}^\ell, \bar{x}^u) \in \mathcal{L}^{\ell, u}(x^t)} \max \left\{ c^\ell(\bar{x}^\ell), c^u(\bar{x}^u) \right\}. \end{aligned}$$

Conversely, we show $c^t(x^t) \leq \min_{(x^\ell, x^u) \in \mathcal{L}^{\ell, u}(x^t)} \max \{c^\ell(x^\ell), c^u(x^u)\}$. To that end, consider a pair of labels $(x^\ell, x^u) \in \mathcal{L}^{\ell, u}(x^t)$, and let M^ℓ and M^u be optimal solutions to $\text{xCol-BM}(\ell, x^\ell)$ and $\text{xCol-BM}(u, x^u)$, respectively. We define the matching $M^t := M^\ell \cup M^u$ in G_t and show that M^t is (t, x^t) -feasible. Equations (3) hold, as $|e \cap M^t| = |e \cap M^\ell| = m_e^\ell = m_e^t$ for all $e \in E[X_t]$. For all $v \in V(G_\ell) \setminus X_t$ it holds that $|\delta_{M^t}(v)| = |\delta_{M^\ell}(v)| = b(v)$ and analogously $|\delta_{M^t}(v)| = |\delta_{M^u}(v)| = b(v)$ for all $v \in V(G_u) \setminus X_t$, proving that equations (4) hold. Concerning constraints (6), for every $v \in X_t$

$$|\delta_{M^t}(v) \setminus E[X_t]| = |\delta_{M^\ell}(v) \setminus E[X_t]| + |\delta_{M^u}(v) \setminus E[X_t]| = \beta_v^\ell + \beta_v^u = \beta_v^t$$

which, in combination with the validity of x^t , directly implies that inequalities (5) are satisfied. Finally, for all $v \in X_t$ it holds that $\text{col}_{M^t}(v) = \text{col}_{M^\ell}(v) \cup \text{col}_{M^u}(v) \subseteq F_v^\ell \cup F_v^u = F_v^t$ proving the validity of constraints (7). Therefore, M^t is (t, x^t) -feasible.

Additionally, $|\text{col}_{M^t}(x^t, v)| = |\text{col}_{M^\ell}(x^\ell, v)|$ for all $v \in V(G_\ell)$ and $|\text{col}_{M^t}(x^t, v)| = |\text{col}_{M^u}(x^u, v)|$ for all $v \in V(G_u)$ as $F_v^\ell = F_v^u = F_v^t$ for all $v \in X_t$. We thus conclude that for all $(x^\ell, x^u) \in \mathcal{L}^{\ell, u}(x^t)$

$$\begin{aligned} c^t(x^t) &\leq \max_{v \in V(G_t)} |\text{col}_{M^t}(x^t, v)| \\ &= \max \left\{ \max_{v \in V(G_\ell)} |\text{col}_{M^t}(x^t, v)|, \max_{v \in V(G_u)} |\text{col}_{M^t}(x^t, v)| \right\} \\ &= \max \left\{ \max_{v \in V(G_\ell)} |\text{col}_{M^\ell}(x^\ell, v)|, \max_{v \in V(G_u)} |\text{col}_{M^u}(x^u, v)| \right\} \\ &= \max \left\{ c^\ell(x^\ell), c^u(x^u) \right\} \end{aligned}$$

and therefore, it holds in particular that

$$c^t(x^t) \leq \min_{(x^\ell, x^u) \in \mathcal{L}^{\ell, u}(x^t)} \max \left\{ c^\ell(x^\ell), c^u(x^u) \right\}.$$

We conclude $c^t(x^t) = \min_{(x^\ell, x^u) \in \mathcal{L}^{\ell, u}(x^t)} \max \{c^\ell(x^\ell), c^u(x^u)\}$. ■

Finally, we show how the optimal solution value to Col-BM on G is obtained from the computed label costs.

Lemma 22. Let r be T 's root with $X_r = \{z\}$, and M^* a perfect b -matching in G of minimum color degree. We define the set $\mathcal{L}^* = \{(m, F, \beta) \in \mathcal{L}^r \mid \beta_z = b(z)\} \subseteq \mathcal{L}^r$ of valid labels at r . Then

$$f_G^{\max}(M^*) = \min_{x \in \mathcal{L}^*} c^r(x).$$

Proof First, we show that $f_G^{\max}(M^*) \geq \min_{x \in \mathcal{L}^*} c^r(x)$. To that end, consider the label $x^r = (m^r, F^r, \beta^r) \in \mathcal{L}^*$ with $F_z^r := |\text{col}_{M^*}(z)|$. Then M^* is by construction (r, x^r) -feasible and thus

$$\min_{x \in \mathcal{L}^*} c^r(x) \leq c^r(x^r) \leq \max_{v \in V(G_t)} |\text{col}_{M^*}(x^r, v)| = \max_{v \in V} |\text{col}_{M^*}(v)| = f_G^{\max}(M^*).$$

Conversely, we show $f_G^{\max}(M^*) \leq \min_{x \in \mathcal{L}^*} c^r(x)$. Let $x^r \in \mathcal{L}^*$ and M^r be an optimal solution to $\text{xCol-BM}(r, x^r)$. We note that equations (4) and (6) ensure that M^r is a perfect b -matching. Therefore it holds that

$$c^r(x^r) = \max_{v \in V(G_t)} |\text{col}_{M^r}(x^r, v)| \geq \max_{v \in V(G_t)} |\text{col}_{M^r}(v)| = f_G^{\max}(M^r) \geq f_G^{\max}(M^*).$$

As x^r was chosen arbitrarily from \mathcal{L}^* , in particular

$$\min_{x \in \mathcal{L}^*} c^r(x) \geq f_G^{\max}(M^*).$$

We conclude $f_G^{\max}(M^*) = \min_{x \in \mathcal{L}^*} c^r(x)$. ■

A perfect b -matching M^* in G of minimum color degree can be obtained by backtracking the chosen minima in the steps of the dynamic program. We can now formulate the main result of this section. For better lucidity, let $B := \max_{v \in V} b(v)$.

Theorem 23. *Col-BM on simple graphs $G = (V, E)$ with bounded treewidth $\text{tw}(G) < W$ is XP with respect to the number of colors q and the width bound W , and can be solved in $O(|V| \cdot 2^{W^2+W(q-1)} \cdot B^W \cdot \max\{2^{W+q}, B^W\})$ time.*

Proof The correctness of the dynamic program follows from Lemma 22 and the label cost computations in Lemmata 18, 19, 20, and 21.

Concerning the algorithm's runtime, recall that a nice tree decomposition (T, \mathcal{X}) of G with $|V(T)| \in O(|V|)$ nodes can be computed in linear time. For each $t \in V(T)$, the number of labels $|\mathcal{L}^t|$ we have to consider at t is in

$$O\left(2^{|E|X_t|} \cdot 2^{q|X_t|} \cdot B^{|X_t|}\right) \subseteq O\left(2^{W^2-W} \cdot 2^{qW} \cdot B^W\right).$$

The computation of label costs for leaves and introduce nodes can be done in $O(1)$ time.

For labels $x \in \mathcal{L}^t$ at forget node $t \in V(T)$ with child node ℓ , we have to compare the label costs of $|\mathcal{L}^\ell(x)| = 2^{|U|} 2^q$ labels. For simple graphs, $|U| \leq |W|$ and thus, the computation of label costs for forget nodes is in $O(2^W 2^q)$ time.

For labels $x = (m, F, \beta) \in \mathcal{L}^t$ at join node $t \in V(T)$ with child nodes ℓ and u , $|\mathcal{L}^{\ell,u}(x)| = \prod_{v \in X_t} (\beta_v + 1) \leq (B+1)^{|X_t|}$. Consequently, the computation of label costs for join nodes is in $O(B^W)$ time.

In conclusion the computation of label costs can be performed in $O(\max\{2^{W+q}, B^W\})$ time. This results in a total runtime in

$$O(|V| \cdot 2^{W^2+W(q-1)} \cdot B^W \cdot \max\{2^{W+q}, B^W\}).$$

■

Corollary 24. *Col-BM on simple graphs $G = (V, E)$ with bounded treewidth $\text{tw}(G) < W$ is FPT with respect to the number of colors q , the width bound W , and the maximum b -value B .*

We note that for all Col-BM instances $B \leq |E|$ and thus, for fixed q and W our dynamic program runs in polynomial time ($O(|V| \cdot B^{2W})$ time). For trees, which are simple graphs with treewidth 1, the runtime obtained from Theorem 23 coincides with the one from Corollary 15.

As soon as we drop the width bound W , we obtain Col-BM on general graphs with a fixed number of colors which is strongly- \mathcal{NP} hard by Theorem 2, even for $B = 2$. The complexity of Col-BM on simple graphs $G = (V, E)$ with bounded treewidth $\text{tw}(G) < W$ and an arbitrary number of colors q remains open.

6 | CONCLUSIONS

In this paper we introduce the minimum color-degree perfect b -matching problem and prove its strong \mathcal{NP} -hardness as well as its $(2 - \varepsilon)$ -inapproximability on bipartite graphs with two colors. However, we identify a class of two-colored complete bipartite graphs on which we can solve Col-BM in quadratic time and propose polynomial-time dynamic programs solving Col-BM with a fixed number of colors on series-parallel graphs and simple graphs with bounded treewidth.

Future work includes generalizing the results for complete bipartite graphs to more colors as well as to more general b -values. Moreover, we will investigate the complexity of Col-BM on series-parallel graphs and graphs of bounded treewidth when the number of colors is not fixed. Furthermore, we plan to examine how special structures in the edge coloring can be exploited. Finally, we intend to devise general exact algorithms and heuristics for Col-BM by exploitation of structures in the underlying polytope.

REFERENCES

- [1] Schrijver A. Combinatorial Optimization: Polyhedra and Efficiency, vol. 24 of Algorithms and Combinatorics. Berlin, Heidelberg, New York: Springer; 2003.
- [2] Kuhn HW. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly* 1955;2(1-2):83–97.
- [3] Tanimoto SL, Itai A, Rodeh M. Some Matching Problems for Bipartite Graphs. *Journal of the ACM* 1978 Oct;25(4):517–525.
- [4] Worldwide Slot Guidelines. 9 ed. International Air Transport Association; 2019.
- [5] Fujita S, Magnant C. Properly colored paths and cycles. *Discrete Applied Mathematics* 2011;159(14):1391 – 1397.
- [6] Garey MR, Johnson DS. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. New York, USA: W. H. Freeman & Co.; 1979.
- [7] Rusu I. Maximum weight edge-constrained matchings. *Discrete Applied Mathematics* 2008;156(5):662 – 672.
- [8] Le VB, Pfender F. Complexity results for rainbow matchings. *Theoretical Computer Science* 2014;524:27 – 33.
- [9] Nomikos C, Pagourtzis A, Zachos S. Randomized and Approximation Algorithms for Blue-Red Matching. In: Kučera L, Kučera A, editors. *Mathematical Foundations of Computer Science 2007* Berlin, Heidelberg: Springer; 2007. p. 715–725.
- [10] Mastrolilli M, Stamoulis G. Bi-criteria and approximation algorithms for restricted matchings. *Theoretical Computer Science* 2014;540-541:115 – 132. *Combinatorial Optimization: Theory of algorithms and complexity*.

-
- [11] Mastrolilli M, Stamoulis G. Constrained Matching Problems in Bipartite Graphs. In: Mahjoub AR, Markakis V, Milis I, Paschos VT, editors. *Combinatorial Optimization* Berlin, Heidelberg: Springer; 2012. p. 344–355.
- [12] Büsing C, Comis M. Budgeted Colored Matching Problems. *Electronic Notes in Discrete Mathematics* 2018;64:245 – 254. 8th International Network Optimization Conference - INOC 2017.
- [13] Monnot J. The labeled perfect matching in bipartite graphs. *Information Processing Letters* 2005;96(3):81 – 88.
- [14] Carrabs F, Cerulli R, Gentili M. The labeled maximum matching problem. *Computers & Operations Research* 2009;36(6):1859 – 1871.
- [15] Wirth HC, Steffan J. Reload cost problems: minimum diameter spanning tree. *Discrete Applied Mathematics* 2001;113(1):73 – 85. <http://www.sciencedirect.com/science/article/pii/S0166218X00003929>, selected Papers: 12th Workshop on Graph-Theoretic Concepts in Computer Science.
- [16] Gourvès L, Lyra A, Martinhon C, Monnot J. The Minimum Reload s-t Path/Trail/Walk Problems. In: Nielsen M, Kučera A, Miltersen PB, Palamidessi C, Tůma P, Valencia F, editors. *SOFSEM 2009: Theory and Practice of Computer Science* Berlin, Heidelberg: Springer Berlin Heidelberg; 2009. p. 621–632.
- [17] Amaldi E, Galbiati G, Maffioli F. On minimum reload cost paths, tours, and flows. *Networks* 2011;57(3):254–260. <https://onlinelibrary.wiley.com/doi/abs/10.1002/net.20423>.
- [18] Baste J, Gözüpek D, Shalom M, Thilikos DM. Minimum Reload Cost Graph Factors. In: Catania B, Královíč R, Nawrocki J, Pighizzini G, editors. *SOFSEM 2019: Theory and Practice of Computer Science* Cham: Springer International Publishing; 2019. p. 67–80.
- [19] Ahmed F, Dickerson JP, Fuge M. Diverse Weighted Bipartite b -Matching. In: *Proceedings of IJCAI-17*; 2017. p. 35–41.
- [20] Berman P, Karpinski M, Scott AD. Approximation Hardness of Short Symmetric Instances of MAX-3SAT. *Electronic Colloquium on Computational Complexity (ECCC)* 2003;.
- [21] Monnot J, Toulouse S. The path partition problem and related problems in bipartite graphs. *Operations Research Letters* 2007;35(5):677 – 684. <http://www.sciencedirect.com/science/article/pii/S0167637707000028>.
- [22] Kano M, Li X. Monochromatic and Heterochromatic Subgraphs in Edge-Colored Graphs - A Survey. *Graphs and Combinatorics* 2008 Sep;24(4):237–263. <https://doi.org/10.1007/s00373-008-0789-5>.
- [23] Kikuno T, Yoshida N, Kakuda Y. A linear algorithm for the domination number of a series-parallel graph. *Discrete Applied Mathematics* 1983;5(3):299–311.
- [24] de Fluiter B, Bodlaender H. Parallel algorithms for series parallel graphs. PhD thesis, University of Utrecht; 1997.
- [25] Büsing C, Comis M. Multi-budgeted matching problems. *Networks* 2018;72(1):25–41.
- [26] Robertson N, Seymour PD. Graph minors. II. Algorithmic aspects of tree-width. *Journal of Algorithms* 1986;7(3):309 – 322.
- [27] Bodlaender HL, Koster AMCA. *Combinatorial Optimization on Graphs of Bounded Treewidth*. *The Computer Journal* 2008 May;51(3):255–269.
- [28] Bodlaender HL. A Linear-Time Algorithm for Finding Tree-Decompositions of Small Treewidth. *SIAM Journal on Computing* 1996;25(6):1305–1317.
- [29] Kloks T. *Treewidth, Computations and Approximations*, vol. 842 of *Lecture Notes in Computer Science*. Berlin, Heidelberg: Springer; 1994.