Discrete Applied Mathematics 246 (2018) 38–48

Contents lists available at ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

On the complexity of rainbow coloring problems ☆

Eduard Eiben [a], Robert Ganian [a], Juho Lauri [b,*]

[a] TU Wien, Vienna, Austria

[b] Tampere University of Technology, Tampere, Finland

ARTICLE INFO

Article history:
Received 25 January 2016
Received in revised form 16 August 2016
Accepted 17 October 2016
Available online 16 November 2016

Keywords:
Rainbow coloring
Rainbow connection number
Graph algorithms
Treewidth

ABSTRACT

An edge-colored graph G is said to be *rainbow connected* if between each pair of vertices there exists a path which uses each color at most once. The *rainbow connection number*, denoted by $\mathrm{rc}(G)$, is the minimum number of colors needed to make G rainbow connected. Along with its variants, which consider vertex colorings and/or so-called strong colorings, the rainbow connection number has been studied from both the algorithmic and graph-theoretic points of view.

In this paper we present a range of new results on the computational complexity of computing the four major variants of the rainbow connection number. In particular, we prove that the Strong Rainbow Vertex Coloring problem is NP-complete even on graphs of diameter 3, and also when the number of colors is restricted to 2. On the other hand, we show that if the number of colors is fixed then all of the considered problems can be solved in linear time on graphs of bounded treewidth. Moreover, we provide a linear-time algorithm which decides whether it is possible to obtain a rainbow coloring by saving a fixed number of colors from a trivial upper bound. Finally, we give a linear-time algorithm for computing the exact rainbow connection numbers for three variants of the problem on graphs of bounded vertex cover number.

1. Introduction

The concept of rainbow connectivity was introduced by Chartrand, Johns, McKeon, and Zhang in 2008 [8] as an interesting connectivity measure motivated by recent developments in the area of secure data transfer. Over the past years, this strengthened notion of connectivity has received a significant amount of attention in the research community. The applications of rainbow connectivity are discussed in detail for instance in the recent survey [25], and various bounds are also available in [10,26].

An edge-colored graph G is said to be *rainbow connected* if between each pair of vertices a, b there exists an $a - b$ path which uses each color at most once; such a path is called *rainbow*. The minimum number of colors needed to make G rainbow connected is called the *rainbow connection number* (rc), and the Rainbow Coloring problem asks to decide if the rainbow connection number is upper-bounded by a number specified in the input. Precise definitions are given in Section 2.

The rainbow connection number and Rainbow Coloring have been studied from both the algorithmic and graph-theoretic points of view. On one hand, the exact rainbow connection numbers are known for a variety of simple graph

☆ A shortened version of this paper has appeared in the proceedings of the 26th International Workshop on Combinatorial Algorithms (IWOCA) (Eiben et al., 2015) [17].

* Corresponding author.
E-mail address: juho.lauri@tut.fi (J. Lauri).

http://dx.doi.org/10.1016/j.dam.2016.10.021

classes, such as wheel graphs [8], complete multipartite graphs [8], unit interval graphs [29], and threshold graphs [6]. On the other hand, Rainbow Coloring is a notoriously hard problem. It was shown by Chakraborty et al. [5] that already deciding if $\mathrm{rc}(G) \leq 2$ is NP-complete, and Ananth et al. [1] showed that for any $k > 2$ deciding $\mathrm{rc}(G) \leq k$ is NP-complete. In fact, Chandran and Rajendraprasad [6] strengthened this result to hold for chordal graphs. In the same paper, the authors gave a linear time algorithm for rainbow coloring split graphs which form a subclass of chordal graphs with at most one more color than the optimum. Basavaraju et al. [2] gave an $(r+3)$-factor approximation algorithm to rainbow color a general graph of radius r. Later on, the inapproximability of the problem was investigated by Chandran and Rajendraprasad [7]. They proved that there is no polynomial time algorithm to rainbow color graphs with less than twice the minimum number of colors, unless P = NP. For chordal graphs, they gave a 5/2-factor approximation algorithm, and proved that it is impossible to do better than 5/4, unless P = NP.

Several variants of the notion of rainbow connectivity have also been considered. Indeed, a similar concept was introduced for vertex-colored graphs by Krivelevich and Yuster [22]. A vertex-colored graph H is *rainbow vertex connected* if there is a path whose internal vertices have distinct colors between every pair of vertices, and this gives rise to the *rainbow vertex connection number* (rvc). The *strong rainbow connection number* (src) was introduced and investigated also by Chartrand et al. [10]; an edge-colored graph G is said to be *strong rainbow connected* if between each pair of vertices a, b there exists a shortest $a-b$ path which is rainbow. The combination of these two notions, *strong rainbow vertex connectivity* (srvc), was studied in a graph theoretic setting by Li et al. [24].

Not surprisingly, the problems arising from the strong and vertex variants of rainbow connectivity are also hard. Chartrand et al. showed that $\mathrm{rc}(G) = 2$ if and only if $\mathrm{src}(G) = 2$ [8], and hence deciding if $\mathrm{src}(G) \leq k$ is NP-complete for $k = 2$. The problem remains NP-complete for $k > 2$ for bipartite graphs [1], and also for split graphs [21]. Furthermore, the strong rainbow connection number of an n-vertex bipartite graph cannot be approximated within a factor of $n^{1/2-\epsilon}$, where $\epsilon > 0$ unless NP = ZPP [1], and the same holds for split graphs [21]. The computational aspects of the rainbow vertex connection numbers have received less attention in the literature. Through the work of Chen et al. [12] and Chen et al. [11], it is known that deciding if $\mathrm{rvc}(G) \leq k$ is NP-complete for every $k \geq 2$. However, to the best of our knowledge, the complexity of deciding whether $\mathrm{srvc}(G) \leq k$ (the k-SRVC problem) has not been previously considered.

In this paper, we present new positive and negative results for all four variants of the rainbow coloring problems discussed above.

- In Section 3, we prove that k-SRVC is NP-complete for every $k \geq 3$ even on graphs of diameter 3. Our reduction relies on an intermediate step which proves the NP-hardness of a more general problem, the k-Subset Strong Rainbow Vertex Coloring problem. We also provide bounds for approximation algorithms (under established complexity assumptions), see Corollary 6, and tighten the hardness result to additionally cover 2-SRVC.
- In Section 4, we show that all of the considered problems can be formulated in monadic second order (MSO) logic. In particular, this implies that for every fixed k, all of the considered problems can be solved in linear time on graphs of bounded treewidth, and the vertex variants can be solved in cubic time on graphs of bounded clique-width.
- In Section 5, we investigate the problem from a different perspective: we ask whether, given an n-vertex graph G and an integer k, it is possible to color G using k colors less than the known upper bound. Here we employ a win–win approach and show that this problem can be solved in time $\mathcal{O}(n)$ for any fixed k.
- In Section 6, we show that in the general case when k is not fixed, three of the considered problems admit linear-time algorithms on graphs of bounded vertex cover number. This is also achieved by exploiting a win–win approach, where we show that either k is bounded by a function of the vertex cover number and hence we can apply the result of Section 4, or k is sufficiently large which allows us to exploit the structure of the graph and solve the problem directly.

A shortened version of this paper has appeared in the proceedings of the 26th International Workshop on Combinatorial Algorithms (IWOCA) [17].

2. Preliminaries

2.1. Graphs and rainbow connectivity

We refer to [15] for standard graph-theoretic notions. We use $[i]$ to denote the set $\{1, 2, \ldots, i\}$. All graphs considered in this paper are simple and undirected. The *degree* of a vertex is the number of its incident edges, and a vertex is a *pendant* if it has degree 1. We will often use the shorthand ab for the edge $\{a, b\}$. For a vertex set X, we use $G[X]$ to denote the subgraph of G induced on X.

A *vertex coloring* of a graph $G = (V, E)$ is a mapping from V to $\mathbb{N}$, and similarly an *edge coloring* of G is a mapping from E to $\mathbb{N}$; in this context, we will often refer to the elements of $\mathbb{N}$ as colors. An *$a-b$ path P of length p* is a finite sequence of the form $(a = v_0, e_0, v_1, e_1, \ldots, b = v_p)$, where $v_0, v_1, \ldots, v_p$ are distinct vertices and $e_0, \ldots, e_{p-1}$ are distinct edges and each edge e_j is incident to v_j and v_{j+1}. An $a-b$ path of length p is a *shortest path* if every $a-b$ path has length at least p. The *diameter* of a graph G is the length of its longest shortest path, denoted by $\mathrm{diam}(G)$. Given an edge (vertex) coloring α of G, a color $x \in \mathbb{N}$ *occurs* on a path P if there exists an edge (an internal vertex) z on P such that $\alpha(z) = x$.

A vertex or edge coloring of G is *rainbow* if between each pair of vertices a, b there exists an $a-b$ path P such that each color occurs at most once on P; in this case we say that G is *rainbow connected* or *rainbow colored*. We denote by $\mathrm{rc}(G)$ the minimum $i \in \mathbb{N}$ such that there exists a rainbow edge coloring $\alpha : E \to [i]$. Similarly, $\mathrm{rvc}(G)$ denotes the minimum $i \in \mathbb{N}$ such that there exists a rainbow vertex coloring $\alpha : V \to [i]$. Furthermore, an edge or vertex coloring of G is a *strong rainbow coloring* if between each pair of vertices a, b there exists a shortest $a - b$ path P such that each color occurs at most once on P. We denote by $\mathrm{src}(G)$ ($\mathrm{srvc}(G)$) the minimum $i \in \mathbb{N}$ such that there exists a strong rainbow edge (vertex) coloring $\alpha : E \to [i]$ ($\alpha : V \to [i]$).

Let G and H be two graphs with n and n' vertices, respectively. The *corona* of G and H, denoted by $G \circ H$, is the disjoint union of G and n copies of H where the ith vertex of G is connected by an edge to every vertex of the ith copy of H. Clearly, the corona $G \circ H$ has $n(1 + n')$ vertices. Coronas of graphs were first studied by Frucht and Harary [18].

2.2. Problem statements

Here we formally state the problems studied in this work.

Rainbow k-Coloring (k-RC)
Instance: A connected undirected graph $G = (V, E)$.
Question: Is $\mathrm{rc}(G) \leq k$?

Strong Rainbow k-Coloring (k-SRC), Rainbow Vertex k-Coloring (k-RVC) and Strong Rainbow Vertex k-Coloring (k-SRVC) are then defined analogously for $\mathrm{src}(G)$, $\mathrm{rvc}(G)$, and $\mathrm{srvc}(G)$, respectively. We also consider generalized versions of these problems, where k is given as part of the input.

Rainbow Coloring (RC)
Instance: A connected undirected graph $G = (V, E)$, and a positive integer k.
Question: Is $\mathrm{rc}(G) \leq k$?

The problems SRC, RVC, and SRVC are also defined analogously. In Section 5 we consider the "saving" versions of the problem, which ask whether it is possible to improve upon the trivial upper bound for the number of colors.

Saving k Rainbow Colors (k-SavingRC)
Instance: A connected undirected graph $G = (V, E)$.
Question: Is $\mathrm{rc}(G) \leq |E| - k$?

Saving k Rainbow Vertex Colors (k-SavingRVC)
Instance: A connected undirected graph $G = (V, E)$.
Question: Is $\mathrm{rvc}(G) \leq |V| - k$?

2.3. Structural measures

Several of our results utilize certain structural measures of graphs. We will mostly be concerned with the *treewidth* and the *vertex cover number* of the input graph. Section 4 also mentions certain implications of our results for graphs of bounded *clique-width*, the definition of which can be found for instance in [14].

A *tree decomposition* of G is a pair $(T, \{X_i : i \in I\})$ where $X_i \subseteq V$, $i \in I$, and T is a tree with elements of I as nodes such that:

1. for each edge $uv \in E$, there is an $i \in I$ such that $\{u, v\} \subseteq X_i$, and
2. for each vertex $v \in V$, $T[\{i \in I \mid v \in X_i\}]$ is a (connected) tree with at least one node.

The *width* of a tree decomposition is $\max_{i \in I} |X_i| - 1$. The *treewidth* [28] of G is the minimum width taken over all tree decompositions of G and it is denoted by $\mathrm{tw}(G)$.

Fact 1 ([4]). *There exists an algorithm which, given a graph G and an integer p, runs in time $2^{p^{\mathcal{O}(1)}} \cdot (|V(G)| + |E(G)|)$, and either outputs a tree decomposition of G of width at most p or correctly determines that $\mathrm{tw}(G) > p$.*

A *vertex cover* of a graph $G = (V, E)$ is a set $X \subseteq V$ such that each edge in G has at least one endvertex in X. The cardinality of a minimum vertex cover in G is denoted as $\tau(G)$. Given a vertex cover X, a *type* T is a subset of $V \setminus X$ such that any two vertices in T have the same neighborhood; observe that any graph contains at most $2^{|X|}$ many distinct types.

2.4. Monadic second order logic

We assume that we have an infinite supply of individual variables, denoted by lowercase letters x, y, z, and an infinite supply of set variables, denoted by uppercase letters X, Y, Z. *Formulas* of MSO_2 logic are constructed from atomic formulas $I(x, y)$, $x \in X$, and $x = y$ using the connectives $\neg$ (negation), $\wedge$ (conjunction) and existential quantification $\exists x$ over individual variables as well as existential quantification $\exists X$ over set variables. Individual variables range over vertices and edges, and set variables range either over sets of vertices or over sets of edges. The atomic formula $I(x, y)$ expresses that vertex x is incident to edge y, $x = y$ expresses equality, and $x \in X$ expresses that x is in the set X. From this, we define the semantics of MSO_2 logic in the standard way.

MSO_1 logic is defined similarly as MSO_2 logic, with the following distinctions. Individual variables range only over vertices, and set variables only range over sets of vertices. The atomic formula $I(x, y)$ is replaced by $E(x, y)$, which expresses that vertex x is adjacent to vertex y.

Free and bound variables of a formula are defined in the usual way. A *sentence* is a formula without free variables. It is known that MSO_2 formulas can be checked efficiently as long as the graph has bounded tree-width.

Fact 2 ([13]). *Let ϕ be a fixed MSO_2 sentence and $p \in \mathbb{N}$ be a constant. Given an n-vertex graph G of treewidth at most p, it is possible to decide whether $G \models \phi$ in time $\mathcal{O}(n)$.*

Similarly, MSO_1 formulas can be checked efficiently as long as the graph has bounded *clique-width* [14] (or, equivalently, *rank-width* [19]). In particular, while the formula can be checked in linear time if a suitable rank- or clique-decomposition is provided, current algorithms for finding (or approximating) such a decomposition require cubic time.

Fact 3 ([14,19]). *Let ϕ be a fixed MSO_1 sentence and $p \in \mathbb{N}$ be a constant. Given an n-vertex graph G of clique-width at most p, it is possible to decide whether $G \models \phi$ in time $\mathcal{O}(n^3)$.*

3. Hardness of strong rainbow vertex *k*-coloring

It is easy to see that $\text{srvc}(G) = 1$ if and only if $\text{diam}(G) = 2$. We will prove that deciding if $\text{srvc}(G) \leq k$ is NP-complete for every $k \geq 3$ already for graphs of diameter 3. This is done by first showing hardness of an intermediate problem, described below. Later on, we will show that deciding $\text{srvc}(G) \leq 2$ is in fact also NP-complete.

In the *k*-Subset Strong Rainbow Vertex Coloring problem (*k*-SSRVC) we are given a graph G which is a corona of a complete graph and K_1, and a set P of pairs of pendants in G. The goal is to decide if the vertices of G can be colored with k colors such that each pair in P is connected by a vertex rainbow shortest path. We will first show this intermediate problem is NP-complete by reducing from the classical *vertex k-coloring problem*: given a graph G, decide if there is an assignment of k colors to the vertices of G such that adjacent vertices receive a different color. The smallest k for which this is possible is known as the *chromatic number* of G. The vertex k-coloring problem is well-known to be NP-complete for every $k \geq 3$.

Lemma 4. *The k-SSRVC problem is* NP*-complete for every $k \geq 3$.*

Proof. Let $G = (V, E)$ be an instance of the vertex k-coloring problem, where $k \geq 3$. We will construct an instance $\langle G', P \rangle$ of the k-SSRVC problem such that $\langle G', P \rangle$ is a YES-instance if and only if G is vertex k-colorable.

The graph $G' = (V', E')$ along with the set of pairs P are constructed as follows:

- $V' = V \cup \{p_v \mid v \in V\}$,
- $E' = \{uv \mid u, v \in V \wedge u \neq v\} \cup \{vp_v \mid v \in V\}$, and
- $P = \{\{p_u, p_v\} \mid uv \in E\}$.

Clearly, $G' = K_{|V|} \circ K_1$. This completes the construction of G'.

Suppose G is vertex k-colorable. Let c be the color assigned to vertex v in V. We assign the color c to both v and p_v in G'. Observe that the shortest path between any pair of vertices in G' is unique. It is then straightforward to verify that any pair in P is strong rainbow vertex connected.

For the other direction, suppose there is a vertex coloring of G' using k colors under which there is a vertex rainbow shortest path between every pair in P. Since any two vertices $\{p_u, p_v\} \in P$ are strong rainbow vertex connected, the two internal vertices on the unique $p_u - p_v$ shortest path have distinct colors. Thus by assigning to the vertex $v \in V$ the color on the corresponding vertex $v' \in (V' \setminus \{p_v \mid v \in V\})$ we get a proper vertex coloring of G. This completes the proof. □

We are now ready to prove the following.

Theorem 5. *The k-*SRVC *problem is* NP*-complete for every integer $k \geq 3$, even when the input is restricted to graphs of diameter* 3.

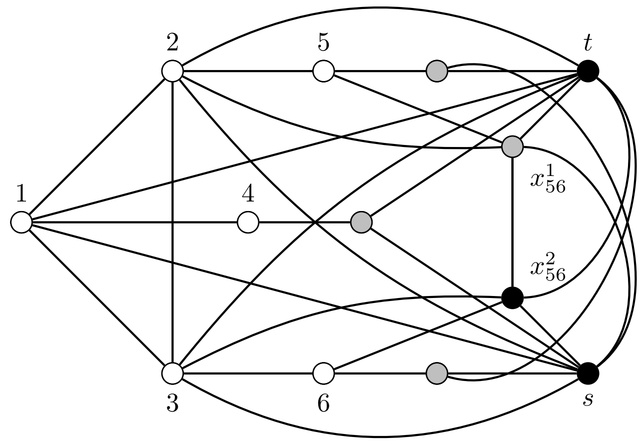

Fig. 1. The graph $K_3 \circ K_1$ transformed to a graph of diameter 3 with $P = \{\{4, 5\}, \{4, 6\}\}$. The color c_1 is represented with gray, and the color c_2 with black. White vertices represent an unknown vertex coloring under which the pairs in P are strong rainbow vertex connected.

Proof. Let $k \geq 3$ and $\langle G = (V, E), P\rangle$ be an instance of the k-SSRVC problem. We will construct a graph $G' = (V', E')$ that is strong rainbow vertex colorable with k colors if and only if $\langle G = (V, E), P\rangle$ is a YES-instance of k-SSRVC.

Let V_1 denote the set of pendant vertices in G. For every vertex $v \in V_1$ we introduce a new vertex x_v. For every pair of pendant vertices $\{u, v\} \notin P$, we add two vertices x^1_{uv} and x^2_{uv}. We also add two new vertices s and t. In the following, we denote by k_v, where $v \in V_1$, the unique vertex that v is adjacent to in G. In addition, for a set A, we write $\binom{A}{2}$ to denote the Cartesian product $A \times A$. Formally, we construct a graph $G' = (V', E')$ such that:

- $V' = V \cup \{x_v \mid v \in V_1\} \cup \{x^1_{uv}, x^2_{uv} \mid \{u, v\} \in \binom{V_1}{2} \setminus P\} \cup \{s, t\}$,
- $E' = E \cup E_1 \cup E_2 \cup E_3 \cup E_4$,
- $E_1 = \{vx_v, sx_v, tx_v \mid v \in V_1\}$,
- $E_2 = \{ux^1_{uv}, x^1_{uv}x^2_{uv}, x^2_{uv}v \mid \{u, v\} \in \binom{V_1}{2} \setminus P\}$,
- $E_3 = \{sx^1_{uv}, tx^2_{uv}, k_ux^1_{uv}, k_vx^2_{uv} \mid \{u, v\} \in \binom{V_1}{2} \setminus P\}$, and
- $E_4 = \{sy, ty \mid y \in V \setminus V_1\}$.

This completes the construction of G'. It is straightforward to verify $\mathrm{diam}(G') = 3$, and this is realized between any pair of vertices in V_1. An example illustrating the reduction is shown in Fig. 1.

First, suppose G' admits a strong rainbow vertex coloring ϕ using k colors. Observe that for each $\{u, v\} \in P$, the shortest path between u and v in G' is unique. Therefore k_u and k_v must receive distinct colors by ϕ. Hence the restriction of ϕ to V witnesses that $\langle G, P\rangle$ is a YES-instance of k-SSRVC.

On the other hand, suppose $\langle G, P\rangle$ is k-subset strong rainbow vertex connected under some coloring $\phi : V \to \{c_1, \dots, c_k\}$. We will describe an extended vertex k-coloring ϕ' under which G' is strong rainbow vertex connected. We retain the original coloring on the vertices of G, i.e., $\phi'(v) = \phi(v)$ for every $v \in V$. The remaining vertices in G' receive colors as follows:

- $\phi'(x_v) = c_1$, for every $v \in V_1$,
- $\phi'(x^1_{uv}) = c_1$, $\phi'(x^2_{uv}) = c_2$, for every $\{u, v\} \in \binom{V_1}{2} \setminus P$, and
- $\phi'(s) = c_2$, and $\phi'(t) = c_2$.

Since each pair of vertices $\{a, b\} \in V'$ at distance at most 2 are always strong rainbow vertex connected regardless of the chosen coloring, and each pair of vertices $\{u, v\} \in \binom{V_1}{2} \setminus P$ are connected by the path through x^1_{uv}, x^2_{uv}, it is straightforward to verify that G' is indeed strong rainbow vertex connected under ϕ'. □

It can be observed that the size of the above reduction does not depend on k, the number of colors. In fact, if the instance of the vertex k-coloring problem has n vertices, then the graph G' we build in Theorem 5 has no more than $\mathcal{O}(n^2)$ vertices. Furthermore, a strong rainbow vertex coloring of G' gives us a solution to the vertex k-coloring problem. Since the chromatic number of an n-vertex graph cannot be approximated within a factor of $n^{1-\epsilon}$ for any $\epsilon > 0$ unless $\mathtt{P} = \mathtt{NP}$ [31], we obtain the following corollary.

Corollary 6. *There is no polynomial time algorithm for approximating the strong rainbow vertex connection number of an n-vertex graph of bounded diameter within a factor of $n^{1/2-\epsilon}$ for any $\epsilon > 0$, unless* $\mathtt{P} = \mathtt{NP}$.

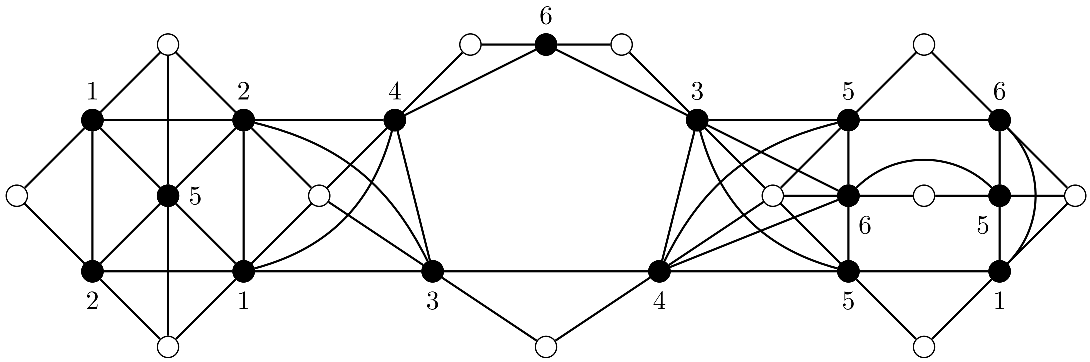

Fig. 2. A 12-vertex graph G after the construction of Lemma 7. The white vertices correspond to the original 12 vertices of G, and the black vertices to the subdivided edges. An optimal strong rainbow coloring with 6 colors is shown.

Each of k-RC, k-SRC, and k-RVC is known to be NP-complete for every $k \geq 2$. In this light, Theorem 5 raises a natural question: is k-SRVC also NP-complete already for $k = 2$? Indeed, the following lemma will establish that this is the case. In contrast to Theorem 5, we employ a more direct reduction from k-SRC.

Lemma 7. *There exists a polynomial time algorithm which, given an instance $G = (V, E)$ of k-SRC, creates an instance $G' = (V', E')$ of k-SRVC such that G is a YES-instance of k-SRC if and only if G' is a YES-instance of k-SRVC.*

Proof. Let $G = (V, E)$ be an instance of the k-SRC problem for any $k \geq 2$. In polynomial time, we will construct an instance $G' = (V', E')$ of the k-SRVC problem such that $\text{src}(G) = k$ if and only if $\text{srvc}(G') = k$.

The graph G' is obtained from G by subdividing each edge $e \in E$ by a new vertex w_e. Then, for every $e, f \in E$ such that $e \neq f$, we make w_e and w_f adjacent in G' if the edges e and f were adjacent in G. Formally, we let $V' = V \cup W$, where $W = \{w_e \mid e \in E\}$, and $E' = \{aw_e, bw_e \mid ab = e \in E\} \cup \{w_e w_f \mid e, f \text{ are adjacent in } G\}$. This completes the construction of G', which is illustrated in Fig. 2. Let us then prove that $\text{src}(G) = k$ if and only if $\text{srvc}(G') = k$.

Suppose $\text{src}(G) = k$, and consider an edge coloring $c : E \to [k]$ under which G is strong rainbow connected. We construct a vertex k-coloring $c' : V' \to [k]$ such that $c'(w_{ij}) = c(ij)$, for every $ij \in E$. The remaining vertices receive an arbitrary color, say $c'(v) = 1$ for every $v \in V$. We claim that any two vertices u and v are strong rainbow vertex connected under c' in G'. There are three cases to consider: both u and v are in V, neither u nor v is in V, and exactly one of u and v is in V. We will show this only for the first case, as the remaining cases follow easily by a similar argument. Without loss of generality, suppose u and v are strong rainbow connected in G via the edges $e_1, \ldots, e_\ell$, where $\ell \leq k$. Then, it is easy to see the path $u, w_1, \ldots, w_\ell, v$ is a rainbow vertex shortest path in G'. A strong rainbow coloring of G transformed into a strong rainbow vertex coloring of G' is shown in Fig. 2.

For the other direction, suppose there is a vertex k-coloring c' under which G' is strong rainbow vertex connected. We construct an edge k-coloring $c : E \to [k]$ for G such that $c(ij) = c'(w_{ij})$, for every $ij \in E$. Observe that for each nonadjacent $u, v \in V(G')$ and each shortest u-v path P_{uv}, it holds that every internal vertex of P_{uv} is in W. Thus, because c' is a strong rainbow vertex coloring of G', we have that c is a strong rainbow coloring of G. This completes the proof. □

By the above, we strengthen Theorem 5 to cover the case $k = 2$.

Corollary 8. *The k-SRVC problem is NP-complete for every integer $k \geq 2$.*

In fact, we remark that Lemma 7 holds also for the non-strong variants of the problems. That is, the lemma establishes also a reduction from k-RC to k-RVC. In the literature, the hardness of k-RVC was established in two stages: Chen et al. [12] first showed 2-RVC is NP-complete by a chain of reductions originating from 3-SAT. Then, by reducing from 2-RVC, Chen et al. [11] proved that k-RVC is NP-complete for every $k \geq 2$. Conceptually, Lemma 7 is a considerable simplification over this chain of reductions. Moreover, it was shown by Chandran and Rajendraprasad [7] that the rainbow connection number of a graph cannot be approximated within a factor of less than 2 unless P = NP. Thus, by combining our remark on Lemma 7 with an argument similar to Corollary 6, we obtain the following.

Corollary 9. *There is no polynomial time algorithm for approximating the rainbow vertex connection number of a graph within a factor less than 2 unless P = NP.*

4. MSO formulations

This section will present formulations of the k-coloring variants of rainbow connectivity in MSO logic, along with their algorithmic implications.

Lemma 10. *For every $k \in \mathbb{N}$ there exists a MSO_1 formula ϕ_k such that for every graph G, it holds that $G \models \phi_k$ iff G is a YES-instance of k-RVC. Similarly, for every $k \in \mathbb{N}$ there exists a MSO_2 formula ψ_k such that for every graph G, it holds that $G \models \psi_k$ iff G is a YES-instance of k-RC.*

Proof. In the case of k-RC, we wish to partition the edges of the graph $G = (V, E)$ into k color classes $C_1, \dots, C_k$ such that each pair of vertices is connected by a rainbow path. Let us consider the following MSO_2 formula ψ_k.

$$\begin{aligned}\psi_k := \exists C_1, \dots, C_k \subseteq E\Big(\forall e \in E\Big(e \in C_1 \vee \dots \vee e \in C_k\Big)\Big)\\ \wedge \Big(\forall i, j \in [k], i \neq j : (C_i \cap C_j = \emptyset)\Big)\\ \wedge \Big(\forall u, v \in V\Big((u \neq v) \Longrightarrow \bigvee_{1 \le i \le k} \Big(\exists e_1, \dots, e_i \in E\big(\mathrm{Path}(u, v, e_1, \dots, e_i)\\ \wedge \mathrm{Rainbow}(e_1, \dots, e_i)\big)\Big)\Big)\Big),\end{aligned}$$

where the auxiliary predicates are defined as

$$\begin{aligned}\mathrm{Path}(u, v, e_1, \dots, e_\ell) := \exists v_1, \dots, v_{\ell-1} \in V\Big(\forall i, j \in [\ell-1], i \neq j : (v_i \neq v_j)\Big)\\ \wedge \mathrm{I}(e_1, u) \wedge \mathrm{I}(e_\ell, v)\\ \wedge \Big(\forall i \in [\ell-1] : (\mathrm{I}(e_i, v_i) \wedge \mathrm{I}(e_{i+1}, v_i))\Big)\end{aligned}$$

and

$$\mathrm{Rainbow}(e_1, \dots, e_\ell) := \forall i \in [\ell]\, \exists j \in [k] : \Big(e_i \in C_j \wedge (\forall p \neq i : e_p \notin C_j)\Big).$$

Here, $\mathrm{Path}(u, v, e_1, \dots, e_\ell)$ expresses that the edges $e_1, \dots, e_\ell$ form a path between the vertices u and v. The predicate $\mathrm{Rainbow}(e_1, \dots, e_\ell)$ expresses that the edges $e_1, \dots, e_\ell$ are each in precisely one color class.

In the case of k-RVC, the MSO_1 formula ϕ_k is defined analogously, with the following distinctions:

1. instead of edges, we partition the vertices of G into color classes;
2. the predicate Path speaks of vertices instead of edges and uses the adjacency relation instead of the incidence relation; and
3. the predicate Rainbow tests the coloring of vertices instead of edges. □

Using a similar approach, we obtain an analogous result for the strong variants of these problems.

Lemma 11. *For every $k \in \mathbb{N}$ there exists a MSO_1 formula ϕ_k such that for every graph G, it holds that $G \models \phi_k$ iff G is a YES-instance of k-SRVC. Similarly, for every $k \in \mathbb{N}$ there exists a MSO_2 formula ψ_k such that for every graph G, it holds that $G \models \psi_k$ iff G is a YES-instance of k-SRC.*

Proof. In the case of k-SRC, we wish to partition the edges of the graph $G = (V, E)$ into k color classes $C_1, \dots, C_k$ such that each pair of vertices is connected by a rainbow shortest path. We will assume the predicates $\mathrm{Path}(u, v, e_1, \dots, e_\ell)$ and $\mathrm{Rainbow}(e_1, \dots, e_\ell)$ are defined precisely as in Lemma 10.

Let us then construct the following MSO_2 formula ψ_k:

$$\begin{aligned}\psi_k := \exists C_1, \dots, C_k \subseteq E\Big(\forall e \in E\Big(e \in C_1 \vee \dots \vee e \in C_k\Big)\Big)\\ \wedge \Big(\forall i, j \in [k], i \neq j : (C_i \cap C_j = \emptyset)\Big)\\ \wedge \Big(\forall u, v \in V\Big((u \neq v) \Longrightarrow \Big(\exists i \in [k]\, \exists e_1, \dots, e_i \in E\big(\mathrm{Path}(u, v, e_1, \dots, e_i)\\ \wedge \mathrm{Rainbow}(e_1, \dots, e_i)\\ \wedge \forall j \in [i-1]\, \neg\big(\exists w_1, \dots, w_j \in E\ \mathrm{Path}(u, v, w_1, \dots, w_j)\big)\big)\Big)\Big)\Big).\end{aligned}$$

To capture the property of being a shortest path, we require there to be a $u - v$ path of length i, and no paths of length less than i. Furthermore, observe that no path of length greater than k can be rainbow. The construction for k-SRVC then uses the same ideas, with the same distinctions as those specified in Lemma 10. □

Theorem 12. *Let $p \in \mathbb{N}$ be fixed. Then the problems k-RC, k-SRC, k-RVC, and k-SRVC can be solved in time $\mathcal{O}(n)$ on n-vertex graphs of treewidth at most p. Furthermore, k-RVC and k-SRVC can be solved in time $\mathcal{O}(n^3)$ on n-vertex graphs of clique-width at most p.*

Proof. The proof follows from Lemmas 10 and 11 in conjunction with Facts 2 and 3. □

In the language of parameterized complexity [16,27], Theorem 12 implies that these problems are fixed-parameter tractable (FPT) parameterized by treewidth, and their vertex variants are FPT parameterized by clique-width.

5. The complexity of saving colors

This section focuses on the saving versions of the rainbow coloring problems introduced in Section 2.2, and specifically gives linear-time algorithms for k-SavingRC and k-SavingRVC. Our results make use of the following facts.

Fact 13 ([20]). *There is a MSO_1 predicate* VertexConnects *such that on a graph $G = (V, E)$* VertexConnects(S, u, v) *is true iff $S \subseteq V$ is a set of vertices of G such that there is a path from u to v that lies entirely in S.*

The above is easily modified to give us the following.

Fact 14. *There is a MSO_2 predicate* EdgeConnects *such that on a graph $G = (V, E)$* EdgeConnects(X, u, v) *is true iff $X \subseteq E$ is a set of edges of G such that there is path from u to v that lies entirely in X.*

Theorem 15. *For each $k \in \mathbb{N}$, the problem k-SavingRC can be solved in time $\mathcal{O}(n)$ on n-vertex graphs.*

Proof. Observe that by coloring each edge of a spanning tree of G with a distinct color we have that $\mathrm{rc}(G) \leq n - 1$. Thus, if $m \geq n + k - 1$, we have a YES-instance of k-SavingRC. Otherwise, suppose $m < n + k - 1$. Then G has a feedback edge set of size at most $k - 1$, and hence, G has treewidth at most k. Furthermore, we assume that $m > 2k$, since otherwise the instance can be solved by brute force in time independent of n. We construct a MSO_2 formula ψ_k such that it holds that $G \models \psi_k$ is true iff G is a YES-instance of k-SavingRC. Using Fact 14, we construct ψ_k as follows:

$$\begin{aligned}\psi_k :=\ & \exists R_1, \ldots, R_k \subseteq E\Big(\forall i, j \in [k], i \neq j : (R_i \cap R_j = \emptyset)\Big)\\ & \wedge \ \Big(\forall i \in [k] : \big(\exists e \in E(e \in R_i)\big)\Big) \wedge |R_1 \cup R_2 \cup \cdots \cup R_k| \geq 2k\\ & \wedge \ \Big(\forall u, v \in V\Big((u \neq v) \implies \Big(\exists X \subseteq E\Big(\mathrm{EdgeConnects}(X, u, v)\\ & \wedge \ \forall e_1, e_2 \in X\Big(\forall i \in [k] : (e_1 \in R_i \wedge e_2 \in R_i) \implies (e_1 = e_2)\Big)\Big)\Big)\Big)\Big).\end{aligned}$$

In the above, the expression $|A| \geq 2k$ is shorthand for the existence of $2k$ pairwise-distinct edges in A, which can be expressed by a simple but lengthy MSO_2 expression. The formula ψ_k expresses that there exist k disjoint sets $R_1, \ldots, R_k$ of edges (each representing a different color set with at least 1 edge) such that their union contains at least $2k$ edges, with the following property: there is a path using at most one edge from each set $R_1, \ldots, R_k$ between every pair of vertices. Formally, this property is stated as the existence of an edge-set X for each pair of vertices u, v such that the graph (V, X) contains an $u - v$ path that cannot repeat edges from any R_i.

Let us argue that $G \models \psi_k$ is true iff G is a YES-instance of k-SavingRC. Assume G contains sets $R_1, \ldots, R_k$ as per ψ_k; then assigning a unique color to each R_i and a unique additional color to each edge in $E \setminus (R_1 \cup \cdots \cup R_k)$ results in a rainbow coloring of G which uses at most $m - k$ colors. On the other hand, in any rainbow $(m - k)$-coloring of G, there must exist j reappearing colors in G (where $2 \leq j \leq k$) and the number of edges with these j colors is at least $j + k$. Let us consider an assignment of edges to $R_1, \ldots, R_j$ such that R_i receives all edges with reappearing color i; for the remaining $k - j$ sets R, we then use arbitrarily selected edges (i.e., they represent arbitrary non-reappearing colors). Since we started with a rainbow coloring, rows 3 and 4 of ψ_k must be satisfied, and since $|R_1, \ldots, R_j| \geq k + j$ it follows that row 2 must also be satisfied.

By the above, it indeed holds that $G \models \psi_k$ is true iff G is a YES-instance of k-SavingRC. The proof then follows by Fact 2. □

To prove a similar result for k-SavingRVC, we will use the following result.

Fact 16 ([3]). *If the treewidth of a connected graph G is at least $2k^3$, then G has a spanning tree with at least k vertices with degree 1.*

Theorem 17. *For each $k \in \mathbb{N}$, the problem k-SavingRVC can be solved in time $\mathcal{O}(n)$ on n-vertex graphs.*

Proof. Using Fact 1, we will test if the treewidth of G is at least $2k^3$. If it is, then by Fact 16 the graph G has a spanning tree with at least k vertices of degree 1. Each of these k vertices can receive the same color, and we conclude we have a YES-instance. Otherwise, suppose the treewidth of G is less than $2k^3$, and we construct a MSO_1 formula ϕ_k such that it holds that $G \models \phi_k$ is true iff G is a YES-instance of k-SavingRVC. The construction is analogous to Theorem 15, but instead of EdgeConnects we use VertexConnects from Fact 13. The proof then follows by Fact 2. □

6. Rainbow coloring graphs with small vertex covers

In this section we turn our attention to the more general problem of determining whether the rainbow connection number is below a number specified in the input. Specifically, we show that RC, RVC, and SRVC admit linear time algorithms on graphs of bounded vertex cover number. In particular, this implies that RC, RVC, and SRVC are FPT parameterized by $\tau(G)$.

Lemma 18. *Let $G = (V, E)$ be a connected graph and $p = \tau(G)$. Then* $\mathrm{rvc}(G) \leq 2p - 1$ *and* $\mathrm{srvc}(G) \leq \frac{p^2+p}{2}$.

Proof. Let us fix a vertex cover X of cardinality p. For the first claim, let S be a spanning tree of G with a minimum number of internal vertices, and observe that the number of internal vertices in S is at most $2p - 1$; indeed, one only needs to add at most $p - 1$ vertices to the vertex cover in order to get a connected subgraph. Let Z be the internal vertices of S. Let α be a vertex coloring which assigns a unique color from $[|Z|]$ to each vertex in Z, and then assigns the color 1 to each vertex in $V \setminus Z$. Then α is a rainbow vertex coloring: for any choice of a and b, there exists an $a - b$ path whose internal vertices are a subset of Z.

For the second claim, consider the set Q constructed as follows: for each distinct $a, b \in X$, if there exists a vertex v in $V \setminus X$ adjacent to both a and b, we choose an arbitrary such v and add it into Q. Let $Z = Q \cup X$, and observe that $|Z| \leq p + \frac{p\cdot(p-1)}{2} = \frac{p^2+p}{2}$. Once again, let α be a vertex coloring which assigns a unique color from $[|Z|]$ to each vertex in Z, and then assigns the color 1 to each vertex in $V \setminus Z$. We claim that α is a strong rainbow vertex coloring. Indeed, consider any $a, b \in V$ and let P be an arbitrary shortest $a - b$ path. Then for every internal vertex v_i of P such that $v_i \notin X$, it must hold that $v_{i-1} \in X$ and $v_{i+1} \in X$. Consider the path P' obtained from P by replacing each internal vertex $v_i \notin X$ by v_i', where v_i' is an element of Q which is adjacent to v_{i-1} and v_{i+1}. Since P' has the same length as P and P' is rainbow colored by α, the claim follows. □

The following lemma will be useful in the proof of Lemma 20, a key component of our approach for dealing with RC on the considered graph classes. A *bridge* is an edge e such that deleting e separates the connected component containing e into two connected components.

Lemma 19. *Let $G = (V, E)$ be a graph and $X \subseteq V$ be a minimum vertex cover of G. Then there exist at most $2|X| - 2$ bridges which are not incident to a pendant outside of X.*

Proof. We prove the lemma by induction on $p = |X|$. If $p = 1$, then the graph is a star and the lemma holds (in a star, every bridge is incident to a pendant).

So, assume the lemma holds for $p - 1$ and assume G has a vertex cover X of size p. Let S be the set of all bridges in G which are not incident to a pendant outside of X. If S contains a bridge e whose both endpoints lie in X, then e separates X into two non-empty subsets X_1 and X_2 and every other bridge has both endpoints either in X_1 or in X_2. Let G_1 and G_2 be the connected components of $G - e$ containing X_1 and X_2, respectively. Observe that X_i is a vertex cover of G_i for $i \in [2]$. Since $|X_1| < p$ and $|X_2| < p$, by our inductive assumption it follows that G_1 contains at most $2|X_1| - 2$ bridges which are not incident to a pendant outside of X, and similarly G_2 contains at most $2|X_2| - 2$ bridges which are not incident to a pendant outside of X. Since each bridge in G is either e or a bridge in G_1 or G_2, it follows that $|S| = 1+2|X_1|-2+2|X_2|-2 = 1+2p-4 < 2p-2$, and hence in this case the lemma holds.

On the other hand, assume S contains a bridge $e = ax$ where $x \in X$, $a \notin X$. Since the connected component of $G - e$ containing a is not a pendant, it follows that a has a neighbor in G which is different from x, and hence this connected component (say G_1) contains at least one vertex from X. Let $X_1 = X \cap V(G_1)$, $X_2 = X \setminus X_1$ and G_2 be the connected component of $G - e$ containing X_2. This implies that in this case e also separates X into two non-empty subsets X_1 and X_2. Furthermore, if there exists another $e' \in S$ which separates X into the same sets X_1 and X_2 as e, then e' must also be incident to a and in particular this other edge e' is unique; every other bridge in S has both endpoints either in X_1 or in X_2. Since $|X_1| < p$ and $|X_2| < p$, by our inductive assumption it follows that G_1 contains at most $2|X_1| - 2$ bridges which are not incident to a pendant outside of X, and similarly G_2 contains at most $2|X_2| - 2$ bridges which are not incident to a pendant outside of X. Since each bridge in G is either e or e' or a bridge in G_1 or G_2, it follows that $|S| \leq 2+2|X_1|-2+2|X_2|-2 = 2+2p-4 \leq 2p-2$, and hence in this case the lemma also holds. □

For ease of presentation, we define the function β as $\beta(p) = 2p - 2 + p \cdot (p^2 + 2p \cdot 2^p)$. Lemma 20 will represent one part of our win–win strategy, as it allows us to precisely compute $\mathrm{rc}(G)$ when the number of bridges is sufficiently large. We remark that an analogous claim does not hold for $\mathrm{src}(G)$ (regardless of the choice of β).

Lemma 20. *Let $G = (V, E)$ be a connected graph and $p = \tau(G)$. Let z be the number of bridges in G. If $z \geq \beta(p)$, then* $\mathrm{rc}(G) = z$.

Proof. Let us fix a vertex cover X of cardinality p. It is known that the number of bridges is a lower bound for $\mathrm{rc}(G)$ [9], i.e., $\mathrm{rc}(G) \geq z$. We will show that z is also an upper bound for $\mathrm{rc}(G)$.

Consider the edge z-coloring α constructed as follows. Since X is a vertex cover and, by Lemma 19 in conjunction with our assumption on z, there are at least $p \cdot (p^2 + 2p \cdot 2^p)$ leaves in G, it follows that there must exist some $x \in X$ adjacent to at least $z' = p^2 + 2p \cdot 2^p$ pendants. Let $\{e_1, \ldots, e_{z'}\}$ be the edges incident to both x and a pendant vertex, and let $\{e_{z'+1}, \ldots, e_z\}$ be all the remaining bridges; then for each bridge we set $\alpha(e_i) = i$.

Let $f_1, \ldots, f_q$ be the edges of $G[X]$ which are not bridges; for each such edge we set $\alpha(f_i) = z' - i$. Observe that for each such f_i we have $\alpha(f_i) > 2p \cdot 2^p$.

Consider the set $\tau = \{T_i \mid T_i$ is a type in G and $|N(T_i)| > 1\}$. Let $Q_i = \{2pi + 1, \ldots, 2pi + 2p\}$. For each $T_i \in \tau$, we let G_i be the subgraph of G on $T_i \cup N(T_i)$ which contains exactly the edges incident to T_i. Then G_i is bipartite, and furthermore can be rainbow colored using at most $2p$ colors as follows: we pick an arbitrary $y \in T_i$ and uniquely color all edges in G_i incident to y using colors $c_1, \ldots, c_p$, and for every other vertex in T_i we color all edges in G_i incident to y' using colors $c_{1+p}, \ldots, c_{2p}$. For each type $T_i \in \tau$, we let α color the edges incident to T_i in this manner using the colors from Q_i.

We will proceed by arguing that α is a rainbow z-coloring of G, but before that we make three key observations. First, there are only two cases when α can use the same color for two distinct edges e, f: either one of e, f is an edge between x and a pendant, or both e, f occur in some G_i. Second, if $|T_i| > 1$, then for every $u \in N(T_i)$ and every $v \in V(G_i)$ and every color c, there exists a rainbow $u - v$ path in G_i under α which does not use c. Third, each G_i is rainbow colored by α.

We now make the following case distinction.

1. Let $a, b \in V$ be such that neither is a pendant adjacent to x. Consider an arbitrary $a - b$ path P such that the number of pairs of edges in P assigned the same color by α is minimized. If P contains two edges e, f such that $\alpha(e) = \alpha(f)$, then both e and f must occur in some G_i. Let t and u be the first and last vertices in $V(G_i)$ on P, respectively. Since G_i is rainbow colored by α, there exists a $t - u$ rainbow path P^* in G_i. Let P' be obtained from P by replacing the path segment between t and u by P^*; by the key observation made above, it follows that P' has a strictly lower number of pairs of edges in P which are assigned the same color by α, hence contradicting our choice of P.
2. Let a be a pendant adjacent to x, and $b \in V$. Let $c = \alpha(xa)$. Consider an arbitrary $a - b$ path P such that the number of pairs of edges in P assigned the same color by α is minimized. If P contains two edges e, f such that $\alpha(e) = \alpha(f) \neq c$, then both e and f must occur in some G_i s.t. $|T_i| > 1$. Let t and u be the first and last vertices in $V(G_i)$ on P, respectively. Since $t \in X \cap N(T_i)$, by our observations above it follows that there exists a rainbow $t - u$ path P^* in G_i which avoids c. Hence the path obtained by replacing the path segment between t and u by P^* once again contradicts our choice of P. On the other hand, if P contains an edge e such that $\alpha(e) = c$, then either e is an edge in $G[X]$ or e is incident to some T_i. In the latter case, the same argument can be used to contradict our choice of P. In the former case it follows by construction of α that c only occurs on the edge (x, a) and on e, and furthermore e is contained in some 2-edge-connected component D of G. Let d, w be the first and last vertices, respectively, in D which occurs in P, and let P' be the path obtained from P by replacing the path segment between d and w by an arbitrary rainbow path segment in D which does not contain e. It is readily verified that the colors which occur in D are only repeated on edges between x and pendants, and in particular such edges cannot occur on P'. Hence P' again contradicts our choice of P.

To summarize, for any $a, b \in V$ there exists a rainbow $a - b$ path under α, and hence α witnesses that $\mathrm{rc}(G) \leq z$. We conclude that $\mathrm{rc}(G) = z$. □

Lemma 21. *Let $G = (V, E)$ be a graph with a vertex cover $X \subseteq V$ of cardinality p. Let z be the number of bridges in G. If $z < \beta(p)$, then $\mathrm{rc}(G) \leq \beta(p) + p^2 + 2^p \cdot 2p$.*

Proof. Consider the following edge coloring α which assigns a unique color to each edge in $G[X]$ and to each edge incident to a pendant. For each nonempty type T_i, we choose an arbitrary vertex y_i and let α assign a unique color for each of the at most p edges incident to y_i. Finally, for each type T_i and each $x \in X$ adjacent to (the vertices of) T_i, α uses a single new color for all edges between x and the vertices in T_i. It is readily verified that α uses no more than $z + p^2 + 2^p \cdot 2p$ colors.

We argue that α is rainbow. Let G_i be the subgraph of G on $T_i \cup N(T_i)$ which contains exactly the edges incident to T_i, and observe that each G_i is rainbow colored by α. Consider any $a, b \in V$ and let P be an $a - b$ path such that the number of pairs of edges in P assigned the same color by α is minimized. By construction of α, two edges e, f in P may only have the same color if e, f are both incident to some T_i. Let t and u be the first and last vertices in $V(G_i)$ on P, respectively. Since G_i is rainbow colored by α, there exists a $t - u$ rainbow path P^* in G_i under α. Let P' be obtained from P by replacing the path segment between t and u by P^*. Then P' has a strictly lower number of pairs of edges in P with the same color, which contradicts our choice of P. □

Theorem 22. *Let $p \in \mathbb{N}$ be fixed. Then the problems* RC, RVC, *and* SRVC *can be solved in time $\mathcal{O}(n)$ on n-vertex graphs of vertex cover number at most p.*

Proof. For RVC and SRVC, we first observe that if k (the queried upper bound on the number of colors) is greater than $2p - 1$ and $\frac{p^2+p}{2}$, respectively, then the algorithm can immediately output YES by Lemma 18. Otherwise we use Theorem 12 and the fact that the vertex cover number is an upper bound on the treewidth to compute a solution in $\mathcal{O}(n)$ time.

For RC, it is well known that the total number of bridges in G, say z, can be computed in linear time on graphs of bounded treewidth. If $z \geq \beta(p)$, then by Lemma 20 we can correctly output YES when $z \leq k$ and NO when $z > k$. On the other hand, if $z < \beta(p)$, then by Lemma 21 the value $\mathrm{rc}(G)$ is upper-bounded by a function of p. We compare k and this upper bound on $\mathrm{rc}(G)$; if k exceeds the upper bound on $\mathrm{rc}(G)$, then we output YES, and otherwise we can use Theorem 12 along with the fact that the vertex cover number is an upper bound on the treewidth to compute a solution in $\mathcal{O}(n)$ time. □

7. Concluding notes

We presented new positive and negative results for the most prominent variants of rainbow coloring. We believe that the techniques presented above, and in particular the win–win approaches used in Sections 5 and 6, can be of use also for other challenging connectivity problems.

It is worth noting that our results in Section 4 leave open the question of whether Rainbow Coloring or its variants can be solved in (uniformly) polynomial time on graphs of bounded treewidth. Hardness results for related problems [30,23] do not imply that finding an optimal coloring of a bounded-treewidth graph is hard, and it seems that new insights are needed to determine the complexity of these problems on graphs of bounded treewidth. Finally, the complexity of the SRC problem still remains open on graphs of bounded vertex cover number.

Acknowledgments

The authors would like to thank the reviewers for their useful comments. In addition, the authors would like to thank Martin Derka for his helpful comments and discussion at the IWOCA'15 conference that took place in Verona, Italy.

This work was partially supported by the Austrian Science Fund (FWF), projects P26696 and W1255-N23 (E.E., R.G.), and by the Emil Aaltonen Foundation (160138 N) (J.L.). Robert Ganian is also affiliated with FI MU, Brno, Czech Republic.

References

[1] P. Ananth, M. Nasre, K.K. Sarpatwar, Rainbow connectivity: Hardness and tractability, in: IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2011, 2011, pp. 241–251.
[2] M. Basavaraju, L. Chandran, D. Rajendraprasad, A. Ramaswamy, Rainbow connection number and radius, Graphs Combin. (2012) 1–11.
[3] H. Bodlaender, On linear time minor tests with depth-first search, J. Algorithms 14 (1) (1993) 1–23.
[4] H.L. Bodlaender, T. Kloks, Efficient and constructive algorithms for the pathwidth and treewidth of graphs, J. Algorithms 21 (2) (1996) 358–402.
[5] S. Chakraborty, E. Fischer, A. Matsliah, R. Yuster, Hardness and algorithms for rainbow connection, J. Comb. Optim. 21 (3) (2009) 330–347.
[6] L.S. Chandran, D. Rajendraprasad, Rainbow colouring of split and threshold graphs, Comput. Comb. (2012) 181–192.
[7] L.S. Chandran, D. Rajendraprasad, Inapproximability of rainbow colouring, in: IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2013, 2013, pp. 153–162.
[8] G. Chartrand, G. Johns, K. McKeon, P. Zhang, Rainbow connection in graphs, Math. Bohem. 133 (1) (2008).
[9] G. Chartrand, F. Okamoto, P. Zhang, Rainbow trees in graphs and generalized connectivity, Networks 55 (4) (2010) 360–367.
[10] G. Chartrand, P. Zhang, Chromatic Graph Theory, CRC Press, 2008.
[11] L. Chen, X. Li, H. Lian, Further hardness results on the rainbow vertex-connection number of graphs, Theoret. Comput. Sci. 481 (0) (2013) 18–23.
[12] L. Chen, X. Li, Y. Shi, The complexity of determining the rainbow vertex-connection of a graph, Theoret. Comput. Sci. 412 (35) (2011) 4531–4535.
[13] B. Courcelle, The monadic second-order logic of graphs I. Recognizable sets of finite graphs, Inform. Comput. (1990) 12–75.
[14] B. Courcelle, J.A. Makowsky, U. Rotics, Linear time solvable optimization problems on graphs of bounded clique-width, Theory Comput. Syst. 33 (2) (2000) 125–150.
[15] R. Diestel, Graph Theory, Springer-Verlag, Heidelberg, 2010.
[16] R.G. Downey, M.R. Fellows, Fundamentals of Parameterized Complexity, Springer, 2013.
[17] E. Eiben, R. Ganian, J. Lauri, On the complexity of rainbow coloring problems, in: Proceedings of the 26th International Workshop on Combinatorial Algorithms, IWOCA 2015, Verona, Italy, October 5–7, 2015, pp. 209–220.
[18] R. Frucht, F. Harary, On the corona of two graphs, Aequationes Math. 4 (3) (1970) 322–325.
[19] R. Ganian, P. Hliněný, On parse trees and Myhill-Nerode-type tools for handling graphs of bounded rank-width, Discrete Appl. Math. 158 (7) (2010) 851–867.
[20] G. Gottlob, S.T. Lee, A logical approach to multicut problems, Inform. Process. Lett. 103 (4) (2007) 136–141.
[21] M. Keranen, J. Lauri, Computing minimum rainbow and strong rainbow colorings of block graphs, 2014. arXiv preprint arXiv:1405.6893.
[22] M. Krivelevich, R. Yuster, The rainbow connection of a graph is (at most) reciprocal to its minimum degree, J. Graph Theory 63 (3) (2010) 185–191.
[23] J. Lauri, Further hardness results on rainbow and strong rainbow connectivity, Discrete Appl. Math. 201 (2016) 191–200.
[24] X. Li, Y. Mao, Y. Shi, The strong rainbow vertex-connection of graphs, Util. Math. 93 (2014) 213–223.
[25] X. Li, Y. Shi, Y. Sun, Rainbow connections of graphs: A survey, Graphs Combin. 29 (1) (2012) 1–38.
[26] X. Li, Y. Sun, Rainbow Connections of Graphs, Springer, 2012.
[27] R. Niedermeier, Invitation to Fixed-Parameter Algorithms, Oxford University Press, Oxford, 2006.
[28] N. Robertson, P. Seymour, Graph minors. II. Algorithmic aspects of tree-width, J. Algorithms 7 (3) (1986) 309–322.
[29] L. Sunil Chandran, A. Das, D. Rajendraprasad, N.M. Varma, Rainbow connection number and connected dominating sets, J. Graph Theory 71 (2) (2012) 206–218.
[30] K. Uchizawa, T. Aoki, T. Ito, A. Suzuki, X. Zhou, On the rainbow connectivity of graphs: Complexity and FPT algorithms, Algorithmica 67 (2) (2013) 161–179.
[31] D. Zuckerman, Linear degree extractors and the inapproximability of max clique and chromatic number, Theory Comput. 3 (6) (2007) 103–128.