

Minimum Sum Coloring with Bundles in Trees and Bipartite Graphs

Takehiro Ito 

Graduate School of Information Sciences, Tohoku University, Sendai, Japan

Naonori Kakimura 

Faculty of Science and Technology, Keio University, Yokohama, Japan

Naoyuki Kamiyama 

Institute of Mathematics for Industry, Kyushu University, Fukuoka, Japan

Yusuke Kobayashi 

Research Institute for Mathematical Sciences, Kyoto University, Japan

Yoshio Okamoto 

Graduate School of Informatics and Engineering,
The University of Electro-Communications, Tokyo, Japan

Abstract

The minimum sum coloring problem with bundles was introduced by Darboui and Friggstad (SWAT 2024) as a common generalization of the minimum coloring problem and the minimum sum coloring problem. During their presentation, the following open problem was raised: whether the minimum sum coloring problem with bundles could be solved in polynomial time for trees. We answer their question in the negative by proving that the minimum sum coloring problem with bundles is NP-hard even for paths. We complement this hardness by providing algorithms of the following types. First, we provide a fixed-parameter algorithm for trees when the number of bundles is a parameter; this can be extended to graphs of bounded treewidth. Second, we provide a polynomial-time algorithm for trees when bundles form a partition of the vertex set and the difference between the number of vertices and the number of bundles is constant. Third, we provide a polynomial-time algorithm for trees when bundles form a partition of the vertex set and each bundle induces a connected subgraph. We further show that for bipartite graphs, the problem with weights is NP-hard even when the number of bundles is at least three, but is polynomial-time solvable when the number of bundles is at most two. The threshold shifts to three versus four for the problem without weights.

2012 ACM Subject Classification Mathematics of computing → Graph algorithms; Theory of computation → Problems, reductions and completeness

Keywords and phrases graph algorithms, minimum sum coloring, minimum coloring, fixed-parameter tractability, NP-hardness

Digital Object Identifier 10.4230/LIPIcs.ISAAC.2025.40

Related Version *Full Version:* <https://doi.org/10.48550/arXiv.2509.15080>

Funding *Takehiro Ito:* JSPS KAKENHI Grant Numbers JP24H00686, JP24H00690.

Naonori Kakimura: JSPS KAKENHI Grant Numbers JP21H03397, JP23K21646, JP22H05001, and JST ERATO Grant Number JPMJER2301, Japan.

Naoyuki Kamiyama: JSPS KAKENHI Grant Number JP24K14825.

Yusuke Kobayashi: JSPS KAKENHI Grant Numbers JP22H05001, JP24K02901.

Yoshio Okamoto: JSPS KAKENHI Grant Number JP23K10982.

Acknowledgements We are grateful to anonymous reviewers for their helpful comments.



© Takehiro Ito, Naonori Kakimura, Naoyuki Kamiyama, Yusuke Kobayashi, and Yoshio Okamoto; licensed under Creative Commons License CC-BY 4.0

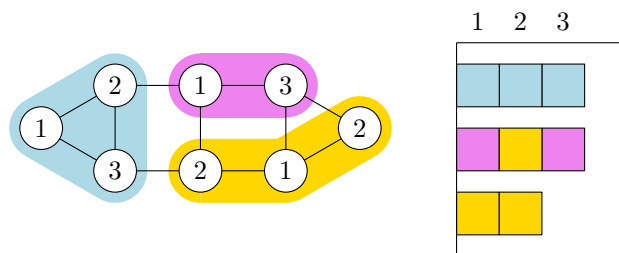
36th International Symposium on Algorithms and Computation (ISAAC 2025).

Editors: Ho-Lin Chen, Wing-Kai Hon, and Meng-Tsung Tsai; Article No. 40; pp. 40:1–40:14

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany



■ **Figure 1** (Left) An instance of the minimum sum coloring problem with bundles. Bundles are distinguished by color. In this example, bundles are disjoint, but they do not have to be disjoint in general. Numbers on vertices represent the colors assigned to them. The maximum color is 3 for the violet bundle, 3 for the blue bundle, and 2 for the orange bundle; their sum is 8. (Right) Scheduling with conflicts. This is a Gantt chart, where each unit-time job corresponds to a square and the numbers correspond to time slots. Colors show bundles. The graph on the left is a conflict graph over the jobs, and the minimum sum coloring problem with bundles corresponds to minimizing the sum of makespans over bundles.

1 Introduction

The minimum sum coloring problem with bundles was introduced by Darbouy and Friggstad [10] as a common generalization of the minimum coloring problem and the minimum sum coloring problem. For the minimum sum coloring problem with bundles, we are given an undirected graph G and a family of vertex subsets of G , called bundles. Then, we want to find a (proper) coloring of G by positive integers such that the sum of the maximum colors in bundles is minimized (a precise definition will be given in the next section). Figure 1 (Left) shows an example. The minimum coloring problem corresponds to the case where the vertex set itself is a unique bundle; the minimum sum coloring problem corresponds to the case where each vertex forms a singleton bundle.

The minimum sum coloring problem with bundles can be seen as a combinatorial model for the following scheduling problem. We want to process n jobs with unit processing time. Among them, some pairs of jobs cannot be processed concurrently due to conflicts such as competing resources or precedence constraints. Those conflicts are modeled by an undirected graph in which the vertices represent given jobs, and two vertices are joined by an edge if and only if the corresponding jobs are in conflict. Furthermore, there are ℓ agents, and each of them is interested in the completion of some set of jobs. Each of these sets of jobs is modeled as a bundle, and the agents want to minimize the completion time of the last job in each of their bundles. We assume that there are sufficiently many machines with identical power. Then, an optimal solution to the minimum sum coloring problem with bundles corresponds to a scheduling of jobs that minimizes the total (or average) completion time of the last jobs in the bundles. Figure 1 (Right) shows the correspondence. With this application in mind, it is natural to assume that the bundles form a partition of the vertex set. However, some of our results can be applied even when the bundles do not form a partition.

The minimum sum coloring problem with bundles is a computationally hard problem. This is expected since the minimum coloring problem is already NP-hard [16]. However, it is known that the minimum coloring problem and the minimum sum coloring problem can be solved in linear time for trees [17]. Darbouy and Friggstad [10], in their presentation at SWAT 2024, asked whether the minimum sum coloring problem with bundles can be solved in polynomial time for trees.

The first main contribution of this paper is to prove that the minimum sum coloring problem with bundles is NP-hard even when the graph is a path and bundles form a partition of the vertex set. This answers the open question by Darbouy and Friggstad [10] mentioned above since paths are trees. On the other hand, we already know from the literature that, if the bundles form a partition, the problem is linear-time solvable when the number of bundles is one (corresponding to the minimum coloring problem) or when it is equal to the number of vertices (corresponding to the minimum sum coloring problem). We also observe that in our reduction, the subgraph induced by a bundle can be disconnected. Therefore, we wonder what happens if the number of bundles is small or large, or if each bundle induces a connected subgraph.

The second contribution of this paper is to provide a fixed-parameter algorithm for the minimum sum coloring problem with bundles in trees when the number of bundles is a parameter. The algorithm also works when each bundle is associated with a non-negative weight that acts as a coefficient in the sum. The algorithm is based on a combinatorial lemma that states that the number of colors in an optimal coloring is bounded by the chromatic number of the graph multiplied by the number of bundles, which holds for general graphs.

The third contribution is to provide a polynomial-time algorithm for the minimum sum coloring problem with bundles in trees when bundles form a partition and the difference between the number of vertices and the number of bundles is small. It should be noted that this is not a fixed-parameter algorithm with respect to that difference.

The fourth contribution is to provide a polynomial-time algorithm for the minimum sum coloring problem with bundles in trees when the family of bundles forms a partition of the vertex set and each bundle induces a connected subgraph. Note that this algorithm does not require the number of bundles to be bounded. We also show that the minimum sum coloring problem with bundles in paths can be solved in polynomial time when each bundle induces a connected subgraph (even if the bundle family does not necessarily form a partition).

Next, we turn our attention to bipartite graphs. Since for bipartite graphs, the minimum coloring (i.e., when the number of bundles is one) is easy and the minimum sum coloring (i.e., when the number of bundles is equal to the number of vertices) is hard [3], we investigate how the number of bundles affects the computational complexity of the minimum sum coloring problem with bundles.

As the fifth contribution, we prove that for bipartite graphs, the minimum sum coloring problem with bundles can be solved in polynomial time when the number of bundles is at most three, and it is NP-hard when the number of bundles is at least four. For the weighted case, the problem can be solved in polynomial time when the number of bundles is at most two, and it is NP-hard when the number of bundles is at least three.

Our results are summarized in Table 1.

Related work

As mentioned in the introduction, the minimum sum coloring problem with bundles was introduced by Darbouy and Friggstad [10] as a common generalization of the minimum coloring problem and the minimum sum coloring problem.

The minimum coloring problem has been one of the central topics in graph theory, graph algorithms, combinatorial optimization, and computational complexity theory. It is known that the minimum coloring problem is NP-complete [16],¹ even for 4-regular planar graphs [9].

¹ Whenever we talk about NP-completeness of a minimization problem, we consider a canonical decision

■ **Table 1** Summary of the results in this paper. Here, n is the number of vertices.

	# bundles	Type of bundles	Weights	Results	Reference
Trees	Unbounded	Independent partition	Uniform	NP-complete	Theorem 3
	Parameter	General	General	FPT	Theorem 5
	n – Parameter	Partition	General	XP	Theorem 7
	Unbounded	Connected partition	General	P	Theorem 8
Bipartite	≤ 3	General	Uniform	P	Theorem 13
	≥ 4	General	Uniform	NP-complete	Theorem 10
	≤ 2	General	General	P	Theorem 12
	≥ 3	General	General	NP-complete	Theorem 11

For several classes of graphs, the minimum coloring problem can be solved in polynomial time. Most notably, for perfect graphs, a polynomial-time algorithm is known that is based on semidefinite programming relaxation [13]. The class of perfect graphs contains bipartite graphs, interval graphs, and chordal graphs, for which the minimum coloring problem can be solved in linear time.

The minimum sum coloring problem was introduced by Kubicka and Schwenk [17], where the NP-completeness of the problem was established. The problem is also known to be NP-complete even for bipartite graphs [3] and interval graphs [19, 18] (and hence for chordal graphs). We recommend a survey by Halldórsson and Kortsarz [14] on approximability for the minimum sum coloring.

The minimum coloring problem and the minimum sum coloring problem are closely related to scheduling with conflict or mutual exclusion scheduling. In a basic version of scheduling with conflict, we are given a set of n jobs with unit processing time, a set of m machines, and a conflict graph G on the set of jobs. Then, we need to assign each job to one of the machines and process the jobs along the timeline in such a way that two jobs are not processed at the same time slot if they are adjacent in G . The objective is to minimize either the makespan or the total completion time. The minimum coloring problem corresponds to the case where $m \geq n$ and the objective is makespan minimization; the minimum sum coloring problem corresponds to the case where $m \geq n$ and the objective is total completion time minimization. Baker and Coffman Jr. [2] proved that the problem of scheduling with conflict can be solved in polynomial time when $m = 2$ and is NP-hard when $m \geq 3$. For more recent and related results, see a paper by Even et al. [11]. We note that there are versions of scheduling with conflicts where we cannot process two jobs in conflict on the same machine [5].

2 Preliminaries

For a positive integer $k \in \mathbb{Z}_{>0}$, we use the notation $[k] = \{1, 2, \dots, k\}$.

All graphs in this paper are finite, undirected, and simple (i.e., having no self-loops or multiple edges). A graph $G = (V, E)$ is *bipartite* if the vertex set V can be partitioned into two disjoint sets A and B in such a way that every edge $e \in E$ has one endpoint in A and the other endpoint in B . A graph G is a *tree* if it is connected and has no cycles. For $v \in V$, let $\delta_G(v)$ denote the set of all edges incident to v .

version of the problem in which we are also given an upper bound C of the optimal value, and want to decide whether the optimal value is at most C .

Let $G = (V, E)$ be a graph. A (proper) *coloring* of G is a map $c: V \rightarrow \mathbb{Z}_{>0}$ such that $c(u) \neq c(v)$ for every edge $uv \in E$. For a vertex subset $S \subseteq V$, an S -*partial coloring* of G is a map $c: V \rightarrow \mathbb{Z}_{\geq 0}$ such that $c(v) > 0$ for $v \in S$, $c(v) = 0$ for $v \in V \setminus S$, and $c(u) \neq c(v)$ for every edge $uv \in E$ with $u, v \in S$.

The problem MINIMUM SUM COLORING WITH BUNDLES is defined as follows.

Problem: MINIMUM SUM COLORING WITH BUNDLES

Input: An undirected graph $G = (V, E)$, a family $\mathcal{B} = \{B_1, B_2, \dots, B_\ell\}$ of non-empty subsets of V , a weight function $w: \mathcal{B} \rightarrow \mathbb{Z}_{>0}$, and a positive integer $C \in \mathbb{Z}_{>0}$.

Question: Determine whether there exists a (proper) coloring $c: V \rightarrow \mathbb{Z}_{>0}$ such that

$$\text{cost}(c) := \sum_{j=1}^{\ell} w(B_j) \max\{c(v) \mid v \in B_j\} \leq C.$$

For an instance $I = (G, \mathcal{B}, w, C)$ of MINIMUM SUM COLORING WITH BUNDLES, we call a coloring c of G *optimal* if $\text{cost}(c) \leq \text{cost}(c')$ for all colorings c' of G .

Each member of $\mathcal{B}$ is called a *bundle*. Throughout the paper, we assume that each vertex $v \in V$ is contained in at least one bundle; otherwise, we can remove v from the instance. The family $\mathcal{B}$ is a *partition* if for each vertex $v \in V$ there exists a unique bundle B_j such that $v \in B_j$. A bundle B is *connected* if the induced subgraph $G[B]$ is connected. A bundle B is *independent* if the induced subgraph $G[B]$ has no edges. The family $\mathcal{B}$ is called *connected* (and *independent*) if all its members are connected (and independent, respectively).

The following observation will be used later explicitly or implicitly.

► **Observation 1.** For any graph $G = (V, E)$ in MINIMUM SUM COLORING WITH BUNDLES, there exists an optimal coloring in which the color of every vertex v is at most $|\delta_G(v)| + 1$.

Proof. If the color of v exceeds $|\delta_G(v)| + 1$, then one of the colors in $\{1, 2, \dots, |\delta_G(v)| + 1\}$ is missing in the neighborhood of v , and v can be recolored with that missing color without increasing the cost. ◀

3 Trees

3.1 NP-Completeness for Paths

As an intermediate step, we first prove that MINIMUM SUM COLORING WITH BUNDLES is NP-complete for perfect matchings. Here, a *perfect matching* is a set of edges in which no pair of edges shares a common endpoint.

► **Theorem 2.** MINIMUM SUM COLORING WITH BUNDLES is NP-complete even when the edge set of the input graph forms a perfect matching, $\mathcal{B}$ is an independent partition of the vertex set, and $w(B) = 1$ for every $B \in \mathcal{B}$.

Proof. It is easy to see that MINIMUM SUM COLORING WITH BUNDLES is in NP. To prove NP-hardness, we reduce INDEPENDENT SET to MINIMUM SUM COLORING WITH BUNDLES. Here, for an undirected graph $G = (V, E)$, we say that $S \subseteq V$ is *independent* if G has no edge connecting the vertices in S .

Problem: INDEPENDENT SET

Input: An undirected graph $G = (V, E)$ and a positive integer $k \in \mathbb{Z}_{>0}$.

Question: Determine whether there exists an independent set $S \subseteq V$ with $|S| \geq k$.

It is well-known that INDEPENDENT SET is NP-complete (see [16]).

Suppose that we are given an instance of INDEPENDENT SET that consists of a graph $G = (V, E)$ and a positive integer k . We construct a new graph $G' = (V', E')$ by replacing each vertex $v \in V$ with $|\delta_G(v)|$ new vertices so that each new vertex has exactly one incident edge. Formally, G' is defined as follows:

$$V' = \{p_{(v,e)} \mid v \in V, e \in \delta_G(v)\}, \quad E' = \{p_{(u,e)}p_{(v,e)} \mid e = uv \in E\}.$$

Let $B_v := \{p_{(v,e)} \mid e \in \delta_G(v)\}$ for each $v \in V$, and define $\mathcal{B} = \{B_v \mid v \in V\}$. Then, it is easy to see that E' forms a perfect matching in G' and $\mathcal{B}$ is a partition of V' . Let $w(B) = 1$ for any $B \in \mathcal{B}$, and let $C = 2|V| - k$. This defines an instance $(G', \mathcal{B}, w, C)$ of MINIMUM SUM COLORING WITH BUNDLES. See Figure 2.

To show the validity of this reduction, it suffices to show that G has an independent set of size at least k if and only if there exists a coloring c of G' such that $\sum_{v \in V} \max\{c(p) \mid p \in B_v\} \leq C$.

We first show the sufficiency (“if” part). Suppose that G' has a coloring $c: V' \rightarrow \mathbb{Z}_{>0}$ such that $\text{cost}(c) \leq C = 2|V| - k$. Let $S := \{v \in V \mid \max\{c(p) \mid p \in B_v\} = 1\}$. Then, since

$$C \geq \text{cost}(c) = \sum_{v \in V} \max\{c(p) \mid p \in B_v\} \geq 2(|V| - |S|) + |S|,$$

we obtain $|S| \geq k$. For any edge $e = uv \in E$, since $c(p_{(u,e)}) \neq c(p_{(v,e)})$ as c is a coloring, at least one of $c(p_{(u,e)}) \geq 2$ and $c(p_{(v,e)}) \geq 2$ holds, which implies that at least one of u and v is in $V \setminus S$. Therefore, S is an independent set in G , which shows the sufficiency.

We next show the necessity (“only if” part). Suppose that G has an independent set $S \subseteq V$ with $|S| \geq k$. Let $c: V' \rightarrow \mathbb{Z}_{>0}$ be a coloring of G' such that

- for each edge $e = uv \in E$, one of $c(p_{(u,e)})$ and $c(p_{(v,e)})$ is 1 and the other is 2, and
- for any $v \in S$ and any $e \in \delta_G(v)$, it holds that $c(p_{(v,e)}) = 1$.

Note that such a coloring c exists, because S is an independent set in G . Since $\max\{c(p) \mid p \in B_v\} = 1$ for $v \in S$ and $\max\{c(p) \mid p \in B_v\} \leq 2$ for $v \in V \setminus S$, we obtain

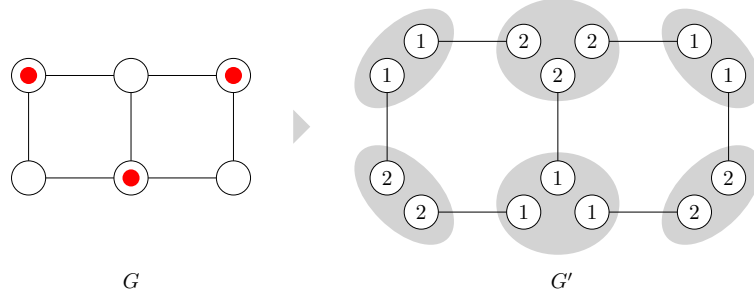
$$\text{cost}(c) = \sum_{v \in V} \max\{c(p) \mid p \in B_v\} \leq 2|V| - |S| \leq C,$$

which shows the necessity.

Therefore, the reduction is valid, and hence MINIMUM SUM COLORING WITH BUNDLES is NP-hard even when the edge set forms a perfect matching, $\mathcal{B}$ is an independent partition of the vertex set, and $w(B) = 1$ for every $B \in \mathcal{B}$. ◀

We then prove the NP-hardness of MINIMUM SUM COLORING WITH BUNDLES on paths by reducing MINIMUM SUM COLORING WITH BUNDLES with the conditions in Theorem 2. Roughly speaking, the reduction involves creating four copies of the instance described in Theorem 2 and carefully connecting their edges to form a single path. See the full version for details.

► **Theorem 3.** MINIMUM SUM COLORING WITH BUNDLES is NP-complete even when G is a path, $\mathcal{B}$ is an independent partition of the vertex set, and $w(B) = 1$ for every $B \in \mathcal{B}$.



■ **Figure 2** Reduction when a graph forms a perfect matching.

3.2 Fixed-Parameter Tractability when $|\mathcal{B}|$ is a Parameter

In the reduction of the previous section, the number of bundles is unbounded. As a positive algorithmic result that is in contrast to Theorem 3, we prove that MINIMUM SUM COLORING WITH BUNDLES in graphs of bounded treewidth can be solved in fixed-parameter tractable time when the number of bundles and the treewidth are parameters. See a textbook (e.g. [8]) for an introduction to treewidths and Courcelle's theorem.

First, we prove the lemma that bounds the number of colors in any optimal coloring. For a graph $G = (V, E)$, we denote by $\chi(G)$ the *chromatic number* of G , which is defined as the minimum of $\max_{v \in V} c(v)$ over all (proper) colorings c of G .

► **Lemma 4.** *Let $I = (G = (V, E), \mathcal{B}, w, C)$ be an instance of MINIMUM SUM COLORING WITH BUNDLES such that each vertex is contained in some bundle. Then, every optimal coloring $c: V \rightarrow \mathbb{Z}_{>0}$ for I satisfies $c(v) \leq \chi(G)|\mathcal{B}|$ for all $v \in V$.*

Proof. The proof proceeds by induction on $|\mathcal{B}|$. Consider the case where $|\mathcal{B}| = 1$ and $\mathcal{B} = \{B\}$. Note that $B = V$ as each vertex is contained in some bundle. Let c be a minimum coloring, i.e., a coloring that attains $\chi(G)$. Then, $\max_{v \in B} c(v) \leq \max_{v \in V} c(v) = \chi(G) = \chi(G)|\mathcal{B}|$.

Now, let $\ell \geq 1$ be a positive integer, and assume that the lemma holds when the number of bundles is ℓ . Then, we prove that the lemma holds when the number of bundles is $\ell + 1$. For the sake of contradiction, suppose that there exists an optimal coloring c^* such that $\max_{v \in B} c^*(v) > \chi(G)(\ell + 1)$ for some $B \in \mathcal{B}$. Fix such a member B , and let U be the set of vertices of G that belong to B , but not to other bundles. Then, consider the instance $I' = (G - U, \mathcal{B} \setminus \{B\}, w|_{V \setminus U}, C)$, where $w|_{V \setminus U}$ is the restriction of w on $V \setminus U$. By the induction hypothesis, there exists an optimal coloring $c': V \setminus U \rightarrow \mathbb{Z}_{>0}$ for I' such that $\max_{v \in V \setminus U} c'(v) \leq \chi(G - U)\ell \leq \chi(G)\ell$. We now extend c' to a coloring c of G by giving colors in $\{\chi(G)\ell + 1, \dots, \chi(G)\ell + \chi(G)\}$ to the vertices of U . This is possible since $\chi(G[U]) \leq \chi(G)$. Then,

$$\begin{aligned} \text{cost}(c) &= \sum_{B' \in \mathcal{B} \setminus \{B\}} w(B') \max_{v \in B'} c(v) + w(B) \max_{v \in B} c(v) \\ &\leq \sum_{B' \in \mathcal{B} \setminus \{B\}} w(B') \max_{v \in B'} c'(v) + w(B) \chi(G)(\ell + 1) \\ &< \sum_{B' \in \mathcal{B} \setminus \{B\}} w(B') \max_{v \in B'} c^*(v) + w(B) \max_{v \in B} c^*(v) = \text{cost}(c^*). \end{aligned}$$

This contradicts the optimality of c^* . ◀

We now focus on the case where the input graph G has treewidth at most tw . Note that $\chi(G) \leq \text{tw} + 1$, because the degeneracy of G is at most tw . Thus, the number of colors in any optimal coloring is at most $(\text{tw} + 1)|\mathcal{B}|$ by Lemma 4.

To solve MINIMUM SUM COLORING WITH BUNDLES for graphs $G = (V, E)$ with treewidth at most tw , we employ Courcelle's theorem [1, 6, 7]. To this end, we introduce the following auxiliary decision problem. In addition to the input $(G, \mathcal{B}, w, C)$ to MINIMUM SUM COLORING WITH BUNDLES, where $\mathcal{B} = \{B_1, B_2, \dots, B_\ell\}$, we also take a sequence $\mathbf{k} = (k_1, k_2, \dots, k_\ell)$ of integers. Then, we want to decide whether there exists a coloring $c: V \rightarrow \mathbb{Z}_{>0}$ such that $\max_{v \in B_j} c(v) \leq k_j$ for all $j \in [\ell]$. If we have a solution to this auxiliary problem for every $\mathbf{k} \in [(\text{tw} + 1)\ell]^\ell$, then the optimal cost for the instance $(G, \mathcal{B}, w, C)$ can be derived as

$$\min_{\mathbf{k}} \left\{ \sum_{j=1}^{\ell} w(B_j)k_j \mid \text{the answer to the instance } (G, \mathcal{B}, w, C, \mathbf{k}) \text{ is yes} \right\}. \quad (1)$$

We now focus on solving the auxiliary problem above. To solve the auxiliary problem using Courcelle's theorem [1, 6, 7], it is sufficient to write down an MSO formula that expresses the decision. An instance is given as $(G, \mathcal{B}, w, C, \mathbf{k})$, and we want to decide whether there exists a coloring c such that $\max_{v \in B_j} c(v) \leq k_j$ for all $j \in [\ell]$. By Lemma 4, we can assume that our coloring c uses colors in $[(\text{tw} + 1)\ell]$. For brevity, let $p = (\text{tw} + 1)\ell$. To express conditions with an MSO formula, we regard a coloring c as a partition $\{C_1, C_2, \dots, C_p\}$ of the vertex set V into independent sets, where $C_i = \{v \in V \mid c(v) = i\}$. Note that in such a partition a set C_i can be empty.

Then, the decision can be expressed by the following MSO formula:

$$\exists C_1, C_2, \dots, C_p \subseteq V: \bigwedge_{i=1}^p \bigwedge_{i'=i+1}^p \forall v \in V: \neg(v \in C_i \wedge v \in C_{i'}) \quad (2)$$

$$\wedge \forall v \in V: \bigvee_{i=1}^p (v \in C_i) \quad (3)$$

$$\wedge \bigwedge_{i=1}^p \forall u, v \in V: (u \in C_i \wedge v \in C_i \rightarrow \neg(uv \in E)) \quad (4)$$

$$\wedge \bigwedge_{j=1}^{\ell} \forall v \in V: \left(v \in B_j \rightarrow \bigvee_{i=1}^{k_j} (v \in C_i) \right). \quad (5)$$

In the formula, the conjunction of Formulae (2) and (3) represents that $\{C_1, C_2, \dots, C_p\}$ is a partition of V . Formula (4) represents that each C_i is independent. Formula (5) represents that the maximum color in bundle B_j is at most k_j for each $j \in [\ell]$. Thus, the formula correctly represents the conditions in the auxiliary problem.

By Courcelle's theorem, this auxiliary problem can be solved in $O(f(\text{tw}, \ell)|V|)$ time for some computable function f . To evaluate the minimum in (1), we solve the auxiliary problem for every choice of $\mathbf{k} \in [(\text{tw} + 1)\ell]^\ell$, thus at most $((\text{tw} + 1)\ell)^\ell$ times. Let $g(\text{tw})|V|$ be the running time of Bodlaender's algorithm [4] for computing a tree decomposition of width tw . Then, the overall running time is $g(\text{tw})|V| + ((\text{tw} + 1)\ell)^\ell \cdot O(f(\text{tw}, \ell)|V|)$. Setting $h(\text{tw}, \ell) = g(\text{tw}) + ((\text{tw} + 1)\ell)^\ell f(\text{tw}, \ell)$, we obtain the following theorem.

► **Theorem 5.** *MINIMUM SUM COLORING WITH BUNDLES can be solved in $O(h(\text{tw}, \ell)|V|)$ time for some computable function $h: \mathbb{Z}_{\geq 0} \times \mathbb{Z}_{\geq 0} \rightarrow \mathbb{Z}_{\geq 0}$, where tw is the treewidth of G and ℓ is the number of bundles.* ◀

3.3 An XP-Algorithm when $|V| - |\mathcal{B}|$ is a Parameter

We consider the case where G is a tree, the bundle family is a partition of the vertex set, and the number of bundles is large. More specifically, if n is the number of vertices and ℓ is the number of bundles that form a partition of the vertex set, we treat $n - \ell$ as a parameter.

Let $B \in \mathcal{B}$ be a bundle. We call B a *singleton bundle* if $|B| = 1$; otherwise, we call B a *non-singleton bundle*.

► **Lemma 6.** *Let $I = (G = (V, E), \mathcal{B}, w, C)$ be an instance of MINIMUM SUM COLORING WITH BUNDLES, where $\mathcal{B}$ is a partition of V . Then, the number of non-singleton bundles in $\mathcal{B}$ is at most $|V| - |\mathcal{B}|$.*

Proof. Let t be the number of non-singleton bundles. Since $\mathcal{B}$ is a partition of V , we obtain $|V| \geq 2t + (|\mathcal{B}| - t) = |\mathcal{B}| + t$. Thus, $t \leq |V| - |\mathcal{B}|$. ◀

Let $I = (G = (V, E), \mathcal{B}, w, C)$ be an instance of MINIMUM SUM COLORING WITH BUNDLES, and assume that G is a tree, $\mathcal{B}$ is a partition of V . Let t be the number of non-singleton bundles in $\mathcal{B}$. By Lemma 6, $t \leq |V| - |\mathcal{B}| = n - \ell$. Denote the non-singleton bundles in $\mathcal{B}$ by $B_1, B_2, \dots, B_t$, and the set of vertices, each of which forms a singleton bundle, by V_s .

A primary idea of our algorithm is to fix upper bounds on the colors used in non-singleton bundles and to optimize the colors in singleton bundles with this restriction. For a non-singleton bundle B_j , $j \in [t]$, let k_j be an upper bound on the colors in B_j ; namely, we seek a coloring c such that $\max_{u \in B_j} c(u) \leq k_j$ for every $j \in [t]$. For brevity, we denote $\mathbf{k} = (k_1, k_2, \dots, k_t)$. By Lemma 4, it is enough to consider the situations with $k_j \leq 2\ell$ for all $j \in [t]$, and thus the number of possible choices for $\mathbf{k}$ is bounded by $(2\ell)^t$ from above.

We regard G as a rooted tree with a root $r \in V$. For each vertex $v \in V$, we denote by $V(v)$ the vertex set of the subtree of G rooted at v . For each vertex $v \in V$, each color k , and an upper-bound tuple $\mathbf{k}$, we define

$$f(v, k; \mathbf{k}) := \min \left\{ \sum_{u \in V_s} w(\{u\})c(u) \mid \begin{array}{l} c \text{ is a } V(v)\text{-partial coloring of } G, \\ c(v) = k, \\ \max_{u \in B_j} c(u) \leq k_j \text{ for all } j \in [t] \end{array} \right\}.$$

Then, the optimal value for the instance I is read by

$$\min_{\mathbf{k}, k} \left(f(r, k; \mathbf{k}) + \sum_{j=1}^t k_j \right),$$

where the minimum is taken over all possible choices of $\mathbf{k} = (k_1, k_2, \dots, k_t)$ and k . Note that we only need to consider the value of k in the interval $k \in [2\ell]$ by Lemma 4.

We now establish a recursive formula for $f(v, k; \mathbf{k})$. First, consider the case where v is a leaf of G . We have two subcases. If v forms a singleton bundle by itself, then

$$f(v, k; \mathbf{k}) = w(\{v\})k.$$

If v belongs to a non-singleton bundle B_j for some $j \in [t]$, which is unique since $\mathcal{B}$ forms a partition of V , then

$$f(v, k; \mathbf{k}) = \begin{cases} 0 & \text{if } k \leq k_j, \\ +\infty & \text{otherwise.} \end{cases}$$

Next, consider the case where v is not a leaf of G . Let $v_1, \dots, v_z$ be the children of v in G . We again have two subcases. If v forms a singleton bundle by itself, then

$$\begin{aligned} f(v, k; \mathbf{k}) &= w(\{v\})k + \min \left\{ \sum_{y=1}^z f(v_y, k'_y; \mathbf{k}) \mid k'_y \neq k \text{ for all } y \in [z] \right\} \\ &= w(\{v\})k + \sum_{y=1}^z \min \{ f(v_y, k'_y; \mathbf{k}) \mid k'_y \neq k \} \end{aligned}$$

since the values $f(v_y, k'_y; \mathbf{k})$ are independent from each other. Similarly, if v belongs to a non-singleton bundle B_j for some $j \in [t]$, then

$$f(v, k; \mathbf{k}) = \begin{cases} \sum_{y=1}^z \min \{ f(v_y, k'_y; \mathbf{k}) \mid k'_y \neq k \} & \text{if } k \leq k_j, \\ +\infty & \text{otherwise.} \end{cases}$$

For each fixed $\mathbf{k}$, we compute $f(v, k; \mathbf{k})$ in a bottom-up manner from leaves to root. When v is not a leaf, the computation of $f(v, k; \mathbf{k})$ can be done in $O(z\ell)$ time for each fixed k since we may suppose that $k'_y \leq 2\ell$ by Lemma 4. In total, for each fixed $\mathbf{k}$, the computation of $f(v, k; \mathbf{k})$ over all v and k can be done in $O(n\ell^2)$ time. Hence, for varying $\mathbf{k}$, the running time of our algorithm is $O((2\ell)^t n\ell^2)$. When $n - \ell$ is a parameter, this is an XP algorithm since $t \leq n - \ell$ by Lemma 6.

We summarize our findings in the following theorem.

► **Theorem 7.** *MINIMUM SUM COLORING WITH BUNDLES can be solved in $O((2\ell)^t n\ell^2)$ time when G is a tree, n is the number of vertices, ℓ is the number of bundles, and t is the number of non-singleton bundles.*

3.4 Polynomial-time Algorithm for the Connected Partition Case

In this subsection, we consider MINIMUM SUM COLORING WITH BUNDLES in trees when the bundle family is a connected partition, namely, when each bundle induces a connected subgraph and each vertex belongs to a unique bundle.

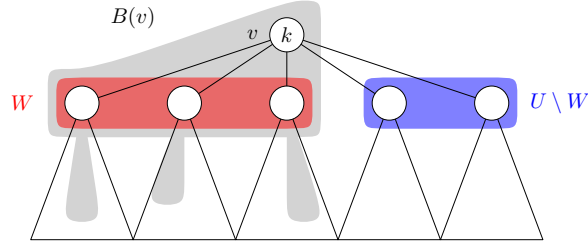
Let $I = (G = (V, E), \mathcal{B}, w, C)$ be an instance of MINIMUM SUM COLORING WITH BUNDLES, and assume that G is a tree, $\mathcal{B}$ is a connected partition of V . Define $n := |V|$. For each vertex $v \in V$, we denote by $B(v)$ a unique bundle in $\mathcal{B}$ containing v . Notice that there exists an optimal coloring c such that $c(v) \in [n]$ for every vertex $v \in V$ by Observation 1.

We regard G as a rooted tree with a root $r \in V$. For each vertex $v \in V$, we denote by $V(v)$ the vertex set of the subtree of G rooted at v . For each vertex $v \in V$, each integer $k \in [n]$, and each integer $g \in [n]$, we define

$$f(v, k, g) := \min \left\{ \sum_{B \in \mathcal{B}} w(B) \max_{u \in B} c(u) \mid \begin{array}{l} c \text{ is a } V(v)\text{-partial coloring of } G, \\ c(u) \leq n \text{ for all } u \in V, \\ c(v) = k, \\ \max\{c(u) \mid u \in B(v)\} = g \end{array} \right\}.$$

We now establish a recursive formula for $f(v, k, g)$. First, consider the case where v is a leaf of G . For $k \in [n]$ and $g \in [n]$, we obtain

$$f(v, k, g) = \begin{cases} w(B(v))g & \text{if } k = g, \\ +\infty & \text{otherwise.} \end{cases}$$



■ **Figure 3** Notation for the case where $\mathcal{B}$ is a partition of V .

Next, consider the case where v is not a leaf of G . Let U be the set of children of v and let $W := U \cap B(v)$. See Figure 3. Notice that $B(u) = B(v)$ for every $u \in W$. Furthermore, since $B(v)$ is connected, $B(v) \cap V(u) = \emptyset$ for every $u \in U \setminus W$. If $k > g$, then we obtain $f(v, k, g) = +\infty$ since $c(v) \leq \max\{c(u) \mid u \in B(v)\}$. Since $\mathcal{B}$ is a connected partition of V , $|\{u \in U \mid B \cap V(u) \neq \emptyset\}| \leq 1$ for each bundle $B \in \mathcal{B} \setminus \{B(v)\}$. Thus, if $k = g$, then

$$\begin{aligned} f(v, k, g) &= w(B(v))g + \sum_{u \in W} \min_{k' \in [n] \setminus \{k\}, g' \in [g]} \left(f(u, k', g') - w(B(u))g' \right) \\ &\quad + \sum_{u \in U \setminus W} \min_{k' \in [n] \setminus \{k\}, g' \in [n]} f(u, k', g'). \end{aligned}$$

Similarly, if $k < g$, then

$$\begin{aligned} f(v, k, g) &= \min_{u' \in W} \left\{ \min_{k' \in [n] \setminus \{k\}} f(u', k', g) \right. \\ &\quad + \sum_{u \in W \setminus \{u'\}} \min_{k' \in [n] \setminus \{k\}, g' \in [g]} \left(f(u, k', g') - w(B(u))g' \right) \\ &\quad \left. + \sum_{x \in U \setminus W} \min_{k' \in [n] \setminus \{k\}, g' \in [n]} f(x, k', g') \right\}. \end{aligned}$$

By computing $f(v, k, g)$ over all k and g with these equations, we can solve MINIMUM SUM COLORING WITH BUNDLES in this case. For each vertex $v \in V$, we can compute $f(v, k, g)$ over all k and g in $O(|\delta_G(v)|^2 n^4)$ time. Thus, we obtain the following theorem.

► **Theorem 8.** *MINIMUM SUM COLORING WITH BUNDLES can be solved in $O(n^6)$ time when G is a tree, the subgraph of G induced by each bundle $B \in \mathcal{B}$ is connected, and $\mathcal{B}$ is a partition of V .*

Finally, we present the following theorem, asserting that if G is a path, we can solve in polynomial time MINIMUM SUM COLORING WITH BUNDLES when $\mathcal{B}$ is connected but not necessarily a partition. See the full version for the proof.

► **Theorem 9.** *MINIMUM SUM COLORING WITH BUNDLES can be solved in $O(|V|^3 |\mathcal{B}|)$ time when $G = (V, E)$ is a path and the subgraph of G induced by each bundle $B \in \mathcal{B}$ is connected.*

4 Bipartite Graphs

In this section, we first prove that MINIMUM SUM COLORING WITH BUNDLES is NP-complete even when G is a bipartite graph, the number of bundles is four, and the weights are uniform. To prove NP-completeness, we use the following problem.

40:12 Minimum Sum Coloring with Bundles in Trees and Bipartite Graphs

Problem: BIPARTITE 3-LIST-COLORING

Input: A bipartite graph $G = (V, E)$ and a list $L(v) \subseteq \{1, 2, 3\}$ of available colors for each vertex $v \in V$.

Question: Determine whether there exists a proper coloring $c: V \rightarrow \{1, 2, 3\}$ such that $c(v) \in L(v)$ for each $v \in V$.

As observed by Golovach and Paulusma [12], a reduction given by Jansen and Scheffler [15] proves the NP-completeness of BIPARTITE 3-LIST-COLORING.

► **Theorem 10.** *MINIMUM SUM COLORING WITH BUNDLES is NP-complete even when G is a bipartite graph, $|\mathcal{B}| = 4$, and $w(B) = 1$ for every $B \in \mathcal{B}$.*

Proof. It is easy to see that the problem is in NP. In what follows, we prove that the problem is NP-hard by reducing BIPARTITE 3-LIST-COLORING.

Suppose that we are given an instance of the problem BIPARTITE 3-LIST-COLORING. Specifically, let $G = (V, E)$ be a bipartite graph, and let $L(v) \subseteq \{1, 2, 3\}$ be a list of available colors for each vertex $v \in V$. For $Z \subseteq \{1, 2, 3\}$, we define $U_Z = \{v \in V \mid L(v) = Z\}$. Then $\{U_Z \mid Z \subseteq \{1, 2, 3\}\}$ forms a partition of V ; that is, $V = \bigcup_{Z \subseteq \{1, 2, 3\}} U_Z$ and $U_Z \cap U_{Z'} = \emptyset$ for any distinct subsets $Z, Z' \subseteq \{1, 2, 3\}$.

We construct an instance $(\tilde{G}, \mathcal{B}, w, C)$ of MINIMUM SUM COLORING WITH BUNDLES, where $\tilde{G} = (\tilde{V}, \tilde{E})$, as follows. For $Z \in \{\{2\}, \{2, 3\}\}$, we denote by U_Z^1 a copy of U_Z , where $u^1 \in U_Z^1$ means a copy of $u \in U_Z$. Similarly, we define two copies $U_{\{1,3\}}^1$ and $U_{\{1,3\}}^2$ of $U_{\{1,3\}}$, and three copies $U_{\{3\}}^1$, $U_{\{3\}}^2$, and $U_{\{3\}}^3$ of $U_{\{3\}}$. Then, the vertex set $\tilde{V}$ of $\tilde{G}$ is defined as

$$\tilde{V} = V \cup U_{\{2\}}^1 \cup \bigcup_{i=1}^3 U_{\{3\}}^i \cup \bigcup_{i=1}^2 U_{\{1,3\}}^i \cup U_{\{2,3\}}^1 \cup \{v_i \mid i \in [16]\}.$$

The edge set $\tilde{E}$ is defined as

$$\tilde{E} = E \cup E_{\{2\}} \cup E_{\{3\}} \cup E_{\{1,3\}} \cup E_{\{2,3\}} \cup E_0,$$

where we define

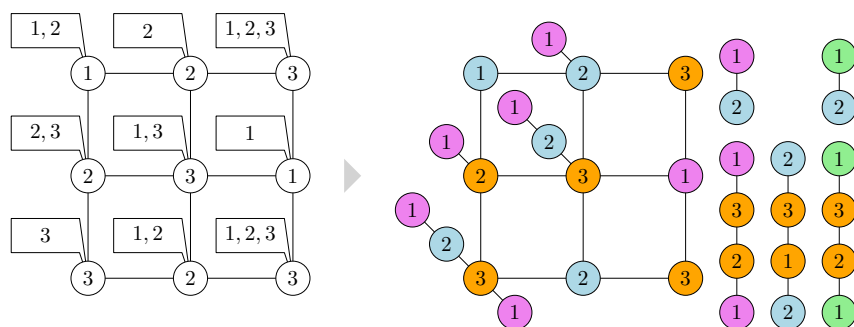
$$\begin{aligned} E_{\{2\}} &= \{uu^1 \mid u \in U_{\{2\}}\}, & E_{\{3\}} &= \{uu^2, u^2u^1, uu^3 \mid u \in U_{\{3\}}\}, \\ E_{\{1,3\}} &= \{uu^2, u^2u^1 \mid u \in U_{\{1,3\}}\}, & E_{\{2,3\}} &= \{uu^1 \mid u \in U_{\{2,3\}}\}, \\ E_0 &= \{v_1v_2, v_{11}v_{12}\} \cup \{v_i v_{i+1} \mid i \in \{3, 4, 5\} \cup \{7, 8, 9\} \cup \{13, 14, 15\}\}. \end{aligned}$$

We note that each of $E_{\{2\}}$, $E_{\{3\}}$, $E_{\{1,3\}}$, and $E_{\{2,3\}}$ consists of disjoint edges and paths. Also, E_0 contains 3 disjoint paths of length 3 and 2 disjoint edges. Since $\tilde{G}$ is obtained from the bipartite graph G by adding edges of $E_{\{2\}} \cup E_{\{3\}} \cup E_{\{1,3\}} \cup E_{\{2,3\}} \cup E_0$, the constructed graph $\tilde{G}$ is bipartite. Moreover, define a family of bundles $\mathcal{B} = \{B_1, B_2, B_3, B_4\}$ as

$$\begin{aligned} B_1 &= U_{\{1\}} \cup U_{\{2\}}^1 \cup U_{\{3\}}^1 \cup U_{\{3\}}^3 \cup U_{\{1,3\}}^1 \cup U_{\{2,3\}}^1 \cup \{v_1, v_3, v_6\}, \\ B_2 &= U_{\{2\}} \cup U_{\{1,2\}} \cup U_{\{3\}}^2 \cup U_{\{1,3\}}^2 \cup \{v_2, v_7, v_{10}, v_{12}\}, \\ B_3 &= U_{\{3\}} \cup U_{\{1,3\}} \cup U_{\{2,3\}} \cup U_{\{1,2,3\}} \cup \{v_4, v_5, v_8, v_9, v_{14}, v_{15}\}, \\ B_4 &= \{v_{11}, v_{13}, v_{16}\}. \end{aligned}$$

We set $w(B_i) = 1$ for $i \in \{1, 2, 3, 4\}$ and $C = 7$. This defines an instance $(\tilde{G}, \mathcal{B}, w, C)$ of MINIMUM SUM COLORING WITH BUNDLES. See Figure 4.

We claim that the instance $(\tilde{G}, \mathcal{B}, w, C)$ of MINIMUM SUM COLORING WITH BUNDLES is a yes-instance if and only if the given instance (G, L) of BIPARTITE 3-LIST-COLORING is a yes-instance. The proof of this claim can be found in the full version. ◀



■ **Figure 4** Reduction for bipartite graphs with uniform weight. (Left) An instance of BIPARTITE 3-LIST-COLORING. The flag attached to each vertex v represents the list $L(v)$. The numbers on vertices represent a proper coloring that respects the lists. (Right) The instance of MINIMUM SUM COLORING WITH BUNDLES that is constructed from the left instance of BIPARTITE 3-LIST-COLORING. Colors represent bundles: the violet vertices form B_1 , the blue vertices form B_2 , the orange vertices form B_3 , and the green vertices form B_4 . The numbers on vertices represent the coloring obtained as in the proof of the if-part.

For the weighted case, MINIMUM SUM COLORING WITH BUNDLES on bipartite graphs is NP-complete even when $|\mathcal{B}| = 3$. The proof is an adaptation of that for Theorem 10, and can be found in the full version.

► **Theorem 11.** *MINIMUM SUM COLORING WITH BUNDLES is NP-complete even when G is a bipartite graph, $|\mathcal{B}| = 3$, and $w(B) \in \{1, 2\}$ for every $B \in \mathcal{B}$.*

To complement Theorems 10 and 11 concerning bipartite graphs, we prove that MINIMUM SUM COLORING WITH BUNDLES can be solved in polynomial time under two specific conditions: when $|\mathcal{B}| \leq 2$ and when $|\mathcal{B}| \leq 3$ with uniform weights. The proofs can be found in the full version.

► **Theorem 12.** *MINIMUM SUM COLORING WITH BUNDLES can be solved in polynomial time when G is bipartite and $|\mathcal{B}| \leq 2$.*

► **Theorem 13.** *MINIMUM SUM COLORING WITH BUNDLES with uniform weight can be solved in polynomial time when G is bipartite and $|\mathcal{B}| \leq 3$.*

5 Concluding Remarks

We have completed a detailed complexity analysis of MINIMUM SUM COLORING WITH BUNDLES for trees and bipartite graphs, and solved an open problem by Darboui and Friggstad [10]. Here, we pose a few open problems. First, we do not know the existence of an FPT algorithm for trees when $|V| - |\mathcal{B}|$ is a parameter and the bundles form a partition of V . Second, we do not know the complexity for bipartite graphs when the bundles form an independent partition or a connected partition. Third, we do not know any better approximation ratio for trees and bipartite graphs than the one proposed by Darboui and Friggstad [10].

References

- 1 Stefan Arnborg, Jens Lagergren, and Detlef Seese. Easy problems for tree-decomposable graphs. *J. Algorithms*, 12(2):308–340, 1991. doi:10.1016/0196-6774(91)90006-K.
- 2 Brenda S. Baker and Edward G. Coffman Jr. Mutual exclusion scheduling. *Theor. Comput. Sci.*, 162(2):225–243, 1996. doi:10.1016/0304-3975(96)00031-X.

- 3 Amotz Bar-Noy and Guy Kortsarz. Minimum color sum of bipartite graphs. *J. Algorithms*, 28(2):339–365, 1998. doi:10.1006/JAGM.1998.0938.
- 4 Hans L. Bodlaender. A linear-time algorithm for finding tree-decompositions of small treewidth. *SIAM J. Comput.*, 25(6):1305–1317, 1996. doi:10.1137/S0097539793251219.
- 5 Hans L. Bodlaender, Klaus Jansen, and Gerhard J. Woeginger. Scheduling with incompatible jobs. *Discret. Appl. Math.*, 55(3):219–232, 1994. doi:10.1016/0166-218X(94)90009-4.
- 6 Bruno Courcelle. The monadic second-order logic of graphs I: recognizable sets of finite graphs. *Inf. Comput.*, 85(1):12–75, 1990. doi:10.1016/0890-5401(90)90043-H.
- 7 Bruno Courcelle. The monadic second-order logic of graphs III: tree-decompositions, minor and complexity issues. *RAIRO Theor. Informatics Appl.*, 26:257–286, 1992. doi:10.1051/ITA/1992260302571.
- 8 Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshantov, Dániel Marx, Marcin Pilipczuk, Michał Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015. doi:10.1007/978-3-319-21275-3.
- 9 David P. Dailey. Uniqueness of colorability and colorability of planar 4-regular graphs are NP-complete. *Discret. Math.*, 30(3):289–293, 1980. doi:10.1016/0012-365X(80)90236-8.
- 10 Seyed Parsa Darbouy and Zachary Friggstad. Approximating minimum sum coloring with bundles. In Hans L. Bodlaender, editor, *19th Scandinavian Symposium and Workshops on Algorithm Theory, SWAT 2024, June 12-14, 2024, Helsinki, Finland*, volume 294 of *LIPICs*, pages 21:1–21:14. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024. doi:10.4230/LIPICs.SWAT.2024.21.
- 11 Guy Even, Magnús M. Halldórsson, Lotem Kaplan, and Dana Ron. Scheduling with conflicts: online and offline algorithms. *J. Sched.*, 12(2):199–224, 2009. doi:10.1007/S10951-008-0089-1.
- 12 Petr A. Golovach and Daniël Paulusma. List coloring in the absence of two subgraphs. *Discret. Appl. Math.*, 166:123–130, 2014. doi:10.1016/J.DAM.2013.10.010.
- 13 Martin Grötschel, László Lovász, and Alexander Schrijver. *Geometric Algorithms and Combinatorial Optimization, Second Edition*. Springer Berlin Heidelberg, 1993. doi:10.1007/978-3-642-78240-4.
- 14 Magnús M. Halldórsson and Guy Kortsarz. Algorithms for chromatic sums, multicoloring, and scheduling dependent jobs. In Teofilo F. Gonzalez, editor, *Handbook of Approximation Algorithms and Metaheuristics, Second Edition, Volume 1: Methodologies and Traditional Applications*, pages 671–684. Chapman and Hall/CRC, 2018. doi:10.1201/9781351236423-38.
- 15 Klaus Jansen and Petra Scheffler. Generalized coloring for tree-like graphs. *Discret. Appl. Math.*, 75(2):135–155, 1997. doi:10.1016/S0166-218X(96)00085-6.
- 16 Richard M. Karp. Reducibility among combinatorial problems. In Raymond E. Miller and James W. Thatcher, editors, *Proceedings of a symposium on the Complexity of Computer Computations, held March 20-22, 1972, at the IBM Thomas J. Watson Research Center, Yorktown Heights, New York, USA*, The IBM Research Symposia Series, pages 85–103. Plenum Press, New York, 1972. doi:10.1007/978-1-4684-2001-2_9.
- 17 Ewa M. Kubicka and Allen J. Schwenk. An introduction to chromatic sums. In Arthur M. Riehl, editor, *Computer Trends in the 1990s - Proceedings of the 1989 ACM 17th Annual Computer Science Conference, Louisville, Kentucky, USA, February 21-23, 1989*, pages 39–45. ACM, 1989. doi:10.1145/75427.75430.
- 18 Dániel Marx. A short proof of the NP-completeness of minimum sum interval coloring. *Oper. Res. Lett.*, 33(4):382–384, 2005. doi:10.1016/J.ORL.2004.07.006.
- 19 Tibor Szkaliczki. Routing with minimum wire length in the dogleg-free Manhattan model is NP-complete. *SIAM J. Comput.*, 29(1):274–287, 1999. doi:10.1137/S0097539796303123.