ELSEVIER

Available online at www.sciencedirect.com

SCIENCE DIRECT®

Theoretical Computer Science 349 (2005) 22–30

Theoretical Computer Science

www.elsevier.com/locate/tcs

Equitable colorings of bounded treewidth graphs

Hans L. Bodlaender[a], Fedor V. Fomin[b,*,1]

[a] *Institute of Information and Computing Sciences, Utrecht University, P.O. Box 80.089, 3508 TB Utrecht, The Netherlands*
[b] *Department of Informatics, University of Bergen, Norway*

Abstract

A proper coloring of a graph G is *equitable* if the sizes of any two color classes differ by at most one. A proper coloring is ℓ-bounded, when each color class has size at most ℓ. We consider the problems to determine for a given graph G (and a given integer ℓ) whether G has an equitable (ℓ-bounded) k-coloring. We prove that both problems can be solved in polynomial time on graphs of bounded treewidth, and show that a precolored version remains NP-complete on trees.

Keywords: Equitable coloring; Treewidth

1. Introduction

There is a wide belief that almost every natural hard problem can be solved efficiently on graphs of bounded treewidth. Of course this belief is not always true, a nice example is the BANDWIDTH MINIMIZATION problem which is NP-hard even on trees of degree three [12,26]. Another part of 'folklore' in the graph algorithms community is that if some (natural) problem can be solved in polynomial time on trees, one should be able to solve it in polynomial time on graphs of bounded treewidth. However, there are some striking and frustrating examples, like $L(2,1)$-COLORING, where an efficient algorithm for trees was known [8], but the complexity of the problem on graphs of treewidth $\geqslant 2$ was an open problem until very recently [28]. For more than 10 years, EQUITABLE k-COLORING and ℓ-BOUNDED k-COLORING were also examples of such problems. Both problems can be solved in polynomial time on trees and forests [10,2,18], i.e. graphs of treewidth 1, and the existence of a polynomial time algorithm for graphs of treewidth $\geqslant 2$ was an open question. In this paper, we introduce the first polynomial time algorithm on graphs of bounded treewidth for both versions of coloring. Due to enormous exponents in the running time, our algorithm is mainly of theoretical interest. Our main technique is quite far from the standard dynamic programming on graphs of bounded treewidth. To convince the reader (and ourselves) that the standard dynamic programming approach is unlikely to be implemented for EQUITABLE k-COLORING on graphs of bounded treewidth, we prove that a precolored version of the problem is NP-complete on trees, hence on graphs of treewidth 1. The main idea behind our polynomial time algorithm is to use recent combinatorial results of Kostochka et al. [22] that allow us to handle graphs with 'large' vertex degrees separately.

Previous results: The EQUITABLE k-COLORING problem has a long history. The celebrated theorem of Hajnal and Szemerédi [14] says that any graph G has an equitable k-coloring for $k \geqslant \Delta(G)+1$. This bound is sharp. One of the

* Corresponding author.
E-mail addresses: hansb@cs.uu.nl (H.L. Bodlaender), fomin@ii.uib.no (F.V. Fomin).
[1] Fedor Fomin is supported by Norges forskningsråd project 160778/V30.

0304-3975/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.tcs.2005.09.027

directions of research in this field was to obtain better upper bounds than $\Delta(G) + 1$ for special graph classes. See the survey [24] for a review of the results in this field.

The k-COLORING problem can be trivially reduced to EQUITABLE k-COLORING problem and thus EQUITABLE k-COLORING is NP-hard. Polynomial time algorithms are known for split graphs [9] and trees [10].

The related ℓ-BOUNDED k-COLORING problem has a number of applications. It is also known as the MUTUAL EXCLUSION SCHEDULING problem (MES) which is the problem of scheduling unit-time tasks non-preemptively on ℓ processors subject to constraints, represented by a graph G, so that tasks represented by adjacent vertices in G must run in disjoint time intervals. This problem arises in load balancing the parallel solution of partial differential equations by domain decomposition. (See [2,27] for more information.) Also the problems of this form have been studied in the operations research literature [3,23]. Other applications are in scheduling in communication systems [16] and in constructing school timetables [20].

When both ℓ and k are variable, the ℓ-BOUNDED k-COLORING problem can be solved in polynomial time on split graphs, complements of interval graphs [25,9], forests and in linear time on trees [2,18]. This is almost all what is known about graph classes where the ℓ-BOUNDED k-COLORING problem is efficiently solvable. When one of the parameters ℓ or k is fixed the situation is different. For example, for fixed ℓ or k the problem is solved on cographs [5,25] and for fixed ℓ on bipartite graphs [5,15] and line graphs [1]. For $\ell = 2$ the problem is equivalent to the MAXIMUM MATCHING problem on the complement graphs and is polynomial. Note that for fixed ℓ the problem can be expressed in the counting monadic second-order logic and for graphs of bounded treewidth, a linear time algorithm for *fixed* ℓ can be constructed [19]. When ℓ is not fixed (i.e. ℓ is part of the input), the situation is not simple even for trees, and the question of existence of a polynomial time algorithm for trees [15] was open for several years.

The problem remains NP-complete on cographs, bipartite and interval graphs [5], on cocomparability graphs and fixed $\ell \geqslant 3$ [25], on complements of line graphs and fixed $\ell \geqslant 3$ [11], and on permutation graphs and $\ell \geqslant 6$ [17]. For $k = 3$ the problem is NP-complete on bipartite graphs [5].

Almost all NP-completeness results for ℓ-BOUNDED k-COLORING for different graph classes mentioned above can also be obtained for EQUITABLE k-COLORING by making use of the following observation.

Proposition 1. *A graph G on n vertices is ℓ-bounded k-colorable if and only if the graph G' obtained by taking disjoint union of G and an independent set of size $\ell k - n$ is equitably k-colorable.*

Our contribution: A standard dynamic programming approach for the COLORING problem needs to keep $O(w^k n)$ entries, where w is the treewidth of a graph and k is the number of colors. Since the chromatic number of a graph is at most $w + 1$ this implies that the classical COLORING problem can be solved in polynomial time on graphs of bounded treewidth. Clearly such a technique does not work for EQUITABLE k-COLORING because the number of colors in an equitable coloring is not bounded by a function of the treewidth. For example, a star on n vertices has treewidth 1 and it cannot be equitable k-colored for any $k < (n-1)/2$. One of the indications that the complexity of EQUITABLE k-COLORING for graphs of bounded treewidth can be different from the ordinary coloring is that by Proposition 1 and [5], the problem is NP-hard on cographs and thus on graphs of bounded clique-width. (Note that chromatic number is polynomial time solvable on graphs of bounded clique-width [21].)

However, one of the properties of equitable colorings making our approach possible is the phenomena observed first by Bollobás and Guy [7] for trees: 'Most' trees can be equitable 3-colored. In other words, for almost all trees the difference between the numbers of colors in an equitable coloring is not 'far' from the chromatic number. Recently Kostochka et al. [22] succeeded to generalize the result of Bollobás and Guy for degenerate graphs and our main contribution—the proof that EQUITABLE k-COLORING can be solved in polynomial time on graphs of bounded treewidth (Section 3)—strongly uses this result. Very roughly, we use the results of Kostochka et al. to establish the threshold when the problem is trivially solved and when it becomes solvable in polynomial time by dynamic programming developed in Section 2. In Section 4, we show that such an approach cannot be extended to PRECOLORED EQUITABLE k-COLORING by showing that the precolored version of the problem is NP-hard on trees.

1.1. Definitions

We denote by $G = (V, E)$ a finite, undirected simple graph. We usually use n to denote the number of vertices in G. For every nonempty $W \subseteq V$, the subgraph of G induced by W is denoted by $G[W]$. The maximum degree of G is denoted by $\Delta(G)$. A graph G is d-*degenerate* if each of its non-empty subgraphs has a vertex of degree at most d. A non-empty subset of vertices $I \subseteq V$ is *independent* in G if no two of its elements are adjacent in G.

Definition. A *tree decomposition* of a graph $G = (V, E)$ is a pair $(\{X_i | i \in I\}, T = (I, F))$, with $\{X_i | i \in I\}$ a family of subsets of V and T a tree, such that
- $\bigcup_{i \in I} X_i = V$.
- For all $\{v, w\} \in E$, there is an $i \in I$ with $v, w \in X_i$.
- For all $i_0, i_1, i_2 \in I$: if i_1 is on the path from i_0 to i_2 in T, then $X_{i_0} \cap X_{i_2} \subseteq X_{i_1}$.

The *width* of tree decomposition $(\{X_i | i \in I\}, T = (I, F))$ is $\max_{i \in I} |X_i| - 1$. The *treewidth* of a graph G is the minimum width of a tree decomposition of G.

Lemma 2 (*Folklore*). *Every graph on n vertices and of treewidth* $\leqslant w$ *has a vertex of degree at most* w, *and has at most* wn *edges.*

A k-*coloring* of the vertices of a graph $G = (V, E)$ is a partition $A_1, A_2, \ldots, A_k$ of V into independent sets (in which some of the A_j may be empty); the k sets A_j are called the *color classes* of the k-coloring. The *chromatic number* $\chi(G)$ is the minimum value k for which a k-coloring exists. A k-coloring $A_1, A_2, \ldots, A_k$ is ℓ-*bounded* if $|A_i| \leqslant l$, $1 \leqslant i \leqslant k$. A k-coloring $A_1, A_2, \ldots, A_k$ is *equitable* if for any $i, j \in \{1, 2, \ldots, k\}$, $|A_i| - |A_j| \leqslant 1$.

In the EQUITABLE k-COLORING problem, we are given a graph $G = (V, E)$, and an integer k, and we ask whether G has an equitable k-coloring.

Theorem 3 (*Kostochka et al. [22]*). *Every* n-*vertex* d-*degenerate graph* G *is equitably* k-*colorable for any* $k \geqslant \max\{62d, 31d(n/(n - \Delta(G) + 1))\}$.

Every graph of treewidth at most d is d-degenerate and Theorem 3 implies the following corollary.

Corollary 4. *Every* n-*vertex graph* G *of treewidth* w *is equitably* k-*colorable for any* $k \geqslant \max\{62w, 31w \times (n/(n - \Delta(G) + 1))\}$.

2. Covering by equitable independent sets

Let $S \subseteq V$ be a set of vertices of a graph $G = (V, E)$. We say that S can be *covered by independent sets of size* $[n/k]$ if there is a set of subsets $A_i \subseteq V$, $i \in \{1, 2, \ldots, p\}$, $p \leqslant |S|$, such that

(i) For every $i \in \{1, 2, \ldots, p\}$, A_i is an independent set;
(ii) For every $i, j \in \{1, 2, \ldots, p\}$, $i \neq j$, $A_i \cap A_j = \emptyset$;
(iii) For every $i \in \{1, 2, \ldots, p\}$, either $|A_i| = \lceil n/k \rceil$, or $|A_i| = \lfloor n/k \rfloor$;
(iv) $S \subseteq \bigcup_{1 \leqslant i \leqslant p} A_i$.

Covering by independent sets is a natural generalization of an equitable coloring: a graph G has an equitable k-coloring if and only if V can be covered by independent sets of size $[n/k]$. We use the following observations in our proof.

Lemma 5. *Let* $S \subseteq V$ *be a vertex subset of a graph* G.
(a) *If* S *cannot be covered by independent sets of size* $[n/k]$, *the graph* G *is not equitably* k-*colorable.*
(b) *If* S *can be covered by* p *independent sets* $A_1, \ldots, A_p$ *of size* $[n/k]$ *and the graph* $G' = G[V \setminus \bigcup_{1 \leqslant i \leqslant p} A_i]$ *is equitably* $(k - p)$-*colorable, the graph* G *is equitably* k-*colorable.*

Proof. (a) Let $B_1, \ldots, B_k$ be the color classes of an equitable k-coloring of G. Consider the collection of sets $\{B_i \mid 1 \leqslant i \leqslant k,\ B_i \cap S \neq \emptyset\}$, and number them $A_1, A_2, \ldots$.

(b) Use color classes $A_1, \ldots, A_p$, and partition the vertices of G' as in the equitable coloring into color classes $A_{p+1}, \ldots, A_k$. $\square$

Let G be a graph of treewidth w. The next theorem implies that when the cardinality of $S \subseteq V$ or the number k is upper-bounded by a function of w, the question if S can be covered by independent sets of size $[n/k]$ can be answered in polynomial time. Because there are graphs that need $\Omega(n)$ colors in an equitable coloring, Theorem 6 does not imply directly that for graphs of bounded treewidth the EQUITABLE k-COLORING problem can be solved in polynomial time.

Theorem 6. *Let w be a constant. Let $G = (V, E)$ be an n-vertex graph of treewidth $\leqslant w$, let S be a subset of V, and let k be an integer. When k or $|S|$ is bounded by a constant, one can either find in polynomial time a covering of S by independent sets of size $[n/k]$, or conclude that there is no such a covering.*

Proof (*Sketch*). This can be shown using standard dynamic programming techniques for graphs of bounded treewidth so we give only outline of the proof. Also we put no efforts in optimizing the running time of the algorithm.

It is well known that every graph of treewidth $\leqslant w$ has a so called *nice* tree decomposition of width $\leqslant w$, i.e. a tree decomposition with a rooted tree T, with root $r \in I$ such that

- T is a binary tree.
- If a node $i \in I$ has two children j_1 and j_2, then $X_i = X_{j_1} = X_{j_2}$. The node i is called the *join* node.
- If a node $i \in I$ has one child j, then either $X_i = X_j \cup \{v\}$ (*introduce node*), or $X_i = X_j - \{v\}$ (*forget node*) for some vertex v of G.

Given a graph G, one can decide in linear time whether the treewidth of G is at most w, and if so, find a nice tree decomposition of G of width at most w (see [6,4]). So, let $(\{X_i \mid i \in I\}, T = (I, F))$ be a nice tree decomposition of G of width at most w with root r. For a node $i \in I$ let $G_i = (V_i, E_i)$ be the subgraph of G induced by vertices that are contained in i and its descendants. Thus $G = G_r$.

We assume that $\min\{|S|, k\} = c$ for some constant c. We now want to check if S can be covered by at most c independent sets of size $[n/k]$. We use a dynamic programming algorithm, where we compute for each node $i \in I$ in the tree decomposition a table of *states*, with each state associated to a Boolean value.

For a node $i \in I$, a state is specified by a $2c$ tuple $[Z_1, Z_2, \ldots, Z_c, \alpha_1, \alpha_2, \ldots, \alpha_c]$, where $Z_1, Z_2, \ldots, Z_c$ are pairwise disjoint subsets of X_i (here some Z_j's can be $\emptyset$) and $\alpha_1, \alpha_2, \ldots, \alpha_c$ are integers from 0 to $[n/k]$. Thus there are $O(w^c[n/k]^c)$ states for each node i. For every state we compute a Boolean value $B([Z_1, Z_2, \ldots, Z_c, \alpha_1, \alpha_2, \ldots, \alpha_c])$. This Boolean value is TRUE if and only if there is a set of subsets $A_j \subseteq V_i$, $j \in \{1, 2, \ldots, p\}$, $p \leqslant c$, such that

(i) For every $j \in \{1, 2, \ldots, p\}$, A_j is an independent set.
(ii) For every $j, l \in \{1, 2, \ldots, p\}$, $l \neq j$, $A_l \cap A_j = \emptyset$.
(iii) For every $j \in \{1, 2, \ldots, p\}$, $|A_j| = \alpha_j$.
(iv) $S \subseteq \bigcup_{1 \leqslant j \leqslant p} A_j$.
(v) For every $j \in \{1, 2, \ldots, p\}$, $A_j \cap X_i = Z_j$.

Clearly, S can be covered by independent sets of size $[n/k]$ if and only if $B([Z_1, Z_2, \ldots, Z_c, [n/k], [n/k], \ldots, [n/k]) = \text{TRUE}$ for some pairwise disjoint subsets $(Z_1, Z_2, \ldots, Z_c)$ of X_r.

Now, in bottom-up order, we compute for each node $i \in I$, for all its states their Boolean value, i.e., a table for $i \in I$ is computed after the tables of the children of i are computed. For a leaf of T, we can compute all state values in $O(w^c)$ time with brute force. A detailed, but not difficult, argument shows that one can compute for join, introduce, and forget nodes its state values in $O(w^c[n/k]^{2c})$ time, when given all state values for the children of the node. Thus, with $O(nw^c[n/k]^{2c})$ time, we can compute the state values of the root of T, and hence decide if there the desired covering exists. Finally, using additional bookkeeping one can also solve the construction variant of the problem and find, if existing, the covering of S by independent sets of size $[n/k]$. □

3. Bounded treewidth

The main result of this paper is the following theorem.

Theorem 7. *The* EQUITABLE k-COLORING *problem can be solved in polynomial time on graphs of bounded treewidth.*

Proof. Let $G = (V, E)$ be a graph of treewidth w and let k be an integer. To determine if G has an equitable k-coloring, we consider the following cases.

Case 1: $\Delta(G) \leqslant n/2 + 1$ and $k \geqslant 62w$. Since

$$\max_{0 \leqslant \Delta(G) \leqslant n/2+1} \frac{n}{n - \Delta(G) + 1} = 2,$$

we have that

$$k \geqslant 62w = \max\{62w, 2 \cdot 31w\} \geqslant \max\left\{62w, 31w\frac{n}{n - \Delta(G) + 1}\right\}$$

and by Corollary 4, G is equitably k-colorable.

Case 2: $\Delta(G) \leqslant n/2 + 1$ and $k \leqslant 62w$. In this case, it follows from Theorem 6 that the question whether G has an equitable k-coloring can be solved in polynomial time.

Case 3: $\Delta(G) > n/2 + 1$. Let $S \subset V$ be the set of vertices in G of degree $\geqslant n/2 + 2$. By Lemma 2, G has at most wn edges, so $|S| \leqslant 4w$. Thus, by Theorem 6, it can be checked in polynomial time whether S can be covered by independent sets of size $[n/k]$. If S cannot be covered, then by part (a) of Lemma 5, G has no equitable k-coloring.

Let $A_i \subset V, i \in \{1, 2, \ldots, p\}, p \leqslant |S|$, be a covering of S by independent sets of size $[n/k]$. We define a new graph $G' = G[V \setminus \bigcup_{1 \leqslant i \leqslant p} A_i]$. The maximum vertex degree in G' is at most $n/2 + 1$ and the treewidth of G' is $\leqslant w$. Graph G' has

$$n' = \left|V \Big\backslash \bigcup_{1 \leqslant i \leqslant p} A_i\right| \geqslant n - p\left\lfloor\frac{n}{k}\right\rfloor \geqslant n - 4w\left(\frac{n}{k} - 1\right) > \left(1 - \frac{4w}{k}\right)n$$

vertices.

Let $k' = k - p$. We need again to distinguish several cases.

Subcase A: $k' \geqslant \max\{62w, 31w(n'/(n' - n/2))\}$. Then

$$k' \geqslant \max\left\{62w, 31w\frac{n'}{n' - n/2}\right\} \geqslant \max\left\{62w, 31w\frac{n'}{n' - \Delta(G') + 1}\right\}$$

and by Corollary 4, G' is equitably k'-colorable. By part (b) of Lemma 5, G has an equitable k-coloring.

Subcase B: $k' < \max\{62w, 31w(n'/(n' - n/2))\}$ and $k' < 62w$. Since $p \leqslant 4w$, we have that $k = k' + p < 66w$. Then by Theorem 6, the question whether G has an equitable k-coloring, can be solved in polynomial time.

Subcase C: $k' < \max\{62w, 31w(n'/(n' - n/2))\}$ and $k' \geqslant 62w$. Then

$$k' < 31w\frac{n'}{n' - (n/2)} < 31w\frac{n}{(1 - 4w/k)n - (n/2)} = \frac{31w}{\frac{1}{2} - (4w/k)}. \tag{1}$$

Using $k = k' + p \geqslant 62w$, we have that

$$\frac{4w}{k} \leqslant \frac{4w}{62w} = \frac{2}{31}$$

and

$$\frac{1}{2} - \frac{4w}{k} \geqslant \frac{27}{62}. \tag{2}$$

By (1) and (2),

$$k' < 31w\frac{62}{27} < 72w$$

and we conclude that $k = k' + p \leqslant 76w$. Again, by Theorem 6 the question if G has an equitable k-coloring, can be solved in polynomial time. This ends the analysis of Case 3, and the proof of the theorem. □

By Proposition 1, Theorem 7 implies directly that there is a polynomial time algorithm for the ℓ-BOUNDED k-COLORING problem restricted to graphs of bounded treewidth.

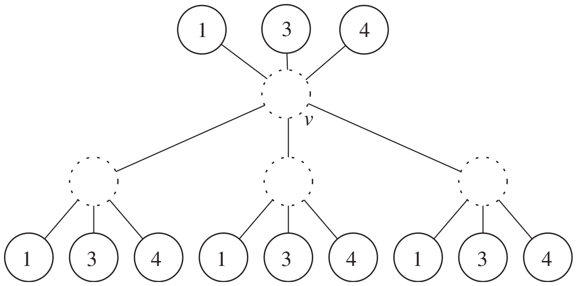

Fig. 1. Tree $G_{i,j}$. Here $a_j = 3$, $m = 4$, $i = 2$ and $N_2 = \{1, 3, 4\}$. In any 5-coloring of $G_{i,j}$, either v should be colored with 5 and all its non-precolored neighbors with 2, or v should be colored with 2 and the neighbors with 5.

4. Precolored equitable coloring

For a graph $G = (V, E)$, a *precoloring* π of a subset $V' \subset V$ in k colors is a mapping $\pi : V' \to \{1, 2, \ldots, k\}$. We say that a coloring $A_1, A_2, \ldots, A_k$ of G *extends* the precoloring π if $u \in A_{\pi(u)}$ for every $u \in V'$. We consider the following problem: PRECOLORED EQUITABLE k-COLORING: for a given graph G, integer k and a given precoloring π of G, determine whether there exists an equitable k-coloring of G extending π.

Theorem 8. PRECOLORED EQUITABLE k-COLORING *is NP-complete on trees.*

Proof. We use a reduction from the problem

3-PARTITION

Instance: A set A of non-negative integers $a_1, \ldots, a_{3m}$, and a bound B, such that for all i with $1 \leqslant i \leqslant 3m$, $(B+1)/4 < a_i < B/2$ and $\sum_{1 \leqslant i \leqslant 3m} a_i = mB$.

Question: Can A be partitioned into m disjoint sets $A_1, A_2, \ldots, A_m$ such that $\sum_{a_i \in A_j} a_i = B$ for every j with $1 \leqslant j \leqslant m$?

3-PARTITION is NP-complete in the strong sense (Problem SP15 in [13]).

Let the set $A = \{a_1, \ldots, a_{3m}\}$ and the bound B be an instance of 3-PARTITION. We construct a tree G and a precoloring of G such that G is equitably $(m+1)$-colorable if and only if A can be 3-partitioned.

For every $i \in \{1, 2, \ldots, m\}$ we define the set $N_i = \{1, 2, \ldots, m\} - \{i\}$ and the precolored star S_i as a star with one nonprecolored central vertex v adjacent to $m - 1$ leaves which are precolored with all colors from N_i. Thus, vertex v can be colored only with color i or $m+1$. For every $i \in \{1, 2, \ldots, m\}$ and $j \in \{1, 2, \ldots, 3m\}$, we define the precolored tree $G_{i,j}$ as a tree obtained by taking the disjoint union of $a_j + 1$ precolored stars S_i and by making the central vertex v of one of them to be adjacent to the central vertices of the other a_j stars. We call the vertex v the central vertex of $G_{i,j}$. Thus, $G_{i,j}$ has $m(a_j + 1)$ vertices; for every color $\ell \in N_i$ there are $(a_j + 1)$ vertices of $G_{i,j}$ precolored with ℓ. Every $(m+1)$-coloring of $G_{i,j}$ either colors v with $m+1$ and the remaining a_j non-precolored vertices with i, or it colors v with i and the remaining a_j non-precolored vertices with $m+1$. (See Fig. 1.)

For every $j \in \{1, 2, \ldots, 3m\}$, we define a precolored tree G_j as follows: we take the disjoint union of precolored trees $G_{i,j}$, $i \in \{1, 2, \ldots, m\}$, add one vertex c_j adjacent to all central vertices of trees $G_{i,j}$ and add one leaf adjacent to c_j precolored with $m+1$. Thus, G_j has $m^2(a_j + 1) + 2$ vertices; for every color $\ell \in \{1, 2, \ldots, m\}$ there are $(m-1)(a_j + 1)$ vertices of G_j precolored with ℓ and there is one vertex precolored with $m+1$. Also in any coloring of G_j, the vertex c_j cannot be colored with $m+1$. Thus, for every $(m+1)$-coloring of G_j the spectra of colors used on neighbors of c_j does not contain all colors from $\{1, 2, \ldots, m\}$.

Finally, G is obtained by taking the disjoint union of precolored trees $G_1, G_2, \ldots, G_{3m}$ and an independent set of cardinality $3m(m-2) + B$ precolored with color $m+1$, and then making this forest a tree by adding appropriate edges between precolored vertices of the same color in different trees. Thus, G has

$$\sum_{1 \leqslant j \leqslant 3m} m^2(a_j + 1) + 3(m-2) + B = m^2(mB + 3m) + 3(m-2) + B$$

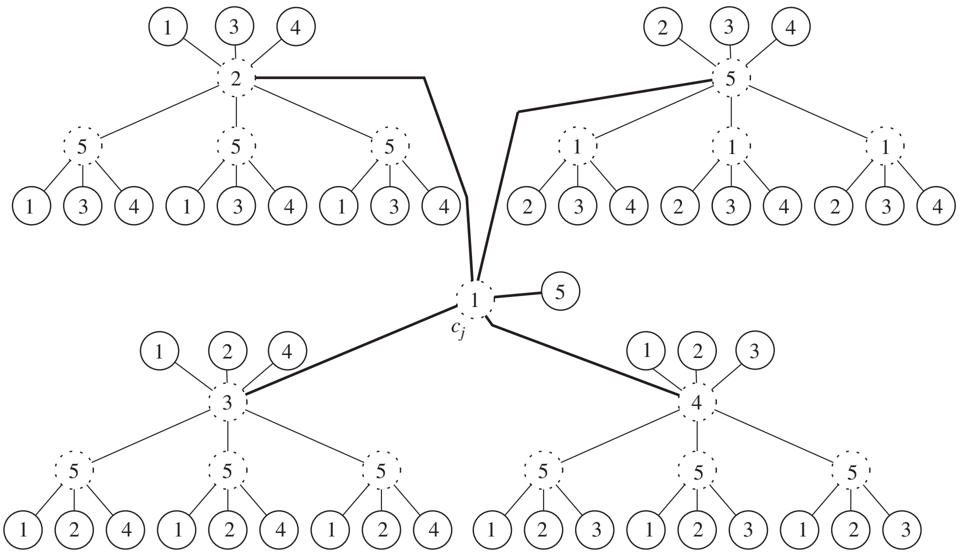

Fig. 2. Tree G_j. Here $a_j = 3, m = 4$. Vertices with solid circles are precolored; vertices with broken circles are non-precolored. The coloring shown displays the case that $a_j \in A_1$.

vertices. For every color $\ell \in \{1, 2, \ldots, m\} - \{i\}$ there are

$$\sum_{1 \leqslant j \leqslant 3m} (m-1)(a_j+1) = (m-1)(mB+3m)$$

vertices of G precolored with ℓ and $3m(m-1)+B$ vertices are precolored with $m+1$.

Claim 1. *A can be partitioned into m disjoint sets $A_1, A_2, \ldots, A_m$ such that $\sum_{a_i \in A_j} a_i = B$, if and only if G has an equitable $(m+1)$-coloring that extends the precoloring.*

Proof of Claim. Suppose that A can be partitioned into m disjoint sets $A_1, A_2, \ldots, A_m$ such that $\sum_{a_i \in A_j} a_i = B$. We define an extension of the precoloring of G as follows. For every fixed $j \in \{1, 2, \ldots, 3m\}$, we choose i such that $a_j \in A_i$. We color the central vertex of $G_{i,j}$ with color $m+1$ and the remaining uncolored a_j vertices of $G_{i,j}$ with color i. In each graph $G_{\ell,j}$, $\ell \neq i$, a_j vertices are colored with $m+1$ and one vertex with ℓ. Also we color vertex c_j with i. Thus in every graph G_j, color i is used a_j+1 times on the set of non-precolored vertices. Any color $\ell \in \{1, 2, \ldots, m\} - \{i\}$ is used once and color $m+1$ is used

$$a_j(m-1)+1$$

times on non-precolored vertices. Thus in graph G, the number of vertices colored with color $\ell \in \{1, 2, \ldots, m\}$ is

$$(m-1)(mB+3m) + \sum_{a_j \in A_\ell} (a_j+1) + \sum_{\{1 \leqslant j \leqslant 3m | a_j \notin A_\ell\}} 1 = (m-1)(mB+3m) + B + 3m.$$

The number of vertices colored with $m+1$ is

$$3m(m-2) + B + \sum_{1 \leqslant j \leqslant 3m} (a_j(m-1)+1) = (m-1)(mB+3m) + B + 3m$$

and we conclude that the obtained coloring is an equitable $(m+1)$-coloring.

An example is given in Fig. 2.

Suppose now that G has an equitable $(m+1)$-coloring that extends the given precoloring. The main observation here is that for every $j \in \{1, 2, \ldots, 3m\}$, at most $a_j(m-1)+1$ vertices of a graph G_j are colored with $m+1$. (Otherwise, the coloring of the central vertices of graphs $G_{i,j}$, $i \in \{1, 2, \ldots, m\}$, uses each of the colors from $\{1, 2, \ldots, m\}$, thus leaving no space for the color of c_j.) If for some $j \in \{1, 2, \dots, 3m\}$, fewer than $a_j(m-1)+1$ vertices of a graph G_j are colored with color $m+1$, then (the coloring is equitable) for some $j' \in \{1, 2, \dots, 3m\}$, at least $a_j m$ vertices of the graph $G_{j'}$ are colored with $m+1$, which is a contradiction.

Thus, we can conclude that for every $j \in \{1, 2, \dots, 3m\}$, there is exactly one subgraph $G_{i,j}$ such that a_j non-precolored vertices of $G_{i,j}$ are colored with i. For all other $i' \in \{1, 2, \dots, m\}$, $i \neq i'$, a_j non-precolored vertices of $G_{i',j}$ are colored with $m+1$. We define

$$A_i = \{a_j \mid a_j \text{ non-precolored vertices of } G_{i,j} \text{ are colored with } i\}.$$

In G, the number of vertices colored with color $i \in \{1, 2, \dots, m\}$ is

$$(m-1)(mB+3m) + B + 3m = (m-1)(mB+3m) + \sum_{a_j \in A_i} (a_j+1) + \sum_{\{1 \leqslant j \leqslant 3m \mid a_j \notin A_i\}} 1.$$

Thus, for every $i \in \{1, 2, \dots, m\}$

$$\sum_{a_j \in A_i} (a_j+1) = B$$

and $A_1, A_2, \dots, A_m$ is a 3-partition of A. □

So we have a polynomial reduction from 3-PARTITION to PRECOLORED EQUITABLE k-COLORING. PRECOLORED EQUITABLE k-COLORING trivially belongs to NP, we can conclude it is NP-complete. □

A direct corollary is that PRECOLORED ℓ-BOUNDED k-COLORING is NP-complete for trees.

We can also formulate the related PRECOLORED EQUITABLE COLORING problem. Here, we are given a precolored graph G, and ask whether there exists an integer k, and an equitable k-coloring of G that extends the precoloring. I.e., k is not part of the instance of the problem.

Proposition 9. PRECOLORED EQUITABLE COLORING *is NP-complete on trees.*

Proof. Clearly, the problem is in NP. We transform from PRECOLORED EQUITABLE k-COLORING on trees. Take a tree instance T of PRECOLORED EQUITABLE k-COLORING. We may assume, by the proof above, that for each color i, $1 \leqslant i \leqslant k$, T has at least one precolored vertex with color i. Suppose T has n vertices. We construct a new tree as follows. For each i, $1 \leqslant i \leqslant k$, we add $n+1$ new vertices, precolored with i. Each of the new vertices is made adjacent to a precolored vertex in T with a different color. We obtain a new tree T'. T' cannot have an equitable r coloring for some $r \neq k$. (An equitable coloring must use at least k colors. It cannot use more than k colors, as we have at most $n-k$ non-precolored vertices, and at least $n+2$ vertices are already precolored with color 1.) Also, T' has an equitable k-coloring if and only if T has an equitable k-coloring. (Use the same colors for the non-precolored vertices in both cases.) So, NP-hardness follows. □

References

[1] N. Alon, A note on the decomposition of graphs into isomorphic matchings, Acta Math. Hungar. 42 (1983) 221–223.

[2] B.S. Baker, E.G. Coffman Jr., Mutual exclusion scheduling, Theoret. Comput. Sci. 162 (1996) 225–243.

[3] J. Blazewicz, K.H. Ecker, E. Pesch, G. Schmidt, J. Weglarz, Scheduling Computer and Manufacturing Processes, second ed., Springer, Berlin, 2001.

[4] H.L. Bodlaender, A linear time algorithm for finding tree-decompositions of small treewidth, SIAM J. Comput. 25 (1996) 1305–1317.

[5] H.L. Bodlaender, K. Jansen, Restrictions of graph partition problems, I, Theoret. Comput. Sci. 148 (1995) 93–109.

[6] H.L. Bodlaender, T. Kloks, Efficient and constructive algorithms for the pathwidth and treewidth of graphs, J. Algorithms 21 (1996) 358–402.

[7] B. Bollobás, R.K. Guy, Equitable and proportional coloring of trees, J. Combin. Theory Ser. B 34 (1983) 177–186.

[8] G.J. Chang, D. Kuo, The $L(2,1)$-labeling problem on graphs, SIAM J. Discrete Math. 9 (1996) 309–316.

[9] B.-L. Chen, M.-T. Ko, K.-W. Lih, Equitable and m-bounded coloring of split graphs, in: Combinatorics and Computer Science, Lecture Notes in Computer Science, Vol. 1120, Springer, Berlin, 1996, pp. 1–5.

[10] B.-L. Chen, K.-W. Lih, Equitable coloring of trees, J. Combin. Theory Ser. B 61 (1994) 83–87.

[11] E. Cohen, M. Tarsi, NP-completeness of graph decomposition problems, J. Complexity 7 (1991) 200–212.
[12] M.R. Garey, R.L. Graham, D.S. Johnson, D.E. Knuth, Complexity results for bandwidth minimization, SIAM J. Appl. Math. 34 (1978) 477–495.
[13] M.R. Garey, D.S. Johnson, Computers and Intractability, A Guide to the Theory of NP-completeness, W.H. Freeman and Co, San Francisco, CA, 1979.
[14] A. Hajnal, E. Szemerédi, Proof of a conjecture of P. Erdős, Combinatorial Theory and its Applications, Vol. II, North-Holland, Amsterdam, 1970, pp. 601–623.
[15] P. Hansen, A. Hertz, J. Kuplinsky, Bounded vertex colorings of graphs, Discrete Math. 111 (1993) 305–312.
[16] S. Irani, V. Leung, Scheduling with conflicts, and applications to traffic signal control, in: Proc. Seventh Annu. ACM-SIAM Symp. on Discrete Algorithms (SODA'96), New York, ACM, New York, 1996, pp. 85–94.
[17] K. Jansen, The mutual exclusion scheduling problem for permutation and comparability graphs, Inform. Comput. 180 (2003) 71–81.
[18] M. Jarvis, B. Zhou, Bounded vertex coloring of trees, Discrete Math. 232 (2001) 145–151.
[19] D. Kaller, A. Gupta, T. Shermer, The χ_t-coloring problem, in: Symp. on Theoretical Aspects of Computer Science (STACS'95), Lecture Notes in Computer Science, Vol. 900, Springer, Berlin, 1995, pp. 409–420.
[20] F. Kitagawa, H. Ikeda, An existential problem of a weight-controlled subset and its application to school timetable construction, Discrete Math. 72 (1988) 195–211.
[21] D. Kobler, U. Rotics, Edge dominating set and colorings on graphs with fixed clique-width, Discrete Appl. Math. 126 (2003) 197–221.
[22] A.V. Kostochka, K. Nakprasit, S.V. Pemmaraju, On equitable coloring of d-generate graphs. SIAM J. Discrete Math. 19 (1) (2005) 83–95.
[23] J. Krarup, D. de Werra, Chromatic optimisation: limitations, objectives, uses, references, European J. Oper. Res. 11 (1982) 1–19.
[24] K.-W. Lih, The equitable coloring of graphs, Handbook of Combinatorial Optimization, Vol. 3, Kluwer Academic Publishers, Boston, MA, 1998, pp. 543–566.
[25] Z. Lonc, On complexity of some chain and antichain partition problems, in: Graph-theoretic Concepts in Computer Science (WG'91), Lecture Notes in Computer Science, Vol. 570, Springer, Berlin, 1992, pp. 97–104.
[26] B. Monien, The bandwidth minimization problem for caterpillars with hair length 3 is NP-complete, SIAM J. Algebraic Discrete Methods 7 (1986) 505–512.
[27] B.F. Smith, P.E. Bjørstad, W.D. Gropp, Domain Decomposition. Parallel Multilevel Methods for Elliptic Partial Differential Equations, Cambridge University Press, Cambridge, 1996.
[28] J. Fiala, P.A. Golovach, J. Kratochvi'l, Distance constrained labelings of graphs of bounded treewidth, in: Automata, Languages and Programming (ICALP 2005), Lecture Notes in Computer Science, Vol. 3580, Berlin, 2005, pp. 360–372.