Fine-Grained Complexity of Rainbow Coloring and Its Variants[*]

Akanksha Agrawal

University of Bergen, Norway
akanksha.agrawal@uib.no

Abstract

Consider a graph G and an edge-coloring $c_R : E(G) \rightarrow [k]$. A rainbow path between $u, v \in V(G)$ is a path P from u to v such that for all $e, e' \in E(P)$, where $e \neq e'$ we have $c_R(e) \neq c_R(e')$. In the Rainbow k-Coloring problem we are given a graph G, and the objective is to decide if there exists $c_R : E(G) \rightarrow [k]$ such that for all $u, v \in V(G)$ there is a rainbow path between u and v in G. Several variants of Rainbow k-Coloring have been studied, two of which are defined as follows. The Subset Rainbow k-Coloring takes as an input a graph G and a set $S \subseteq V(G) \times V(G)$, and the objective is to decide if there exists $c_R : E(G) \rightarrow [k]$ such that for all $(u, v) \in S$ there is a rainbow path between u and v in G. The problem Steiner Rainbow k-Coloring takes as an input a graph G and a set $S \subseteq V(G)$, and the objective is to decide if there exists $c_R : E(G) \rightarrow [k]$ such that for all $u, v \in S$ there is a rainbow path between u and v in G. In an attempt to resolve open problems posed by Kowalik et al. (ESA 2016), we obtain the following results.

- For every $k \geq 3$, Rainbow k-Coloring does not admit an algorithm running in time $2^{o(|E(G)|)}n^{\mathcal{O}(1)}$, unless ETH fails.
- For every $k \geq 3$, Steiner Rainbow k-Coloring does not admit an algorithm running in time $2^{o(|S|^2)}n^{\mathcal{O}(1)}$, unless ETH fails.
- Subset Rainbow k-Coloring admits an algorithm running in time $2^{\mathcal{O}(|S|)}n^{\mathcal{O}(1)}$. This also implies an algorithm running in time $2^{o(|S|^2)}n^{\mathcal{O}(1)}$ for Steiner Rainbow k-Coloring, which matches the lower bound we obtain.

1998 ACM Subject Classification G.2.2 Graph Algorithms, I.1.2 Analysis of Algorithms

Keywords and phrases Rainbow Coloring, Lower bound, ETH, Fine-grained Complexity

Digital Object Identifier 10.4230/LIPIcs.MFCS.2017.60

1 Introduction

Graph connectivity is one of the fundamental properties in graph theory. Several connectivity measures like k-vertex connectivity, k-edge connectivity, hamiltonicity, etc. have been studied for graphs. Chartrand et al. [8] defined an interesting connectivity measure, called rainbow connectivity, which is defined as follows. Let G be a graph and $c_R : E(G) \rightarrow [k]$ be an edge-coloring of G. A *rainbow path* between $u, v \in V(G)$ is a path P from u to v such that for all $e, e' \in E(P)$, where $e \neq e'$ we have $c_R(e) \neq c_R(e')$. A graph with an edge-coloring is *rainbow-connected* if for every pair of vertices there is a rainbow path between them. In the Rainbow k-Coloring problem we are given a graph G, and the objective is to decide if there exists an edge-coloring $c_R : E(G) \rightarrow [k]$ such that for all $u, v \in V(G)$, there is a rainbow

[*] Due to space limitations most proofs have been omitted.
The research leading to these results received funding from the European Research Council under the European Union's Seventh Framework Programme (FP/2007-2013)/ ERC Grant Agreements no. 306992.

42nd International Symposium on Mathematical Foundations of Computer Science (MFCS 2017).
Editors: Kim G. Larsen, Hans L. Bodlaender, and Jean-Francois Raskin; Article No. 60; pp. 60:1–60:14
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

path between u and v in G. The problem has received attention both from graph theoretic and algorithmic point of view, the details of which can be found, for instance in [9, 23, 24].

Rainbow k-Coloring problem is notoriously hard. It was conjectured by Caro et al. [4] to be NP-complete already for $k = 2$. Indeed, by giving a polynomial time reduction from 3-SAT, this was confirmed by Chakraborty et al. [5]. Building on their results, Ananth et al. [3] later showed that Rainbow k-Coloring remains NP-complete for every $k \geq 2$. An alternate hardness proof was also given by Le and Tuza [21]. For the complexity of the problem on restricted graph classes, see e.g., [5, 6, 7, 8].

Impagliazzo et al. [16] introduced the Exponential time hypothesis (ETH), which has been used as a basis for proving qualitative lower bounds for computational problems. The ETH states that 3-SAT does not admit an algorithm running in time $2^{o(n)}n^{\mathcal{O}(1)}$, where n is the number of variables in the input 3-CNF formula. It has been shown that assuming ETH, several NP-hard problems like Independent Set, Hitting set, and Chromatic Number do not admit subexponential time algorithms (see the survey [25]).

Kowalik et al. [20] studied the fine-grained complexity of Rainbow k-Coloring and some of its variants. In particular, they showed that Rainbow k-Coloring admits neither an algorithm running in time $2^{o(|V(G)|^{3/2})}|V(G)|^{\mathcal{O}(1)}$, nor an algorithm running in time $2^{o(|E(G)|/\log |E(G)|)}|V(G)|^{\mathcal{O}(1)}$, unless ETH fails. They also studied a variant of Rainbow k-Coloring, called Subset Rainbow k-Coloring (to be defined shortly), which was introduced by Chakraborty et al. [5]. They showed that Subset Rainbow k-Coloring does not admit an algorithm running in time $2^{o(|E(G)|)}|V(G)|^{\mathcal{O}(1)}$ assuming ETH. In contrast, they designed an FPT algorithm for the problem running in time $|S|^{\mathcal{O}(|S|)}n^{\mathcal{O}(1)}$, where S is a part of the input. For $k = 2$, they obtained a faster algorithm running in time $2^{\mathcal{O}(|S|)}n^{\mathcal{O}(1)}$. Finally, they proposed yet another (parametric) variant of Rainbow k-Coloring, which they called Steiner Rainbow k-Coloring. Their lower bound result for Rainbow k-Coloring implies that Steiner Rainbow k-Coloring does not admit an algorithm running in time $2^{o|S|^{3/2}}n^{\mathcal{O}(1)}$. Moreover, their algorithm for Subset Rainbow k-Coloring gives an algorithm for Steiner Rainbow k-Coloring running in time $2^{\mathcal{O}(|S|^2 \log |S|)}n^{\mathcal{O}(1)}$.

Our results. We attempt to tighten the gaps in the study of fine-grained complexity of Rainbow k-Coloring and some of its variants, initiated by Kowalik et al. [20]. We now describe our results in detail.

The first problem that we study is Steiner Rainbow k-Coloring, which is formally defined below.

Steiner Rainbow k-Coloring **Parameter:** $|S|$
Input: A graph G and a vertex subset $S \subseteq V(G)$.
Question: Does there exist an edge-coloring $c_R : E(G) \to [k]$ such that for every $u, v \in S$, there is a rainbow path between u and v in G?

In Section 3, we show that for every $k \geq 3$, Steiner Rainbow k-Coloring does not admit an algorithm running in time $2^{o(|S|^2)}n^{\mathcal{O}(1)}$, under ETH. This resolves an open problem posed by Kowalik et al. [20]. To prove the result, we give a reduction from k-Coloring on graphs of maximum degree $2(k-1)$ which does not admit an algorithm running in time $2^{o(n)}n^{\mathcal{O}(1)}$, assuming ETH. Our reduction starts by computing a harmonious coloring of the (bounded degree) input instance of k-Coloring, which forms an essential step in the construction of S for the instance of Steiner Rainbow k-Coloring that we create. The idea of using harmonious coloring for proving lower bounds of the form $2^{o(\ell^2)}n^{\mathcal{O}(1)}$ was used by Agrawal et al. [1] to prove a lower bound for Split Contraction, when parameterized by the vertex cover number ℓ, of the input graph. Also, the idea of partitioning vertices of the input graph based on some coloring scheme was used by Cygan et al. [10] to prove ETH-based lower bounds for Graph Homomorphism and Subgraph Isomorphism.

The next problem we study is Rainbow k-Coloring, which is formally defined below.

Rainbow k-Coloring
Input: A graph G.
Question: Does there exist an edge-coloring $c_R : E(G) \to [k]$ such that for every $u, v \in V(G)$, there is a rainbow path between u and v in G?

Kowalik et al. [20] conjectured that for every $k \geq 2$, Rainbow k-Coloring does not admit an algorithm running in time $2^{o(|E(G)|)}n^{\mathcal{O}(1)}$, unless ETH fails. In Section 4, we resolve this conjecture for every $k \geq 3$. Again, we proceed with a reduction from k-Coloring on bounded degree graphs. Although, the general scheme of reduction is the same as the one we five for Steiner Rainbow k-Coloring, in this case the reduction is more involved. Furthermore, we require to distinguish between the cases for k being odd and even in the gadget construction. Also, to keep our gadgets simpler, we separate the case for $k = 3$ and $k > 3$.

Finally, we study the complexity of Subset Rainbow k-Coloring, which is formally defined below.

Subset Rainbow k-Coloring **Parameter:** $|S|$
Input: A graph G and a subset $S \subseteq V(G) \times V(G)$.
Output: An edge-coloring $c_R : E(G) \to [k]$ such that for every $(u, v) \in S$, there is a rainbow path between u and v in G, if it exists. Otherwise, return *no*.

In Section 5 we design an FPT algorithm running in time $2^{\mathcal{O}(|S|)}n^{\mathcal{O}(1)}$ for Subset Rainbow k-Coloring, for every fixed k. This resolves the conjecture of Kowalik et al. [20] regarding the existence of an algorithm running in time $2^{\mathcal{O}(|S|)}n^{\mathcal{O}(1)}$ for Subset Rainbow k-Coloring, and is an improvement over their algorithm, which runs in time $|S|^{\mathcal{O}(|S|)}n^{\mathcal{O}(1)}$, for $k \geq 3$. Our algorithm is based on the technique of color coding, which was introduced by Alon et al. [2]. Observe that Steiner Rainbow k-Coloring is a special case of Subset Rainbow k-Coloring. Hence, as a corollary we obtain an algorithm running in time $2^{\mathcal{O}(|S|^2)}n^{\mathcal{O}(1)}$ for Steiner Rainbow k-Coloring, which matches the lower bound we prove in Section 3.

2 Preliminaries

In this section, we state some basic definitions and introduce terminology from graph theory and algorithms. We also establish some of the notation that will be used throughout.

We denote the set of natural numbers by $\mathbb{N}$. For $k \in \mathbb{N}$, by $[k]$ we denote the set $\{1, 2, \ldots, k\}$. We use standard terminology from the book of Diestel [13] for the graph related terminologies which are not explicitly defined here. We consider finite simple graphs. For a graph G, by $V(G)$ and $E(G)$ we denote the vertex and edge sets of the graph G, respectively. For $v \in V(G)$, by $N_G(v)$ we denote the set $\{u \in V(G) \mid (v, u) \in E(G)\}$. We drop the subscript G from $N_G(v)$ when the context is clear. For $C, C' \subseteq V(G)$, we say that there is an edge between C and C' in G if there exists $u \in C$ and $v \in C'$ such that $(u, v) \in E(G)$. A *path* $P = (v_1, v_2, \ldots, v_\ell)$ is a graph with vertex and edge sets as $\{v_1, v_2, \ldots, v_\ell\}$ and $\{(v_i, v_{i+1}) \mid i \in [l-1]\}$, respectively.

A *harmonious coloring* of a graph G is a vertex coloring $\varphi : V(G) \to [k]$, with color classes $C_1, C_2, \ldots, C_k$ such that for each $i \in [k]$, C_i is an independent set in G and for all $i, j \in [k]$, where $i \neq j$ there is at most one edge between C_i and C_j in G. We use the following result for computing a harmonious coloring on bounded degree graphs.

▶ **Proposition 1** ([11, 14, 22, 26]). *Given a G with the degree of each vertex bounded by d, where d is a fixed constant. A harmonious coloring of G can be computed in time $\mathcal{O}(n^{\mathcal{O}(1)})$ using $\mathcal{O}(\sqrt{n})$ colors with each color class having at most $\mathcal{O}(\sqrt{n})$ vertices.*

3 Lower bound for Steiner Rainbow k-Coloring

In this section, we show that for every $k \geq 3$, Steiner Rainbow k-Coloring does not admit an algorithm running in time $2^{o(|S|^2)}n^{\mathcal{O}(1)}$, unless ETH fails. Towards this we give an appropriate reduction from k-Coloring on graphs of maximum degree $2(k-1)$. We note that k-Coloring does not admit an algorithm running in time $2^{o(n)}n^{\mathcal{O}(1)}$ unless ETH fails [17]. Moreover, assuming ETH, 3-Coloring does not admit an algorithm running in time $2^{o(n)}n^{\mathcal{O}(1)}$ on graphs of maximum degree 4 [18, 11]. This follows from the fact that 3-Coloring does not admit such an algorithm, and a reduction from an instance G of 3-Coloring to an equivalent instance G' of 3-Coloring, where G' is a graph with maximum degree 4 with $|V(G')| \in \mathcal{O}(|V(G)|)$ (see [15, Theorem 4.1]). In fact, we can show that k-Coloring does not admit an algorithm running in time $2^{o(n)}n^{\mathcal{O}(1)}$ on graphs of maximum degree $2(k-1)$ (folklore). This result can be obtained (inductively) by giving a reduction from an instance G of $(k-1)$-Coloring on graphs of degree at most $2(k-2)$ to an instance of k-Coloring on a graphs of bounded average degree (by adding global vertex), and then using an approach similar to that in Theorem 4.1 in [15] we can obtain an (equivalent) instance of k-Coloring where the degree of the graph is bounded by $2(k-1)$.

Given an instance G of k-Coloring on n vertices and degree bounded by $2(k-1)$, we start by computing a harmonious coloring φ of G with $t \in \mathcal{O}(\sqrt{n})$ color classes such that each color class contains at most $\mathcal{O}(\sqrt{n})$ vertices using Proposition 1. Let $C_1, C_2, \ldots, C_t$ be the color classes of φ. Recall that for $i, j \in [t]$ with $i \neq j$ there is at most one edge between C_i and C_j in G. Moreover, C_i is an independent set in G, where $i \in [t]$. We create an instance G' of k-Coloring which has a harmonious coloring φ' with color classes $C'_1, C'_2, \ldots, C'_t$ such that for all $i, j \in [t]$, $i \neq j$ we have exactly one edge between C_i and C_j. Initially, we have $G = G'$ and $C'_i = C_i$, for all $i \in [t]$. For each $i, j \in [t]$, $i \neq j$ such that there is no edge between C_i and C_j in G we add two new vertices a_{ij} and a_{ji} to $V(G')$ and add the edge (a_{ij}, a_{ji}) to $E(G')$. Furthermore, we add a_{ij} to C'_i and a_{ji} to C'_{ji}. Observe that $|V(G')| \in \mathcal{O}(n)$, $|E(G')| \in \mathcal{O}(n)$, and for each $i \in [t]$, $|C'_i| \in \mathcal{O}(\sqrt{n})$. Also, for each $i, j \in [t]$, $i \neq j$ there is exactly one edge between C'_i and C'_j in G'. It is easy to see that G is a *yes* instance of k-Coloring if and only if G' is a *yes* instance of k-Coloring.

Hereafter, we will be working with the instance G' of k-Coloring, together with its harmonious coloring φ' with color classes $C'_1, C'_2, \ldots, C'_t$. Moreover, for $i, j \in [t]$, $i \neq j$ there is exactly one edge between C'_i and C'_j in G'.

We now move to the description of creating an equivalent instance $(\tilde{G}, S)$ of Steiner Rainbow k-Coloring, where $k \geq 3$. Initially, we have $V(\tilde{G}) = V(G')$. For $(u, v) \in E(G')$ we add $k-3$ new vertices $x_1^{uv}, x_2^{uv}, \ldots, x_{k-3}^{uv}$ to $\tilde{G}$ and add all the edges in the path $(u, x_1^{uv}, \ldots, x_{k-3}^{uv}, v)$ to $E(\tilde{G})$. Note that for $k = 3$ we do not any new vertex and directly add the edge (u, v) to $\tilde{G}$. For each $i \in [t]$ we add a vertex c_i to $\tilde{G}$ and add all the edges in $\{(c_i, v) \mid v \in C'_i\}$ to $E(\tilde{G})$. Finally, we set $S = \{c_i \mid i \in [t]\}$. Notice that $|S| \in \mathcal{O}(\sqrt{n})$. In the following lemma we establish that G' is a *yes* instance of k-Coloring if and only if $(\tilde{G}, S)$ is a *yes* instance of Steiner Rainbow k-Coloring.

▶ **Lemma 2.** *G' is a* yes *instance of k-Coloring if and only if $(\tilde{G}, S)$ is a* yes *instance of Steiner Rainbow k-Coloring.*

Proof. In the forward direction, let G' be a *yes* instance of k-Coloring, and $c : V(G') \to [k]$ be one of its solution. We create a coloring $c_R : E(\tilde{G}) \to [k]$ as follows. For $i \in [t]$ and $v \in C'_i$ we set $c_R(c_i, v) = c(v)$. For $i, j \in [t]$, $i \neq j$ let u, v be the (unique) vertices in C'_i and C'_j such that $(u, v) \in E(G')$. We now describe the value of c_R for edges in the path $P = (u, x_1^{uv}, \ldots, x_{k-3}^{uv}, v)$. Notice that $|E(P)| = k - 2$, and we arbitrarily assign distinct integers in $[k] \setminus \{c_R(c_i, u), c_R(c_j, v)\}$ to $c_R(e)$, where $e \in E(P)$. Since c is a proper coloring of G', $c_R(c_i, u) = c(u) \neq c(v) = c_R(c_j, v)$. This together with the definition of c_R for edges in P implies that there is a rainbow path, namely $(c_i, u, x_1^{uv}, \ldots, x_{k-3}^{uv}, v, c_j)$ in $\tilde{G}$ between c_i and c_j. This concludes the proof in the forward direction.

In the reverse direction, let $(\tilde{G}, S)$ be a *yes* instance of Steiner Rainbow k-Coloring, and $c_R : E(\tilde{G}) \to [k]$ be one of its solution. We create a coloring $c : V(G') \to [k]$ as follows. For $i \in [t]$ and $v \in C'_i$, we let $c(v) = c_R(c_i, v)$. We show that c is a solution to k-Coloring in G'. Consider $(u, v) \in E(G')$, and let $u \in C'_i$ and $v \in C'_j$. Note that we have $i \neq j$. Let P be a rainbow path between c_i and c_j in $\tilde{G}$. By the construction of $\tilde{G}$, we have $N_{\tilde{G}}[c_i] \cap N_{\tilde{G}}[c_j] = \emptyset$. Moreover, since P is a rainbow path, it can contain at most k edges. Recall that $N_{\tilde{G}}(c_i) = C'_i$, $N_{\tilde{G}}(c_j) = C'_j$, and there is exactly one path with at most $k - 2$ edges between a vertex in C'_i and a vertex in C'_j, namely $(c_i, u, x_1^{uv}, \ldots, x_{k-3}^{uv}, v, c_j)$. This together with the construction of c implies that $c(u) \neq c(v)$. This concludes the proof. ◀

▶ **Theorem 3.** Steiner Rainbow k-Coloring *does not admit an algorithm running in time* $2^{o(|S|^2)} n^{\mathcal{O}(1)}$, *unless* ETH *fails. Here, n is the number of vertices in the input graph.*

4 Lower bound for Rainbow k-Coloring

In this section, we show that for every $k \geq 3$, Rainbow k-Coloring does not admit an algorithm running in time $2^{o(|E(G)|)} n^{\mathcal{O}(1)}$, unless ETH fails. We give different reductions for the case when $k = 3$ (Section 4.1), k is an even number greater than 3 (Section 4.2), and k is an odd number greater than 4 (Section 4.3). We note that although the approach used for the proving lower bound for Rainbow 3-Coloring is extensible to Rainbow k-Coloring when k is odd, it unnecessarily adds to complexity of the reduction. Moreover, the approach we follow for showing the lower bound result for $k > 3$, where k is an odd number, introduces some technical issues when we try to extend it for $k = 3$.

Towards proving our lower bound result, we give an appropriate reduction from k-Coloring on graphs of maximum degree $2(k-1)$, which does not admit an algorithm running in time $2^{o(n)} n^{\mathcal{O}(1)}$ unless ETH fails. The key idea behind the reduction is the same as that presented in Section 3, but for this case it is more involved. Before moving on to the description of the reductions we define a graph that will be useful in our reductions.

A *clique sequence* $\mathbb{Z}_{n,t} = (Z_1, Z_2, \ldots Z_t)$ of order (n, t) is a graph defined as follows. We have $V(\mathbb{Z}_{k,t}) = \uplus_{i \in [t]} Z_i$, where $|Z_i| = n$ for all $i \in [t]$. For each $i \in [t]$, all the edges in $\{(z, z') \mid z, z' \in Z_i\}$ are present in $E(\mathbb{Z}_{n,t})$, *i.e.* Z_i is a clique. Furthermore, for all $i \in [t-1]$ all the edges in $\{(z, z') \mid z, \in Z_i, x' \in Z_{i+1}\}$ are present in $E(\mathbb{Z}_{n,t})$.

4.1 Lower bound for Rainbow 3-Coloring

In this section, we show that Rainbow 3-Coloring does not admit an algorithm running in time $2^{o(|E(G)|)} n^{\mathcal{O}(1)}$, where n is the number of vertices in the input graph G.

Let G be an instance of 3-Coloring on n vertices with maximum degree bounded by 4. We start by computing (in polynomial time) a harmonious coloring φ of G with $t \in \mathcal{O}(\sqrt{n})$ color classes such that each color class contains at most $\mathcal{O}(\sqrt{n})$ vertices using Proposition 1.

Let $C_1, C_2, \ldots, C_t$ be the color classes of φ. From the discussion in Section 3, we assume that for $i, j \in [t]$, $i \neq j$ there is exactly one edge between C_i and C_j in G. We construct an instance G' of Rainbow 3-Coloring as follows.

- *Color class gadget.* Consider $i \in [t]$. The color class gadget $\mathcal{C}_i$ comprises of the set C_i, two vertices c_i, b_i, and a clique U_i on 3 vertices with vertex set $\{u^i_1, u^i_2, u^i_3\}$. We add all the edges in $\{(v, c_i), (v, b_i), (v, u^i_1), (v, u^i_2), (v, u^i_3) \mid v \in C_i\}$ to $E(\mathcal{C}_i)$. Also, we add the edge (b_i, c_i) to $E(\mathcal{C}_i)$.
- *Connection between color class gadgets.* Consider $i, j \in [t]$ such that $i \neq j$. We add all the edges in $\{(b_i, u^j_\ell) \mid \ell \in [3]\}$ to $E(G')$. Furthermore, we add all the edges $\{(u^i_\ell, u^j_{\ell'}) \mid \ell, \ell' \in [3]\}$ to $E(G')$. Note that $\{u^{i'}_\ell \mid i' \in [t], \ell \in [3]\}$ induces a clique in G'.
- *Encoding edges.* For $i, j \in [t]$, $i \neq j$ we add the unique edge (u, v) between C_i and C_j with $u \in C_i$ and $v \in C_j$ to G'. Note that this is same as adding all the edges in $E(G)$ to $E(G')$.

This finishes the description of the instance G' of Rainbow 3-Coloring. We note that some of the edges in G' are not necessary for the correctness of the reduction. However, they are added to reduce the number of pairs for which we need to argue about the existence of a rainbow path. Before moving on to the proof of equivalence between these instances, we create an edge-coloring $c_R : E(G') \to [3]$. Here, we create c_R based on a solution c to 3-Coloring in G, assuming that G is a *yes* instance of 3-Coloring. We will follow computation modulo k, and therefore color 0 is same as color k.

▶ **Definition 4.** Given a solution c to 3-Coloring in G, we construct $c_R : E(G') \to [3]$ as follows.

1. For $i \in [t]$, and $v \in C_i$ set $c_R(v, c_i) = c(v)$, $c_R(v, b_i) = c(v)$, and for $\ell \in [3]$, $c_R(v, u^i_\ell) = \ell$.
2. For $i, j \in [t]$, $i \neq j$ let (u, v) be the unique edge between C_i and C_j. We set $c_R(u, v)$ to be the unique integer in $[3] \setminus \{c(u), c(v)\}$. Here, the uniqueness is guaranteed by the fact that c is a proper 3-coloring of G, promising that $c(u) \neq c(v)$.
3. For $i \in [t]$ set $c_R(b_i, c_i) = 3$, $c_R(u^i_1, u^i_2) = 3$, $c_R(u^i_2, u^i_3) = 2$, and $c_R(u^i_3, u^i_1) = 1$.
4. For $i, j \in [t]$, $i \neq j$ and $\ell \in [3]$ set $c_R(b_i, u^j_\ell) = \ell - 1$.
5. For $i, j \in [t]$, $i \neq j$ and $\ell \in [3]$ set $c_R(u^i_\ell, u^j_\ell) = \ell$. Furthermore, for $\ell' \in [3] \setminus \{\ell\}$ we set $c_R(u^i_\ell, u^j_{\ell'}) = \hat{\ell}$, where $\hat{\ell}$ is the unique integer in $[3] \setminus \{\ell, \ell'\}$.

Next, we prove some lemmata that will be useful in establishing the equivalence between the instance G of 3-Coloring and the instance G' of Rainbow 3-Coloring.

▶ **Lemma 5.** *For $i, j \in [t]$, where $i \neq j$, let (u^*, v^*) be the unique edge between C_i and C_j with $u^* \in C_i$ and $v^* \in C_j$. There is exactly one path, namely (c_i, u^*, v^*, c_j) in G', between c_i and c_j that has at most 3 edges.*

▶ **Lemma 6.** *Let G be a yes instance of 3-Coloring, and c be one of its solution. Furthermore, let $c_R : E(G') \to [3]$ be the coloring given by Definition 4 for the coloring c of G. Then for all $i \in [t]$, and $u, v \in \mathcal{C}_i$ there is a rainbow path between u and v in G'.*

▶ **Lemma 7.** *Let G be a yes instance of 3-Coloring, and c be one of its solution. Furthermore, let $c_R : E(G') \to [3]$ be the coloring given by Definition 4 for the coloring c of G. Then for all $i, j \in [t]$, $i \neq j$ for all $u \in \mathcal{C}_i$ and $v \in \mathcal{C}_j$ there is a rainbow path between u and v in G'.*

We now establish equivalence between the instance G of 3-Coloring and the instance G' of Rainbow 3-Coloring.

▶ **Lemma 8.** *G is a yes instance of* 3-Coloring *if and only if G' is a yes instance of* Rainbow 3-Coloring.

Proof. In the forward direction, let G be a *yes* instance of 3-Coloring, and $c : V(G) \to [3]$ be one of its solution. Let $c_R : E(G') \to [3]$ be the coloring given by Definition 4 for the given coloring c of G. From Lemma 6 and 7 it follows that c_R is a solution to Rainbow 3-Coloring in G'.

In the reverse direction, let G' be a *yes* instance of Rainbow 3-Coloring, and $c_R : E(G') \to [3]$ be one of its solution. We create a coloring $c : V(G) \to [3]$ as follows. For $i \in [t]$ and $v \in C_i$, we let $c(v) = c_R(c_i, v)$. We show that c is a valid solution to 3-Coloring in G. Consider $(u, v) \in E(G)$, and let $u \in C_i$ and $v \in C_j$. Note that we have $i \neq j$. Let P be a rainbow path between c_i and c_j in G'. Note that P can have at most 3 edges. By Lemma 5 we know that $P = (c_i, u, v, c_j)$, therefore by construction of c, we have $c_R(c_i, u) = c(u) \neq c(v) = c_R(c_i, v)$. This concludes the proof. ◀

▶ **Theorem 9.** Rainbow 3-Coloring *does not admit an algorithm running in time $2^{o(|E(G)|)}n^{\mathcal{O}(1)}$, unless* ETH *fails. Here, n is the number of vertices in the input graph.*

4.2 Lower Bound for Rainbow k-Coloring, $k > 3$ and even

In this section, we show that Rainbow k-Coloring does not admit an algorithm running in time $2^{o(|E(G)|)}n^{\mathcal{O}(1)}$, for every even k where $k > 3$. Here, n is the number of vertices in the input graph.

Let G be an instance of k-Coloring on n vertices with maximum degree bounded by $2(k-1)$. Here, $k > 3$ and k is an even number. We start by computing (in polynomial time) a harmonious coloring φ of G with $t \in \mathcal{O}(\sqrt{n})$ color classes such that each color class contains at most $\mathcal{O}(\sqrt{n})$ vertices using Proposition 1. Let $C_1, C_2, \ldots, C_t$ be the color classes of φ with exactly one edge between C_i and C_j in G, where $i, j \in [t]$. We modify the graph G and its harmonious coloring φ, to obtain a more structured instance, which will be useful later. For each $i \in [t]$, we add k new vertices $v^*_{i1}, v^*_{i2}, \ldots, v^*_{ik}$ to $V(G)$, and add them to C_i. We continue to call the modified graph as G and its harmonious coloring as φ with color classes $C_1, C_2, \ldots, C_t$. We note that $\{v^*_{ij} \mid i \in [t], j \in [k]\}$ induce an independent set in G. The purpose of adding these k new vertices is to ensure that if G is a *yes* instance of k-Coloring then there is a k-coloring c of G, such that for each $i \in [t]$ and $j \in [k]$, we have $c^{-1}(j) \cap C_i \neq \emptyset$. This will be helpful in simplifying some of the arguments later. Observe that the original instance is a *yes* instance of a k-Coloring is and only if the modified instance is a *yes* instance of k-Coloring. Moreover, given a k-coloring of G (modified graph), in polynomial time we can obtain another k-coloring c' of G such that for all $i \in [t], j \in [k]$ we have $c(v^*_{ij}) = j$. Also, we have $|V(G)| \in \mathcal{O}(n)$, and $|E(G)| \in \mathcal{O}(n)$, where n is the number of vertices in the original instance. Hereafter, whenever we talk about a solution c to k-Coloring in G (if it exists) we will assume (without explicitly mentioning) that for all $i \in [t]$ and $p \in [k]$ we have $C_i \cap c^{-1}(p) \neq \emptyset$. We now move to the description of the reduction.

We proceed by describing color class gadget $\mathcal{C}_i$, corresponding to the color class C_i, where $i \in [t]$, and gadgets to encode edges in G. Then we state the connection between various color class gadgets and edge gadgets. We let $k = 2\ell$, where $\ell \in \mathbb{N}$ and $\ell > 1$. We create an instance G' of Rainbow k-Coloring as described below.

- *Color class gadget.* Consider $i \in [t]$. The color class gadget $\mathcal{C}_i$ comprises of the set C_i, a vertex c_i, and a clique sequence $\mathbb{Z}_i = (U^i_1 \cup D^i_1, \ldots, U^i_{\ell-1} \cup D^i_{\ell-1})$ of order $(2k, \ell-1)$. Here, for each $i \in [\ell-1]$ we have $|U_i| = |D_i| = k$. For $r \in [\ell-1]$ we let $U^i_r = \{u^i_{rp} \mid p \in [k]\}$, and $D^i_r = \{d^i_{rp} \mid p \in [k]\}$. We add all the edges in $\{(c_i, v) \mid v \in C_i\}$ to $E(\mathcal{C}_i)$. Also, we add all the edges in $\{(v, w) \mid v \in C_i, w \in U^i_1 \cup D^i_1\}$ to $E(\mathcal{C}_i)$.

- *Connection between color class gadgets.* For each $i, j \in [t]$ where $i \neq j$, we add all the edges in $\{(w, w') \mid w \in U^i_{\ell-1} \cup D^i_{\ell-1}, w' \in U^j_{\ell-1} \cup D^j_{\ell-1}\}$ to $E(G')$.
- *Edge gadget.* Consider $i, j \in [t]$ with $i < j$. Recall that there is exactly one edge between C_i and C_j. Corresponding to this edge we create a path $P = (x^{ij}_1, \dots, x^{ij}_{\ell-2}, z_{ij}, x^{ji}_{\ell-2}, \dots, x^{ji}_1)$ on $k - 3$ vertices, and add it to G'. We note that whenever we say vertex z_{ji} it refers to the vertex z_{ij} *i.e.* z_{ij} and z_{ji} denotes the same vertex.
- *Connection between color class gadgets and edge gadgets.* Consider $i, j \in [t]$, where $i < j$. Let (u^*_i, v^*_j) be the unique edge between C_i and C_j with $u^*_i \in C_i$ and $v^*_j \in C_i$. We add the edges $(u^*_i, x^{ij}_1), (x^{ji}_1, v^*_j)$ to $E(G')$. Notice that when $\ell = 2$ x^{ij}_1 does not exists. In this case, we add the edges $(u^*_i, z), (z, v^*_j)$ to $E(G')$. For each $r \in [\ell - 2]$ we add all the edges in $\{(x^{ij}_r, w) \mid w \in U^i_r \cup D^i_r\}$ to $E(G')$. Similarly, we add all the edges in $\{(x^{ji}_r, w) \mid w \in U^j_r \cup D^j_r\}$ to $E(G')$. Also, we add all the edges in $\{(z_{ij}, u) \mid u \in U^i_{\ell-1} \cup D^i_{\ell-1} \cup U^j_{\ell-1} \cup D^j_{\ell-1}\}$ to $E(G')$.

This finishes the construction of instance G' of Rainbow k-Coloring for the given instance G of k-Coloring. Before moving on to proving the equivalence between these instances, we create an edge-coloring $c_R : E(G') \to [k]$. Here, we create c_R based on a solution c to k-Coloring in G, assuming that is G a *yes* instance of k-Coloring. We will follow computation modulo k (color 0 is same as color k).

▶ **Definition 10.** Given a solution c to k-Coloring in G, we construct $c_R : E(G') \to [k]$ as follows.

1. For $i \in [t]$, and $v \in C_i$ we set $c_R(v, c_i) = c(v)$.
2. For $i, j \in [t]$, $i < j$ let (u^*_i, v^*_j) be the unique edge between C_i and C_j. Consider the path $P = (u^*_i, x^{ij}_1, \dots x^{ij}_{\ell-2}, z_{ij}, x^{ji}_{\ell-2}, \dots x^{ji}_1, v^*_j)$. We arbitrarily assign unique integers in $[k] \setminus \{c(u^*_i), c(v^*_j)\}$ to $c_R(e)$, for each $e \in E(P)$.
3. For $i \in [t]$, a vertex $v \in C_i$, and $p \in [k]$ we set $c_R(v, u^i_{1p}) = p - 1$, and $c_R(v, d^i_{1p}) = p$.
4. For $i \in [t]$, $r \in [\ell - 1]$, and $p, q \in [k]$ we set $c_R(d^i_{rp}, u^i_{rq}) = p$.
5. For $i, j \in [t]$, where $i \neq j$, $r \in [\ell - 1]$, and $p \in [k]$ we set $c_R(x^{ij}_r, u^i_{rp}) = p$, and $c_R(x^{ij}_r, d^i_{rp}) = p + 1$.
6. For $i \in [t]$, $r \in [\ell - 2]$, $p, q \in [k]$ we set $c_R(d^i_{(r+1)p}, d^i_{rq}) = p$, and $c_R(u^i_{rp}, u^i_{(r+1)q}) = p$.
7. For $i, j \in [t]$ where $i \neq j$, $p, q \in [k]$ we set $c_R(u^i_{(\ell-1)p}, d^j_{(\ell-1)q}) = p$, $c_R(u^i_{(\ell-1)p}, z_{ij}) = p$, and $c_R(d^i_{(\ell-1)p}, z_{ij}) = p + 1$.
8. For $i \in [t]$, $r \in [\ell - 2]$, $p, q \in [k]$ we set $c_R(u^i_{rp}, d^i_{(r+1)q}) = q$ and $c_R(u^i_{(r+1)p}, d^i_{rq}) = p$.
9. For all $i \in [t]$, $r \in [\ell - 1]$, $p, q \in [k]$, where $p \neq q$ we set $c_R(u^i_{rp}, u^i_{rq}) = k$.
10. For all the remaining edges in $E(G')$, c_R assigns it an integer in $[k]$ arbitrarily.

Next, we prove some lemmata that will be useful in establishing equivalence between the instance G of k-Coloring and the instance G' of Rainbow k-Coloring.

▶ **Lemma 11.** *For $i, j \in [t]$, where $i \neq j$, let P be a path between c_i and c_j with at most k edges in G'. If $\ell > 2$ then P contains the edge $(x^{ij}_{\ell-2}, z_{ij})$. Otherwise, P contains the edge (u, z_{ij}), where u is the unique vertex in C_i that in adjacent to a vertex in C_j.*

▶ **Lemma 12.** *For $i, j \in [t]$, where $i \neq j$ let (u^*, v^*) be the unique edge between C_i and C_j with $u^* \in C_i$ and $v^* \in C_j$. There is exactly one path, namely $(c_i, u^*, x^{ij}_1, \dots, x^{ij}_{\ell-2}, z_{ij}, x^{ji}_{\ell-2}, \dots, x^{ji}_1$ $, v^*, c_j)$ in G' between c_i and c_j that has at most k edges.*

▶ **Lemma 13.** *Let G be a yes instance of k-*Coloring*, and c be one of its solution. Furthermore, let $c_R : E(G') \rightarrow [k]$ be the coloring given by Definition 10 for the coloring c of G. For all $i \in [t]$, and $u, v \in V(\mathcal{C}_i) \cup \{z_{ij} \mid j \in [k] \setminus \{i\}\} \cup \{x_r^{ij} \mid j \in [t] \setminus \{i\}, r \in [\ell - 2]\}$ there is a rainbow path between u and v in G'.*

▶ **Lemma 14.** *Let G be a yes instance of k-*Coloring*, and c be one of its solution. Furthermore, let $c_R : E(G') \rightarrow [k]$ be the coloring given by Definition 10 for the coloring c of G. For all $i, j \in [t]$ where $i \neq j$, $u \in V(\mathcal{C}_i) \cup \{z_{ij'} \mid j' \in [k] \setminus \{i\}\} \cup \{x_r^{ij'} \mid j' \in [t] \setminus \{i\}, r \in [\ell-2]\}$ and $v \in V(\mathcal{C}_j) \cup \{z_{ji'} \mid i' \in [k] \setminus \{j\}\} \cup \{x_r^{ji'} \mid i' \in [t] \setminus \{j\}, r \in [\ell - 2]\}$ there is a rainbow path between u and v in G'.*

We now establish equivalence between the instance G of k-Coloring and the instance G' of Rainbow k-Coloring.

▶ **Lemma 15.** *G' is a yes instance of k-*Coloring* if and only if G' is a yes instance of* Rainbow *k-*Coloring*.*

▶ **Theorem 16.** Rainbow k-Coloring *does not admit an algorithm running in time $2^{o(|E(G)|)} n^{\mathcal{O}(1)}$, unless* ETH *fails. Here, n is the number of vertices in the input graph, and k is an even number greater than* 3.

4.3 Lower Bound for Rainbow k-Coloring, $k > 3$ and odd

In this section, we show that Rainbow k-Coloring does not admit an algorithm running in time $2^{o(|E(G)|)} n^{\mathcal{O}(1)}$, for every odd k where $k > 3$. Here, n is the number of vertices in the input graph.

Let G be an instance of k-Coloring on n vertices with maximum degree bounded by $2(k-1)$. Here, $k > 3$ and k is an odd number. We start by computing (in polynomial time) a harmonious coloring φ of G with $t \in \mathcal{O}(\sqrt{n})$ color classes such that each color class contains at most $\mathcal{O}(\sqrt{n})$ vertices using Proposition refprop:compute-harmonious-coloring. Let $C_1, C_2, \ldots, C_t$ be the color classes of φ. From the discussion in Section 3, we assume that for $i, j \in [t]$, $i \neq j$ there is exactly one edge between C_i and C_j in G. As discussed in Section 4.2, we modify the graph G and its harmonious coloring φ, to obtain a more structured (equivalent) instance of k-Coloring. This is achieved by adding k new vertices $v_{i1}^*, v_{i2}^*, \ldots, v_{ik}^*$ to C_i (and G) for each $i \in [t]$. The purpose of adding these k new vertices is to ensure that if G is a *yes* instance of k-Coloring then there is a k-coloring c of G, such that for each $i \in [t]$ and $j \in [k]$, we have $c^{-1}(j) \cap C_i \neq \emptyset$. Hereafter, whenever we talk about a solution c to k-Coloring in G (if it exists) we will assume (without explicitly mentioning) that for all $i \in [t]$ and $p \in [k]$ we have $C_i \cap c^{-1}(p) \neq \emptyset$.

We move to the description of the reduction. We first describe the color class gadget $\mathcal{C}_i$, corresponding to each color class C_i, where $i \in [t]$, and gadgets to encode edges in G. We also have a link vertex which is connected to all color class gadgets (but not all vertices). After this, we state connections between color class gadgets and edge gadgets. We let $k = 2\ell + 1$, where $\ell \in \mathbb{N}$ and $\ell \geq 2$. We create an instance G' of Rainbow k-Coloring as follows.

- *Color class gadget.* Consider $i \in [t]$. The color class gadget $\mathcal{C}_i$ comprises of the set C_i, a vertex c_i, and a clique sequence $\mathbb{Z}_i = (U_1^i \cup D_1^i, \ldots, U_{\ell-1}^i \cup D_{\ell-1}^i)$ of order $(2k, \ell - 1)$. Here, for each $i \in [\ell - 1]$ we have $|U_i| = |D_i| = k$. For $r \in [\ell - 1]$ we let $U_r^i = \{u_{rp}^i \mid p \in [k]\}$ and $D_r^i = \{d_{rp}^i \mid p \in [k]\}$. We add all the edges in $\{(c_i, v) \mid v \in C_i\}$ to $E(\mathcal{C}_i)$. Also, we add all the edges in $\{(v, w) \mid v \in C_i, w \in U_1^i \cup D_1^i\}$ to $E(\mathcal{C}_i)$.
- *Link vertex and its connection to color class gadgets.* We add a vertex z to G'. For each $i \in [t]$, we add all the edges in $\{(z, w) \mid w \in U_{\ell-1}^i \cup D_{\ell-1}^i\}$ to $E(G')$.

- *Edge gadget.* Consider $i, j \in [t]$ with $i \neq j$. Recall that there is exactly one edge between C_i and C_j. Corresponding to this edge we create a path $P = (x_1^{ij}, \ldots, x_{\ell-1}^{ij}, x_{\ell-1}^{ji}, \ldots, x_1^{ji})$ on $k-3$ vertices, and add it to G'.
- *Connection between color class gadgets and edge gadgets.* Consider $i, j \in [t]$, where $i \neq j$. Let (u_i^*, v_j^*) be the unique edge between C_i and C_j with $u_i^* \in C_i$ and $v_j^* \in C_i$. We add the edges $(u_i^*, x_1^{ij}), (x_1^{ji}, v_j^*)$ to $E(G')$. For each $r \in [\ell-1]$ we add all the edges in $\{(x_r^{ij}, w) \mid w \in U_r^i \cup D_r^i\}$ to $E(G')$. Similarly, we add all the edges in $\{(x_r^{ji}, w) \mid w \in U_r^j \cup D_r^j\}$ to $E(G')$.

This finishes the construction of the instance G' of Rainbow k-Coloring for the given instance G of k-Coloring. Before moving on to proving the equivalence between these instances, we create an edge-coloring $c_R : E(G') \to [k]$. Here, we create c_R based on a solution c to k-Coloring in G, assuming that is G a *yes* instance of k-Coloring. We will follow computation modulo k (color 0 is same as color k).

▶ **Definition 17.** Given a solution c to k-Coloring in G, we construct $c_R : E(G') \to [k]$ as follows.

1. For $i \in [t]$, and $v \in C_i$ we set $c_R(v, c_i) = c(v)$.
2. For $i, j \in [t]$, $i \neq j$ let (u_i^*, v_j^*) be the unique edge between C_i and C_j. Consider the path $P = (u_i^*, x_1^{ij}, \ldots x_{\ell-1}^{ij}, x_{\ell-1}^{ji}, \ldots x_1^{ji}, v_j^*)$. We arbitrarily assign unique integers in $[k] \setminus \{c(u_i^*), c(v_j^*)\}$ to $c_R(e)$, for each $e \in E(P)$.
3. For $i \in [t]$, a vertex $v \in C_i \cup \{x_1^{ij} \mid j \in [t] \setminus \{i\}\}$, and $p \in [k]$ we set $c_R(v, u_{1p}^i) = p-1$, and $c_R(v, d_{1p}^i) = p$.
4. For $i \in [t]$, $r \in [\ell-1]$, and $p, q \in [k]$ we set $c_R(d_{rp}^i, u_{rq}^i) = p$.
5. For $i, j \in [t]$, where $i \neq j$, $r \in [\ell-1]$, and $p \in [k]$ we set $c_R(x_r^{ij}, u_{rp}^i) = p$, and $c_R(x_r^{ij}, d_{rp}^i) = p+1$.
6. For $i \in [t]$, $r \in [\ell-2]$, $p, q \in [k]$ we set $c_R(d_{(r+1)p}^i, d_{rq}^i) = p$, and $c_R(u_{rp}^i, u_{(r+1)q}^i) = p$.
7. For $i \in [t]$, $p \in [k]$ we set $c_R(u_{(\ell-1)p}^i, z) = p$, and $c_R(d_{(\ell-1)p}^i, z) = p-1$.
8. For $i \in [t]$, $r \in [\ell-2]$, $p, q \in [k]$ we set $c_R(u_{rp}^i, d_{(r+1)q}^i) = q$ and $c_R(u_{(r+1)p}^i, d_{rq}^i) = p$.
9. For all $i \in [t]$, $r \in [\ell]$, $p, q \in [k]$, where $p \neq q$ we set $c_R(u_{rp}^i, u_{rq}^i) = k$.
10. For all the remaining edges in $E(G')$, c_R assigns it an integer in $[k]$ arbitrarily.

Next, we prove some lemmata that will be useful in establishing the equivalence between the instance G of k-Coloring and the instance G' of Rainbow k-Coloring.

▶ **Lemma 18.** *For $i, j \in [t]$, where $i \neq j$, let P be a path between c_i and c_j with at most k edges in G'. Then $(x_{\ell-1}^{ij}, x_{\ell-1}^{ij}) \in E(P)$.*

▶ **Lemma 19.** *For $i, j \in [t]$, where $i \neq j$ let (u_i^*, v_j^*) be the unique edge between C_i and C_j with $u_i^* \in C_i$ and $v_j^* \in C_j$. There is exactly one path, namely $(c_i, u_i^*, x_1^{ij}, \ldots, x_{\ell-1}^{ij}, x_{\ell-1}^{ji}, \ldots, x_1^{ji}, v_j^*$ $, c_j)$ in G' between c_i and c_j that has at most k edges.*

▶ **Lemma 20.** *Let G be a yes instance of k-Coloring, and c be one of its solution. Furthermore, let $c_R : E(G') \to [k]$ be the coloring given by Definition 17 for the coloring c of G. For all $i \in [t]$, and $u, v \in V(\mathcal{C}_i) \cup \{x_r^{ij} \mid j \in [t] \setminus \{i\}, r \in [\ell-1]\} \cup \{z\}$ there is a rainbow path between u and v in G'.*

▶ **Lemma 21.** *Let G be a yes instance of k-Coloring, and c be one of its solution. Furthermore, let $c_R : E(G') \to [k]$ be the coloring given by Definition 17 for the coloring c of G. For all $i, j \in [t]$ where $i \neq j$, $u \in V(\mathcal{C}_i) \cup \{x_r^{ij'} \mid j' \in [t] \setminus \{i\}, r \in [\ell-1]\}$ and $v \in \mathcal{C}_j \cup \{x_r^{ji'} \mid i' \in [t] \setminus \{j\}, r \in [\ell-1]\}$ there is a rainbow path between u and v in G'.*

▶ **Lemma 22.** *G' is a yes instance of k-*Coloring *if and only if G' is a yes instance of* Rainbow k-Coloring*.*

▶ **Theorem 23.** Rainbow k-Coloring *does not admit an algorithm running in time $2^{o(|E(G)|)}n^{\mathcal{O}(1)}$, unless* ETH *fails. Here, n is the number of vertices in the input graph, and k is an odd number greater than 3.*

5 FPT Algorithm for Subset Rainbow k-Coloring

In this section, we design an FPT algorithm running in time $\mathcal{O}(2^{|S|}n^{\mathcal{O}(1)})$ for Subset Rainbow k-Coloring, when parameterized by $|S|$. Our algorithm is based on the technique of color coding, which was first introduced by Alon et al. [2]. We first describe a randomized algorithm for Subset Rainbow k-Coloring, which we derandomize using splitters.

The intuition behind the algorithm is as follows. Let (G, S) be an instance of Subset Rainbow k-Coloring on n vertices and m edges. For a solution $c_R : E(G) \to [k]$, to Subset Rainbow k-Coloring in (G, S) the following holds. For each $(u, v) \in S$, there exist a path P from u to v in G with at most k edges such that for all $e, e' \in E(P)$, where $e \neq e'$ we have $c_R(e) \neq c_R(e')$. Therefore, at most $k|S|$ edges in G seems to be "important" for us, *i.e.* if we color at most $k|S|$ edges "nicely" then we would obtain the desired soultion. To capture this, we start by randomly coloring edges in G, hoping that with sufficiently high probability we obtain a coloring that colors the desired set of edges "nicely". Once we have obtained such a "nice" coloring, we employ the algorithm of Kowalik and Lauri [19] to check if there is a rainbow path for each $(u, v) \in S$. We note that we use the algorithm given by [19] instead of the one in [28] because the latter requires exponential space.

Algorithm Rand-SRC. Let $c : E(G) \to [k]$ be a coloring of $E(G)$, where each edge is colored with one of the colors in $[k]$ uniformly and independently at random. If for each $(u, v) \in S$, there is rainbow path between u and v in G' with edge-coloring c then the algorithm return c as a solution to Subset Rainbow k-Coloring in (G, S). Otherwise, it returns *no*. We note that for a given graph G with edge-coloring c, and vertices u and v, in time $2^k n^{\mathcal{O}(1)}$ time we can check if there is a rainbow path between u and v in G' by using the algorithm given by Corollary 5 in [19]. This completes the description of the algorithm.

We now proceed to show how we can obtain an algorithm with constant success probability.

▶ **Theorem 24.** *There is an algorithm that, given an instance (G, S) of* Subset Rainbow k-Coloring*, in time $2^{\mathcal{O}(|S|k \log k)}n^{\mathcal{O}(1)}$ either returns no or outputs a solution to* Subset Rainbow k-Coloring *in (G, S). Moreover, if the input is a yes instance of* Subset Rainbow k-Coloring*, then it returns a solution with positive constant probability.*

We start by defining some terminologies which will be useful in derandomization of our algorithm (see [12, 27]). An (n, p, ℓ)-*splitter* $\mathcal{F}$, is a family of functions from $[n]$ to ℓ such that for every $S \subseteq [n]$ of size at most p there is a function $f \in \mathcal{F}$ such that f splits S evenly. That is, for all $i, j \in [\ell]$, $|f^{-1}(i)|$ and $|f^{-1}(j)|$ differs by at most 1. Observe that when $\ell \geq p$ then for any $S \subseteq [n]$ of size at most p and a function $f \in \mathcal{F}$ that splits S, we have $|f^{-1}(i) \cap S| \leq 1$, for all $i \in [\ell]$. An (n, ℓ, ℓ)-splitter is called as an (n, ℓ)-*perfect hash family*. Moreover, for any $\ell \geq 1$, we can construct an (n, ℓ)-perfect hash family of size $e^{\ell}\ell^{\mathcal{O}(\log \ell)} \log n$ in time $e^{\ell}\ell^{\mathcal{O}(\log \ell)}n \log n$ [27].

We next move to the description of derandomization of the algorithm presented in Theorem 24. For the sake of simplicity in explanation, we associate each $e \in E(G)$ with a unique integer, say i_e in $[m]$, and whenever we refer to e as an integer, we actually refer to the integer

i_e. We start by computing an $(m, k|S|)$-perfect hash family $\mathcal{F}$ of size $e^{k|S|}(k|S|)^{\mathcal{O}(\log k|S|)} \log m$ in time $e^{k|S|}(k|S|)^{\mathcal{O}(\log k|S|)} m \log m$ using the algorithm of Naor et al. [27]. We will create a family of function $\mathcal{F}'$ from $[m]$ to $[k]$ of size $e^{k|S|}(k|S|)^{\mathcal{O}(\log k|S|)} k^{k|S|} \log m$. Towards this, consider an $f \in \mathcal{F}$ and a partition $\mathcal{P} = \{P_1, P_2, \dots, P_{k'}\}$ of $[k|S|]$ into k' sets, where $k' \leq k$. We let $f_{\mathcal{P}}$ to be the function obtained from f as follows. For each $i \in [k']$ we have $f_{\mathcal{P}}^{-1}(i) = \cup_{x \in P_i} f^{-1}(x)$. For every such pair f and $\mathcal{P}$, we add the function $f_{\mathcal{P}}$ to the set $\mathcal{F}'$. We will call such an $\mathcal{F}'$ as $(m, k|S|, k)$-*unified perfect hash* family. Observe that $\mathcal{F}'$ has size at most $e^{k|S|}(k|S|)^{\mathcal{O}(\log k|S|)} k^{k|S|} \log m$. We now describe the derandomized algorithm SRC, which is a result of derandomization of Rand-SRC.

Algorithm SRC. Given an instance (G, S) of Subset Rainbow k-Coloring, the algorithm start by computing an $(m, k|S|, k)$-unified perfect hash family $\mathcal{F}'$. If there exists $c : E(G) \rightarrow [k]$, where $c \in \mathcal{F}'$ such that for each $(u, v) \in S$, there is rainbow path between u and v in G' with the edge-coloring c then we return c as a solution to Subset Rainbow k-Coloring in (G, S). Otherwise, we return that (G, S) is a *no* instance of Subset Rainbow k-Coloring. We note that for a given graph G with edge-coloring c, and vertices u and v, in time $2^k n^{\mathcal{O}(1)}$ time we can check if there is a rainbow path between u and v in G' by using the algorithm given by Corollary 5 in [19]. This completes the description of the algorithm.

▶ **Theorem 25.** *Given an instance (G, k) of* Subset Rainbow k-Coloring*, the algorithm* SRC *either correctly reports that (G, k) is a no instance of* Subset Rainbow k-Coloring *or returns a solution to* Subset Rainbow k-Coloring *in (G, S). Moreover,* SRC *runs in time $2^{\mathcal{O}(|S|)} n^{\mathcal{O}(1)}$, for every fixed k. Here, $n = |V(G)|$.*

▶ **Corollary 26.** Steiner Rainbow k-Coloring *admits an algorithm running in time $2^{\mathcal{O}(|S|^2)} n^{\mathcal{O}(1)}$.*

6 Conclusion

In this paper, we proved that for all $k \geq 3$, Rainbow k-Coloring does not admit an algorithm running in time $2^{o(|E(G)|)} n^{\mathcal{O}(1)}$, unless ETH fails. This (partially) resolves the conjecture of Kowalik et al. [20], which states that for every $k \geq 2$, Rainbow k-Coloring does not admit an algorithm running in time $2^{o(|E(G)|)} n^{\mathcal{O}(1)}$. It would be an interesting direction to study whether or not Rainbow k-Coloring admits an algorithm running in time $2^{o(|E(G)|)} n^{\mathcal{O}(1)}$, for $k = 2$. We also studied the problem Steiner Rainbow k-Coloring, and proved that for every $k \geq 3$ the problem does not admit an algorithm running in time $2^{o(|S|^2)} n^{\mathcal{O}(1)}$, unless ETH fails. We complemented this by designing an algorithm for Subset Rainbow k-Coloring running in time $2^{\mathcal{O}(|S|)} n^{\mathcal{O}(1)}$, which implies an algorithm running in time $2^{\mathcal{O}(|S|^2)} n^{\mathcal{O}(1)}$ for Steiner Rainbow k-Coloring. It would be interesting to study whether or not Steiner Rainbow k-Coloring admits an algorithm running in time $2^{o(|S|^2)} n^{\mathcal{O}(1)}$, for $k = 2$. Kowalik et al. [20] also conjectured that for every $k \geq 2$, Rainbow k-Coloring does not admit an algorithm running in time $2^{o(n^2)} n^{\mathcal{O}(1)}$, which is another interesting direction of research.

Acknowledgements. The author is thankful to Saket Saurabh for helpful discussions.

References

1 Akanksha Agrawal, Daniel Lokshtanov, Saket Saurabh, and Meirav Zehavi. Split contraction: The untold story. In *34th Symposium on Theoretical Aspects of Computer Science, (STACS)*, pages 5:1–5:14, 2017.

2 Noga Alon, Raphael Yuster, and Uri Zwick. Color-coding. *Journal of the ACM (JACM)*, 42(4):844–856, 1995.

3 Prabhanjan Ananth, Meghana Nasre, and Kanthi K. Sarpatwar. Rainbow Connectivity: Hardness and Tractability. In *Foundations of Software Technology and Theoretical Computer Science (FSTTCS)*, volume 13, pages 241–251, 2011.

4 Yair Caro, Arie Lev, Yehuda Roditty, Zsolt Tuza, and Raphael Yuster. On rainbow connection. *Electronic Journal of Combinatorics*, 15(1):R57, 2008.

5 Sourav Chakraborty, Eldar Fischer, Arie Matsliah, and Raphael Yuster. Hardness and algorithms for rainbow connection. *Journal of Combinatorial Optimization*, 21(3):330–347, 2011.

6 L. Sunil Chandran and Deepak Rajendraprasad. Rainbow colouring of split and threshold graphs. In *18th Annual International Conference: Computing and Combinatorics, (COCOON)*, pages 181–192, 2012.

7 L. Sunil Chandran and Deepak Rajendraprasad. Inapproximability of rainbow colouring. In *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS)*, pages 153–162, 2013.

8 Gary Chartrand, Garry L Johns, Kathleen A McKeon, and Ping Zhang. Rainbow connection in graphs. *Mathematica Bohemica*, 133(1):85–98, 2008.

9 Gary Chartrand and Ping Zhang. *Chromatic graph theory.* CRC press, 2008.

10 Marek Cygan, Fedor V. Fomin, Alexander Golovnev, Alexander S. Kulikov, Ivan Mihajlin, Jakub Pachocki, and Arkadiusz Socala. Tight bounds for graph homomorphism and subgraph isomorphism. In *Proceedings of the 27th Annual ACM-SIAM Symposium on Discrete Algorithms, (SODA)*, pages 1643–1649, 2016.

11 Marek Cygan, Fedor V. Fomin, Alexander Golovnev, Alexander S. Kulikov, Ivan Mihajlin, Jakub Pachocki, and Arkadiusz Socala. Tight bounds for graph homomorphism and subgraph isomorphism. In *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1643–1649, 2016.

12 Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms.* Springer, 2015.

13 Reinhard Diestel. *Graph Theory, 4th Edition*, volume 173 of *Graduate texts in mathematics.* Springer, 2012.

14 Keith Edwards. The harmonious chromatic number and the achromatic number. *Surveys in Combinatorics*, pages 13–48, 1997.

15 Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W. H. Freeman & Co., New York, NY, USA, 1979.

16 Russell Impagliazzo and Ramamohan Paturi. On the complexity of k-sat. *Journal of Computer and System Sciences*, 62(2):367–375, 2001.

17 Russell Impagliazzo, Ramamohan Paturi, and Francis Zane. Which problems have strongly exponential complexity? *Journal of Computer and System Sciences*, 63(4):512–530, 2001.

18 Christian Komusiewicz. Tight running time lower bounds for vertex deletion problems. *arXiv preprint arXiv:1511.05449*, 2015.

19 Lukasz Kowalik and Juho Lauri. On finding rainbow and colorful paths. *Theoretical Computer Science*, 628(C):110–114, 2016.

20 Lukasz Kowalik, Juho Lauri, and Arkadiusz Socala. On the fine-grained complexity of rainbow coloring. In *24th Annual European Symposium on Algorithms, (ESA)*, pages 58:1–58:16, 2016.

21 V.B. Le and Z. Tuza. *Finding Optimal Rainbow Connection is Hard*. Preprints aus dem Institut für Informatik / CS. Inst. für Informatik, 2009. URL: `https://books.google.no/books?id=0ErVPgAACAAJ`.

22 Sin-Min Lee and John Mitchem. An upper bound for the harmonious chromatic number. *Journal of Graph Theory*, 11(4):565–567, 1987.

23 Xueliang Li, Yongtang Shi, and Yuefang Sun. Rainbow connections of graphs: A survey. *Graphs and Combinatorics*, 29(1):1–38, 2013.

24 Xueliang Li and Yuefang Sun. *Rainbow connections of graphs*. Springer Science & Business Media, 2012.

25 Daniel Lokshtanov, Dániel Marx, and Saket Saurabh. Lower bounds based on the exponential time hypothesis. *Bulletin of the EATCS*, pages 41–71, 2011.

26 Colin McDiarmid and Luo Xinhua. Upper bounds for harmonious coloring. *Journal of Graph Theory*, 15(6):629–636, 1991.

27 M. Naor, L. J. Schulman, and A. Srinivasan. Splitters and near-optimal derandomization. In *Proceedings of the 36th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 182–191, 1995.

28 Kei Uchizawa, Takanori Aoki, Takehiro Ito, Akira Suzuki, and Xiao Zhou. On the rainbow connectivity of graphs: Complexity and fpt algorithms. *Algorithmica*, 67(2):161–179, 2013.