Algorithms and Bounds for Very Strong Rainbow Coloring

L. Sunil Chandran[1], Anita Das[2], Davis Issac[3(✉)] [ID], and Erik Jan van Leeuwen[4]

[1] Department of Computer Science and Automation,
Indian Institute of Science, Bangalore, India
`sunil@csa.iisc.ernet.in`
[2] Infosys Ltd., Bangalore, India
`anita_das01@infosys.com`
[3] MPI für Informatik, Saarland Informatics Campus,
Saarbrücken, Germany
`dissac@mpi-inf.mpg.de`
[4] Department of Information and Computing Sciences,
Utrecht University, Utrecht, The Netherlands
`e.j.vanleeuwen@uu.nl`

Abstract. A well-studied coloring problem is to assign colors to the edges of a graph G so that, for every pair of vertices, all edges of at least one shortest path between them receive different colors. The minimum number of colors necessary in such a coloring is the strong rainbow connection number ($\mathbf{src}(G)$) of the graph. When proving upper bounds on $\mathbf{src}(G)$, it is natural to prove that a coloring exists where, for *every* shortest path between every pair of vertices in the graph, all edges of the path receive different colors. Therefore, we introduce and formally define this more restricted edge coloring number, which we call *very strong rainbow connection number* ($\mathbf{vsrc}(G)$).

In this paper, we give upper bounds on $\mathbf{vsrc}(G)$ for several graph classes, some of which are tight. These immediately imply new upper bounds on $\mathbf{src}(G)$ for these classes, showing that the study of $\mathbf{vsrc}(G)$ enables meaningful progress on bounding $\mathbf{src}(G)$. Then we study the complexity of the problem to compute $\mathbf{vsrc}(G)$, particularly for graphs of bounded treewidth, and show this is an interesting problem in its own right. We prove that $\mathbf{vsrc}(G)$ can be computed in polynomial time on cactus graphs; in contrast, this question is still open for $\mathbf{src}(G)$. We also observe that deciding whether $\mathbf{vsrc}(G) = k$ is fixed-parameter tractable in k and the treewidth of G. Finally, on general graphs, we prove that there is no polynomial-time algorithm to decide whether $\mathbf{vsrc}(G) \leq 3$ nor to approximate $\mathbf{vsrc}(G)$ within a factor $n^{1-\varepsilon}$, unless P = NP.

1 Introduction

The chromatic number is one of the most widely studied properties in graph theory. It has inspired a wealth of combinatorial and algorithmic results, as well as a host of variants. A variant that has recently attracted much interest is

the *rainbow connection number* of a graph, which is an edge coloring property introduced by Chartrand et al. [8] in 2008. Formally, the rainbow connection number $\mathbf{rc}(G)$ of a graph G is the smallest number of colors needed such that there exists a coloring of $E(G)$ with these colors such that, for every pair of vertices, there exists at least one path P between them, such that all edges of P receive different colors. We also say that this path P is *rainbow colored*. The rainbow connection number has attracted much attention, and the exact number is known for a variety of simple graph classes [5,6,8] and the complexity of computing this number was broadly investigated [1,2,4,6,7]. See also the surveys by Li et al. [17–19]. Most recently, in ESA 2016, it was shown that for any $k \geq 2$, deciding whether $\mathbf{rc}(G) \leq k$ (k-Rc) cannot be solved in $2^{o(n^{3/2})}$ or $2^{o(m/\log m)}$ time, where $n = |V(G)|$ and $m = |E(G)|$, unless ETH fails [15].

   To prove an upper bound on $\mathbf{rc}(G)$, the choice of the path P that is rainbow colored is crucial. The analysis would seem simpler when we are able to choose P as a shortest path between its two endpoints. This leads to the definition of the *strong rainbow connection number* of a graph. Formally, the strong rainbow connection number $\mathbf{src}(G)$ of a graph G is the smallest number of colors needed such that there exists a coloring of $E(G)$ with these colors such that, for every pair of vertices, there exists at least one *shortest* path P between them, such that all edges of P receive different colors. Clearly, $\mathbf{src}(G) \geq \mathbf{rc}(G)$, and both parameters are at least the diameter of G. Moreover, $\mathbf{rc}(G) = 2$ if and only if $\mathbf{src}(G) = 2$ [4]. Nontrivial upper bounds on $\mathbf{src}(G)$ are known for several simple graph classes such as cycles, wheels, and complete bipartite graphs [8] and block graphs [16]. It is also known that deciding whether $\mathbf{src}(G) \leq k$ (k-Src) is NP-hard even for $k = 2$ [4]. The problem of deciding whether $\mathbf{src}(G) \leq k$ remains NP-complete even for bipartite graphs and split graphs [1,14]. In fact, $\mathbf{src}(G)$ cannot be approximated in polynomial time within a factor $n^{1/2-\varepsilon}$ for any $\varepsilon > 0$, unless P = NP, even for split and bipartite graphs [1,14].[1]

   The lack of combinatorial bounds on $\mathbf{src}(G)$ for specific graph classes G (the recent survey by Li and Sun [19] cites only three papers) is somewhat surprising compared to the vast literature for $\mathbf{rc}(G)$ (see the surveys [17–19]). Li and Sun [19] explain this by the fact that $\mathbf{src}(G)$ is not a monotone graph property, and thus investigating $\mathbf{src}(G)$ is much harder than investigating $\mathbf{rc}(G)$. Hence, it is a major open question to prove upper bounds on $\mathbf{src}(G)$.

   In this paper, we make significant progress on this question. We observe that to prove upper bounds on $\mathbf{src}(G)$, it suffices to prove the existence of a coloring where all edges of not just one, but of *all* shortest paths between two vertices receive different colors. Therefore, we define the *very strong rainbow connection number* $\mathbf{vsrc}(G)$ of a graph G, which is the smallest number of colors for which there exists a coloring of $E(G)$ such that, for every pair of vertices and *every shortest* path P between them, all edges of P receive different colors. We call a coloring that achieves this property a *very strong rainbow coloring* of the graph. We also call the problem of deciding whether $\mathbf{vsrc}(G) \leq k$ the k-Vsrc problem.

---

[1] [1,14] mention NP $\neq$ ZPP as the complexity assumption but one can use P $\neq$ NP because of [23].

Our Results. We prove the first combinatorial upper bounds on $\mathbf{vsrc}(G)$ for several graph classes. These immediately imply upper bounds on $\mathbf{src}(G)$ for the same graph classes. In particular, we show upper bounds that are linear in $|V(G)|$ (improving from the trivial bound of $|E(G)|$) if G is a chordal graph, a circular arc graph, or a disk graph. We also make progress on the following conjecture:

Conjecture 1.1 ([16]). For any connected graph G, $\mathbf{src}(G) \leq |V(G)| - \chi(G) + 1$ where $\chi(G)$ denotes the chromatic number of G.

We show that the conjecture holds for the class of chordal graphs in Corollary 2.4.

Conversely, we prove that a bound on $\mathbf{vsrc}(G)$ implies that G should be highly structured: the neighborhood of every vertex can be partitioned into $\mathbf{vsrc}(G)$ cliques. For further details, we refer to Sect. 2.

In the second part of the paper, we address the computational complexity of k-VSRC. To start our investigation, we prove hardness results on general graphs.

Theorem 1.2. *3-VSRC is NP-complete. Moreover, there is no polynomial-time algorithm that approximates $\mathbf{vsrc}(G)$ within a factor $|V(G)|^{1-\varepsilon}$ for any $\varepsilon > 0$, unless $P = NP$.*

This result implies that k-VSRC is not fixed-parameter tractable when parameterized by k, unless P = NP. In order to prove the theorem, we show a nontrivial connection to the clique partition number of a graph.

We remark that, in contrast to the NP-complete 2-RC and 2-SRC problems, 2-VSRC can be solved in polynomial time (see Sect. 5 for the proof). Together with Theorem 1.2, this gives a dichotomy result for the complexity of k-VSRC.

Proposition 1.3. *Let G be any graph. Then 2-VSRC can be decided in polynomial time.*

We then study the complexity of determining $\mathbf{vsrc}(G)$ for graphs of bounded treewidth. This is a major open question also for $\mathbf{src}(G)$ and $\mathbf{rc}(G)$ [16], which are only known to be solvable in polynomial time on graphs of treewidth 1. We mention that no results for graphs of higher treewidth are known, even for outerplanar or cactus graphs. However, for the slightly different problem of deciding whether an already given coloring forms a (strong) rainbow coloring of a given graph, a polynomial-time algorithm for cactus graphs and an NP-hardness result for outerplanar graphs are known [22]. With this in mind, we focus on cactus graphs and make the first progress towards understanding the complexity of rainbow coloring problems, in particular of computing $\mathbf{vsrc}(G)$, on graphs of treewidth 2 with the following result.

Theorem 1.4. *Let G be any cactus graph. Then $\mathbf{vsrc}(G)$ can be computed in polynomial time.*

Our algorithm relies on an extensive characterization result for the behavior of very strong rainbow colorings on cactus graphs. Since a cactus graph consists of

bridges, even cycles, and odd cycles, we analyze the behavior of any very strong rainbow coloring of the graph with respect to these structures. We show that color repetition can mostly occur only within an odd cycle or even cycle. Odd cycles can repeat some colors from outside but we characterize how they can be repeated. However, our arguments are not sufficient to derive a completely combinatorial bound. Instead, we must find a maximum matching in a well-chosen auxiliary graph to compute the very strong rainbow connection number.

We also observe that $\mathbf{vsrc}(G)$ can be computed efficiently for graphs having bounded treewidth, when $\mathbf{vsrc}(G)$ itself is small. In contrast to known results for the (strong) rainbow connection number [9], we present an algorithm that does not rely on Courcelle's theorem. (See Sect. 5 for details.)

Theorem 1.5. k-Vsrc *is fixed-parameter tractable when parameterized by* $k+t$, *where* $t-1$ *is the treewidth of the input graph.*

Preliminaries. We consider simple, undirected graphs and use standard notation for graphs. Given a universe $\mathcal{U} = \{x_1, x_2, \ldots, x_n\}$ and a family $\mathcal{F} = \{S_1, S_2, \ldots, S_t\}$ of subsets of $\mathcal{U}$, the *intersection graph* $G(\mathcal{F})$ of $\mathcal{F}$ has vertex set $\{v_1, \ldots, v_t\}$, and there is an edge between two vertices v_i, v_j if and only if $S_i \cap S_j \neq \emptyset$. We call $\mathcal{F}$ a *representation* of $G(\mathcal{F})$. An *interval graph* is an intersection graph of intervals on the real line. The interval graph is *proper* if it has a representation by intervals where no interval is properly contained in another. A *circular arc graph* is an intersection graph of arcs of a circle. A *chordal graph* is an intersection graph of subtrees of a tree. A *block* of a graph is a maximal 2-connected component. In a *cactus graph*, each block of the graph is a cycle or an edge; equivalently, every edge belongs to at most one cycle.

For a graph G, let $\hat{G}$ denote the graph obtained by adding a new vertex $\hat{u}$ to G such that $\hat{u}$ is adjacent to all vertices of G, i.e., $\hat{u}$ is a universal vertex in $\hat{G}$.

Finally, we use $\omega(G)$ to denote the maximum size of any clique in graph G. We use $d(u, v)$ to denote the length of a shortest path between vertices u and v.

2   Combinatorial Results

We show several upper and lower bounds on $\mathbf{vsrc}(G)$, both for general graphs and for graphs G that belong to a specific graph class. Crucial in our analysis are connections between very strong rainbow colorings and decompositions of the input graph into cliques. We use $\mathbf{cp}(G)$ to denote the *clique partition number* (or clique cover number) of G, the smallest number of subsets of $V(G)$ that each induce a clique in G and whose union is $V(G)$. $\hat{G}$ used in the following lemma (defined in the preliminaries) is important for our hardness reductions.

Lemma 2.1. *Let G be any graph. Then*

1. $\mathbf{src}(G) \leq \mathbf{vsrc}(G) \leq \mathbf{cp}(G)(\mathbf{cp}(G)+1)/2.$
2. $\mathbf{src}(\hat{G}) \leq \mathbf{vsrc}(\hat{G}) \leq \mathbf{cp}(G)(\mathbf{cp}(G)+1)/2.$

Proof. Let $\mathcal{C} = C_1, \ldots, C_r$ be the set of cliques in an optimal clique partition of G; that is, $r = \mathbf{cp}(G)$. For a vertex v, let $c(v)$ denote the clique in $\mathcal{C}$ that contains v. We define the set of colors as $\mathcal{P}_{\leq 2}(\mathcal{C}) \setminus \{\emptyset\}$, the set of subsets of $\mathcal{C}$ of size 1 or 2. We then color any edge $uv \in E(G)$ by $\{c(u), c(v)\}$. For sake of contradiction, suppose that this does not constitute a very strong rainbow coloring of G. Then there exist two vertices $s, t \in V(G)$, a shortest path P between s and t, and two edges $uv, wx \in E(P)$ that received the same color. If $c(u) = c(v)$, then $c(w) = c(x)$, meaning that P uses two edges of the same clique. Then P can be shortcut, contradicting that P is a shortest path between s and t. Hence, $c(u) \neq c(v)$ and thus $c(w) \neq c(x)$. Without loss of generality, $c(u) = c(w)$ and thus $c(v) = c(x)$. Then either the edge uw or the edge vx will shortcut P, a contradiction. Hence, $\mathbf{vsrc}(G) \leq \mathbf{cp}(G)(\mathbf{cp}(G)+1)/2$ by the set of colors used. To see the second part of the lemma, color edges $\hat{u}v$ incident on the universal vertex $\hat{u}$ in $\hat{G}$ by $c(v)$ in addition to the above coloring. Suppose this was not a very strong rainbow coloring of $\hat{G}$. Then there exists vertices u, v such that $u\hat{u}v$ is a shortest path and $u\hat{u}$ and $v\hat{u}$ are colored the same. But then u and v are in the same clique C_i in $\mathcal{C}$. But then uv can shortcut $u\hat{u}v$, a contradiction. □

The following lemma is more consequential for our upper bounds. We use $\mathbf{is}(G)$ to denote the smallest size of the universe in any intersection graph representation of G, and $\mathbf{ecc}(G)$ to denote the smallest number of cliques needed to cover all edges of G. It is known that $\mathbf{is}(G) = \mathbf{ecc}(G)$ [20].

Lemma 2.2. *Let G be any graph. Then* $\mathbf{vsrc}(G) \leq \mathbf{is}(G) = \mathbf{ecc}(G)$.

Proof. Let $\mathcal{U} = \{x_1, x_2, \ldots, x_n\}$ be a universe and let $\mathcal{F} = \{S_1, S_2, \ldots, S_m\}$ be a family of subsets of $\mathcal{U}$, such that G is the intersection graph of $\mathcal{F}$ and $|\mathcal{U}| = \mathbf{is}(G)$. Let v_i be the vertex of G corresponding to the set S_i. We consider $x_1, x_2, \ldots, x_n$ as colors, and color an edge between vertices v_i and v_j with any $x \in S_i \cap S_j$ (note that this intersection is nonempty by the presence of the edge). Suppose for sake of contradiction that this is not a very strong rainbow coloring of G. Then there exist two vertices $s, t \in V(G)$, a shortest path P between s and t, and two edges $v_i v_j$ and $v_a v_b$ in P that received the same color x. By the construction of the coloring, this implies that $x \in S_i \cap S_j \cap S_a \cap S_b$. Hence, v_i, v_j, v_a, v_b induce a clique in G. But then the path P can be shortcut, a contradiction. □

A similar lemma for $\mathbf{src}(G)$ was proved independently by Lauri [16, Proposition 5.3].

Corollary 2.3. *Let G be any graph. Then* $\mathbf{vsrc}(G) \leq \min\{\lfloor |V(G)|^2/4 \rfloor, |E(G)|\}$.

Proof. Directly from $\mathbf{ecc}(G) \leq \min\{\lfloor |V(G)|^2/4 \rfloor, |E(G)|\}$ for any graph [10]. □

Corollary 2.4. *Let G be any graph.*

1. *If G is chordal, then* $\mathbf{src}(G) \leq \mathbf{vsrc}(G) \leq |V(G)| - \omega(G) + 1$.
2. *If G is circular-arc, then* $\mathbf{src}(G) \leq \mathbf{vsrc}(G) \leq |V(G)|$.
3. $\mathbf{src}(L(G)) \leq \mathbf{vsrc}(L(G)) \leq |V(G)|$, *where $L(G)$ is the line graph of G.*

These bounds are (almost) tight in general.

Proof. In each of the three cases, we express the graph as an intersection graph over a suitable universe, and then by Lemma 2.2, we get that the size of the universe is an upper bound on **vsrc** of the graph.

Every chordal graph is the intersection graph of subtrees of a tree [12]. It is also known that the number of vertices of this tree only needs to be at most $|V(G)| - \omega(G) + 1$. (For completeness, we provide a proof of this in the full version.)

For a circular arc graph G, consider any set of arcs whose intersection graph is G. We now construct a different intersection representation. Take the set of second (considering a clockwise ordering of points) endpoints of all arcs as the universe $\mathcal{U}$. Take $S_i \subseteq \mathcal{U}$ as the set of clockwise endpoints contained in the i-th arc. It is easy to see that G is the intersection graph of $\mathcal{F} = \{S_1, S_2, \ldots, S_n\}$.

Finally, consider $L(G)$. We construct an intersection representation with universe $V(G)$. For each $uv \in E(G)$, let $S_{uv} = \{u, v\}$. Then $L(G)$ is the intersection graph of $\mathcal{F} = \{S_e : e \in E(G)\}$.

The (almost) tightness follows from $\mathbf{vsrc}(G) = |V(G)| - 1$ and $\mathbf{vsrc}(L(G)) = |V(G)| - 2$ for any path G. Paths are both chordal and circular-arc. □

In the remainder, we consider a natural generalization of line graphs. A graph is k-*perfectly groupable* if the neighborhood of each vertex can be partitioned into k or fewer cliques. It is well known that line graphs are 2-perfectly groupable. A graph is k-*perfectly orientable* if there exists an orientation of its edges such that the outgoing neighbors of each vertex can be partitioned into k or fewer cliques. Clearly, any k-perfectly groupable graph is also k-perfectly orientable. Many geometric intersection graphs, such as disk graphs, are known to be k-perfectly orientable for small k [13].

Corollary 2.5. *Let G be any k-perfectly orientable graph. Then, $\mathbf{src}(G) \leq \mathbf{vsrc}(G) \leq k|V(G)|$.*

Proof. Consider any orientation of the edges of G such that the outgoing neighbors of each vertex can be partitioned into k or fewer cliques. For a given vertex v, let $C(v)$ denote the set of cliques induced by its outgoing neighbors, where v is added to each of those cliques. Observe that $\bigcup_{v \in V(G)} C(v)$ is an edge clique cover of G, because every edge is outgoing from some vertex v and will thus be covered by a clique in $C(v)$. Hence, $\mathbf{vsrc}(G) \leq \mathbf{ecc}(G) \leq k|V(G)|$. □

Since any k-perfectly groupable graph is also k-perfectly orientable, the above bound also applies to k-perfectly groupable graphs. In this context, we prove an interesting converse of the above bound.

Lemma 2.6. *Let G be any graph. If $\mathbf{vsrc}(G) \leq k$, then G is k-perfectly groupable.*

Proof. Consider an optimal very strong rainbow coloring μ of G. Consider an arbitrary vertex v of G and let c be any color used in μ. Define the set $Q(c) = \{u \in N(v) : \mu(vu) = c\}$. Suppose there exist two non-adjacent vertices u, w in $Q(c)$. Then uvw is a shortest path between u and w, and thus uv and vw cannot have the same color, a contradiction. Hence, for each color c used in μ, $Q(c)$ is a clique. Since the number of colors is at most k, the edges incident on v can be covered with at most k cliques. Hence, G is k-perfectly groupable. $\qquad\square$

3   Hardness Results

The hardness results lean heavily on the combinatorial bounds of the previous section. In this section, we use $\hat{G}$ (see the preliminaries for the definition) extensively. We need the following bound, which strengthens Lemma 2.1.

Lemma 3.1. *Let G be any graph. If $\mathbf{cp}(G) \leq 3$, then $\mathbf{vsrc}(\hat{G}) \leq 3$.*

Proof. Let C_1, C_2, and C_3 be three cliques into which $V(G)$ is partitioned. We will color $\hat{G}$ with three colors, say c_1, c_2, and c_3, as follows. For each edge with both endpoints in C_i for $1 \leq i \leq 3$, color it with c_i. For each edge vw with $v \in C_i$, $w \in C_j$ such that $1 \leq i < j \leq 3$, color it with c_k, where $k \in \{1, 2, 3\} \setminus \{i, j\}$. Finally, for each edge $\hat{u}v$ with $v \in C_i$ for $1 \leq i \leq 3$, color it with c_i.

Suppose this is not a very strong rainbow coloring of $\hat{G}$. Since the diameter of $\hat{G}$ is at most 2, there exists a shortest path xyz with xy and yz having the same color. However, if xy and yz have the same color, at least two of x, y and z are in the same C_i for $1 \leq i \leq 3$ and the third one is either $\hat{u}$ or in C_i itself. Then, we can shortcut xyz by xz, a contradiction. Hence $\mathbf{vsrc}(\hat{G}) \leq 3$. $\qquad\square$

Proof (of Theorem 1.2). We first prove that 3-VSRC is NP-complete. We reduce from the NP-hard 3-COLORING problem [11]. Let G be an instance of 3-COLORING. Let H be the complement of G. We claim that $\mathbf{vsrc}(\hat{H}) = 3$ if and only if G is 3-colorable. Indeed, if $\mathbf{vsrc}(\hat{H}) \leq 3$, then $\hat{H}$ is 3-perfectly groupable by Lemma 2.6. In particular, the neighborhood of $\hat{u}$ (the universal vertex in $\hat{H}$) can be partitioned into at most 3 cliques. These cliques induce disjoint independent sets in G that cover $V(G)$, and thus G is 3-colorable. For the other direction, note that if G is 3-colorable, then $\mathbf{cp}(H) \leq 3$, and by Lemma 3.1, $\mathbf{vsrc}(\hat{H}) \leq 3$.

To prove the hardness of approximation, we recall that there exists a polynomial-time algorithm that takes a SAT formula ψ as input and produces a graph G as output such that if ψ is not satisfiable, then $\mathbf{cp}(G) \geq |V(G)|^{1-\varepsilon}$, and if ψ is satisfiable, then $\mathbf{cp}(G) \leq |V(G)|^{\varepsilon}$ [23, Proof of Theorem 2]. Consider the graph $\hat{G}$ and let n denote the number of its vertices. Then

$$\psi \text{ not satisfiable} \Rightarrow \mathbf{cp}(G) \geq (n-1)^{1-\varepsilon} \Rightarrow \mathbf{vsrc}(\hat{G}) \geq (n-1)^{1-\varepsilon}$$

$$\psi \text{ satisfiable} \Rightarrow \mathbf{cp}(G) \leq (n-1)^{\varepsilon} \Rightarrow \mathbf{vsrc}(\hat{G}) \leq (n-1)^{2\varepsilon}$$

because Lemma 2.6 implies that $\mathbf{vsrc}(\hat{G}) \geq \mathbf{cp}(G)$, and by Lemma 2.1. The result follows by rescaling ε. $\qquad\square$

4   Algorithm for Cactus Graphs

Let G be the input cactus graph. We first prove several structural properties of cactus graphs, before presenting the actual algorithm.

4.1   Definitions and Structural Properties of Cactus Graphs

We make several structural observations related to cycles. For a vertex v and a cycle C containing v, we define $\mathsf{S}(v, C)$ as the vertices of G that are reachable from v without using any edge of C.

Observation 4.1. *For any cycle C in G, $\{\mathsf{S}(v, C) : v \in V(C)\}$ is a partition of $V(G)$.*

From Observation 4.1, we have that for any fixed $u \in V(G)$ and any fixed cycle C of G, there exists a unique vertex $v \in V(C)$ such that $u \in \mathsf{S}(v, C)$. We denote that unique vertex v by $\mathsf{g}(u, C)$.

Observation 4.2. *Let $u \in V(G)$ and let C be a cycle in G. Let $w \in V(C)$ and let $x_1 x_2 \ldots x_r$ be a path from u to w where $x_1 = u$ and $x_r = w$. Let i^* be the smallest i such that $x_i \in V(C)$. Then, $x_{i^*} = \mathsf{g}(u, C)$. In simpler words, any path from u to any vertex in C enters C through $\mathsf{g}(u, C)$.*

Observation 4.3. *For any cycle C in G and for any $uv \in E(G) \setminus E(C)$, $\mathsf{g}(u, C) = \mathsf{g}(v, C)$.*

We now consider even cycles. For an edge uv in an even cycle C, we define its *opposite edge*, denoted by $\mathsf{eopp}(uv)$, as the unique edge $xy \in E(C)$ such that $d(u, x) = d(v, y)$. Note that $\mathsf{eopp}(\mathsf{eopp}(e)) = e$. Call the pair of edges e and $\mathsf{eopp}(e)$ an *opposite pair*. Each even cycle C has exactly $\frac{|C|}{2}$ opposite pairs.

Lemma 4.4. *Let C be an even cycle. For any vertex $x \in V(G)$ and edge $uv \in E(C)$, either there is a shortest path between x and u that contains uv or there is a shortest path between x and v that contains uv.*

Proof. Let $w = \mathsf{g}(x, C)$. Then, w cannot be equidistant from u and v, because otherwise C is an odd cycle. Suppose that $d(w, u) < d(w, v)$. Then a shortest path from w to u appended with the edge uv gives a shortest path between w and v. Now, due to Observation 4.2, if we append a shortest path between x and w with a shortest path between w and v, we get a shortest path between x and v. Thus there is a shortest path between x and v that contains uv. If $d(w, u) > d(w, v)$, then we get the other conclusion of the lemma. $\square$

We then consider odd cycles in more detail. For any edge e in an odd cycle C, there is a unique vertex in C, which is equidistant from both endpoints of e. We call this vertex the *opposite vertex* of e and denote it as $\mathsf{vopp}(e)$. We call $\mathsf{OS}(e) = G[\mathsf{S}(\mathsf{vopp}(e), C)]$ the *opposite subgraph* of e. See Fig. 1.

Lemma 4.5. *Let C be an odd cycle and $uv \in E(C)$. For any vertex $x \in V(G) \setminus V(\mathsf{OS}(uv))$, either there is a shortest path between x and u that contains uv or there is a shortest path between x and v that contains uv.*

Proof. Let $w = \mathsf{g}(x, C)$. Since $x \notin V(\mathsf{OS}(uv))$, $w \neq \mathsf{vopp}(uv)$. Hence, w cannot be equidistant from u and v. So, the same arguments as in Lemma 4.4 complete the proof. □

Lemma 4.6. *Let e be any edge in an odd cycle of G for which $\mathsf{vopp}(e)$ has degree more than 2. Then $\mathsf{OS}(e)$ contains a bridge, or an even cycle, or an edge e' in an odd cycle for which $\mathsf{vopp}(e')$ has degree 2.*

Proof. Suppose this is not the case. We define a sequence $e_1, e_2, \ldots$ of edges by the following procedure. Let $e_1 = e$. Given e_i, we define e_{i+1} as follows. By assumption and the definition of cactus graphs, e_i is contained in an odd cycle, which we denote by C_i, and $\mathsf{vopp}(e_i)$ has degree more than 2. Choose e_{i+1} as any edge incident on $\mathsf{vopp}(e_i)$ that is not in C_i. However, observe that $\mathsf{OS}(e_{i+1}) \subset \mathsf{OS}(e_i)$ by the choice of e_{i+1}. Hence, this is an infinite sequence, which contradicts the finiteness of $E(G)$. □

4.2 Properties of Very Strong Rainbow Colorings of Cactus Graphs

We initially partition the edges of G into three sets: E_{bridge}, E_{even}, and E_{odd}. The set E_{bridge} consists of those edges that are not in any cycle. In other words, E_{bridge} is the set of bridges in G. By definition, each of the remaining edges is part of exactly one cycle. We define E_{even} as the set of all edges that belong to an even cycle, and E_{odd} as the set of all edges that belong to an odd cycle. Note that E_{bridge}, E_{even}, and E_{odd} indeed induce a partition of $E(G)$. We then partition E_{odd} into two sets: E_{opp} and E_{rem}. An edge $e \in E_{\mathsf{odd}}$ is in E_{opp} if $\mathsf{vopp}(e)$ is not a degree-2 vertex and in E_{rem} otherwise. See Fig. 1. We analyze each of these sets in turn, and argue how an optimal VSRC might color them.
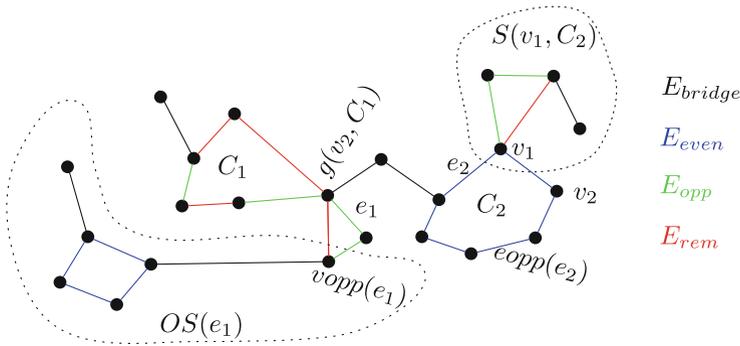
Fig. 1. An example of a cactus graph and related definitions.

Two edges e_1 and e_2 are called *conflicting* if there is a shortest path in the graph which contains both e_1 and e_2. Two conflicting edges must have different colors in any VSRC. We now exhibit several classes of conflicting pairs of edges.

Lemma 4.7. [2] *Any VSRC of G colors the edges of E_{bridge} with distinct colors.*

Proof. Consider $uv, xy \in E_{\text{bridge}}$. We prove that uv and xy are conflicting, i.e. there is a shortest path in G which contains both uv and xy. Since uv is a bridge, we can assume without loss of generality that any path between u and y uses the edge uv. Similarly, since xy is a bridge, we can assume without loss of generality that any path between y and u uses the edge xy. Hence, the shortest path from u to y uses both uv and xy. Hence, uv and xy are conflicting.   □

Lemma 4.8. *Let $e_1 \in E_{\text{bridge}}$ and $e_2 \in E_{\text{even}}$. Then any VSRC of G colors e_1 and e_2 with different colors.*

Proof. Let C be the cycle containing e_2. Let $e_1 = xy$ and $e_2 = uv$. Since xy is a bridge, we can assume w.l.o.g. that any path from x to any vertex in C contains xy. Due to Lemma 4.4, we can assume w.l.o.g. that there is a shortest path from x to v that contains uv. Thus we have a shortest path which contains both uv and xy, which means that uv and xy are conflicting.   □

Observation 4.9. *Let e_1 and e_2 be edges in an even cycle C of G such that $e_1 \neq \text{eopp}(e_2)$. Then any VSRC of G colors e_1 and e_2 with different colors.*

Lemma 4.10. *Let e_1 and e_2 be edges in two different even cycles C_1 and C_2 of G. Then any VSRC of G colors uv and xy with different colors.*

Proof. Let $e_1 = uv$ and $e_2 = xy$. Let $z = \text{g}(u, C_2)$ and $w = \text{g}(x, C_1)$. By Observation 4.3, $\text{g}(v, C_2) = z$ and $\text{g}(y, C_1) = w$. Due to Lemma 4.4, we can assume w.l.o.g. that there is a shortest path P_1 between z and x containing xy and that there is a shortest path P_2 between w and u containing uv. Let P_3 be a shortest path between w and z. Then $P_1 \cup P_3 \cup P_2$ gives a shortest path between u and x that contains both uv and xy. Hence, e_1 and e_2 are conflicting.   □

Lemma 4.11. *Let $e_1 \in E_{\text{bridge}} \cup E_{\text{even}}$ and $e_2 \in E_{\text{rem}}$. Then any VSRC of G colors e_1 and e_2 with different colors.*

Proof. Let $e_1 = xy$ and $e_2 = uv$, let C be the odd cycle containing e_2, and let $w = \text{g}(x, C)$. By Observation 4.3, $w = \text{g}(y, C)$. In other words, $x, y \in \text{S}(w, C)$. Note that w is not a degree-2 vertex, because there are at least two vertices in $\text{S}(w, C)$. Hence, $w \neq \text{vopp}(uv)$ by the definition of E_{rem}. Hence, by Lemma 4.5, w.l.o.g. there is a shortest path P_1 from w to u that contains uv.

We now consider two cases, depending on whether $e_1 \in E_{\text{bridge}}$ or $e_1 \in E_{\text{even}}$. First, suppose that $e_1 \in E_{\text{bridge}}$. Since xy is a bridge, we can assume w.l.o.g. that any shortest path from x to w contains xy. Let P_2 be such a shortest path. By Observation 4.2, if we append a shortest path from x to w with a shortest path

---

[2] This lemma holds for any graph, not necessarily cactus.

from w to u, we get a shortest path from x to u. Thus, $P_1 \cup P_2$ is a shortest path from x to u containing xy and uv. Hence, e_1 and e_2 are conflicting.

Suppose that $e_1 \in E_{\mathsf{even}}$. Let C' be the even cycle containing e_1. Let $z = \mathsf{g}(v, C')$. From Lemma 4.4, we can assume w.l.o.g. that there is a shortest path from z to x that contains xy. Let this shortest path be P_3. Let P_4 be a shortest path between w and z. By Observation 4.2, $P_3 \cup P_4 \cup P_1$ is a shortest path between x and u that contains xy and uv. Hence, e_1 and e_2 are conflicting.   □

Lemma 4.12. *Let C_1 and C_2 be two distinct odd cycles and let $e_1 \in E(C_1) \cap E_{\mathsf{rem}}$ and $e_2 \in E(C_2) \cap E_{\mathsf{rem}}$. Then any VSRC of G colors e_1 and e_2 with different colors.*

Proof. Let $e_1 = xy$ and $e_2 = uv$, and let $w = \mathsf{g}(x, C_2)$. By Observation 4.3, $w = \mathsf{g}(y, C_2)$. Let $z = \mathsf{g}(u, C_1)$. By Observation 4.3, $z = \mathsf{g}(v, C_1)$. That is, $x, y \in \mathsf{S}(w, C_2)$ and $u, v \in \mathsf{S}(z, C_1)$. Note that w and z are not degree-2 vertices, because there are at least two vertices in $\mathsf{S}(w, C_2)$ and $\mathsf{S}(z, C_1)$. Hence, $w \neq \mathsf{vopp}(uv)$ and $z \neq \mathsf{vopp}(xy)$ by the definition of E_{rem}. Hence, by Lemma 4.5, we can assume w.l.o.g. that there is a shortest path P_1 from u to w that contains uv and there is a shortest path P_2 from z to x that contains xy. Let P_3 be a shortest path from w to z. By Observation 4.2, $P_1 \cup P_2 \cup P_3$ is a shortest path from x to u containing xy and uv. Hence, e_1 and e_2 are conflicting.   □

Finally, we prove the existence of some non-conflicting pairs of edges.

Lemma 4.13. *For any $e_1 \in E_{\mathsf{opp}}$ and $e_2 \in \mathsf{OS}(e_1)$, e_1 and e_2 are not conflicting.*

Proof. Let $e_1 = uv$, $e_2 = xy$, and let C be the odd cycle containing e_1. For sake of contradiction, suppose that uv and xy are conflicting. Assume w.l.o.g. that there is a shortest path P from x to v which contains uv and xy. From Observation 4.2, P contains a subpath P' from $\mathsf{g}(x, C)$ to v. Clearly, P' contains uv. Also, $\mathsf{g}(x, C) = \mathsf{vopp}(uv)$, because $x \in \mathsf{OS}(uv)$. However, recall that $\mathsf{vopp}(uv)$ is equidistant from u and v. Hence, any shortest path from $\mathsf{vopp}(uv)$ to v does not contain uv, which contradicts the existence of P'.   □

4.3   Algorithm

Based on the results of the previous two subsections, we now describe the algorithm for cactus graphs. First, we color the edges of E_{bridge} with unique colors. By Lemma 4.7, no VSRC can use less colors to color E_{bridge}.

Next, we color the edges in E_{even} using colors that are distinct from those we used before. This will not harm the optimality of the constructed coloring, because of Lemma 4.8. Moreover, we use different colors for different even cycles, which does not harm optimality by Lemma 4.10. We then introduce a set of $\frac{|C|}{2}$ new colors for each even cycle C. For an opposite pair, we use the same color, and we color each opposite pair with a different color. Thus we use $\frac{|C|}{2}$ colors for each even cycle C. By Observation 4.9, no VSRC can use less colors to color C.

Next, we will color the edges in E_{rem} using colors that are distinct from those we used before. This will not harm the optimality of the constructed coloring, because of Lemma 4.11. For each odd cycle, we use a different set of colors. This will not harm the optimality of the constructed coloring, because of Lemma 4.12.

For each odd cycle C, we construct an auxiliary graph H_C for $E_{\text{rem}} \cap C$ as follows. Let $V(H_C) = E_{\text{rem}} \cap C$ and let $E(H_C) = \{e_1 e_2 \ : \ e_1, e_2 \in V(H_C); e_1 \text{ and } e_2 \text{ are not conflicting in } G\}$.

Lemma 4.14. $\Delta(H_C) \leq 2$.

Proof. It is easy to observe that in any odd cycle C, for any $e \in E(C)$, there are only two other edges in C that are not conflicting with e. $\square$

Let M_C be a maximum matching of H_C. We can compute M_C in linear time, since $\Delta(H_C) \leq 2$. For an $e_1 e_2 \in M_C$, color e_1 and e_2 with the same, new color. Then color each $e \in E_{\text{rem}} \cap C$ that is unmatched in M_C, each using a new color.

Lemma 4.15. *The procedure for coloring $E_{\text{rem}} \cap C$ gives a coloring of the edges in $E_{\text{rem}} \cap C$ such that no conflicting edges are colored the same. Moreover, no VSRC of G can use less colors to color $E_{\text{rem}} \cap C$ than used by the above procedure.*

Proof. Suppose two conflicting edges $e_1, e_2 \in E_{\text{rem}} \cap C$ were colored the same. Then the corresponding vertices e_1 and e_2 were matched to each other in M_C. Hence, e_1 and e_2 are adjacent in H_C, meaning that e_1 and e_2 did not conflict each other in G, which is a contradiction. Hence, we have proved that no conflicting edges were given the same color by the procedure.

Now, consider any VSRC μ of G which colored $E_{\text{rem}} \cap C$ with fewer colors than by our procedure. Observe that for any edge e in an odd cycle, there are only two other edges (say e_a and e_b) that are not conflicting with e. Moreover, e_a and e_b are conflicting with each other. This means that μ can use each color for at most two edges of $E_{\text{rem}} \cap C$. Suppose there are k_1 colors that are assigned to two edges in $E_{\text{rem}} \cap C$ by μ. Each pair of edges colored the same should be non-conflicting and hence have an edge between them in H_C. So, taking all pairs colored the same induces a matching of size k_1 of H_C. Then $k_1 \leq |M_C|$, because M_C is a maximum matching of H_C. But then the number of colors used by μ is equal to $k_1 + (|E_{\text{rem}} \cap C| - 2k_1) = |E_{\text{rem}} \cap C| - k_1$. The number of colors used by our procedure is $|M_C| + |E_{\text{rem}} \cap C| - 2|M_C| = |E_{\text{rem}} \cap C| - |M_C| \leq |E_{\text{rem}} \cap C| - k_1$. Hence, we use at most the number of colors used by μ. $\square$

Finally, we color the edges of E_{opp} without introducing new colors. Indeed, for every $e \in E_{\text{opp}}$, it follows from Lemma 4.6 that there exists an edge $e' \in E(\text{OS}(e)) \cap (E_{\text{bridge}} \cup E_{\text{even}} \cup E_{\text{rem}})$, which does not conflict with e by Lemma 4.13. Since e' is already colored, say by color c, then we can simply re-use that color c for e. Indeed, suppose for sake of contradiction that there is a shortest path P between two vertices x, y that contains e and that contains another edge e'' using the color c. By Lemma 4.13, $e'' \notin \text{OS}(e)$. This implies that $e'' \notin E_{\text{bridge}} \cup E_{\text{even}} \cup E_{\text{rem}}$ by the choice of c and the construction of the coloring. Hence, $e'' \in E_{\text{opp}}$. However, by a similar argument, e'' can only receive color c if $e' \in \text{OS}(e'')$.

But then $\mathsf{OS}(e) \subseteq \mathsf{OS}(e'')$ or $\mathsf{OS}(e'') \subseteq \mathsf{OS}(e)$, and thus e and e'' are not conflicting by Lemma 4.13, a contradiction to the existence of P.

Proof (of Theorem 1.4). It follows from the above discussion that the constructed coloring is a very strong rainbow coloring of G. Moreover, it uses $\mathbf{vsrc}(G)$ colors. Clearly, the coloring can be computed in polynomial time.   □

5   Other Algorithmic Results

In this section, we first show that 2-VSRC can be solved in polynomial time. Then we show that for k-VSRC is fixed parameter tractable when parameterized by $k + \mathbf{tw}(G)$, where $\mathbf{tw}(G)$ denotes the treewidth of G.

   For proving both the results, we use an auxiliary graph G' defined as follows: add a vertex v_e to G' for each edge e in G; add an edge between vertices v_{e_1} and v_{e_2} in G' if and only if edges e_1 and e_2 appear together in some shortest path of G. The latter condition can be easily checked in polynomial time. Observe that $\mathbf{vsrc}(G) \leq k$ if and only if G' admits a proper k-coloring. Since 2-COLORING is solvable in polynomial time, this implies that 2-VSRC is polynomial time solvable and hence we have proved Proposition 1.3.

   It is worth noting that the chromatic number of the auxiliary graph G' constructed in the above proof always corresponds to the very strong rainbow connection number of G. However, in the transformation from G to G', we lose a significant amount of structural information. For example, if G is a path or a star ($\mathbf{tw}(G) = 1$), then G' is a clique ($\mathbf{tw}(G') = |V(G'| - 1 = |V(G)| - 2$), where we use $\mathbf{tw}(G)$ to denote the treewidth of G. However, if $\mathbf{vsrc}(G) \leq k$, then we can prove that $|V(G')| \leq k^{(k+1)} \cdot (\mathbf{tw}(G) + 1)^{(k+1)}$ as shown below.

Lemma 5.1. *Let G be any connected graph and let $\mathbf{vsrc}(G) \leq k$ and $\mathbf{tw}(G) \leq t - 1$. Then $\Delta(G) \leq kt$ and $|V(G)| \leq (kt)^k$.*

Proof. By Lemma 2.6, the fact $\mathbf{vsrc}(G) \leq k$ implies that G is k-perfectly groupable. Hence, the neighborhood of each vertex can be partitioned into k or fewer cliques. Since $\mathbf{tw}(G) \leq t - 1$, each clique of G has size at most t [21]. Hence, $\Delta(G) \leq kt$. Now observe that $\mathbf{vsrc}(G) \leq k$ implies that the diameter of G is at most k. Combined, these two facts imply that $|V(G)| \leq (kt)^k$.   □

Proof (of Theorem 1.5). Again, let $\mathbf{vsrc}(G) \leq k$ and $\mathbf{tw}(G) \leq t - 1$. We now construct the auxiliary graph G' as above. Now, we only need to compute the chromatic number of G'. We aim to use the algorithm by Björklund et al. [3] which computes the chromatic number of a graph on n vertices in $2^n n^{\mathcal{O}(1)}$ time. To bound $|V(G')|$, we observe that by Lemma 5.1, $|V(G)| \leq (kt)^k$ and $\Delta(G) \leq kt$. Hence, $|V(G')| = |E(G)| \leq (kt)^{(k+1)}$. Therefore, the chromatic number of G', and thereby $\mathbf{vsrc}(G)$, can be determined in $\mathcal{O}(2^{(kt)^{(k+1)}} (kt)^{\mathcal{O}(k+1)})$ time.   □

References

1. Ananth, P., Nasre, M., Sarpatwar, K.K.: Rainbow connectivity: Hardness and tractability. In: Chakraborty, S., Kumar, A. (eds.) Proceedings of FSTTCS 2011. LIPIcs, vol. 13, pp. 241–251. Schloss Dagstuhl (2011)
2. Basavaraju, M., Chandran, L.S., Rajendraprasad, D., Ramaswamy, A.: Rainbow connection number and radius. Graphs Comb. **30**(2), 275–285 (2014)
3. Björklund, A., Husfeldt, T., Koivisto, M.: Set partitioning via inclusion-exclusion. SIAM J. Comput. **39**, 546–563 (2009)
4. Chakraborty, S., Fischer, E., Matsliah, A., Yuster, R.: Hardness and algorithms for rainbow connection. J. Comb. Optim. **21**(3), 330–347 (2011)
5. Chandran, L.S., Das, A., Rajendraprasad, D., Varma, N.M.: Rainbow connection number and connected dominating sets. J. Graph Theory **71**(2), 206–218 (2012)
6. Chandran, L.S., Rajendraprasad, D.: Rainbow colouring of split and threshold graphs. In: Gudmundsson, J., Mestre, J., Viglas, T. (eds.) COCOON 2012. LNCS, vol. 7434, pp. 181–192. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-32241-9_16
7. Chandran, L.S., Rajendraprasad, D.: Inapproximability of rainbow colouring. In: Seth, A., Vishnoi, N.K. (eds.) Proceedings of FSTTCS 2013. LIPIcs, vol. 24, pp. 153–162. Schloss Dagstuhl (2013)
8. Chartrand, G., Johns, G.L., McKeon, K.A., Zhang, P.: Rainbow connection in graphs. Math. Bohem. **133**(1), 85–98 (2008)
9. Eiben, E., Ganian, R., Lauri, J.: On the complexity of rainbow coloring problems. Discret. Appl. Math. (2016, in press). https://doi.org/10.1016/j.dam.2016.10.021
10. Erdős, P., Goodman, A.W., Pósa, L.: The representation of a graph by set intersections. Canad. J. Math **18**(106–112), 86 (1966)
11. Gary, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-Completeness. W. H. Freeman and Company, New York (1979)
12. Gavril, F.: The intersection graphs of subtrees in trees are exactly the chordal graphs. JCTB **16**(1), 47–56 (1974)
13. Kammer, F., Tholey, T.: Approximation algorithms for intersection graphs. Algorithmica **68**(2), 312–336 (2014)
14. Keranen, M., Lauri, J.: Computing minimum rainbow and strong rainbow colorings of block graphs. arXiv preprint arXiv:1405.6893 (2014)
15. Kowalik, Ł., Lauri, J., Socala, A.: On the fine-grained complexity of rainbow coloring. In: Sankowski, P., Zaroliagis, C.D. (eds.) Proceedings of ESA 2016. LIPIcs, vol. 57, pp. 58:1–58:16. Schloss Dagstuhl (2016)
16. Lauri, J.: Chasing the Rainbow Connection: Hardness, Algorithms, and Bounds, vol. 1428. Tampere University of Technology Publication, Tampere (2016)
17. Li, X., Shi, Y., Sun, Y.: Rainbow connections of graphs: a survey. Graphs Comb. **29**(1), 1–38 (2013)
18. Li, X., Sun, Y.: Rainbow Connections of Graphs. Springer Science & Business Media, Boston (2012). https://doi.org/10.1007/978-1-4614-3119-0
19. Li, X., Sun, Y.: An updated survey on rainbow connections of graphs - a dynamic survey. Theory Appl. Graphs **0**, 3 (2017)
20. Roberts, F.S.: Applications of edge coverings by cliques. Discret. Appl. Math. **10**(1), 93–109 (1985)
21. Robertson, N., Seymour, P.D.: Graph minors. II. Algorithmic aspects of tree-width. JCTB **7**, 309–322 (1986)

22. Uchizawa, K., Aoki, T., Ito, T., Suzuki, A., Zhou, X.: On the rainbow connectivity of graphs: complexity and FPT algorithms. Algorithmica **67**(2), 161–179 (2013)
23. Zuckerman, D.: Linear degree extractors and the inapproximability of max clique and chromatic number. In: Proceedings of STOC 2006, pp. 681–690. ACM (2006)