Journal of Combinatorial Optimization (2020) 39:156–169
https://doi.org/10.1007/s10878-019-00461-7

Tree-coloring problems of bounded treewidth graphs

Bi Li[1] · Xin Zhang[1]

Published online: 19 October 2019

Abstract
This paper studies the parameterized complexity of the tree-coloring problem and equitable tree-coloring problem. Given a graph $G = (V, E)$ and an integer $r \geq 1$, we give an FPT algorithm to decide whether there is a tree-r-coloring of graph G when parameterized by treewidth. Moreover, we prove that to decide the existence of an equitable tree-r-coloring of graph G is W[1]-hard when parameterized by treewidth; and that it is polynomial solvable in the class of graphs with bounded treewidth.

Keywords Tree-coloring · Equitable tree-coloring · Nice tree decomposition · Bounded treewidth

1 Introduction

A *tree-coloring* of a graph G is a vertex coloring of G such that the subgraph induced by each color class is a forest. Given an integer $r \geq 1$, a *tree-r-coloring* of G is a tree-coloring of G with at most r colors. Moreover, an *equitable tree-r-coloring* of G is a tree-r-coloring of G such that the sizes of any two color classes differ by at most one. Note that this implies that equitable tree-r-coloring of any graph with at least r vertices contains exactly r colors. For any graph G, the minimum r such that G has an equitable tree-r-coloring is called the *equitable vertex arboricity* of G; and *the threshold of equitable vertex arboricity* of G, denoted as $va^*_{eq}(G)$, is the minimum r such that for any $r' \geq r$, G has an equitable tree-r'-coloring. We mainly consider the following two problems:

The first author is supported by the National Natural Science Foundation of China (Nos. 11701440, 11626181), the Natural Science Basic Research Plan in Shaanxi Province of China(No. 2017JQ1031), and the second author is supported by the National Natural Science Foundation of China (No. 11871055).

✉ Bi Li
libi@xidian.edu.cn

Xin Zhang
xzhang@xidian.edu.cn

1 School of Mathematics and statistics, Xidian University, Xi'an, China

TREE- COLORING PROBLEM (TCP)
Instance: A graph G and a positive integer r.
Question: Is there a tree-r-coloring of G?

EQUITABLE TREE-r-COLORING PROBLEM (ETCP)
Instance: A graph G and an integer r.
Question: Is there an equitable tree-r-coloring of graph G?

Related work The model of graph coloring has many practical applications. For example, it can be used in the case when a system with binary conflict relations needs to be divided into some equal size of conflict-free subsystems; also it was applied in solving scheduling problems (Meyer 1973; Kubale 1989; Furmanczyk 2006). In the theoretical viewpoint, as many coloring problems, both the TCP and ETCP are NP-Complete (Nakprasit and Nakprasit 2016). Kronk and Mitchem (1975) proved that every graph has a tree-r-coloring for some $r \leq \lceil \frac{\Delta(G)+1}{2} \rceil$, where $\Delta(G)$ is the maximum degree of the graph G. It is well known that every planar graph has a tree-3-coloring (Chartrand and Kronk 1969). Wu et al. (2013) proved that every planar graph with girth at least 5 has an equitable tree-3-coloring, while Esperet et al. (2015) showed that every planar graph has an equitable tree-r-coloring for every $r \geq 4$ (i.e., $va^*_{eq}(G) \leq 4$). Moreover, Chen et al. (2017) proved that any 5-degenerate graph G has an equitable tree-r-coloring for any $r \geq \lceil \frac{\Delta(G)+1}{2} \rceil$ (i.e., $va^*_{eq}(G) \leq \lceil \frac{\Delta(G)+1}{2} \rceil$). Many NP-Complete coloring problems have been studied in the parameterized complexity view (Fiala et al. 2011).

Main contributions In the rest of the paper, we prove that TREE- COLORING PROBLEM is *Fixed Parameter Tractable* (FPT) when parameterized by *treewidth* in Sect. 2. Moreover, the W[1]-hardness of the EQUITABLE TREE-r-COLORING PROBLEM when parameterized by treewidth is proved in Sect. 3; and for the positive side, it is proved that ETCP is polynomial solvable in the class of graphs with bounded treewidth or vertex cover number. Finally, we summarize our results and some remaining open questions in Sect. 4.

Preliminary Note that it is sufficient to consider connected graph. A *tree-decomposition* of a graph (Robertson and Seymour 1986) G is a way to represent G by a family of subsets of its vertex-set organized in a tree-like manner and satisfying some connectivity property. The *treewidth* of G measures the proximity of G to a tree. More formally, a tree-decomposition of $G = (V, E)$ is a pair $(T, \mathcal{X})$ where $\mathcal{X} = \{X_i | i \in V(T)\}$ is a family of subsets of V, called *bags*, and T is a tree, such that:

- $\bigcup_{i \in V(T)} X_i = V$;
- for any edge $uv \in E$, there is a bag X_i (for some node $i \in V(T)$) containing both u and v;
- for any vertex $v \in V$, the set $\{i \in V(T) | v \in X_i\}$ induces a subtree of T.

The *width* of a tree-decomposition $(T, \mathcal{X})$ is $max_{i \in V(T)} |X_i| - 1$ and its *size* is the order $|V(T)|$ of T. The treewidth of G, denoted by $tw(G)$, is the minimum width over all possible tree-decompositions of G.

It is well known that every graph of treewidth at most w has a so-called *nice tree decomposition* of width at most w, i.e. a tree decomposition with a rooted tree T, with root $l \in V(T)$ such that

- T is a binary tree.
- If a node $i \in V(T)$ 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 $i \in V(T)$ 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 and an integer $k > 0$, to decide whether there exists a tree decomposition of width at most k of G is FPT (Bodlaender et al. 2013). Given a tree decomposition of width at most k of G, a nice tree decomposition of the same width of G can be obtained in linear time.

A *k-tree* is the graph constructed recursively: a complete graph of $k + 1$ vertices is a k-tree; for any given k-tree G and a clique of size k in G, adding a new vertex adjacent to all vertices of this clique gives another k-tree. Any subgraph of a k-tree is called a *partial k-tree*. It is proved that any graph G is a partial k-tree if and only if $tw(G) \leq k$. More details can be found in Bodlaender (1998).

A proper r-coloring (the vertices of every color class induces an independent set) of the vertices of a graph G is called *acyclic* if every subgraph induced by vertices of any two color classes is acyclic, i.e., a forest (Grünbaum 1973). The minimum r such that G has an r-acyclic-coloring is called *acyclic chromatic number*, denoted as $\chi_a(G)$.

By induction, it is easy to prove that for any k-tree $G = (V, E)$, $\chi_a(G) \leq k + 1$: if $|V| = k + 1$, then each color class of the $k + 1$ colors contains exactly one vertex. Assume that any k-tree G with n vertices has a $k+1$-acyclic-coloring. Note that every vertex of a clique has different color. Adding a new vertex u adjacent to a clique K of size k in G, then color u with the only one left color different from the k colors of the k vertices in K. One can check that, this is a $k + 1$-acyclic-coloring satisfying the assumption. Thus $\chi_a(G) \leq tw(G) + 1$. Since Esperet et al. (2015) proved that $va^*_{eq}(G) \leq \chi_a(G) - 1$ for every graph G, we deduce the following

Theorem 1 *Given a graph G with at least one edge, we have* $va^*_{eq}(G) \leq tw(G)$. □

2 FPT algorithms for tree-coloring problem

In this section, we prove the following theorem.

Theorem 2 *The* Tree- coloring Problem *is FPT when parameterized by treewidth.*

For easier description, we give some notations, inspired from the notations in Bodlaender and Fomin (2005), we will see that adding connectivity (compare forest with independent set) in the standard dynamic programming makes the proof more complicated.

Given a graph $G = (V, E)$ of treewidth k, for any subset $S \subseteq V$, $G[S]$ denotes the induced subgraph of G. Let $(T, \mathcal{X})$ be a nice tree decomposition of G of width k. Without confusion, we identify any vertex in T with its corresponding bag in $\mathcal{X}$. Suppose that the root bag of T is $R \in \mathcal{X}$. For any bag $X \in \mathcal{X}$, let V_X be the vertex set of all vertices in bags X and its descendant in T; and G_X denotes the induced subgraph $G[V_X]$. For any set U, a partition $\mathcal{U}$ of U is a set of subsets of U such that the union of all subsets in $\mathcal{U}$ is U and any two of the subsets do not intersect.

For a bag $X \in \mathcal{X}$, let $\mathcal{F} = \{F_1, F_2, \ldots, F_f\}$ be a partition of X, where $1 \le f \le k$; and let $\mathcal{T} = \{\mathcal{T}_i | i = 1, 2, \ldots, f\}$, where for each $1 \le i \le f$, $\mathcal{T}_i = \{T_{i_1}, T_{i_2}, \ldots, T_{i_{t_i}}\}$ is a partition of F_i, $1 \le t_i \le k$. We call such $\mathcal{F}, \mathcal{T}$ as a *pair* of X. For every such pair of X, we compute a Boolean value $B_X[\mathcal{F}, \mathcal{T}]$. This Boolean value is TRUE if and only if there exists a-tree-r-coloring of G_X such that, for $1 \le i \le f$:

(I) every F_i is contained in one color class differently from each other. This implies that each $G[F_i]$ is an induced forest in G;
(II) in each subgraph induced by the color class of F_i, every T_{i_j}, $1 \le j \le t_i$, is contained in one connected component, i. e. a maximal induced subtree, differently from each other.

Lemma 3 *Let $X \in \mathcal{X}$ be a forget node in T and let Y be the unique child of X; and $X = Y - \{v\}$ for some vertex $v \in V_X$. Let $\mathcal{F} = \{F_1, F_2, \ldots, F_f\}$ be a partition of X, where $1 \le f \le \min\{k, r\}$; and let $\mathcal{T} = \{\mathcal{T}_i | i = 1, 2, \ldots, f\}$, where for each $1 \le i \le f$, $\mathcal{T}_i = \{T_{i_1}, T_{i_2}, \ldots, T_{i_{t_i}}\}$ is a partition of F_i, $1 \le t_i \le k$.*

Then $B_X[\mathcal{F}, \mathcal{T}] = TRUE$ if and only if there exist partition $\mathcal{F}' = \{F'_1, F'_2, \ldots, F'_{f'}\}$ of Y and $\mathcal{T}' = \{\mathcal{T}'_i$ is a partition of F'_i $| i = 1, 2, \ldots, f'\}$ such that:

(i) *$B_Y[\mathcal{F}', \mathcal{T}'] = TRUE$;*
(ii) *if $v \in F'_i$ and $F'_i = \{v\}$, then $\mathcal{F}' \backslash \{F'_i\} = \mathcal{F}$ and $\mathcal{T}' \backslash \{\mathcal{T}'_i\} = \mathcal{T}$;*
(iii) *if $v \in F'_i$ and $F'_i \supset \{v\}$, then there exists $F \in \mathcal{F}$, with the partition $\mathcal{T}_j \in \mathcal{T}$ of F, satisfying that $\mathcal{F}' \backslash \{F'_i\} = \mathcal{F} \backslash \{F\}$ and $\mathcal{T}' \backslash \{\mathcal{T}'_i\} = \mathcal{T} \backslash \{\mathcal{T}_j\}$; and that $F'_i - \{v\} = F$ and any $T' \in \mathcal{T}'_i$ satisfying one of the three cases: (1) T' is a union of some element of $\mathcal{T}_j$ and $\{v\}$; (2) $T' \in \mathcal{T}_j$; (3) $T' = \{v\}$.*

Proof If $B_X[\mathcal{F}, \mathcal{T}] = TRUE$, then let $c_X : V_X \to \{1, 2, \ldots, r\}$ be a tree-r-coloring of G_X satisfying the two properties (I) (II). Since $G_X = G_Y$, c_X is a tree-r-coloring of G_Y. According to the color classes, one gets the partition $\mathcal{F}' = \{F'_1, F'_2, \ldots, F'_{f'}\}$ of Y. In each subgraph induced by the color class of F'_h, $h = 1, 2, \ldots, f'$, according to the connected components intersecting with Y, one gets the partition $\mathcal{T}'_h$ of F'_h. Let $\mathcal{T}' = \{\mathcal{T}'_h | h = 1, 2, \ldots, f'\}$. One can check that $\mathcal{F}', \mathcal{T}'$ satisfy (i), (ii), (iii) in the lemma.

If there exist $\mathcal{F}', \mathcal{T}'$ satisfying (i), (ii), (iii) in the lemma, let $c_Y : V_Y \to \{1, 2, \ldots, r\}$ be a tree-r-coloring of G_Y satisfying the two properties (I) (II). Then c_Y is also a tree-r-coloring of G_X since $G_X = G_Y$. If $v \in F'_i$ and $F'_i = \{v\}$, then $\mathcal{F} \subset \mathcal{F}'$ and $\mathcal{T} \subset \mathcal{T}'$. So the tree-$r$-coloring c_Y of G_X satisfies (I) (II), i.e. $B_X[\mathcal{F}, \mathcal{T}] = TRUE$. Otherwise, $v \in F'_i$ and $F'_i \supset \{v\}$ as described in (iii) in the lemma. Then each $F_i \in \mathcal{F} \backslash \{F\} = \mathcal{F}' \backslash \{F'_i\}$ with its corresponding partition $\mathcal{T}_i \in \mathcal{T} \backslash \{\mathcal{T}_j\} = \mathcal{T}' \backslash \{\mathcal{T}'_i\}$ satisfies (I) (II). In the rest, we prove that $F = F'_i - \{v\}$ and its partition $\mathcal{T}_j$ satisfy (I) (II). Since $F \subset F'_i$, F is contained in the same color class with F'_i, differently from other sets in $\mathcal{F}$. For any $T \in \mathcal{T}_j$, it satisfies either $T \in \mathcal{T}'_i$ or $T = T' - \{v\}$ for some $T' \in \mathcal{T}'_i$. So in both cases, T is contained in one connected component of the subgraph induced by the color class of F, which is the same as the one of F'_i. The lemma is proved. □

Lemma 4 *Let $X \in \mathcal{X}$ be an introduce node in T and let Y be the unique child of X; and $X = Y \cup \{v\}$ for some vertex $v \in V_X$. Let $\mathcal{F} = \{F_1, F_2, \ldots, F_f\}$ be a partition of X, where $1 \leq f \leq \min\{k, r\}$; and let $\mathcal{T} = \{\mathcal{T}_i | i = 1, 2, \ldots, f\}$, where for each $1 \leq i \leq f$, $\mathcal{T}_i = \{T_{i_1}, T_{i_2}, \ldots, T_{i_{t_i}}\}$ is a partition of F_i, $1 \leq t_i \leq k$. Without loss of generality, assume that $v \in F_1$ and $v \in T_{1_1}$. Let the neighborhood of v in $G[F_1]$ be $N_1(v) = \{v_1, v_2, \ldots, v_d\}$, $0 \leq d \leq k$ and $d = 0$ when $N_1(v) = \emptyset$.*

- (i) *If $F_1 \neq \{v\}$, then $B_X[\mathcal{F}, \mathcal{T}] = TRUE$ if and only if for the partition $\mathcal{F}' = \{F_1 \backslash \{v\}, F_2, \ldots, F_f\}$ of Y, there exists a partition $\{T_1, T_2, \ldots, T_d\}$ of $T_{1_1} \backslash \{v\}$ such that (1) for each $1 \leq p \leq d$, $v_p \in T_p$; (2) and that $B_Y[\mathcal{F}', \mathcal{T}'] = TRUE$, where $\mathcal{T}' = \{\{T_1, T_2, \ldots, T_d, T_{1_2}, T_{1_3}, \ldots, T_{1_{t_1}}\}\} \cup \{\mathcal{T}_i | i = 2, 3, \ldots, f\}$.*
- (ii) *Otherwise, $B_X[\mathcal{F}, \mathcal{T}] = TRUE$ if and only if $B_Y[\{F_2, F_3, \ldots, F_f\}, \{\mathcal{T}_2, \mathcal{T}_3, \ldots, \mathcal{T}_f\}] = TRUE$.*

Proof (i) First we prove the case $F_1 \neq \{v\}$, which implies $F_1 \supset \{v\}$. If $B_X[\mathcal{F}, \mathcal{T}] = TRUE$, then let $c_X : V_X \to \{1, 2, \ldots, r\}$ be a tree-r-coloring of G_X satisfying the two properties (I) (II). Restrict c_X in V_Y. Then $c_X|_{V_Y}$ is a tree-r-coloring of $G_Y = G_X \backslash \{v\}$. According to the color classes, one gets the partition $\mathcal{F}' = \{F_1 \backslash \{v\}, F_2, \ldots, F_f\}$ of Y, since $Y = X - \{v\}$ and $v \in F_1$. In each subgraph induced by the color class of F_i, every T_{i_j}, $1 \leq j \leq t_i$, is contained in one connected component differently from each other. Since $v \in T_{1_1}$, we have $N_1(v) \subseteq T_{1_1}$. In $T_{1_1} \backslash \{v\}$, the connected component C_X^v, containing T_{1_1} in the subgraph induced by the color class of F_1 in G_X, is divided into d connected components in $C_X^v \backslash \{v\}$, each of which contains a subset of T_{1_1} and exactly one vertex of $N_1(v)$. This gives a partition of $T_{1_1} \backslash \{v\}$, put as $\{T_1, T_2, \ldots, T_d\}$ satisfying that $v_p \in T_p$ for each $1 \leq p \leq d$ and that T_p is contained in one connected component, differently from each other, of the subgraph induced by the color class of $F_1 \backslash \{v\}$ in G_Y. So $B_Y[\mathcal{F}', \mathcal{T}'] = TRUE$.

If there exists a partition $\{T_1, T_2, \ldots, T_d\}$ of $T_{1_1} \backslash \{v\}$ such that (1) $v_p \in T_p$ for each $1 \leq p \leq d$; (2) $B_Y[\mathcal{F}', \mathcal{T}'] = TRUE$, where $\mathcal{T}' = \{\{T_1, T_2, \ldots, T_d, T_{1_2}, T_{1_3}, \ldots, T_{1_{t_1}}\}\} \cup \{\mathcal{T}_i | i = 2, 3, \ldots, f\}$, let $c_Y : V_Y \to \{1, 2, \ldots, r\}$ be a tree-r-coloring of G_Y satisfying the two properties (I) (II) for the pair $\mathcal{F}', \mathcal{T}'$. Then define $c_X : V_X = V_Y \cup \{v\} \to \{1, 2, \ldots, r\}$ such that $c_X(u) = c_Y(u)$ for any $u \in V_Y$; and $c_X(v) = c_Y(w)$, where w is any vertex in $F_1 \backslash \{v\} \neq \emptyset$. So the tree-$r$-coloring c_X of G_X satisfies (I) (II), i.e. $B_X[\mathcal{F}, \mathcal{T}] = TRUE$.

(ii) If $F_1 = \{v\}$, then the necessariness follows directly from the proof above. In the following, we prove the sufficiency. Since $B_Y[\{F_2, F_3, \ldots, F_f\}, \{\mathcal{T}_2, \mathcal{T}_3, \ldots, \mathcal{T}_f\}] = TRUE$, let $c_Y : V_Y \to \{1, 2, \ldots, r\}$ be a tree-r-coloring of G_Y satisfying the two properties (I) (II) for the pair $\{F_2, F_3, \ldots, F_f\}, \{\mathcal{T}_2, \mathcal{T}_3, \ldots, \mathcal{T}_f\}$. Then define $c_X : V_X = V_Y \cup \{v\} \to \{1, 2, \ldots, r\}$ such that $c_X(u) = c_Y(u)$ for any $u \in V_Y$; and $c_X(v)$ is any color different from the colors of $F_2, F_3, \ldots, F_f$. Since $f \leq r$, we have that $f - 1 \leq r - 1$ and that $c_X(v)$ exists. The color class of v induces a forest in G_X, since $X \cap Y = Y$ is a separator between $\{v\}$ and $V_X \backslash Y$ and v is not adjacent to any vertex in $V_X \backslash Y$. So c_X is a tree-r-coloring of G_X satisfying (I) (II), i.e. $B_X[\mathcal{F}, \mathcal{T}] = TRUE$. □

In the following, we deal with the case that X is a join node in T. To be simple, we consider only the pair $\mathcal{F}, \mathcal{T}$ of X, whose Boolean value is possible true. The following fact tells the Boolean value of some pairs cannot be true.

Fact 1 *For any pair $\mathcal{F}, \mathcal{T}$ of X, if one of the following cases occurs, then $B[\mathcal{F}, \mathcal{T}] = FALSE$: (1) $|\mathcal{F}| > r$; (2) for some $F \in \mathcal{F}$, $G[F]$ is not a forest; (3) for some $F \in \mathcal{F}$, a maximal induced subtree in $G[F]$ is not contained in any $H \in \mathcal{T}_F$, where $\mathcal{T}_F \in \mathcal{T}$ is a partition of F.*

Define *proper pair* of X as the pair $\mathcal{F}, \mathcal{T}$ such that none of the three cases in Fact 1 occurs. So in any proper pair $\mathcal{F}, \mathcal{T}$ of X, for each $F \in \mathcal{F}$, a partition $\mathcal{T}_F \in \mathcal{T}$ of F, $G[F]$ is a forest; and any connected component of $G[F]$ is contained in some $H \in \mathcal{T}_F$, which also implies that each $H \in \mathcal{T}_F$ induces a forest in $G[F]$.

Suppose that vertex subset $U \subset V$ induces a forest in G. Let $\mathcal{U} = \{U_1, U_2, \ldots, U_t\}$ be the set of all vertex sets of the connected components of $G[U]$. Denote $\mathcal{L} = \{\mathcal{U}_1, \mathcal{U}_2, \ldots, \mathcal{U}_s\}$ as a partition of $\mathcal{U}$. We call such a partition $\mathcal{R}$ as a *join partition* of $\mathcal{L}$ with respect to U, if $\mathcal{R} = \{\{U_{i_1}, U_{i_2}, \ldots, U_{i_s}\}, \{U_{i_{s+1}}\}, \{U_{i_{s+2}}\}, \ldots, \{U_{i_t}\}\}$, where for each $j = 1, 2, \ldots, s$, $U_{i_j} \in \mathcal{U}_j$ and $\{U_{i_{s+1}}, U_{i_{s+2}}, \ldots, U_{i_t}\} = \mathcal{U} \backslash \{U_{i_1}, U_{i_2}, \ldots, U_{i_s}\}$. Join $\mathcal{L}$ and $\mathcal{R}$ to obtain another partition, denoted as $\mathcal{L} \uplus \mathcal{R}$, of $\mathcal{U}$ in the following way: in $\mathcal{L} \uplus \mathcal{R}$, if any two sets $L, R \in \mathcal{L} \cup \mathcal{R}$ intersects, then replace L, R with $L \cup R$ in $\mathcal{L} \uplus \mathcal{R}$; and repeat this until any two sets do not intersect. One sees that $\mathcal{L} \uplus \mathcal{R} = \{U\}$. This is because $\{U_{i_1}, U_{i_2}, \ldots, U_{i_s}\}$ intersects all sets in $\mathcal{L}$ and then their union gives U.

Given two proper pairs $\mathcal{F}, \mathcal{T}$ and $\mathcal{F}, \mathcal{T}'$ of X, we say that $\mathcal{T}'$ is smaller that $\mathcal{T}$, denoted as $\mathcal{T}' \prec \mathcal{T}$, if they satisfy that, for any $F \in \mathcal{F}$ and a partition $\mathcal{T}_F \in \mathcal{T}$ (resp. $\mathcal{T}'_F \in \mathcal{T}'$) of F and any $H \in \mathcal{T}_F$, there exist $T'_1, T'_2, \ldots, T'_{q_H} \in \mathcal{T}'_F$ such that $H = \cup_{i=1,2,\ldots,q_H} T'_i$. Without confusion, we also say that $\{T'_1, T'_2, \ldots, T'_{q_H}\}$ is a partition of H in $\mathcal{T}'$. Then the following claim holds.

Claim 4.1 *Given a proper pair $\mathcal{F}, \mathcal{T}$ of X, for any $F \in \mathcal{F}$ and a partition $\mathcal{T}_F \in \mathcal{T}$ of F and $H \in \mathcal{T}_F$, let $\mathcal{F}, \mathcal{T}'$ be a proper pair of X such that $\mathcal{T}' \prec \mathcal{T}$. Let $\mathcal{L}$ be a partition of H in $\mathcal{T}'$ and $\mathcal{R}$ be a join partition of $\mathcal{L}$ respect to H. Then $\mathcal{L} \uplus \mathcal{R} = \{H\}$.*

Definition 1 Given a proper pair $\mathcal{F}, \mathcal{T}$ of X, two proper pairs of X, $\mathcal{F}, \mathcal{T}'$ and $\mathcal{F}, \mathcal{T}''$ consist a join pair of $\mathcal{F}, \mathcal{T}$ if they satisfy that: (1) $\mathcal{T}' \prec \mathcal{T}$ and $\mathcal{T}'' \prec \mathcal{T}$; and that (2) for any $F \in \mathcal{F}$ and a partition $\mathcal{T}_F \in \mathcal{T}$ of F, for any $H \in \mathcal{T}_F$, let $\mathcal{L}$ (resp. $\mathcal{R}$) be the partition of H in $\mathcal{T}'$ (resp. $\mathcal{T}''$), then $\mathcal{R}$ is a join partition of $\mathcal{L}$ respect to H.

Then we can give the equivalent condition for $B_X[\mathcal{F}, \mathcal{T}] = TRUE$ for join node X in the following lemma.

Lemma 5 *Let $X \in \mathcal{X}$ be a join node in T and let Y, Z be the two children of X (Note that $X = Y = Z$). Then for any proper pair $\mathcal{F}, \mathcal{T}$ of X, $B_X[\mathcal{F}, \mathcal{T}] = TRUE$ if and only if there exist a join pair $\mathcal{F}, \mathcal{T}'$ and $\mathcal{F}, \mathcal{T}''$ of $\mathcal{F}, \mathcal{T}$, such that $B_Y[\mathcal{F}, \mathcal{T}'] = TRUE$ and $B_Z[\mathcal{F}, \mathcal{T}''] = TRUE$.*

Proof If $B_X[\mathcal{F}, \mathcal{T}] = TRUE$, let $c_X : V_X \to \{1, 2, \ldots, r\}$ be a tree-r-coloring of G_X satisfying the two properties (I) (II). Restrict c_X in V_Y. Then $c_X|_{V_Y}$ is a tree-r-coloring of $G_Y = G_X \backslash \{V_Z \backslash Z\}$. According to the color classes, one gets the partition

$\mathcal{F} = \{F_1, F_2, \ldots, F_f\}$ of $Y = X$. In each subgraph induced by the color class of F_i, according to the connected components intersecting with Y, one gets the partition $\mathcal{T}_i'$ of F_i. Let $\mathcal{T}' = \{\mathcal{T}_i' | i = 1, 2, \ldots, f\}$. One can check that $B_Y(\mathcal{F}, \mathcal{T}') = TRUE$. Note that $\mathcal{T}' \prec \mathcal{T}$. Similarly, restricting c_X in V_Z we get $\mathcal{T}'' \prec \mathcal{T}$ such that $B_Z(\mathcal{F}, \mathcal{T}') = TRUE$. One sees that $\mathcal{F}, \mathcal{T}'$ and $\mathcal{F}, \mathcal{T}''$ consist a join pair of $\mathcal{F}, \mathcal{T}$.

If there exist a join pair $\mathcal{F}, \mathcal{T}'$ and $\mathcal{F}, \mathcal{T}''$ of $\mathcal{F}, \mathcal{T}$, such that $B_Y[\mathcal{F}, \mathcal{T}'] = TRUE$ and $B_Z[\mathcal{F}, \mathcal{T}''] = TRUE$, then let $c_Y : V_Y \to \{1, 2, \ldots, r\}$ (resp. $c_Z : V_Z \to \{1, 2, \ldots, r\}$) be a tree-$r$-coloring of G_Z satisfying the two properties (I) (II). Without loss of generality, assume that $c_Y(v) = c_Z(v)$ for each $v \in X = Y = Z$. Then define $c_X : V_X = V_Y \cup V_Z \to \{1, 2, \ldots, r\}$ such that $c_X(y) = c_Y(y)$ for any $y \in V_Y$; and $c_X(z) = c_Z(z)$ for any $z \in V_Z \backslash Y$. Since $X, \mathcal{F}, \mathcal{T}'$ and $\mathcal{F}, \mathcal{T}''$ are a join pair of $\mathcal{F}, \mathcal{T}$, c_X is a tree-r-coloring of G_X with the pair $\mathcal{F}, \mathcal{T}$ satisfying the two properties (I) (II). So $B_X[\mathcal{F}, \mathcal{T}] = TRUE$. □

Algorithm 1: FPT Algorithm for TCP with parameter treewidth

Input: $G = (V, E)$, and a nice tree decomposition $(T, \mathcal{X})$ of G with width k; r colors denoted by $\{1, 2, \ldots, r\}$;
Output: $Answ$;

$Answ \leftarrow YES$;
for *a leaf bag* X *in* T **do**
  if *there exists no tree-*r*-coloring in the induced subgraph* $G[W]$, **then**
    Output NO
  else
    compute $B_X[\mathcal{F}, \mathcal{T}]$ for each proper pair $\mathcal{F}, \mathcal{T}$ of X;
    (Then we say the bag X is computed.)
for *a bag* X *in* T *with all its children bags computed* **do**
  if X *is a forget node* **then**
    compute $B_X[\mathcal{F}, \mathcal{T}]$ for each proper pair $\mathcal{F}, \mathcal{T}$ of X according to Lemma 3;
  if X *is an introduce node* **then**
    compute $B_X[\mathcal{F}, \mathcal{T}]$ for each proper pair $\mathcal{F}, \mathcal{T}$ of X according to Lemma 4;
  if X *is a join node* **then**
    compute $B_X[\mathcal{F}, \mathcal{T}]$ for each proper pair $\mathcal{F}, \mathcal{T}$ of X according to Lemma 5;
if $B_R[\mathcal{F}, \mathcal{T}] = NO$ *for each proper pair* $\mathcal{F}, \mathcal{T}$ *of the root bag* R **then**
  Output NO;
Output $Answ$.

[Proof of Theorem 2] From Lemmas 3–5, we see that Algorithm 1 outputs YES if and only if there exists a tree-r-coloring of the graph G. In the rest, we analyze the time complexity of Algorithm 1. There are $O(n)$ bags in $\mathcal{X}$. Each bag X has at most $k + 1$ vertices, so there are at most $f(k)$ proper pairs of X, for some function f. Then the time complexity of Algorithm 1 is $O(f^2(k)n)$. The theorem is proved. □

Note that in the above proof, $f(k) \leq B(k+1)$, which is the (k+1)-st *Bell number* counting the number of partitions of a set with k+1 elements. From Theorem 1 Algortihm 1 always outputs YES if $r \geq k$.

Since the treewidth of any graph is at most its vertex cover number, from the above theorem, the following corollary is true:

Corollary 6 *The* TREE- COLORING PROBLEM *is FPT when parameterized by the vertex cover number.*

3 Complexity of equitable-tree-coloring problem

3.1 W[1]-hardness of ETCP parameterized by treewdith

For the formal definition of the W-hierarchy (not needed in our proof) and related problems, please see in Downey and Thilikos (2011) for a survey. In this subsection, we prove the following result:

Theorem 7 *The Equitable Tree-Coloring Problem is W[1]-hard when parameterized by treewidth.*

Proof We do reduction from the following problem:

EQUITABLE COLORING PROBLEM (ECP)
Instance: A graph G with treewidth k and an integer r.
Parameter: $k+r$
Question: Is there an equitable coloring of graph G with at most r colors?

The above ECP is proved to be W[1]-hard in Fellows et al. (2011).

Given a graph $G = (V, E)$ with treewidth k and integer r, construct a graph H with $V(H) = V(G) \cup V(K_r)$ and $E(H) = E(G) \cup E(K_r) \cup \{uv | u \in V(G), v \in V(K_r)\}$, where K_r is a complete graph with r vertices. Then the treewidth of H is at most $k+r$. To prove the theorem, it is sufficient to show that there is an equitable coloring of G with at most r colors if and only if there is an equitable tree-r-coloring of H.

If there is an equitable coloring c of G with at most r colors, then in H we color $V(G)$ as c and color each vertex in $V(K_r)$ with one color differently. Then this is an equitable tree-r-coloring of H.

Now suppose that there is an equitable tree-r-coloring of H. In the following, we prove that there is an equitable coloring of G with at most r colors. Let c be an equitable tree-r-coloring of H, which maximizes the number of color classes of c inducing respectively independent sets in G.

Claim 7.1 $c|_{V(G)}$ *is a proper coloring of* G.

If each color class induces an independent set in G, then $c|_{V(G)}$ is a proper coloring of G. Otherwise, there is a color class V_i containing two vertices $u, v \in V(G)$ and $uv \in E(G)$. This implies that $V_i \cap V(K_r) = \emptyset$, since for any vertex $w \in V(K_r)$, $\{u, v, w\}$ induces a triangle. So $V(K_r)$ are colored by at most $r-1$ colors. Then there exists a color class $V_j = \{x, y\}$, where $x, y \in V(K_r)$, because that add any other vertex of $V(H)$ to V_j inducing a triangle. Since c is equitable and $|V_i| \geq 2$, $|V_j| = 2$, we have that $|V_i| = 2$ or 3. If $|V_i| = 2$, then $V_i = \{u, v\}$. Replacing the color class V_i, V_j by $V_i' = \{x, u\}$, $V_j' = \{y, v\}$ in c, we get an equitable tree-r-coloring of H, which has two more color classes inducing independent sets in G respectively. If $|V_i| = 3$, then let $V_i = \{u, v, z\}$, where $z \in V(G)$. Without loss of generality, suppose that $vz \notin E(G)$. Replacing the color class V_i, V_j by $V_i' = \{x, u\}$, $V_j' = \{y, v, z\}$ in c, we get an equitable tree-r-coloring of H, which has two more color classes inducing independent sets in G respectively.

- If each color class of c containing exactly one vertex of K_r, then $c|_{V(G)}$ is an equitable coloring of G with at most r colors.
- If there is a color class V_i containing no vertex of K_r, i.e. $V_i \cap V(K_r) = \emptyset$, then there exists a color class $V_j = \{x, y\}$, where $x, y \in V(K_r)$, as proved above; and replace V_i, V_j in c with V_i', V_j' such that $x \in V_i'$, $y \in V_j'$, $V_i' \cup V_j' = V_i \cup V_j$ and $|V_i'| = |V_i|$, $|V_j'| = |V_j|$. In this way, from c we can obtain an equitable tree-r-coloring c' of H such that each color class of c' contains at least one vertex of K_r. Since there are r vertices in K_r and r color classes in c', each color class contains exactly one vertex of K_r in c'. Then we are in the above case.
- Otherwise, there is a color class contains at least two vertices of K_r. Note that, this is equivalent to the above case, in which there is a color class containing no vertex of K_r, since there are exactly r color classes.

The theorem is proved. □

3.2 Polynomial time algorithm for ETCP in class of graphs with bounded treewidth

Consider the ETCP problem in this section. The basic idea is similar with the one for ECP in Bodlaender and Fomin (2005): the main point is to solve the case in which , the number of colors is bounded by some function of treewidth, maximum degree and the order (the number of vertices) of the graph, because of the result in Kostochka et al. (2005).

Theorem 8 [Kostochka et al. (2005)] *Every n-vertex d-degenerate graph G with maximum degree Δ is equitably r-colorable for any $r \geq \max\{62d, 31d(n/(n-\Delta+1))\}$.*

Every graph of treewidth at most d is d-degenerate and every equitable r-coloring of a graph is also an equitable tree-r-coloring, so Theorem 8 implies the following corollary.

Corollary 9 *Every n-vertex graph of treewidth k with maximum degree Δ is equitably tree-r-colorable for any $r \geq \max\{62k, 31k(n/(n-\Delta+1))\}$.*

In Bodlaender and Fomin (2005), to solve the case with 'smaller' (bounded by some function) r, for any given subset S of vertices, they found several equitable independent sets of each size $\lceil \frac{n}{r} \rceil$ or $\lceil \frac{n}{r} \rceil - 1$ to cover S. They proved that it can be done in polynomial time in the class of graphs with bounded treewidth if $|S|$ or r is bounded, by applying the dynamic programming based on a nice tree decomposition of the graph. Instead, we will find several equitable forests of each size $\lceil \frac{n}{r} \rceil$ or $\lceil \frac{n}{r} \rceil - 1$ to cover S, which is more difficult because that some connectivity needs to be considered comparing with the independent sets. To be easier understood, we give the details in the following.

Let $S \subseteq V$ be a set of vertices of a graph $G = (V, E)$. We say that S can be *covered by forests of size* $\lceil \frac{n}{r} \rceil$ *or* $\lceil \frac{n}{r} \rceil - 1$ if there is a set of subsets $A_i \subseteq V, i \in \{1, 2, \ldots, p\}$, $p \leq |S|$, such that

(1) For every $i \in \{1, 2, \ldots, p\}$, A_i induces a forest in G;
(2) For every $i, j \in \{1, 2, \ldots, p\}$,$i \neq j$, $A_i \cap A_j = \emptyset$;
(3) For every $i \in \{1, 2, \ldots, p\}$, $|A_i| = \lceil \frac{n}{r} \rceil$ or $\lceil \frac{n}{r} \rceil - 1$;
(4) $S \subseteq \cup_{1 \leq i \leq p} A_i$.

Covering by forests is a natural generalization of an equitable tree-coloring: a graph G has an equitable r-tree-coloring if and only if V can be covered by forests size $\lceil \frac{n}{r} \rceil$ or $\lceil \frac{n}{r} \rceil - 1$. As seen in Sect. 2, we find at most r forests 'covering' V without restricting the size of each forest. Details for restricting the size of each forest will be given later.

The following lemma will be used in the proof.

Lemma 10 *Let $S \subseteq V$ be a vertex subset of a graph G.*

(a) *If S cannot be covered by forests of size $\lceil \frac{n}{r} \rceil$ or $\lceil \frac{n}{r} \rceil - 1$, then graph G is not equitably r-colorable.*
(b) *If S can be covered by p forests $A_1, \ldots, A_p$ of size $\lceil \frac{n}{r} \rceil$ or $\lceil \frac{n}{r} \rceil - 1$ and the graph $G' = G[V \setminus \cup_{1 \leq i \leq p} A_i]$ is equitably $(r - p)$-colorable, the graph G is equitably r-colorable.*

Proof (a) Let $B_1, \ldots, B_r$ be the color classes of an equitable tree-r-coloring of G. Consider the collection of sets $\{B_i | 1 \leq i \leq r, B_i \cap S \neq \emptyset\}$. Then S is covered by this collection of forests of size $\lceil \frac{n}{r} \rceil$ or $\lceil \frac{n}{r} \rceil - 1$. It is a contradiction.

(b) Use color classes $A_1, \ldots, A_p$, and partition the vertices of G' as in the equitable tree-$(r - p)$-coloring into color classes $A_{p+1}, \ldots, A_r$. This gives an equitable tree-r-coloring of G. □

Theorem 11 *Let k be a constant. Let $G = (V, E)$ be an n-vertex graph of treewidth at most k, let S be a subset of V, and let r be an integer. When r or $|S|$ is bounded by a constant, one can either find in polynomial time a covering of S by forests of size $\lceil \frac{n}{r} \rceil$ or $\lceil \frac{n}{r} \rceil - 1$, or conclude that there is no such a covering.*

In Sect. 2, we describe a polynomial algorithm, which either finds a collection of $\leq r$ forests covering V, or concludes that there is no such covering. Note that the size of each forest is not precise. To prove Theorem 11, we need to record the size of each forest in the covering, additionally. The proof will be given after the main result of this section described in the following theorem.

Theorem 12 *The Equitable Tree-Coloring Problem is polynomial solvable in the class of graphs of bounded treewidth.*

Proof The proof is similar with the one in Bodlaender and Fomin (2005). It is put here for the convince of the readers.

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

Case 1 $\Delta \le n/2 + 1$ and $r \ge 62k$. Since $\max_{0\le\Delta\le n/2+1} \frac{n}{n-\Delta+1} = 2$, we have that

$$r \ge 62k = \max\{62k, 2 \cdot 31k\} \ge \max\left\{62k, 31k\frac{n}{n-\Delta+1}\right\}$$

and by Corollary 9, G is equitably tree-r-colorable.

Case 2 $\Delta \le n/2 + 1$ and $r \le 62k$. In this case, it follows from Theorem 11 that the question whether G has an equitable tree-r-coloring can be solved in polynomial time.

Case 3 $\Delta > n/2 + 1$. Let $S \subseteq V$ be the set of vertices in G of degree at least $n/2 + 2$. Since the treewidth of G is k, G has at most kn edges. So $|S| \le 4k$. Thus, by Theorem 11, it can be checked in polynomial time whether S can be covered by forests of size $\lceil \frac{n}{r} \rceil$ or $\lceil \frac{n}{r} \rceil - 1$. If S cannot be covered, then by part (a) of Lemma 10, G has no equitable tree-r-coloring. Let $A_i \subseteq V$, $i \in \{1, 2, \ldots, p\}$, $p \le |S|$, be a covering of S by forests of size $\lceil \frac{n}{r} \rceil$ or $\lceil \frac{n}{r} \rceil - 1$. We define a new graph $G' = G[V \setminus \cup_{1\le i\le p} A_i]$. The maximum vertex degree Δ' in G' is at most $n/2 + 1$ and the treewidth of G' is at most k. Graph G' has

$$n' = |V \setminus \cup_{1\le i\le p} A_i| \ge n - p\left(\lceil \frac{n}{r} \rceil - 1\right) \ge n - 4k\left(\frac{n}{r} - 1\right) > \left(1 - \frac{4k}{r}\right)n$$

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

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

$$r' \ge \max\left\{62k, 31k\frac{n'}{n' - n/2}\right\} \ge \max\left\{62k, 31k\frac{n'}{n' - \Delta' + 1}\right\}$$

and by Corollary 9, G' is equitably tree-r'-colorable. By part (b) of Lemma 10, G has an equitable tree-r-coloring.

Subcase B $r' < \max\{62k, 31k(n'/(n' - n/2))\}$ and $r' < 62k$. Since $p \le 4k$, we have that $r = r' + p < 66k$. Then by Theorem 11, the question whether G has an equitable tree-r-coloring, can be solved in polynomial time.

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

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

Using $r = r' + p \ge 62k$, we have that $4k/r \le 4k/62k = 2/31$ and $1/2 - 4k/r \ge 27/62$. So $r' < 72k$ and we conclude that $r = r' + p \le 76k$. Again, by Theorem 11 the question if G has an equitable tree-r-coloring, can be solved in polynomial time. This ends the analysis of Case 3, and the proof of the theorem.

□

Now we give the proof of Theorem 11, which plays an important role in the above proof. This is also the main difference of our algorithm for ETCP with the algorithm for ECP in Bodlaender and Fomin (2005).

[Proof of Theorem 11] We assume that $\min\{|S|, r\} = c$ for some constant c. We now want to check if S can be covered by at most c forests of each size $\lceil \frac{n}{r} \rceil$ or $\lceil \frac{n}{r} \rceil - 1$. Recall some notations in Sect. 2: let $(T, \mathcal{X})$ be a nice tree decomposition of G with k. Without confusion, we identify any vertex in T with its corresponding bag in $\mathcal{X}$. Suppose that the root bag of T is $R \in \mathcal{X}$. For any bag $X \in \mathcal{X}$, let V_X be the vertex set of all vertices in bags X and its descendant in T; and G_X denotes the induced subgraph $G[V_X]$. We use a dynamic programming algorithm, where we compute for each node $X \in \mathcal{X}$ in the tree decomposition a table of *triples*, with each state associated to a Boolean value.

For a bag $X \in \mathcal{X}$, let $\mathcal{F} = \{F_1, F_2, \ldots, F_c\}$ be a partition of X, where F_i can be empty for any $1 \leq i \leq c$; and let $\mathcal{T} = \{\mathcal{T}_i | i = 1, 2, \ldots, c\}$, where for each $1 \leq i \leq c$, $\mathcal{T}_i = \{T_{i_1}, T_{i_2}, \ldots, T_{i_{t_i}}\}$ is a partition of F_i, $1 \leq t_i \leq k$. Moreover, let α_{i_j} be integers from 0 to $\lceil \frac{n}{r} \rceil$, $1 \leq i \leq c$ and $1 \leq j \leq t_i + 1$. For convenience, put α as a vector consisting of all these α_{i_j}. We call such $\mathcal{F}, \mathcal{T}, \alpha$ as a *triple* of X. Recall that $B(k+1)$ denotes the (k+1)-st *Bell number*. Thus there are at most $B^2(k+1)\lceil \frac{n}{r} \rceil^{c(k+1)}$ triples for each X. For every such triple of X, we compute a Boolean value $B_X[\mathcal{F}, \mathcal{T}, \alpha]$. This Boolean value is TRUE if and only if there is a set of subsets $A_i \subseteq V_X$, $i \in \{1, 2, \ldots, p\}$, $p \leq c$, such that:

- (I') For every $i \in \{1, 2, \ldots, p\}$, A_i induces a forest in G;
- (II') For every $j, l \in \{1, 2, \ldots, p\}$, $j \neq l$, $A_j \cap A_l \neq \emptyset$;
- (III') $S \cap V_X \subseteq \cup_{1 \leq i \leq p} A_i$;
- (IV') For every $i \in \{1, 2, \ldots, p\}$, $A_i \cap X = F_i$;
- (V') In each subgraph induced by A_i, every T_{i_j}, $1 \leq j \leq t_i$, is contained in one connected component C_{i_j}, i. e. a maximal induced subtree, differently from each other. Furthermore, for every $i \in \{1, 2, \ldots, p\}$ and $1 \leq j \leq t_i$, $|C_{i_j}| = \alpha_{i_j}$, i.e. the connected component containing T_{i_j} contains α_{i_j} vertices; while, $\alpha_{i_{t_i+1}}$ is the sum number of vertices of all connected components in $G[A_i]$ not intersecting with X.

Clearly, S can be covered by forests of each size $\lceil \frac{n}{r} \rceil$ or $\lceil \frac{n}{r} \rceil - 1$ if and only if $B_R[\mathcal{F}, \mathcal{T}, \alpha] = TRUE$ for some triple of R satisfying for every $1 \leq i \leq p$, $\sum_{1 \leq j \leq t_i + 1} \alpha_{i_j} = \lceil \frac{n}{r} \rceil$ or $\lceil \frac{n}{r} \rceil - 1$. Now, in bottom-up order, we compute for each bag $X \in \mathcal{X}$, for all its triples their Boolean values as shown in Sect. 2, but pay additionally attention to the vector α, which records the size of the subtrees of the forests. A detailed, but not difficult, argument shows that one can compute for join, introduce, and forget bags its triple values in $O(B^2(k+1)\lceil \frac{n}{r} \rceil^{2c(k+1)})$ time, when given all triple values for the children of the bag. Thus, with $O(nB^2(k+1)\lceil \frac{n}{r} \rceil^{2c(k+1)})$ time, we can compute the triple values of the root R, and hence decide if there the desired covering exists.

Table 1 Parameterize complexity of the Tree- coloring Problem and Equitable tree-r-coloring Problem. "?" denotes that the Equitable tree-r-coloring Problem parameterized by vertex cover number is not known to be FPT or W[1]-hard, even though this problem is polynomial solvable in graphs with bounded vertex cover number

	vetex cover number	treewidth
Tree- coloring Problem	FPT	FPT
Equitable tree-r-coloring Problem	?	W[1]-hard

Finally, using additional bookkeeping one can also solve the construction variant of the problem and find, if existing, the covering of S by forests of each size $\lceil \frac{n}{r} \rceil$ or $\lceil \frac{n}{r} \rceil - 1$. □

Since the treewidth of any graph is at most its vertex cover number, from the above theorem, the following corollary is true:

Corollary 13 *The Equitable Tree-Coloring Problem is polynomial solvable in the class of graphs of bounded vertex cover number.*

4 Conclusion

In this article, we gave some parameterize complexity of the Tree- coloring Problem and Equitable tree-r-coloring Problem. Table 1 summarizes our results as well as the remaining open questions.

Besides, we look back at Theorem 1, which immediately imply the following

Theorem 14 $va^*_{eq}(G) \leq k$ *for every graph* G *with treewidth at most* k.

Now we claim that the upper bound for $va^*_{eq}(G)$ in the above theorem is sharp. Let H be a complete graph on k vertices, and let S be a set of independent vertices that are adjacent to every vertex of H. By G, we denote the resulting graph, which is clearly a k-tree, and thus has treewidth at most k. We next prove that G admits no equitable tree-$(k-1)$-colorings, and thus the upper bound k for $va^*_{eq}(G)$ in Theorem 14 cannot be improved. Suppose otherwise that c is an equitable tree-$(k-1)$-coloring of G. Since there are $k-1$ colors in c and H has k vertices, at least two vertices of H are monochromatic. But every color class in c induces a forest, so every color in c appears on H at most twice. Therefore, there is a color in c appearing on H exactly twice, and moreover, this color cannot be used by any vertex in S (otherwise a monochromatic triangle occurs). This gives that there is a color in c that are used by G exactly twice. Since c is equitable, every color in c appears on G at most three times, which implies that the number of colored vertices is at most $2 + 3(k-2) = 3k-4$. Choose $|S|$ to be at least $2k-3$, we conclude a contradiction.

A graph G is k-degenerate if every subgraph of G has minimum degree at most k. Clearly, a graph with treewidth at most k is k-degenerate. Therefore, it is interesting to ask whether Theorem 14 can be generalized to k-degenerate graphs. To end this paper, we leave a conjecture.

Conjecture 15 *$va^*_{eq}(G) \leq k$ for every k-degenerate graph G.*

Conjecture 15 holds for all 2-degenerate graphs G, which is an almost trivial result, since it is easy to see that $\chi_a(G) \leq 3$ and thus $va^*_{eq}(G) \leq \chi_a(G) - 1 \leq 2$. To our knowledge, for k-degenerate graphs G, the best known upper bound for $va^*_{eq}(G)$ is 3^{k-1}, see (Esperet et al. 2015, Theorem 9). Therefore, finding two constant c and t such that $va^*_{eq}(G) \leq ck^t$ for every k-degenerate graph G is also a problem that can be considered further, instead of proving Conjecture 15 directly.

References

Bodlaender HL (1998) A partial k-arboretum of graphs with bounded treewidth. Theor Comput Sci 209(1–2):1–45

Bodlaender HL, Drange P, Dregi M, Fomin F, Lokshtanov D, Pilipczuk M (2013) An $o(c^k n)$ 5-approximation algorithm for treewidth. In: 2013 IEEE 54th annual symposium on foundations of computer science (FOCS), pp 499–508

Bodlaender HL, Fomin FV (2005) Equitable colorings of bounded treewidth graphs. Theor Comput Sci 349(1):22–30

Chartrand G, Kronk HV (1969) The point-arboricity of planar graphs. J Lond Math Soc 1:612–616

Chen G, Gao Y, Shan S, Wang G, Wu J (2017) Equitable vertex arboricity of 5-degenerate graphs. J Comb Optim 34(2):426–432

Downey RG, Thilikos DM (2011) Confronting intractability via parameters. Comput Sci Rev 5(4):279–317

Esperet L, Lemoine L, Fre M (2015) Equitable partition of graphs into induced forests. Discrete Math 338(8):1481–1493

Fellows MR, Fomin FV, Lokshtanov D, Rosamond F, Saurabh S, Szeider S, Thomassen C (2011) On the complexity of some colorful problems parameterized by treewidth. Inf Comput 209(2):143–153

Fiala J, Golovach PA, Kratochvíl J (2011) Parameterized complexity of coloring problems: treewidth versus vertex cover. Theor Comput Sci 412(23):2513–2523

Furmanczyk H (2006) Equitable coloring of graph products. Opusc Math 26(1):31–44

Grünbaum B (1973) Acyclic colorings of planar graphs. Israel J Math 14(4):390–408

Kostochka AV, Nakprasit K, Pemmaraju SV (2005) On equitable coloring of d-degenerate graphs. SIAM J Discrete Math 19(1):83–95

Kronk H, Mitchem J (1975) Critical point-arboritic graphs. J Lond Math Soc 2–9(3):459–466

Kubale M (1989) Interval vertex-coloring of a graph with forbidden colors. Discrete Math 74(1):125–136 (**Special double issue**)

Nakprasit KM, Nakprasit K (2016) Complexity of equitable tree-coloring problems. ArXiv e-prints

Meyer W (1973) Equitable coloring. Am Math Mon 80(8):920–922

Robertson N, Seymour PD (1986) Graph minors. II. Algorithmic aspects of tree-width. J Algorithms 7(3):309–322

Wu J-L, Zhang X, Li H (2013) Equitable vertex arboricity of graphs. Discrete Math 313(23):2696–2701