Marshall Plan Scholarship: Final Report

Research Exchange with DePaul University, Chicago, USA

March 20, 2017 - June 30, 2017

Eduard Eiben
(eiben@ac.tuwien.ac.at)

Home Institution
Technische Universität Wien
Algorithms and Complexity Group
Favoritenstrae 9-11
A-1040 Wien
Thesis Advisor: Stefan Szeider

Host Institution
DePaul University
College of Computing and Digital Media
243 South Wabash Avenue
Chicago, IL 60604
Host Advisor: Iyad Kanj

November 8, 2017

How to navigate a robot through obstacles?

Eduard Eiben* Iyad Kanj†

Abstract

We consider the following motion-planning problem: Given a set of obstacles in the plane, can we navigate a robot between two designated points without crossing more than k different obstacles? Equivalently, can we remove k obstacles so that there is an obstacle-free path between the two designated points? This problem is known to be NP-hard, even when each obstacle is either a square or a straight-line segment. It can be formulated and generalized into the following graph problem: Given a planar graph G whose vertices are colored by color sets, two designated vertices $s, t \in V(G)$, and $k \in \mathbb{N}$, is there an s-t path in G that uses at most k colors? If each obstacle is connected, the resulting graph from this formulation satisfies the property that each color induces a connected subgraph.

In this work, we study the complexity and design algorithms for this motion-planning problem. We first show that the problem is W[SAT]-hard parameterized by k, and is W[1]-complete on graphs of pathwidth 4 parameterized by both k and the length of the path. We then focus on the case where each color is connected. We first show that this problem is NP-hard, even when restricted to 2-outerplanar graphs of pathwidth 3. We then exploit the planarity of the graph and the connectivity of the colors to prove the following graph-theoretic structural result. For any vertex v in the graph, there exists a set of paths whose cardinality is upper bounded by some function of k, that "represents" the valid s-t paths containing subsets of colors from v. We then employ this structural result to design an FPT algorithm for the problem parameterized by both k and the treewidth of the graph.

Keywords. parameterized complexity and algorithms; planar graphs; treewidth; motion planning.

1 Introduction

1.1 Problem Definition and Motivation

Motion planning is an important subject with applications in Robotics, Computational Geometry, Graphics, and Gaming, among others [20]. The goal in motion planning problems is generally to move a robot from a starting position to a final position, while avoiding collision with a set of obstacles. This is usually referred to as the *piano-mover*'s problem. This work is concerned with a variant of the *piano-mover*'s problem, where the obstacles are in the Euclidean plane and the robot is represented as a point in the plane. Since determining if there is an obstacle-free path for the robot in this case is solvable in polynomial time, if no such path exists, it is natural to seek a path that intersects as few obstacles as possible. More formally, in the setting under consideration, we are given a set of obstacles in the plane, and $k \in \mathbb{N}$, and we need to determine if there is a path for the robot from the starting position to the final position that crosses at most k different

*Algorithms and Complexity Group, TU Wien, Austria. Email: eiben@ac.tuwien.ac.at
†School of Computing, DePaul University, Chicago, USA. Email: ikanj@cs.depaul.edu

obstacles; equivalently, we need to determine if we can remove at most k obstacles so that there is an obstacle-free path for the robot. By considering the auxiliary plane graph that is the dual of the plane subdivision determined by (the regions formed by) the obstacles, the problem can be formulated and generalized into the following graph problem. We are given a planar graph G, each of whose vertices is colored by a (possibly empty) color set, two designated vertices $s, t \in V(G)$, and $k \in \mathbb{N}$, and we need to decide if there is an s-t path in G that uses at most k colors. See Figure 1 for illustrations. We assume that the regions formed by the obstacles can be computed in polynomial time. The obstacles may or may not contain their interiors. If the intersection of two obstacles is not a 2-D region, we can thicken the borders of the obstacles without changing the sets of obstacles they intersect, so that their intersection becomes a 2-D region.

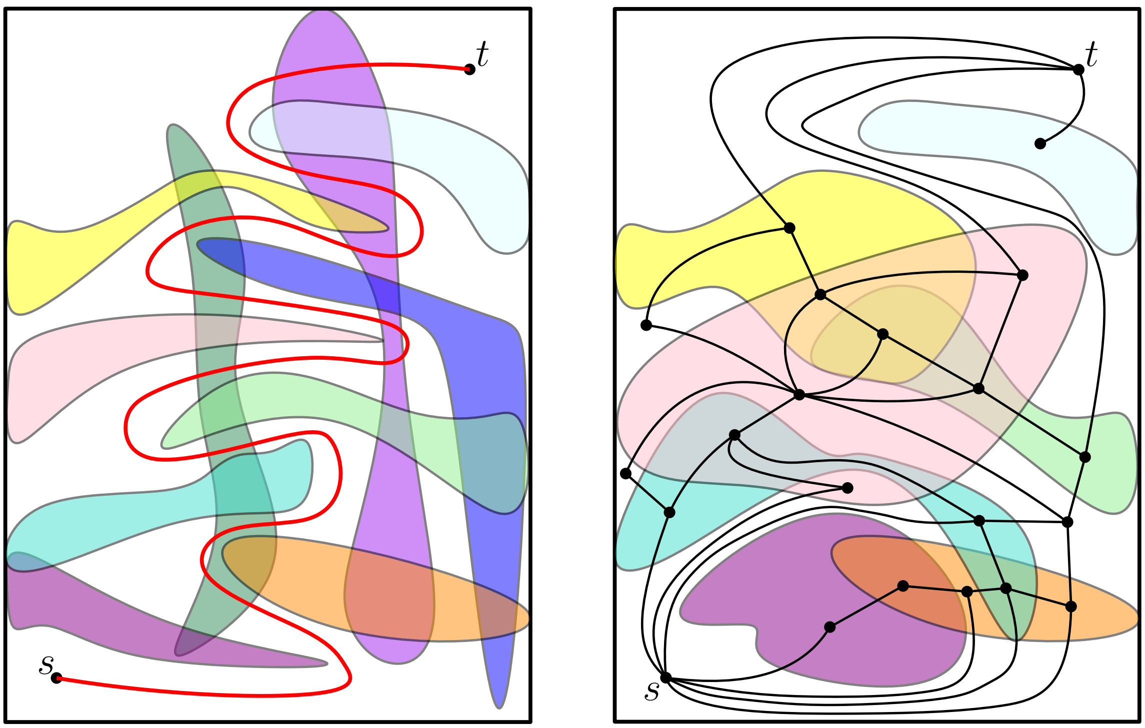

Figure 1: Illustration of instances of the problem under consideration drawn within a bounding box. The figure on the left shows an instance in which the optimal path crosses two obstacles, zigzagging between the other obstacles. The figure on the right shows an instance and its auxiliary plane graph.

Both the geometric and combinatorial problems were studied under the name Minimum Constraint Removal [7, 8, 11, 13]. Hauser [13] refers to the geometric problem as the Continuous Minimum Constraint Removal problem, and to the more general combinatorial one as the Planar Discrete Minimum Constraint Removal problem. Hauser [13] considered the Discrete Minimum Constraint Removal problem on general graphs, and showed it to be NP-hard via a reduction from the Set Cover problem. He also showed how this reduction from Set Cover can be slightly modified to yield instances of Planar Discrete Minimum Constraint Removal problem, thus showing its NP-hardness [13]. Although it was not mentioned in [13], the reduction from Set Cover to Planar Discrete Minimum Constraint Removal is an FPT-reduction, implying the W[2]-hardness of this problem as well. Hauser [13] also implemented and tested several algorithms for both the Continuous Minimum Constraint Removal and the Discrete Minimum Constraint Removal problems. Gorbenko and Popov [11] proposed a heuristic algorithm that is based on reducing the Discrete Minimum Constraint Removal problem to Sat, and then using SAT-solvers. Erickson and LaValle [8] showed that the Continuous Minimum Constraint Removal problem is NP-hard, even when each obstacle is either a square or a straight-line segment. A recent work of the authors of this paper, among others, refines the result of Erickson and LaValle [8] to show that the problem remains NP-hard even if all the obstacles are axes-parallel rectangles, and even if all the obstacles are line segments such that no three intersect at the same point [7]. In the same recent work, exact and heuristic algorithms for the problem have also been developed [7]. There is also a related problem that is solvable in polynomial time, which has received considerable attention [2, 14, 15], where the goal is to find a shortest path w.r.t. the Euclidean length between two given points in the plane that intersects at most k obstacles. We mention that both the Continuous Minimum Constraint Removal and the Discrete Minimum Constraint Removal problems generalize a set of problems, in which the objective is to determine a minimal set of reasons to why a task cannot be performed (*e.g.*, see [1, 21]). They also fall into the category of many computationally-hard problems on colored graphs, where the objective is to compute a graph structure satisfying certain (desired) properties that uses the minimum number of colors.

1.2 Our Contributions

We consider the Planar Discrete Minimum Constraint Removal problem, that we refer to in this paper as Obstacle Removal, and a restriction of it that we refer to as Connected Obstacle Removal. The Connected Obstacle Removal problem is the restriction of Obstacle Removal to instances satisfying that, for every color in the graph, the set of vertices on which this color appears induces a connected subgraph. Clearly, these problems model and generalize the two variants of the Continuous Minimum Constraint Removal problem, distinguished based on whether or not the obstacles are connected regions of the plane; we refer to these two geometric counterpart problems as Geomertic Obstacle Removal and Geomertic Connected Obstacle Removal. We note that we do not treat the more general Discrete Minimum Constraint Removal problem (*i.e.*, on general graphs), because, as we point out in Remark 3.11, this problem is computationally very hard, even when restricted to instances in which each color is connected.

We start in Section 3 by studying the complexity and the parameterized complexity of Obstacle Removal and Connected Obstacle Removal. Our first hardness result shows that both problems are NP-hard, even when restricted to graphs of small outerplanarity and pathwidth, and that it is unlikely that they can be solved in subexponential time:

- Obstacle Removal is NP-complete, even for outerplanar graphs of pathwidth at most 2 and in which every vertex contains at most one color (Theorem 3.1).
- Connected Obstacle Removal is NP-complete even for 2-outerplanar graphs of pathwidth at most 3 (Corollary 3.2).
- Unless ETH fails, Connected Obstacle Removal (and hence Obstacle Removal) is not solvable in subexponential time, even for 2-outerplanar graphs of pathwidth at most 3 and in which each color appears at most 4 times (Corollary 3.3).

The reduction used to prove the first result above produces combinatorial instances of Obstacle Removal that can be realized as geometric instances of Geomertic Obstacle Removal, in which the number of obstacles that overlap at any region is at most 2, and the auxiliary graph of the instance satisfies the properties in the statement of the result. Thus, this hardness result extends to the aforementioned restriction of GEOMERTIC OBSTACLE REMOVAL. This reduction, which is modified to yield the other two results above for CONNECTED OBSTACLE REMOVAL, produces combinatorial instances that can be realized as geometric instances of GEOMERTIC CONNECTED OBSTACLE REMOVAL whose auxiliary graph satisfies the statements in the result, and in which no more than four obstacles overlap at any region, again showing that the hardness results extend to these restrictions of GEOMERTIC CONNECTED OBSTACLE REMOVAL.

We then study the parameterized complexity of OBSTACLE REMOVAL and CONNECTED OBSTACLE REMOVAL. It is easy to see that all these problems, including DISCRETE MINIMUM CONSTRAINT REMOVAL, are in the parameterized class XP. Our first set of results shows that the color-connectivity property is crucial for any hope for an FPT-algorithm, as we show that even very restricted instances and combined parameterizations of OBSTACLE REMOVAL are W[1]-complete:

- OBSTACLE REMOVAL, restricted to instances of pathwidth at most 4, and in which each vertex contains at most one color and each color appears on at most 2 vertices, is W[1]-complete parameterized by k (Theorem 3.8).
- OBSTACLE REMOVAL, parameterized by both k and the length of the sought path ℓ, is W[1]-complete (Theorem 3.7).

Without any restrictions, the OBSTACLE REMOVAL problem sits high in the parameterized complexity hierarchy:

- OBSTACLE REMOVAL, parameterized by k, is W[SAT]-hard (Theorem 3.10) and is in W[P] (Theorem 3.9).

By producing a generic construction that can be used to realize any combinatorial instance of OBSTACLE REMOVAL as a geometric instance of GEOMERTIC OBSTACLE REMOVAL, the above results about OBSTACLE REMOVAL extend to the restriction of GEOMERTIC OBSTACLE REMOVAL to instances whose auxiliary graphs satisfy the properties in the statements of the results. The only thing that may be affected by this geometric realization is the number of obstacles that overlap at any region, which corresponds to the number of colors at the vertex in the graph that corresponds to the region; this number might increase by at most 4.

As we note in Remark 3.11, the color connectivity property without the planarity of the input graph is a hopeless case: we can tradeoff planarity for color-connectivity by adding a single vertex that serves as a color-connector, thus establishing the W[SAT]-hardness of the connected obstacle removal problem on apex graphs. Therefore, after establishing the aforementioned hardness results, we focus our attention on CONNECTED OBSTACLE REMOVAL. We show the following result:

- CONNECTED OBSTACLE REMOVAL, parameterized by both k and the treewidth ω of the input graph, is in FPT (Theorem 5.12).

The folklore dynamic programming approach based on tree decomposition, that is used for the HAMILTONIAN PATH/CYCLE problems, does not work for CONNECTED OBSTACLE REMOVAL for the following reasons. As opposed to the HAMILTONIAN PATH/CYCLE problems, where it is sufficient to keep track of how the path/cycle interacts with each bag in the tree decomposition, this is not sufficient in the case of CONNECTED OBSTACLE REMOVAL because we also need to keep track of which color sets are used on both sides of the bag. Although (by color-connectivity) any subset of colors appearing on both sides of the bag must appear on vertices in the bag as well, there can be too many such subsets (up to $|C|^k$, where C is the set of colors), and certainly we cannot afford to enumerate all of them if we seek an FPT algorithm. To overcome this issue, we prove in Section 4 structural results that exploit the planarity of the graph and the connectivity of the colors to show the following. For any vertex $w \in V(G)$, and for any pair of vertices $u, v \in V(G)$, the set of (valid) u-v paths in $G - w$ that use colors appearing on vertices in the face of $G - w$ containing w can be "represented" by a minimal set of paths $\mathcal{P}$ whose cardinality is a function of k. To derive such an upper bound on the cardinality of $\mathcal{P}$, we select a maximal set $\mathcal{M}$ of color-disjoint paths in $\mathcal{P}$, and show that the cardinality of $\mathcal{P}$ is upper bounded by that of $\mathcal{M}$ multiplied by some function of k. The problem then reduces to upper bounding $|\mathcal{M}|$. To do so, we use an inductive proof whose main ingredient is showing that the subgraph induced by the paths in $\mathcal{M}$ has a u-v vertex-separator of cardinality $O(k)$. We then upper bound $|\mathcal{M}|$ by upper bounding the number of different traces of the paths of $\mathcal{M}$ on this small separator, and inducting on both sides of the separator.

In Section 5, we extend the notion of a minimal set of paths w.r.t. a single vertex to a "representative set" of paths w.r.t. a specific bag, and a specific enumerated configuration for the bag, in a tree decomposition of the input graph. This enables us to use the upper bound on the cardinality of a minimal set of paths, derived in Section 4, to upper bound the size of a representative set of paths w.r.t. a bag and a configuration. This, in turn, yields an upper bound on the size of the table stored at a bag, in the dynamic programming algorithm, by a function of both k and the treewidth of the input graph, thus yielding the desired result.

In Section 6 we extend the FPT results for CONNECTED OBSTACLE REMOVAL w.r.t. the combined parameters k and ω—the treewidth of the input graph, to show that the Connected Obstacle Removal problem parameterized by both k and the length ℓ, is FPT:

- BOUNDED-LENGTH CONNECTED OBSTACLE REMOVAL parameterized by both k and the length of the path is in FPT(Theorem 6.13).

We then present some applications of the above result to show that CONNECTED OBSTACLE REMOVAL, restricted to instances in which the number of occurrences of each color is bounded, is FPT:

- BOUNDED-INTERSECTION CONNECTED OBSTACLE REMOVAL is FPT (Theorem 6.15).

Clearly, all our FPT results extend to the restriction of GEOMERTIC CONNECTED OBSTACLE REMOVAL to instances whose auxiliary graphs satisfy the properties in the statements of the results.

The above result has applications pertaining to instances of GEOMERTIC CONNECTED OBSTACLE REMOVAL whose auxiliary graph is an instance of BOUNDED-INTERSECTION CONNECTED OBSTACLE REMOVAL. In particular, an interesting case that was studied in the literature corresponds to the case in which the obstacles are convex polygons, each intersecting at most a constant number of other polygons. The complexity of this problem was left as an open question in [8], and remains unresolved. The above results shows that this problem is FPT.

We finally mention that it remains open whether CONNECTED OBSTACLE REMOVAL is FPT parameterized by k only.

2 Preliminaries

We assume familiarity with the basic notations and terminologies in graph theory and parameterized complexity. We refer the reader to the standard books [5, 6] for more information on these subjects.

Graphs. All graphs in this paper are simple (*i.e.*, loop-less and with no multiple edges). Let G be an undirected graph. For an edge $e = uv$ in G, *contracting* e means removing the two vertices u and v from G, replacing them with a new vertex w, and for every vertex y in the neighborhood of v or u in G, adding an edge wy in the new graph, not allowing multiple edges. Given a vertex-set $S \subseteq V(G)$, *contracting* S means contracting the edges between the vertices in S to obtain a single vertex at the end.

A graph is *planar* if it can be drawn in the plane without edge intersections (except at the endpoints). An *apex graph* is a graph in which the removal of a single vertex results in a planar graph. A *plane graph* has a fixed drawing. Each maximal connected region of the plane minus the drawing is an open set; these are the *faces*. One is unbounded, called the *outer face*. An *outerplane graph* is a plane graph for which every vertex is incident to the outer face; and *outerplanar graph* is a graph that has such a plane embedding. An *i-outerplane graph* (resp. *i-outerplaner graph*), for $i > 1$, is defined inductively as a graph such that the removal of its outer face results in an $(i-1)$-*outerplane graph* (resp. $(i-1)$-*outerplaner graph*) graph.

Let S be a set of points in the plane, and let C_1, C_2 be two non self-intersecting curves that meet S in precisely their common endpoints a and b. We say that C_1 and C_2 are *isotopic* w.r.t. S (also known as *homotopic rel. boundary*) if there is a continuous deformation from C_1 to C_2 through curves between a and b such that no intermediate curve in this deformation meets a vertex of S in its interior.

Let $W_1 = (u_1, \ldots, u_p)$ and $W_2 = (v_1, \ldots, v_q)$, $p, q \in \mathbb{N}$, be two walks such that $u_p = v_1$. Define the *gluing* operation $\circ$ that when applied to W_1 and W_2 produces that walk $W_1 \circ W_2 = (u_1, \ldots, u_p, v_2, \ldots, v_q)$.

For a graph G and two vertices $u, v \in V(G)$, we denote by $d_G(u, v)$ the *distance* between u and v in G, which the length of a shortest path between u and v in G.

Treewidth, Pathwidth and Tree Decomposition.

Definition 2.1. Let $G = (V, E)$ be a graph. A *tree decomposition* of G is a pair $(\mathcal{V}, \mathcal{T})$ where $\mathcal{V}$ is a collection of subsets of V such that $\bigcup_{X_i \in \mathcal{V}} = V$, and $\mathcal{T}$ is a rooted tree whose node set is $\mathcal{V}$, such that:

1. for every edge $\{u, v\} \in E$, there is an $X_i \in \mathcal{V}$, such that $\{u, v\} \subseteq X_i$; and
2. for all $X_i, X_j, X_k \in \mathcal{V}$, if the node X_j lies on the path between the nodes X_i and X_k in the tree $\mathcal{T}$, then $X_i \cap X_k \subseteq X_j$.

   The *width* of the tree decomposition $(\mathcal{V}, \mathcal{T})$ is defined to be $\max\{|X_i| \mid X_i \in \mathcal{V}\} - 1$. The *treewidth* of the graph G is the minimum width over all tree decompositions of G.

A *path decomposition* of a graph G is a tree decomposition $(\mathcal{V}, \mathcal{T})$ of G, where $\mathcal{T}$ is a path. The *pathwidth* of a graph G is the minimum width over all path decompositions of G.

A tree decomposition $(\mathcal{V}, \mathcal{T})$ is *nice* if it satisfies the following conditions:

1. Each node in the tree $\mathcal{T}$ has at most two children.
2. If a node X_i has two children X_j and X_k in the tree $\mathcal{T}$, then $X_i = X_j = X_k$; in this case node X_i is called a *join node*.
3. If a node X_i has only one child X_j in the tree $\mathcal{T}$, then either $|X_i| = |X_j| + 1$ and $X_j \subset X_i$, and in this case X_i is called an *insert node*; or $|X_i| = |X_j| - 1$ and $X_i \subset X_j$, and in this case X_i is called a *forget node*.
4. If X_i is a leaf node or the root, then $X_i = \emptyset$.

Boolean Circuits and Parameterized Complexity. A *circuit* is a directed acyclic graph. The vertices of indegree 0 are called the (input) *variables*, and are labeled either by *positive literals* x_i or by *negative literals* $\overline{x}_i$. The vertices of indegree larger than 0 are called the *gates* and are labeled with Boolean operators AND or OR. A special gate of outdegree 0 is designated as the *output* gate. We do not allow NOT gates in the above circuit model, since by De Morgan's laws, a general circuit can be effectively converted into the above circuit model. A circuit is said to be *monotone* if all its input literals are positive. The *depth* of a circuit is the maximum distance from an input variable to the output gate of the circuit. A circuit represents a Boolean function in a natural way. The size of a circuit C, denoted $|C|$, is the size of the underlying graph (*i.e.*, number of vertices and edges). An *occurrence* of a literal in C is an edge from the literal to a gate in C. Therefore, the total number of occurrences of the literals in C is the number of outgoing edges from the literals in C to its gates.

We say that a truth assignment τ to the variables of a circuit C *satisfies* a gate g in C if τ makes the gate g have value 1, and that τ *satisfies the circuit* C if τ satisfies the output gate of C. A circuit C is *satisfiable* if there is a truth assignment to the input variables of C that satisfies C. The *weight* of an assignment τ is the number of variables assigned value 1 by τ.

A *parameterized problem* Q is a subset of $\Omega^* \times \mathbb{N}$, where Ω is a fixed alphabet. Each instance of the parameterized problem Q is a pair (x, k), where $k \in \mathbb{N}$ is called the *parameter*. We say that the parameterized problem Q is *fixed-parameter tractable* (FPT) [6], if there is a (parameterized) algorithm, also called an FPT-*algorithm*, that decides whether an input (x, k) is a member of Q in time $f(k) \cdot |x|^{O(1)}$, where f is a computable function. Let FPT denote the class of all fixed-parameter tractable parameterized problems.

A parameterized problem Q is FPT-*reducible* to a parameterized problem Q' if there is an algorithm, called an FPT-*reduction*, that transforms each instance (x, k) of Q into an instance (x', k') of Q' in time $f(k) \cdot |x|^{O(1)}$, such that $k' \leq g(k)$ and $(x, k) \in Q$ if and only if $(x', k') \in Q'$, where f and g are computable functions. By FPT-*time* we denote time of the form $f(k) \cdot |x|^{O(1)}$, where f is a computable function and $|x|$ is the input instance size.

Based on the notion of FPT-reducibility, a hierarchy of parameterized complexity, *the* W-*hierarchy* $\bigcup_{t \geq 0} \mathsf{W}[t]$, where $\mathsf{W}[t] \subseteq \mathsf{W}[t+1]$ for all $t \geq 0$, has been introduced, in which the 0-th level $\mathsf{W}[0]$ is the class FPT. The hardness and completeness have been defined for each level $\mathsf{W}[i]$ of the W-hierarchy for $i \geq 1$ [6]. It is commonly believed that $\mathsf{W}[1] \neq \mathsf{FPT}$ (see [6]). The $\mathsf{W}[1]$-hardness has served as the main working hypothesis of fixed-parameter intractability.

The class $\mathsf{W}[\mathrm{SAT}]$ contains all parameterized problems that are FPT-reducible to the weighted satisfiability of Boolean formulas. It contains the classes $\mathsf{W}[\mathsf{t}]$, for every $t \geq 0$. Boolean formulas can be represented (in polynomial time) by Boolean circuits that are in the *normalized* form (see [6]). In the normalized form every (nonvariable) gate has outdegree at most 1, and the gates are structured into alternating levels of ORs-of-ANDs-of-ORs.... Therefore, the underlying undirected graph of the circuit with the input variables removed is a tree; the input variables can be connected to any gate in the circuit, including the output gate. The class $\mathsf{W}[\mathsf{P}]$ contains all parameterized problems that are FPT-reducible to the weighted satisfiability of Boolean circuits of polynomial size, and contains the class $\mathsf{W}[\mathrm{SAT}]$.

The *Exponential Time Hypothesis* (ETH) states that the satisfiability of k-CNF Boolean formulas, where $k \geq 3$, is not decidable in subexponential-time $\mathcal{O}(2^{o(n)})$, where n is the number of variables in the formula. ETH has become a standard hypothesis in complexity theory for proving hardness results that is closely related to the computational intractability of a large class of well-known NP-hard problems, measured from a number of different angles, such as subexponential-time complexity, fixed-parameter tractability, and approximation.

The asymptotic notation $\mathcal{O}^*$ suppresses a polynomial factor in the input length.

Obstacle Removal and Connected Obstacle Removal. For a set S, we denote by 2^S the power set of S. Let $G = (V, E)$ be a graph, let $C \subset \mathbb{N}$ be a finite set referred to as a set of colors, and let $\chi : V \longrightarrow 2^C$. A vertex v in V is said to be *empty* if $\chi(v) = \emptyset$. We say that a color c *appears on*, or is *contained in*, a subset of vertices S if $c \in \bigcup_{v \in S} \chi(v)$. For two vertices $u, v \in V(G)$, a u-v path $P = (u = v_0, \ldots, v_r = v)$ in G is said to be *ℓ-valid* if $|\bigcup_{i=0}^{r} \chi(v_i)| \leq \ell$; that is, if the total number of colors appearing on the vertices of P is at most ℓ. A color $c \in C$ is *connected* in G, or simply *connected*—if it is clear from the context which graph is meant, if $\bigcup_{c \in C(v)} \{v\}$ induces a connected subgraph of G. The graph G is said to be *color-connected*, if for each $c \in C$, c is connected in G.

The Obstacle Removal problem is formally defined as follows:

Obstacle Removal
Given: A planar graph G; a set of colors C; $\chi : V \longrightarrow 2^C$; and two designated vertices $s, t \in V(G)$
Parameter: k
Question: Does there exist a k-valid s-t-path in G?

We denote by Connected Obstacle Removal the restriction of Obstacle Removal to instances in which the input graph G is color-connected.

For an instance (G, C, χ, s, t, k) of Obstacle Removal or Connected Obstacle Removal, if s and t are nonempty vertices, we can remove their colors and decrement k by $|\chi(s) \cup \chi(t)|$ because their colors appear on every s-t path. If afterwards k becomes negative, then there is no k-valid s-t path in G. Moreover, if s and t are adjacent, then the path (s, t) is a path with the minimum number of colors among all s-t paths in G. Therefore, we will assume the following:

Assumption 2.2. *For an instance (G, C, χ, s, t, k) of* Obstacle Removal *or* Connected Obstacle Removal, *we can assume that s and t are nonadjacent empty vertices.*

Definition 2.3. Let s, t be two designated vertices in G, and let x, y be two adjacent vertices in G such that $\chi(x) = \chi(y)$. We define the following operation to x and y, referred to as a *color contraction* operation, that results in a graph G', a color function χ', and two designated vertices s', t' in G', obtained as follows:

- G' is the graph obtained from G by contracting the edge xy, which results in a new vertex z;
- $s' = s$ (resp. $t' = t$) if $s \notin \{x, y\}$ (resp. $t \notin \{x, y\}$), and $s' = z$ (resp. $t' = z$) otherwise; and
- $\chi' : V(G') \longrightarrow 2^C$ is the function defined as $\chi'(w) = \chi(w)$ if $w \neq z$, and $\chi'(z) = \chi(x) = \chi(y)$.

G is *irreducible* if there does not exist two vertices in G to which the color contraction operation is applicable.

Lemma 2.4. *Let G be a color-connected plane graph, C a color set, $\chi : V \longrightarrow 2^C$, $s, t \in V(G)$, and $k \in \mathbb{N}$. Suppose that the color contraction operation is applied to two vertices in G to obtain G', χ', s', t', as described in Definition 2.3. Then G' is a color-connected plane graph, and there is a k-valid s-t path in G if and only if there is a k-valid s'-t' path in G'.*

Proof. Let x and y be the two adjacent vertices in G to which the color contraction operation is applied, and let z be the new vertex resulting from this contraction. It is clear that after the contraction operation the obtained graph G' is a plane color-connected graph.

Suppose that there is a k-valid s-t path in G, and let $P = (s = v_0, \ldots, v_r = t)$ be such a path. We can assume that P is an induced path. If no vertex in $\{x, y\}$ is on P, then $P' = P$ is a k-valid s'-t' in G'. If exactly one vertex in $\{x, y\}$, say x, is on P, then since the color set of every vertex other than x on P is the same before and after the contraction operation, and since $\chi'(z) = \chi(x)$, the path P' obtained from P by replacing x with z is a k-valid s'-t' in G'. (Note that if $x = s$ then $s' = z$, and replacing x with z on P is obsolete in this case.) Finally, if both x and y are on P, then since P is induced, x and y must appear consecutively on P. Without loss of generality, assume $x = v_i$ and $y = v_{i+1}$, for some $i \in \{0, \dots, r-1\}$. Since the color set of every vertex other than x and y on P is the same before and after the operation, and since $\chi'(z) = \chi(x) = \chi(y)$, the path $P' = (s' = v_0, \dots, v_{i-1}, z, v_{i+1}, \dots, t = v_r)$ is a k-valid s'-t' path in G'.

Conversely, suppose that there is a k-valid s'-t' path in G', and let $P' = (s' = v'_0, \dots, v'_p = t')$, where $p > 0$, be such a path. If z does not appear on P' then P' is a k-valid s-t path in G. Otherwise, $z = v'_i$ for some $i \in \{0, \dots, p\}$. If $i = 0$ and P' consists only of vertex z, then since $\chi(x) = \chi(y) = \chi'(z)$, either $s = t$, and in which case there is a trivial k-valid s-t path in G, or $s \neq t$, and in this case $P = (x, y)$ is a k-valid s-t path in G. Otherwise, when $i = 0$ we must have $s = x$ or $s = y$, $v'_i \in G$ for $i \in [p]$, and $t' = t$; without loss of generality, assume that $s = x$. Since z is adjacent to v'_1, either $x = s$ or y (or both) is adjacent to v'_1. Since $\chi(x) = \chi(y) = \chi'(z)$, if x is adjacent to v'_1 then $P = (s = x, v'_1, \dots, v'_p = t')$ is a k-valid s-t path in G, and if y is adjacent to v'_1 then $P = (s = x, y, v'_1, \dots, v'_p = t')$ is a k-valid s-t path in G. The case is similar if $i = p$. Suppose now that $i \neq 0$ and $i \neq p$. If x (resp. y) is adjacent to both v'_{i-1} and v'_{i-1}, then the path $P = (s, v'_1, \dots, v'_{i-1}, x, v'_{i+1}, \dots, v'_p = t)$ (resp. $P = (s, v'_1, \dots, v'_{i-1}, y, v'_{i+1}, \dots, v'_p = t)$ is a k-valid s-t-path in G; otherwise, one vertex in $\{x, y\}$, say x, must be adjacent to v'_{i-1}, and the other vertex y must be adjacent to v'_{i+1}. In this case the path $P = (s, v'_1, \dots, v'_{i-1}, x, y, v'_{i+1}, \dots, v'_p = t)$ is a k-valid s-t-path in G. □

3 Hardness Results

In this section, we study the complexity and the parameterized complexity of Obstacle Removal and Connected Obstacle Removal. We start by showing that both problems are NP-hard, even when restricted to graphs of small outerplanarity and pathwidth.

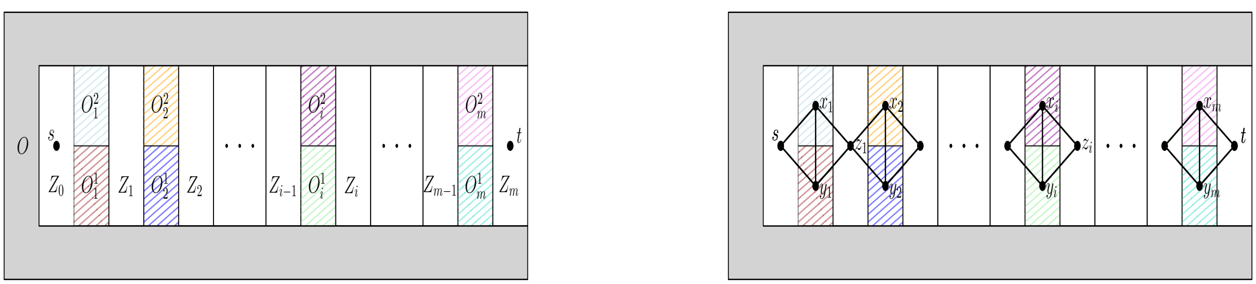

Figure 2: Illustration of the construction in the proof of Theorem 3.1. The left figure shows the geometric instance of Obstacle Removal, and the right figure the graph associated with it.

Theorem 3.1. Obstacle Removal, *restricted to outerplanar graphs of pathwidth at most 2 and in which every vertex contains at most one color, is* NP*-complete.*

Proof. It is clear that Obstacle Removal is in NP. To show its NP-hardness, we reduce from the NP-hard problem Vertex Cover [10]. Let (G, k) be an instance of Vertex Cover, where $V(G) = \{v_1, \dots, v_n\}$ and $E(G) = \{e_1, \dots, e_m\}$. In the rest of the proof, when we write $e = uw$ for an edge e in $E(G)$, we assume that $u = v_i$ and $w = v_j$ such that $i < j$ (*i.e.*, the vertex of smaller index always appears first). Although not necessary for the proof, we first describe a geometric instance I of Obstacle Removal whose associated graph is the desired instance of Obstacle

REMOVAL. The regions of I are $O \cup \{Z_0, \ldots, Z_m\} \cup \bigcup_{i=1}^{m} \{O_i^1, O_i^2\}$, depicted in Figure 2 (left figure). The obstacles of I are defined as follows. For each vertex $v_j \in V(G)$, the obstacle corresponding to v_j is the polygon whose boundary is the boundary of the region formed by the union of O, each O_i^1 such that $e_i = v_j v_q$, and each O_i^2 such $e_i = v_p v_j$. More formally, the obstacle corresponding to v_j is $\partial(O \cup \bigcup_{e_i = v_j v_q} O_i^1 \cup \bigcup_{e_i = v_p v_j} O_i^2)$. The graph associated with I, G_I, is defined as follow. Each (empty) region Z_i, $i = 0, \ldots, m$, corresponds to a vertex $z_i \in V(G_I)$, where Z_0 corresponds to s and Z_m to t. Each region O_i^1, $i \in [m]$, corresponds to a vertex y_i, and each region O_i^2, $i \in [m]$, corresponds to a vertex x_i. The set of edges $E(G_I)$ is $E(G_I) = \{z_{i-1}x_i, z_{i-1}y_i, x_i y_i, z_i x_i, z_i y_i \mid i \in [m]\}$. The color function $\chi : V(G_I) \longrightarrow 2^C$, where $C = [n]$, is defined as follows: $\chi(z_i) = \emptyset$, for $i = 0, \ldots, m$; $\chi(x_i) = \{j\}$ and $\chi(y_i) = \{p\}$, where $e_i = v_p v_j$, for $i \in [m]$. This completes the construction of G_I; see Figure 2 (right figure) for illustration. It is easy to see that G_I is outerplanar and has pathwidth at most 2.

Define the reduction from VERTEX COVER to OBSTACLE REMOVAL that takes an instance (G, k) to the instance (G_I, C, χ, s, t, k). Clearly, this reduction is polynomial-time computable. Suppose that Q, where $|Q| = r \leq k$, is a vertex cover of G. Consider the s-t path $P = (s, w_1, z_1, , \ldots, w_m, z_m)$ in G_I, where $w_i = y_i$ if edge $e_i = v_p v_q$ is covered by v_p, and $w_i = x_i$ otherwise, for $i \in [m]$. Clearly this is a k-valid s-t path in G_I since each edge e_i is covered by a vertex in Q, each w_i is colored by the index of one of the vertices in Q, and each vertex in G_I (and hence each w_i) contains at most one color. Conversely, suppose that P is a k-valid s-t path in G_I. By construction of G_I, P has to contain at least one vertex from $\{x_i, y_i\}$, for each $i \in [m]$. If P contains both x_i and y_i, for some $i \in [m]$, then clearly, from the construction of P, P must contain either (z_{i-1}, x_i, y_i, z_i) or (z_{i-1}, y_i, x_i, z_i), as a subpath, and we can shortcut this subpath by removing one of x_i, y_i, to obtain another k-valid s-t path in G_I. Therefore, without loss of generality, we may assume that P contains exactly one vertex w_i from $\{x_i, y_i\}$, for $i \in [m]$. Now define the set of vertices Q in G as the vertices in G whose indices are the colors appearing on (the w_i's in) P. More formally, define $Q = \{v_p \mid w_i = x_i \in P \wedge e_i = v_q v_p\} \cup \{v_p \mid w_i = y_i \in P \wedge e_i = v_p v_q\}$. Since P is a k-valid path in G_I, the total number of colors appearing on $\{w_1, \ldots, w_m\}$ is at most k. Notice that the color of each of x_i, y_i is the index of a vertex in G that covers edge e_i. It follows that the set Q of vertices in G, that are the indices of the colors on P, form a k-vertex cover of G. □

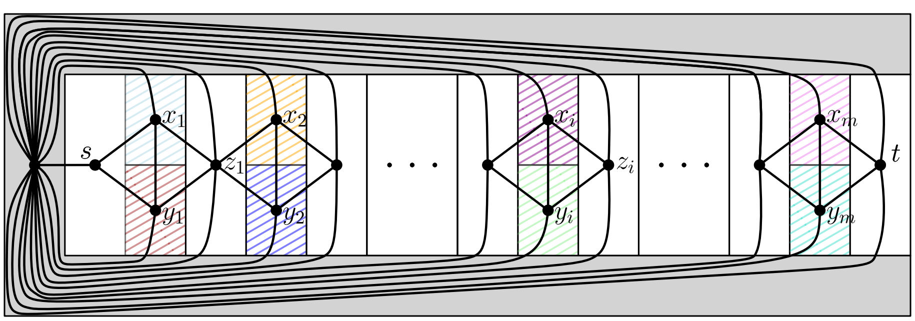

Figure 3: Illustration for the proof of Corollary 3.2.

Corollary 3.2. CONNECTED OBSTACLE REMOVAL*, restricted to 2-outerplanar graphs of pathwidth at most 3, is* NP*-complete.*

Proof. This follows directly from the NP-hardness reduction in the proof of Theorem 3.1 by observing the following. The graph G_I resulting from the reduction is outerplanar. We can add a new vertex to the outer face of G_I (see Figure 3) containing all colors that appear on G_I, and add edges between the new vertex and all vertices in G_I. The obtained graph is color-connected and has pathwidth at most 3. □

Assuming ETH, the following corollary rules out the existence of subexponential-time algorithms for CONNECTED OBSTACLE REMOVAL (and hence for OBSTACLE REMOVAL), even for restrictions of the problem to graphs of small outerplanarity, pathwidth, and maximum number of occurrences of each color:

Corollary 3.3. *Unless ETH fails,* CONNECTED OBSTACLE REMOVAL, *restricted to 2-outerplanar graphs of pathwidth at most 3 and in which each color appears at most 4 times, is not solvable in* $\mathcal{O}(2^{o(n)})$ *time, where* n *is the number of vertices in the graph.*

Proof. It is well known, and follows from [17] and the standard reduction from INDEPENDENT SET to VERTEX COVER, that unless ETH fails, VERTEX COVER, restricted to graphs of maximum degree at most 3, denoted VC-3, is not solvable in subexponential time. Starting from an instance of VC-3 with n vertices, and observing that the reduction in the proof of Theorem 3.1 results in an instance of CONNECTED OBSTACLE REMOVAL whose number of vertices is $O(n)$, of pathwidth at most 3, and in which each color appears at most 4 times, proves the result. □

Next, we shift our attention to studying the parameterized complexity of OBSTACLE REMOVAL and CONNECTED OBSTACLE REMOVAL. To show the NP-hardness of OBSTACLE REMOVAL, Hauser [13] gave a reduction from SET COVER to OBSTACLE REMOVAL. This reduction is in fact an FPT-reduction, which implies that OBSTACLE REMOVAL is W[2]-hard. We will strengthen this result, and show in the remainder of this section that OBSTACLE REMOVAL is W[SAT]-hard. We will also prove the membership of the problem in W[P], which adds a natural W[SAT]-hard problem to this class. The W[SAT]-hardness result shows that the problem is hopeless in terms of it having FPT-algorithms. We start by showing that the problem remains W[1]-hard, even when restricted to instances of small pathwidth (and hence small treewidth) and maximum number of occurrences of each color. We then show that the problem remains W[1]-hard even when parameterized by both k and the length of the sought path.

Remark 3.4. Before we prove our hardness results for OBSTACLE REMOVAL, we remark that we can obtain equivalent hardness results for GEOMERTIC OBSTACLE REMOVAL using the following generic realization of instances of OBSTACLE REMOVAL as instances of GEOMERTIC OBSTACLE REMOVAL. Given an instance (G, C, χ, s, t, k) of OBSTACLE REMOVAL, we define an equivalent instance of GEOMERTIC OBSTACLE REMOVAL as follows. We start by fixing a straight-line plane embedding Π of G, which always exists by Fáry's theorem [9]. Moreover, we can compute such an embedding in linear time [4]. We define the starting and finishing positions for the robot as the images of vertices s and t under Π, respectively. To force the robot to walk along the edges of G, we correspond with every edge a "corridor" by putting on both sides of the image of every edge $k+1$ trapezoids as shown in Figure 6. The only possible way to move between vertices of the graph G without intersecting more than k obstacles is to move within these corridors. Finally, for each color $c \in C$ and every vertex $v \in V(G)$ such that $c \in \chi(v)$, we create a rectangle around the image of the vertex v under Π that intersects all the trapezoids corresponding to the edges incident to v. We define the obstacle corresponding to the color c in the geometric instance to be the union of these rectangles. This disallows the use of less than $k+1$ trapezoid obstacles to go through a vertex v of G without intersecting all the obstacles representing the color set $\chi(v)$. Note that the only thing that is affected by this geometric realization is the number of obstacles that overlap at

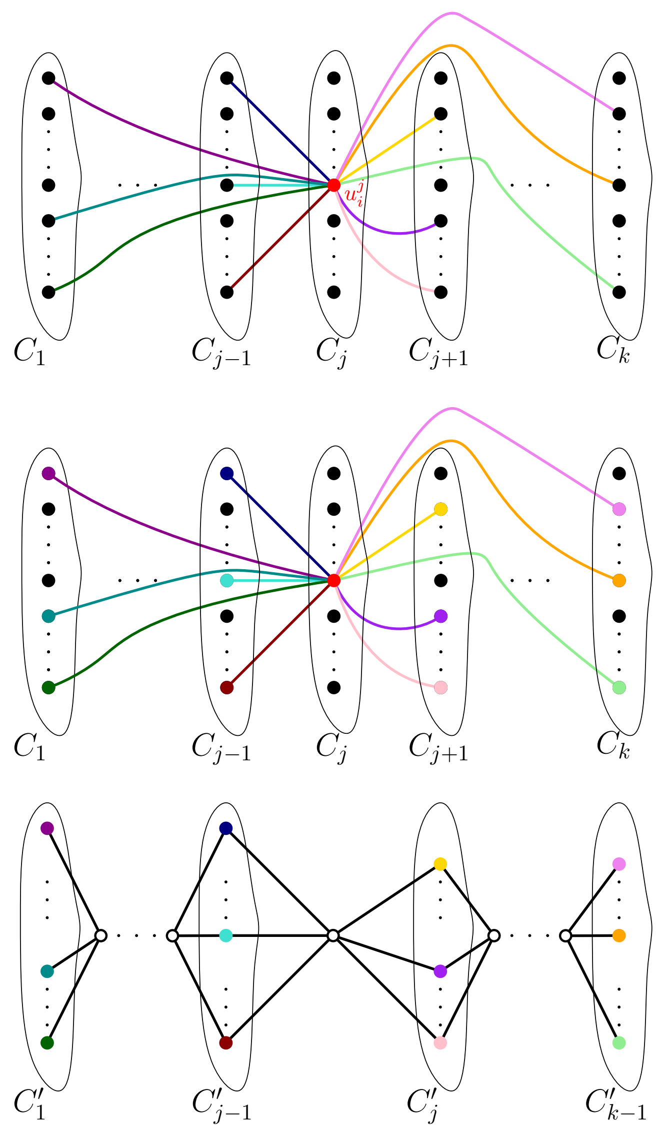

Figure 4: Illustration for the construction of the gadget $G_{i,j}$ in the proof of Lemma 3.5.

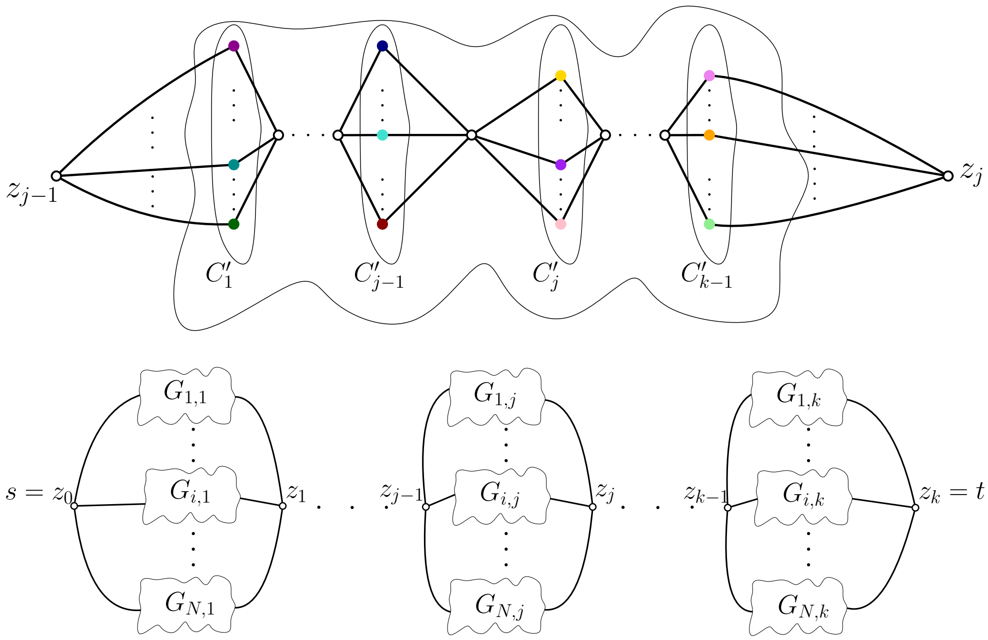

Figure 5: Illustration for the construction of G' in the proof of Lemma 3.5.

a region, which corresponds to the number of colors on the vertex in the graph that corresponds to the region; this number might increase by at most 4.

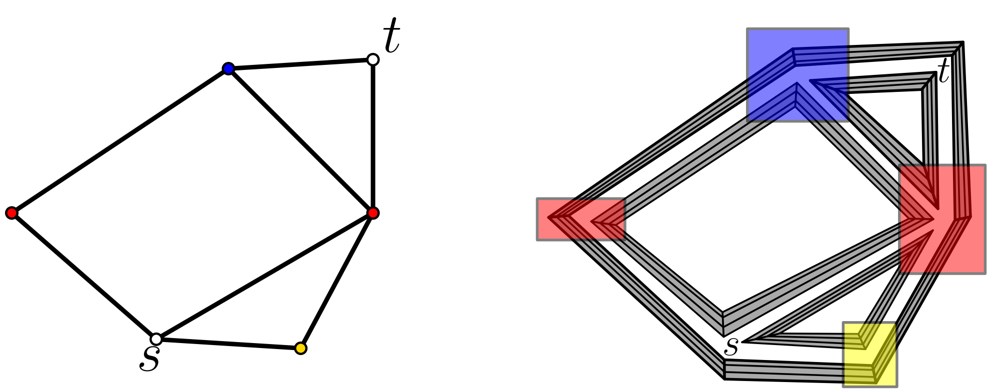

Figure 6: Illustration for the realization of an instance of Obstacle Removal as an instance of Geomertic Obstacle Removal.

Lemma 3.5. Obstacle Removal, *restricted to instances of pathwidth at most 4 and in which each vertex contains at most one color and each color appears on at most 2 vertices, is* W[1]-hard parameterized by k.

Proof. We reduce from the W[1]-hard problem Multi-Colored Clique [12]. Let (G, k) be an instance of Multi-Colored Clique, where $V(G)$ is partitioned into the color classes $C_1, \dots, C_k$. Let $C_j = \{u_i^j \mid i \in [|C_j|]\}$. We describe how to construct an instance $(G', C', \chi', s, t, k')$ of Obstacle Removal. For an edge $e \in G$, associate a distinct color c_e, and define $C' = \{c_e \mid e \in E(G)\}$. To simplify the description of the construction, we start by defining a gadget that will serve as a building block for this construction.

For a vertex u_i^j in color class C_j, we define the gadget $G_{i,j}$ as follows. Create a copy of each color class $C_{j'}$, $j' \neq j$, and remove from each $C_{j'}$ all copies of vertices that are not neighbors of u_i^j in G. Let the resulting copies of the color classes be $C'_1, \ldots, C'_{k-1}$. We define the color of a copy v' of a neighbor v of u_i^j as $\chi'(v') = \{c_e\}$, where $e = u_i^j v$. Next, we introduce $k-2$ empty vertices y_r, $r \in [k-2]$. For $r \in [k-2]$, we connect all vertices in C'_r to y_r, and connect y_r to all vertices in C'_{r+1}. This completes the construction of gadget $G_{i,j}$; we refer to C'_1 and C'_{k-1} as the first and last color classes in gadget $G_{i,j}$, respectively. See Figure 4 for illustration of $G_{i,j}$. Observe that every path from a vertex in C'_1 to a vertex in C'_{k-1} contains exactly one vertex from each C'_r, $r \in [k-1]$, and contains all vertices y_r, $r \in [k-2]$. Therefore, any such path contains the colors of exactly $k-1$ distinct edges that are incident to u_i^j.

We finish the construction of G' by introducing $k+1$ new empty vertices $z_0, \ldots, z_k$, and connecting them as follows. For each color class C_j, $j \in [k]$, and each vertex $u_i^j \in C_j$, we create the gadget $G_{i,j}$, connect z_{j-1} to each vertex in the first color class of $G_{i,j}$, and connect each vertex in the last color class of $G_{i,j}$ to z_j. Let G' be the resulting graph. Finally, we set $s = z_0$, $t = z_k$, and $k' = \binom{k}{2}$. See Figure 5 for illustration. This completes the construction of the instance $(G', C', \chi', s, t, k')$ of Obstacle Removal. Observe that each vertex in G' contains at most one color, and that each color c_e of an edge $e = u_i^j u_{i'}^{j'}$ in G, appears on exactly two vertices in G': the copy of $u_{i'}^{j'}$ in the gadget $G_{i,j}$ of u_i^j, and the copy of u_i^j in the gadget $G_{i',j'}$ of $u_{i'}^{j'}$.

Clearly, the reduction that takes an instance (G, k) of Multi-Colored Clique and produces the instance $(G', C', \chi', s, t, k')$ of Obstacle Removal is computable in FPT-time. To show its correctness, suppose that (G, k) is a yes-instance of Multi-Colored Clique, and let Q be a k-clique in G. Then Q contains a vertex from each C_j, for $j \in [k]$. For a vertex $u_i^j \in Q$, let $G_{i,j}$ be its gadget, and define the path P_j as follows. In each color class in $G_{i,j}$, pick the unique vertex that is a copy of a neighbor of u_i^j in Q; define P_j to be the path in $G_{i,j}$ induced by the picked vertices, plus the empty vertices y_r, $r \in [k-2]$, that appear in $G_{i,j}$. Finally, define P to be the s-t path in G' whose edges are: the (unique) edge between z_{r-1} and an endpoint of P_r, P_r, and the (unique) edge between an endpoint of P_r and z_r, for $r \in [k]$. To show that P is k'-valid, observe that all the nonempty vertices in P are vertices whose color is the color of an edge between two vertices in Q. This shows that the number of colors that appear on P is at most $k' = \binom{k}{2}$, and hence, P is k'-valid. It follows that $(G', C', \chi', s, t, k')$ is a yes-instance of Obstacle Removal.

Conversely, suppose that P is a k'-valid s-t path in G'. Then P' must start at s, visit the gadgets of exactly k vertices $u_{i_j}^j \in C_j$, for $j \in [k], i_j \in [|C_j|]$, and end at t. We claim that $Q = \{u_{i_j}^j \mid j \in [k]\}$ is a clique in G. Recall that the subpath of P that traverses a gadget $G_{i,j}$ of $u_{i_j}^j$ contains the colors of exactly $k-1$ edges that are incident to $u_{i_j}^j$. Therefore, the total number of occurrences of colors (counting multiplicities) on P is precisely $(k-1)k$. Since P is $\binom{k}{2}$-valid, and each color c_e of an edge e in G appears exactly twice in G', it follows that each color that appears on P appears exactly twice on P. This is only possible if the gadgets corresponding to the two endpoints of the edge are traversed by P, and hence, both endpoints of the edge are in Q. Therefore, P contains the colors of $k' = \binom{k}{2}$ edges, whose both endpoints are in Q. Since $|Q| = k$, it follows that Q is a k-clique in G, and that (G, k) is a yes-instance of Multi-Colored Clique. □

Lemma 3.6. Obstacle Removal, *parameterized by both k and the length of the path ℓ, is in* W[1].

Proof. To prove membership in W[1], we use the characterization of the class W[1] given by Chen *et al.* [3]:

> A parameterized problem Q is in W[1] if and only if there is a computable function h and a nondeterministic FPT algorithm $\mathbb{A}$ for a nondeterministic-RAM machine deciding Q, such that, for each instance (x, k) of Q (k is the parameter), all nondeterministic steps of $\mathbb{A}$ take place during the last $h(k)$ steps of the computation.

Therefore, to show that Obstacle Removal is in W[1], it suffices to exhibit such a nondeterministic FPT algorithm $\mathbb{A}$. $\mathbb{A}$ works as follows: it guesses a set C' of k colors and guesses a sequence of $\ell - 1$ internal vertices $v_1, \ldots, v_{\ell-1}$ of the path. Then it verifies that $(s = v_0, v_1, \ldots, v_{\ell-1}, v_\ell = t)$ is a path in G, and that $\chi(v_i) \subseteq C'$, for $i = 0, \ldots, \ell$. It is not difficult to see that this verification can be implemented in $h(k)$ steps, where h is a computable function. □

By Lemma 2.4, we can assume that in an instance of Obstacle Removal, no two adjacent vertices are empty. With this assumption in mind, if the instance satisfies that each vertex contains at most one color and that each color appears on at most 2 vertices, then any k-valid s-t path has length at most $4k + 1$. It follows from Lemma 3.5 and Lemma 3.6 that:

Theorem 3.7. Obstacle Removal*, parameterized by both k and the length of the path ℓ, is* W[1]-complete.

Theorem 3.8. Obstacle Removal*, restricted to instances of pathwidth at most 4 and in which each vertex contains at most one color and each color appears on at most 2 vertices, is* W[1]-complete parameterized by k.

Next, we show that Obstacle Removal sits high up in the parameterized complexity hierarchy. We start by showing its membership in W[P]:

Theorem 3.9. Obstacle Removal*, parameterized by k, is in* W[P].

Proof. We give an FPT-reduction from Obstacle Removal to Weighted Boolean Circuit Satisfiability (WBCS) on polynomial size (monotone) circuits. Given an instance (G, C, χ, s, t, k) of Obstacle Removal, we construct an instance (B, k) of WBCS, where B is a circuit whose output gate is an or-gate, as follows. By Assumption 2.2, we can assume that s and t are nonadjacent empty vertices. By Lemma 2.4, we can also assume that no two adjacent vertices are empty. For each color $c \in C$, we create a variable x_c; those are the input variables to B. In addition to the output gate, B contains $n = |V(G)|$ layers of gates, where each layer, except the first, consists of two rows of gates, U_i, L_i, for $i = 2, \ldots, n$, and the first layer consists of one row L_1 of gates. The layers of B are defined as follows.

Each gate in L_1 is an and-gate g_v that corresponds to a neighbor v of s; the input to g_v is the set of input variables corresponding to the colors in $\chi(v)$. Suppose that row L_i in layer i, $i \geq 1$, has been defined, and we describe how U_{i+1} and L_{i+1} are defined. For every vertex $v \in V(G)$ with a neighbor u such that u has a corresponding and-gate g_u^2 in L_i, we create an or-gate g_v^1 in U_{i+1} and an and-gate g_v^2 in L_{i+1} corresponding to v; we connect the output of each and-gate g_u^2 in L_i corresponding to neighbor u of v to the input of or-gate g_v^1 in U_{i+1}, and connect the output of the or-gate g_v^1 and each input variable x_c such that $c \in \chi(v)$ to the and-gate g_v^2 in L_{i+1}. If $v = t$, then we connect the output of the and-gate g_v^2 to the output gate of the circuit. This completes the description of B. Clearly, the reduction that takes (G, C, χ, s, t, k) to (B, k) runs in polynomial time, and hence in FPT-time. Next, we prove its correctness.

First observe that the only gates in B that are connected to its output gate are the and-gates that correspond to t. Second, every gate in B corresponds to a vertex that is reachable from s in

G. Moreover, for every AND-gate g corresponding to a vertex v, and every s-v path in G, the truth assignment that assigns 1 to the variables corresponding to the colors of this path satisfies g.

Suppose now that (G, C, χ, s, t, k) is a yes-instance of Obstacle Removal. Then there is an s-t k-valid path P in G. Based on the above observations, the assignment that assigns $x_c = 1$ if and only if $c \in \chi(P)$ is a satisfying assignment to B of weight at most k. Conversely, suppose that B has a satisfying assignment τ of weight at most k. Then there is an AND-gate g corresponding to t that is satisfied by τ, and there is a path in B from a gate corresponding to neighbor of s in L_1 to g, all of whose gates are satisfied by τ. It is easy to verify that this path in B corresponds to an s-t path all of whose colors correspond to the input variables assigned 1 by τ, and hence this path is k-valid. □

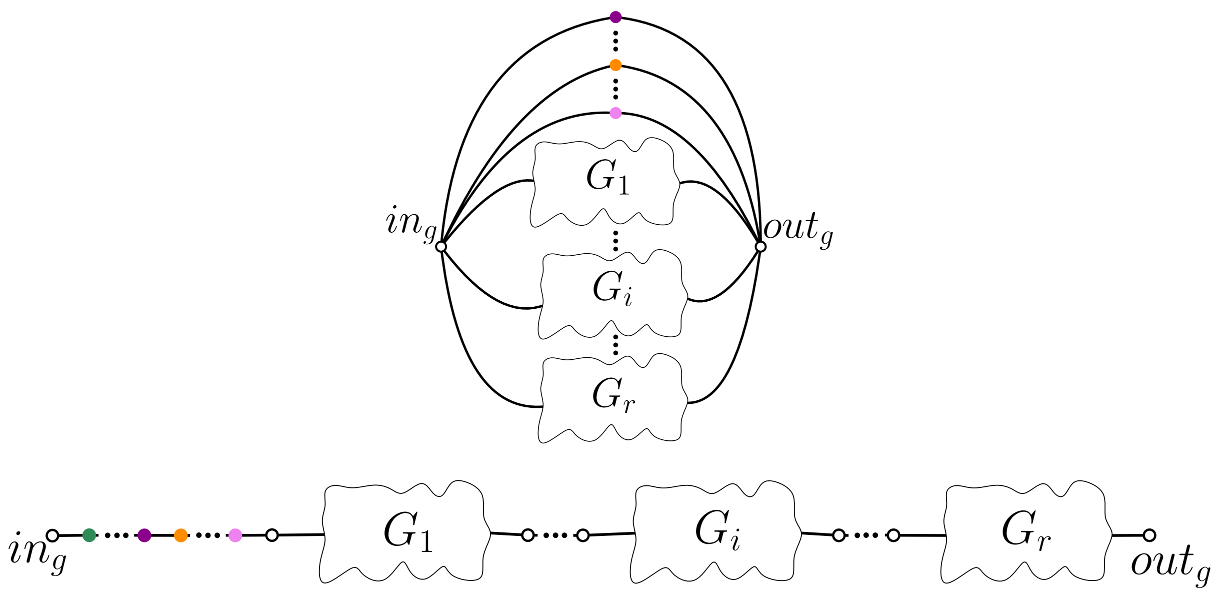

Figure 7: Illustrations of the construction of the gadgets for an OR-gate (top) and an AND-gate (bottom) in the proof of Theorem 3.10.

Theorem 3.10. Obstacle Removal, *parameterized by* k, *is* W[SAT]-hard.

Proof. We give an FPT-reduction from the W[SAT]-complete problem Monotone Weighted Boolean Formulas Satisfiability (M-WSAT) [6].

Recall that a Boolean formula corresponds to a circuit in the normalized form. Therefore, we can assume that the input instance of M-WSAT is (B, k), where B is a monotone Boolean circuit in which each (non-variable) gate has fan-out at most 1, and the gates of B are structured into alternating levels of ORs-of-ANDs-of-ORs. We construct an instance (G, C, χ, s, t, k) of Obstacle Removal as follows.

First, we let $C = [n]$, where color i will represent input variable x_i in B. We define G from B by defining a gadget for each gate in B recursively, starting the recursive definition at the output gate of B. For a gate g in B, its gadget is defined as follows by distinguishing the type of g.

If g is an AND-gate, let $g_1, \ldots, g_r$ be the OR-gates, and $x_{i_1}, \ldots, x_{i_p}$ be the input variables that feed into g. The gadget of g is defined as follows. First, create two empty vertices in_g and out_g, which will serve as the "entry" and "exit" vertices of the gadget for g, respectively. For each x_{i_j}, $j \in [p]$, create a vertex v_j colored with color i_j and an entry vertex v_0 and an exit vertex v_{p+1}; form a path

G_0 consisting of the vertices $v_0, v_1, \ldots, v_p, v_{p+1}$. For each OR-gate g_i, $i \in [r]$, recursively construct the gadget G_i for g_i. Connect all these gadgets $G_0, \ldots, G_r$ serially in arbitrary order, starting by identifying in_g with the entry vertex of the first gadget, the exit vertex of the first gadget with the entry of the second, ..., and the exit vertex of the last gadget with out_g. See Figure 7 (bottom) for illustration.

If g is an OR-gate, let $g_1, \ldots, g_r$ be the OR-gates, and $x_{i_1}, \ldots, x_{i_p}$ be the input variables that feed into g. The gadget of g is defined as follows. First, create two empty vertices in_g and out_g, which will serve as the "entry" and "exit" vertices of the gadget for g, respectively. For each x_{i_j}, $j \in [p]$, create a vertex v_j colored with color i_j, and connect each v_j to in_g and out_g. For each AND-gate g_i, $i \in [r]$, recursively construct the gadget G_i for g_i. Connect all these gadgets $G_1, \ldots, G_r$ in parallel by identifying all the entry vertices of $G_1, \ldots, G_r$ with in_g and all their exit vertices with out_g. This complete the description of G. It is not difficult to see that since B with its input variables removed is a tree, the above construction runs in polynomial time and results in a planar graph G. See Figure 7 (top) for illustration.

Finally, set s and t to be the entry and exit vertices of the gadget corresponding to the output gate of B. Clearly, the reduction that takes (B, k) and produces (G, C, χ, s, t, k) runs in FPT-time. Next, we prove its correctness.

We will prove the following statement: For any gate g in B, and any assignment τ to B that assigns variables $x_{i_1}, \ldots, x_{i_p}$ the value 1, and all other variables the value 0, τ satisfies g if and only if there is a path P in G from the entry vertex to the exit vertex of the gadget corresponding to g such that P uses a subset of the colors $\{i_1, \ldots, i_p\}$. Clearly, proving the aforementioned statement implies that there is a k-valid s-t path in G if and only if there is an assignment of weight at most k that satisfies the output gate of B, and hence satisfies B.

We prove the above statement by induction on the depth of the gate g in B. The base case is when g has depth 1. In this case the input to g consists only of input variables. Suppose first that g is an OR-gate, and let τ be an assignment that assigns exactly variables $x_{i_1}, \ldots, x_{i_p}$ the value 1. Then τ satisfies g if and only if x_{i_j} is an input variable to g, for some $j \in [p]$, which is true if and only if there is a path from the entry vertex of the gadget for g to its exit vertex that uses color i_j. Suppose now that g is an AND-gate, and let τ be an assignment that assigns exactly variables $x_{i_1}, \ldots, x_{i_p}$ the value 1. Then τ satisfies g if and only if the input variables to g form a subset S of $\{x_{i_1}, \ldots, x_{i_p}\}$; let $\eta(S)$ be the indices of the variables in S. Since the gadget for g consists of a path P between the entry and exit vertices of the gadget for g such that $\chi(P) = \eta(S)$, the statement follows.

Suppose, by the inductive hypothesis, that the statement we are proving is true for any gate g of depth $1 \leq a < \ell$, and let g be a gate of depth ℓ. Let $x_{j_1}, \ldots, x_{j_q}$ be the input variables to g, and $g_1, \ldots, g_r$ be the input gates to g. We again distinguish two cases based on the type of g.

Gate g is an OR-gate. Let τ be an assignment that assigns exactly variables $x_{i_1}, \ldots, x_{i_p}$ the value 1. Suppose first that τ satisfies g. Then either τ satisfies an input variable x_{j_z}, $z \in [q]$, or τ satisfies an input AND-gate g_y, $y \in [r]$. If τ satisfies x_{j_z} then there is a path between the entry and exit vertices of the gadget for g that uses color j_z. Otherwise, τ satisfies g_y, $y \in [r]$, and by the inductive hypothesis applied to g_y, there is a path P_y between the entry and exit vertices of the gadget for g_y such that $\chi(P_y) \subseteq \{i_1, \ldots, i_p\}$. From the way the gadget for g was constructed, it follows that P_y is also a path between the entry and exit vertices of the gadget for g. To prove the converse, suppose that there is a path P_g between the entry and exit vertices of the gadget for g that uses a subset of colors in $\{i_1, \ldots, i_p\}$. Either P_g is a path whose only internal vertex corresponds to an input variable, and in such case the input variable is in $\{x_{i_1}, \ldots, x_{i_p}\}$, and g is satisfied; or P_g is a path between the entry and exit vertices of the gadget for an AND-gate g_y that feeds into g, and by the inductive hypothesis, τ satisfies g_y and also g.

Gate g is an AND-gate. Let τ be an assignment that assigns exactly variables $x_{i_1}, \ldots, x_{i_p}$ the value 1. Suppose first that τ satisfies g. Then τ assigns 1 to every input variable x_{j_z} to g, $z \in [q]$. Hence, there is a path P between the entry and exit vertices of the gadget corresponding to $x_{j_1}, \ldots, x_{j_q}$ such that $\chi(P) \subseteq \{i_1, \ldots, i_p\}$. Assignment τ also satisfies each OR-gate g_y, where $y \in [r]$. By the inductive hypothesis, there is a path P_y between the entry and exit vertices of the gadget for g_y such that $\chi(P_y) \subseteq \{i_1, \ldots, i_p\}$. From the construction of g, it follows that the path between the entry and exit vertices of the gadget for g, which is $P_g = P \circ P_1 \circ \cdots \circ P_r$, satisfies $\chi(P_g) \subseteq \{i_1, \ldots, i_p\}$. Conversely, suppose that there is a path P_g between the entry and exit vertices of the gadget for g such that $\chi(P_g) \subseteq \{i_1, \ldots, i_p\}$. Then P_g can be decomposed into a subpath P that traverses the vertices corresponding to $x_{j_1}, \ldots, x_{j_q}$, and subpaths $P_1, \ldots, P_r$, where P_y is a subpath between the entry and exit vertices of the gadget for g_y. Since P traverses the vertices corresponding to $x_{j_1}, \ldots, x_{j_q}$, it follows that $\{x_{j_1}, \ldots, x_{j_q}\} \subseteq \{x_{i_1}, \ldots, x_{i_p}\}$. Since P_y, $y \in [r]$, is a subpath between the entry and exit vertices of the gadget for g_y, by the inductive hypothesis, it follows that τ satisfies g_y. It follows that τ assigns 1 to all input variables to g and satisfies all the input OR-gates to g, and hence, τ satisfies g. ☐

Remark 3.11. *A noteworthy remark that we close this section with, is to comment on the role that planarity plays in the parameterized complexity of* CONNECTED OBSTACLE REMOVAL. *If one drops the planarity requirement on the instances of* CONNECTED OBSTACLE REMOVAL *(*i.e., *considers* CONNECTED OBSTACLE REMOVAL *on general graphs), then it follows from the proof of Theorem 3.10 that the resulting problem is* W[SAT]*-hard. This can be seen by adding a single vertex containing all colors, that serves as a "color-connector," to the instance of* OBSTACLE REMOVAL *produced by the* FPT*-reduction; this modification results in an instance of the connected obstacle removal problem on apex graphs, establishing the* W[SAT]*-hardness of this problem on apex graphs.*

4 Structural Results

Let G be a color-connected plane graph, C a set of colors, and $\chi : V \longrightarrow 2^C$. In this section, we present structural results that are the cornerstone of the FPT-algorithm for CONNECTED OBSTACLE REMOVAL presented in the next section. The ultimate goal of this section is to show that, for any vertex $w \in V(G)$, and for any pair of vertices $u, v \in V(G)$, the set of k-valid u-v paths in $G - w$ that use colors external to w can be "represented" by a minimal set whose size is a function of k. This result is the key ingredient of the dynamic programming FPT-algorithm in the next section, that is based on tree decomposition of the input graph; it allows us to extend the notion of a minimal set of k-valid u-v paths w.r.t. a single vertex w, to all the vertices of a bag in the tree decomposition, yielding a representative set for the whole bag. Throughout this section, we shall assume that G is color-connected. We start with the following simple observation:

Observation 4.1. *Let $x, y \in V(G)$ be such that there exists a color $c \in \mathcal{C}$ that appears on both x and y. Then any x-y vertex-separator in G contains a vertex on which c appears.*

Proof. This follows because color c is connected. ☐

Let G' be a plane graph, let $w \in V(G')$, and let f be the face in $G' - w$ such that w is interior to f; we call f the *external face* with respect to w in G', and the vertices incident to f *external vertices* with respect to w in G'. A color $c \in C$ is said be an *external color* with respect to w in G', or simply *external* to w in G', if c appears on an external vertex with respect to w in G'; otherwise, c is said be *internal* to w in G'. The following observation is easy to see:

Observation 4.2. *Let G be a color-connected graph, and let $w \in V(G)$. Let H be any subgraph of $G - w$. If c is an external color to w in $G - w$ and c appears on some vertex in H, then c is an external color to w in H. This also implies that the set of internal colors to w in H is a subset of the set of internal colors to w in $G - w$.*

Definition 4.3. Let $P = (w_1, \ldots, w_r)$ be a path in a graph G, and let $x, y \in V(G)$. Suppose that we apply the color contraction operation to x and y, and let z be the new vertex resulting from this contraction. We define an operation, denoted Λ_{xy}, that when applied to path P results in another path $\Lambda_{xy}(P)$ defined as follows:

1. If $\{x, y\} \cap \{w_1, \ldots, w_r\} = \emptyset$ then $\Lambda_{xy}(P) = P$.
2. If $\{x, y\} \cap \{w_1, \ldots, w_r\} = \{w_i\}$, where $i \in [r]$, then $\Lambda_{xy}(P) = (w_1, \ldots, w_{i-1}, z, w_{i+1}, \ldots, w_r)$.
3. If $\{x, y\} \cap \{w_1, \ldots, w_r\} = \{w_i, w_j\}$, where $i < j$, then $\Lambda_{xy}(P) = (w_1, \ldots, w_{i-1}, z, w_{j+1}, \ldots, w_r)$.

For a set of paths $\mathcal{P}$, we define $\Lambda_{xy}(\mathcal{P}) = \{\Lambda_{xy}(P) \mid P \in \mathcal{P}\}$.

Definition 4.4. Let $u, v, w \in V(G)$. A set $\mathcal{P}$ of k-valid u-v paths in $G - w$ is said to be *minimal* with respect to w if:

(i) There does not exist two paths $P_1, P_2 \in \mathcal{P}$ such that $\chi(P_1) \cap \chi(w) = \chi(P_2) \cap \chi(w)$;

(ii) there does not exist two paths $P_1, P_2 \in \mathcal{P}$ such that $\chi(P_1) \subseteq \chi(P_2)$; and

(iii) for any $P \in \mathcal{P}$, there does not exist a u-v path P' in $G - w$ such that $\chi(P') \subsetneq \chi(P)$.

Clearly, for any $u, v, w \in V(G)$, a minimal set of k-valid u-v paths in $G - w$ exists.

Observation 4.5. *Let $u, v, w \in V(G)$. Any set of u-v paths that is minimal with respect to w contains at most one path whose vertices contain only internal colors w.r.t. w in $G - w$.*

Proof. Since the external face f of w in $G - w$ is a Jordan curve that separates w from any vertex in $G - w$ that is not incident to f, by Observation 4.1, any color that appears both on w and on a vertex in $G - w$ must appear on a vertex incident to f, and hence, must be external to w by definition. Therefore, any path P containing only internal colors to w satisfies $\chi(P) \cap \chi(w) = \emptyset$. The observation now follows from property (i) in Definition 4.4. ☐

Lemma 4.6. *Let $u, v, w \in V(G)$, and let $\mathcal{P}$ be a minimal set of k-valid u-v paths in $G - w$. Suppose that we apply the color contraction operation to an edge $xy \in G - w$, and let G', χ' be the graph and color function obtained from the contraction operation, respectively. Let $\mathcal{P}' = \Lambda_{xy}(\mathcal{P})$. Then $\mathcal{P}'$ is minimal w.r.t w in G'.*

Proof. Let H' be the subgraph of $G' - w$ induced by the edges of the paths in $\mathcal{P}'$, and denote by z the new vertex obtained from the contraction of the edge xy. We start by showing the following claim:

Claim 1. *For every $P \in \mathcal{P}$, it holds that $\chi(\Lambda_{xy}(P)) = \chi(P)$.*

Let $P = (u = w_1, \ldots, w_r = v)$. Since $\chi'(z) = \chi(x) = \chi(y)$, it follows from Definition 4.3 that if $|\{x, y\} \cap \{w_1, \ldots, w_r\}| \leq 1$, then $\chi(\Lambda_{xy}(P)) = \chi(P)$. Now assume that $\{x, y\} \cap \{w_1, \ldots, w_r\} = \{w_i, w_j\}$, where $i < j$, and suppose to get a contradiction that $\chi(\Lambda_{xy}(P)) \neq \chi(P)$. Since $\Lambda_{xy}(P) = (w_1, \ldots, w_{i-1}, z, w_{j+1}, \ldots, w_r)$, it follows that $\chi(\Lambda_{xy}(P)) \subsetneq \chi(P)$. However, $G - w$ contains the u-v path $P' = (w_1, \ldots, w_{i-1}, w_i, w_j, w_{j+1}, \ldots, w_r)$, which satisfies $\chi(P') = \chi(\Lambda_{xy}(P)) \subsetneq \chi(P)$; this, together with $P \in \mathcal{P}$, contradicts the minimality of $\mathcal{P}$.

We now proceed to verify that $\mathcal{P}'$ is indeed minimal with respect to w. Properties (i) and (ii) in Definition 4.4 follow directly from Claim 1 and the minimality of $\mathcal{P}$. To prove that property (iii) holds, assume that there is a path $P' \in \mathcal{P}'$, and a path Q' in $G' - w$ between the endpoints of P' such that $\chi(Q') \subsetneq \chi(P')$. Let P be the path in $\mathcal{P}$ such that $\Lambda_{xy}(P) = P'$. It is straightforward to verify that $G - w$ contains a u-v path Q that is either identical to Q', or obtained from Q' by replacing z by either a single vertex x or y, or by the pair x, y. Clearly, $\chi(Q) = \chi(Q')$. Since $\chi(Q') \subsetneq \chi(P') = \chi(P)$ by Claim 1, it follows that $\chi(Q) \subsetneq \chi(P)$, contradicting the minimality of $\mathcal{P}$. It follows that Property (iii) holds, and the proof is complete. □

To upper bound the cardinality of a minimal set of k-valid u-v path w.r.t. a vertex w by a function of k, we first select a maximal subset of color-disjoint paths in this set, and upper bound the cardinality of this subset; we do so by showing that this subset induces a graph that has a small vertex-separator, and then applying an inductive counting argument based on this separator. We then show, again using an inductive proof, that the upper bound on the cardinality of this color-disjoint subset of paths implies an upper bound on the cardinality of the whole minimal set of k-valid u-v paths w.r.t. w.

For the rest of this section, we let $u, v, w \in V(G)$, and let $\mathcal{P}$ be a set of minimal k-valid u-v paths in $G - w$. Let $\mathcal{M}$ be a set of minimal k-valid color-disjoint u-v paths in $G - w$. Let H be the subgraph of $G - w$ induced by the edges of the paths in $\mathcal{P}$, and let M be that induced by the edges of the paths in $\mathcal{M}$.

Observation 4.7. *If $P \in \mathcal{M}$ contains a color c that is external to w in M, then c appears on a vertex in P that is incident to the external face to w in M.*

Proof. By definition, c appears on a vertex x incident to the external face with respect to w in M. Since the paths in $\mathcal{M}$ are pairwise color-disjoint and c appears on P, it follows that x is a vertex of P. □

Lemma 4.8. *Let G' be a plane graph, and let $x, y, z \in V(G')$. Let $x_1, \ldots, x_r$, $r \geq 3$, be the neighbors of x in counterclockwise order. Suppose that, for each $i \in [r]$, there exists an x-y path P_i containing x_i such that P_i does not contain z and does not contain any x_j, $j \in [r]$ and $j \neq i$. Then there exist two paths P_i, P_j, $i, j \in [r]$ and $i \neq j$, such that the two paths P_i, P_j induce a Jordan curve separating $\{x_1, \ldots, x_r\} \setminus \{x_i, x_j\}$ from z.*

Proof. The proof is by induction on $r \geq 3$. The base case is when $r = 3$. Consider the faces induced by the two paths P_1 and P_2 in the embedding. If z and x_3 are in two separate faces, then clearly P_1 and P_2 induce a Jordan curve separating x_3 from z, and we are done. Therefore, we can assume that z and x_3 are in the same face induced by P_1 and P_2. Since P_1 does not contain x_2, we can continuously deform P_1 into an isotopic non self-intersecting curve P'_1 w.r.t. x_3, x_2, z, that includes xx_1, intersects edges xx_2 and xx_3 only at x, and intersects P_2 only at x and y. Similarly, if P_2 and P_3 do not separate z from x_1, then z and x_1 are in the same face induced by P_2 and P_3 and we can define a curve P'_3 that is isotopic to P_3 w.r.t. x_2, x_1, z, and such that P'_3 contains xx_3, intersects xx_2 and xx_1 only at x, and intersects P_2 only at x and y. Now if z and x_2 are in different faces induced by P'_1 and P'_3, then P'_1 and P'_3 separate z from x_2, and since P_1 is isotopic to P'_1 w.r.t. z

and x_2, and P'_3 is isotopic to P_3 w.r.t. z and x_2, it follows that P_1 and P_3 induce a Jordan curve that separates x_2 from z. Assume now that z and x_2 are in the same face f induced by P'_1 and P'_3. Since P_2 intersects with each of P'_1 and P'_3 precisely at x and y, it follows that P_2 splits f into two faces f_1, f_2, where xx_2, xx_1 are two consecutive edges on the boundary of f_1 and xx_2, xx_3 are two consecutive edges on the boundary of f_2. Then, z must be interior to exactly one of the two faces f_1, f_2. If z is interior to f_1, let f'_1 be the face induced by P'_1 and P_2 and containing z. Then f'_1 contains f_1, and does not contain x_3 (because P'_1 intersects xx_3 only at x). Therefore, f'_1, and hence, P'_1 and P_2 induce a Jordan curve that separates z from x_3. It follows that P_1, which is isotopic to P'_1 w.r.t. x_2, x_3, z, and P_2 induce a Jordan curve that separates z from x_3. Similarly, if z is interior to f_2, then P'_3, P_2 induce a Jordan curve that separates z from x_1, and hence, P_3 and P_2 induce a Jordan curve that separates z from x_1.

Assume inductively that the statement of the lemma is true for any $3 \leq \ell < r$. By the inductive hypothesis applied to $x_1, \ldots, x_{r-1}$, there exist two paths P_i, P_j, $i, j \in [r-1]$ and $i \neq j$, such that the two paths P_i, P_j induce a Jordan curve separating $\{x_1, \ldots, x_{r-1}\} \setminus \{x_i, x_j\}$ from z. If x_r and z are not in the same face induced by P_i, P_j, then P_i, P_j separate x_r from z as well, and we are done. Assume now that z and x_r are in the same face f induced by P_i, P_j. Since P_i, P_j separate z from $\{x_1, \ldots, x_{r-1}\} \setminus \{x_i, x_j\}$, none of $\{x_1, \ldots, x_{r-1}\} \setminus \{x_i, x_j\}$ is interior to f, and hence, x_r is the only neighbor of x between x_i and x_j w.r.t. the rotation system of G', which implies w.l.o.g. that $x_1 = x_i$ and $x_{r-1} = x_j$. By the inductive hypothesis applied to x_1, x_{r-1}, x_r there are two paths in P_1, P_{r-1}, P_r that induce a Jordan curve that separates z from one of x_1, x_{r-1}, x_r. Since P_1 and P_{r-1} do not separate x_r from z, one of these two path must be P_r; assume, w.l.o.g., that the two paths are P_1 and P_r. Since x_1 and x_r are consecutive neighbors in the rotation system, and since P_1, P_r do not contain any of $x_2, \ldots, x_{r-1}$, it follows that $x_2, \ldots, x_{r-1}$ are in the same face induced by P_1, P_r, and this face does not contain z because P_1, P_r separate z from x_{r-1}. It follows that P_1, P_r induce a Jordan curve that separates z from $x_2, \ldots, x_{r-1}$. This completes the inductive proof. □

Lemma 4.9. *Let G' be a plane graph with a face f, and let $u, v \in V(G')$. Let $u_1, \ldots, u_r$, $r \geq 3$, be the neighbors of u. Suppose that, for each $i \in [r]$, there exists a u-v path P_i in G' containing u_i and a vertex incident to f different from v, and such that P_i does not contain any u_j, $j \in [r]$, $j \neq i$. Then there exist two paths P_i, P_j, $i, j \in [r]$, $i \neq j$, such that $V(P_i) \cup V(P_j) - \{v\}$ is a vertex-separator separating $\{u_1, \ldots, u_r\} \setminus \{u_i, u_j\}$ from v.*

Proof. Create a new vertex y interior to f. Each path P_i, $i \in [r]$, contains a vertex y_i incident to f and different from v; we define a new path P'_i from u to y, consisting of the prefix of P_i up to y_i, and extending this prefix by adding a new edge between y_i and the new vertex y. Note that we can extend the rotation system of G' in a straightforward manner to obtain a rotation system for the plane graph resulting from adding y and the edges $y_i y$ to G', $i \in [r]$. Since v is the endpoint of P_i and $v \neq y_i$, it follows that v is not contained in P'_i, for $i \in [r]$. By Lemma 4.8, there exist two paths P'_i, P'_j, $i, j \in [r]$, and $i \neq j$, such that the two paths P'_i, P'_j induce a Jordan curve separating $\{u_1, \ldots, u_r\} \setminus \{u_i, u_j\}$ from v in $G' + y$. It follows that $V(P'_i) \cup V(P'_j)$ is a vertex separator separating $\{u_1, \ldots, u_r\} \setminus \{u_i, u_j\}$ from v in $G' + y$, and hence, $V(P'_i) \cup V(P'_j) - \{y\} \subseteq V(P_i) \cup V(P_j) - \{v\}$ is a vertex-separator separating $\{u_1, \ldots, u_r\} \setminus \{u_i, u_j\}$ from v in G'. □

Lemma 4.10. *Let x, y be two vertices in an irreducible subgraph G' of G, and let f be a face in G'. Then there are at most two color-disjoint x-y paths in G' that contain only colors that appear on f.*

Proof. Suppose, to get a contradiction, that there are three color-disjoint x-y paths P_1, P_2, P_3 in G' that contain only colors that appear on f. We create a new vertex z interior to f and add edges between z and each vertex incident to f. Note that we can extend the rotation system of G' in a straightforward manner to obtain a rotation system for the plane graph resulting from adding z and the edges incident to it to G'. Clearly, none of P_1, P_2, P_3 contains z. Because the paths P_1, P_2, P_3 are color-disjoint, both x and y must be empty vertices. Let v_1, v_2, v_3 be the neighbors of x on P_1, P_2, P_3, respectively. Since x is an empty vertex and G' is irreducible, none of v_1, v_2, v_3 is an empty vertex, and hence each v_i, $i \in [3]$, must contain a color c_i that appears on f. Since P_1, P_2, P_3 are pairwise color-disjoint, it follows that no vertex in $\{v_1, v_2, v_3\} \setminus \{v_i\}$ is contained in P_i, for $i \in [3]$. By Lemma 4.8, there is a v_i, $i \in [3]$, such that the two paths in $\{P_1, P_2, P_3\} - P_i$ induce a Jordan curve in $G' + z$ separating v_i and z, and hence separating v_i from each vertex incident to f. Since c_i appears on both v_i and a vertex incident to f, by Observation 4.1, it follows that c_i must appear on a vertex in $V(P_1) \cup V(P_2) \cup V(P_3) - V(P_i)$. This is a contradiction since c_i appears on P_i and the paths P_1, P_2, P_3 are pairwise color-disjoint. $\square$

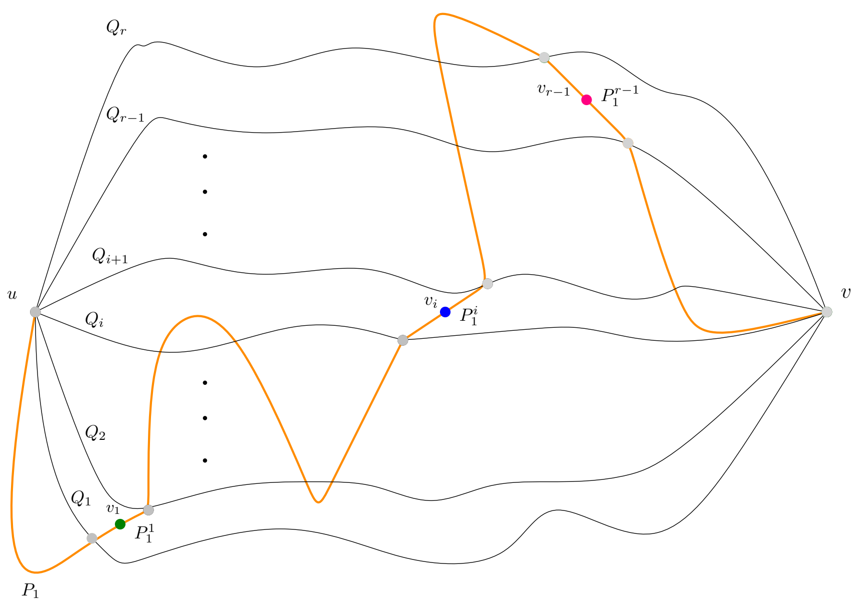

Figure 8: Illustration for the proof of Lemma 4.11.

Lemma 4.11. *Suppose that M is irreducible, then there exist paths $P_1, P_2, P_3 \in \mathcal{M}$ such that $M - P_1 - P_2 - P_3$ has a u-v vertex-separator of cardinality at most $2k + 3$.*

Proof. By Observation 4.5, $\mathcal{M}$ contains at most one path that contains only internal colors with respect to w in M. Therefore, it suffices to show that $\mathcal{M}$ contains two paths P_1', P_2' such that $M - P_1' - P_2'$ has a u-v vertex-separator of cardinality at most $2k + 3$, assuming that every path in $\mathcal{M}$ contains an external color w.r.t. w in M.

By Observation 4.7, every path in $\mathcal{M}$ passes through an external vertex w.r.t. w in M that contains an external color to w in M. Because the paths in $\mathcal{M}$ are pairwise color-disjoint, u and v are empty vertices, and hence, every path in $\mathcal{M}$ passes through a vertex on the external face of w in M that is different from u and v. Let $u_1, \ldots, u_q$ be the neighbors of u in M, and note that since u is empty and M is irreducible, each u_i, $i \in [q]$, contains a color. Let $P_1, \ldots, P_q$ be the paths in $\mathcal{M}$ containing $u_1, \ldots, u_q$, respectively, and note that since the paths in $\mathcal{M}$ are color-disjoint, no P_i passes through u_j, for $j \neq i$. By Lemma 4.9, there are two paths in $P_1, \ldots, P_q$, say P_1, P_2 without loss of generality, such that $V_{12} = V(P_1) \cup V(P_2) - \{v\}$ is a vertex separator that separates $\{u_3, \ldots, u_q\}$ from v.

We proceed to prove the lemma by contradiction and assume that $M^- = M - P_1 - P_2$ does not have a u-v vertex-separator of cardinality $2k+3$. By Menger's theorem [5], there exists a set $\mathcal{D}$ of $r' \geq 2k+3$ vertex-disjoint u-v paths in M^-. Since V_{12} separates $\{u_3, \ldots, u_q\}$ from v in M, every u-v in M^- intersects (shares a vertex with) at least one of P_1, P_2 at a vertex other than v. It follows that there exists a path in $\{P_1, P_2\}$, say P_1, that intersects at least $k+2$ paths in $\mathcal{D}$ at vertices other than v. Since the paths in $\mathcal{D}$ are vertex-disjoint and incident to u, we can order the paths in $\mathcal{D}$ that intersect P_1 around u (in counterclockwise order) as $\langle Q_1, \ldots, Q_r \rangle$, $r \geq k+2$, where Q_{i+1} is counterclockwise from Q_i, for $i \in [r-1]$. P_1 intersects each path Q_i, $i \in [r]$, possibly multiple times. Moreover, since the paths in $\mathcal{M}$ are pairwise color-disjoint, each intersection between P_1 and a path Q_i, $i \in [r]$, must occur at an empty vertex. We choose $r-1$ subpaths of P_1, $P_1^1, \ldots, P_1^{r-1}$, satisfying the property that the endpoints of P_1^i are on Q_i and Q_{i+1}, for $i = 1, \ldots, r-1$, and the endpoints of P_1^i are the only vertices on P_1^i that appear on a path Q_j, for $j \in [r]$. It is easy to verify that the subpaths $P_1^1, \ldots, P_1^{r-1}$ of P_1 can be formed by following the intersection of P_1 with the sequence of (ordered) paths $Q_1, \ldots, Q_r$. See Figure 8 for illustration.

Recall that the endpoints of $P_1^1, \ldots, P_1^{r-1}$ are empty vertices. Since M is irreducible, no two empty vertices are adjacent, and hence, each subpath P_1^i must contain an internal vertex v_i that contains at least one color. We claim that no two vertices v_i, v_j, $1 \leq i < j \leq r-1$, contain the same color. Suppose not, and let v_i, v_j, $i < j$, be two vertices containing a color c. Since v_i, v_j are internal to P_1^i and P_1^j, respectively, $Q_1, \ldots, Q_r$ are vertex-disjoint u-v paths, and by the choice of the subpaths $P_1^1, \ldots, P_1^{r-1}$, the paths Q_i and Q_{i+1} form a Jordan curve, and hence a vertex separator in G, separating v_i from v_j. By Observation 4.1, color c must appear on a vertex in Q_p, $p \in \{i, i+1\}$, and this vertex is clearly not in P_1 since P_1 intersects Q_p at empty vertices. Since every vertex in M appears on a path in $\mathcal{M}$, and c appears on $P_1 \in \mathcal{M}$ and on a vertex not in P_1, this contradicts that the paths in $\mathcal{M}$ are pairwise color-disjoint, and proves the claim.

Since no two vertices v_i, v_j, $1 \leq i < j \leq r$, contain the same color, this implies that the number of subpaths $P_1^1, \ldots, P_1^{r-1}$, $r-1$, is upper bounded by the number of distinct colors that appear on P_1, which is at most k. It follows that r, and hence, the number of vertex-disjoint u-v paths in M is at most $k+1$, contradicting our assumption above and proving the lemma. ☐

Lemma 4.12. *Let S be a minimal u-v vertex-separator in M. Let M_u, M_v be a partition of $M - S$ containing u and v, respectively, and such that there is no edge between M_u and M_v. For any vertex $x \in S$, M_u is contained in a single face of $M_v + x$.*

Proof. Let $x \in S$. It suffices to show that the subgraph F of M induced by $V(M_u) \cup (S \setminus \{x\})$ is connected. This suffices because $V(F)$ and $V(M_v + x)$ are disjoint, and hence every face in $M_v + x$ separates the vertices in $V(F)$ inside the face from those outside of it. We will show that F is connected by showing that there is a path in F from each vertex in F to $u \in V(F)$. Let $z \in V(F)$. If $z \in S$, then by minimality of S, there is a path from u to z whose internal vertices are all in M_u, and hence this path is in F. If $z \notin S$, let P be a u-v path containing z. If P passes through z before passing through any vertex in S, then clearly there is a path from u to z in F. Otherwise, P passes through a vertex $y \in S$ before passing through z. In this case, there exists a vertex $y' \in S$, such that $y' \neq y$ and P passes through y' after passing through z. Either y or y', say y', is different from x. From the above discussion, there is a path P' from u to y' in F, which when combined with the subpath of P between y' and z yields a path from u to z in F. ☐

Lemma 4.13. $|\mathcal{M}| \leq g(k)$, *where* $g(k) = \mathcal{O}(c^k k^{2k})$, *for some constant* $c > 1$.

Proof. By Observation 4.5, there can be at most one path in $\mathcal{M}$ that contains only internal colors w.r.t. w in $G - w$. Therefore, it suffices to upper bound the number of paths in $\mathcal{M}$ that contain at least one external color to w in $G - w$. Without loss of generality, in the rest of the proof, we shall assume that $\mathcal{M}$ does not include a path that contains only internal colors w.r.t. w in $G - w$, and upper bound $|\mathcal{M}|$ by $g(k)$; adding 1 to $g(k)$ we obtain an upper bound on $|\mathcal{M}|$ with this assumption lifted. Note that by Observation 4.2, the previous assumption implies that every path in $\mathcal{M}$ contains a color that is external to w in M.

The proof is by induction on k, over every color-connected plane graph G, every triplet of vertices u, v, w in G, and every minimal set w.r.t. w of k-valid pairwise color-disjoints sets of u-v paths $\mathcal{M}$ in $G - w$. If $k = 1$, then any path in $\mathcal{M}$ contains exactly one external color w.r.t. w in M. By Lemma 4.10, at most two paths in $\mathcal{M}$ contain only external colors. It follows that for $k = 1$ $|\mathcal{M}| \le 2 \le g(1)$, if we choose the hidden constant in the $\mathcal{O}$ asymptotic notation to be at least 2.

Suppose by the inductive hypothesis that for any $1 \le i < k$, we have $|\mathcal{M}| \le g(i)$. We can assume that M is irreducible; otherwise, we apply the color contraction operation to any edge xy in $\mathcal{M}$ to which the operation is applicable, and replace $\mathcal{M}$ with the set of paths $\Lambda_{xy}(\mathcal{M})$, which is pairwise color-disjoint, contains the same number of paths as $\mathcal{M}$, and is minimal w.r.t. w by Lemma 4.6.

By Lemma 4.11, there are at most 3 paths in $\mathcal{M}$, such that the subgraph of M induced by the remaining paths of $\mathcal{M}$ has a u-v vertex-separator S satisfying $|S| \le 2k + 3$. To simplify the argument, in what follows, we assume that we already removed these 3 paths from $\mathcal{M}$ and that M already has a u-v vertex-separator S satisfying $|S| \le 2k + 3$. We will add 3 to the count of $|\mathcal{M}|$ at the end to account for these removed paths. We can assume, without loss of generality, that S is minimal (w.r.t. containment). S separates M into two subgraphs M_u and M_v such that $u \in V(M_u)$, $v \in V(M_v)$, and there is no edge between M_u and M_v. We partition $\mathcal{M}$ into the following groups, where each group excludes the paths satisfying the properties of the groups defined earlier: (1) The set of paths in $\mathcal{M}$ that contain a nonempty vertex in S; (2) the set of paths $\mathcal{M}_u^k$ consisting of each path P in $\mathcal{M}$ such that all colors on P appear on vertices in M_u (these colors could still appear on vertices in M_v as well); (3) the set of paths $\mathcal{M}_v^k$ consisting of each path P in $\mathcal{M}$ such that all colors on P appear on vertices in M_v; and (4) the set $\mathcal{M}^{<k}$ of remaining paths in $\mathcal{M}$, satisfying that each path contains a nonempty external vertex to w in M and contains less than k colors from each of M_u and M_v. Note that by Observation 4.7, each path in $\mathcal{M}$ belongs to one of the 4 groups above.

Since the paths in $\mathcal{M}$ are pairwise color-disjoint, no nonempty vertex in S can appear on two distinct paths from group (1). Therefore, the number of paths in group (1) is at most $|S| \le 2k + 3$. Observe, that the vertices in S contained in any path in groups (2)-(4) are empty vertices.

To upper bound the number of paths in group (2), for each path P, there is a last vertex x_P (*i.e.*, farthest from u) in P that is in S. Fix a vertex $x \in S$, and let us upper bound the number of paths P in group (2) for which $x = x_P$. Let P_v be the subpath of P from x to v. Note that since v is empty and all the vertices in S that are contained in paths in group (2) are empty, and since M is irreducible, P_v must contain at least one color. Since all colors appearing on P appear on vertices in M_u, all colors appearing on P_v appear in M_u. By Lemma 4.12, M_u is contained in a single face f of $M_v + x$. Since f is a vertex-separator that separates $V(M_u)$ from $V(P_v)$ in G, by Observation 4.1, every color that appears on P_v appears on f. By Lemma 4.10, there are at most two x-v paths that contain only colors that appear on f. This shows that there are at most two paths in group (2) for which x is the last vertex in S. Since $|S| \le 2k + 3$, this upper bounds the number of paths in group (2) by $2(2k + 3) = 4k + 6$. By symmetry, the number of paths in group (3) is upper bounded by $4k + 6$.

Finally, we upper bound the number of paths in group (4). Let $S = \{s_2, \ldots, s_{r-1}\}$, where $r \le 2k + 5$, and extend S by adding the two vertices $s_1 = u$ and $s_r = v$ to form the set $A = \{s_1, s_2, \ldots, s_r\}$. For every two (distinct) vertices $s_j, s_{j'} \in A$, $j, j' \in [r], j < j'$, we define a set of paths $\mathcal{P}_{jj'}$ in $G - w$ whose endpoints are s_j and $s_{j'}$ as follows. For each path P in group (4), partition (the edges in) P into subpaths $P_1, \ldots, P_q$ satisfying the property that the endpoints of each P_i, $i \in [q]$, are in A, and no internal vertex to P_i is in A. Since each P is a u-v path, clearly, P can be partitioned as such. For each P_i, $i \in [q]$, such that P_i contains a vertex that contains an external color to w in $G - w$, let P'_i (possibly equal to P_i) be a subpath in $G - w$ between the endpoints of P_i satisfying that $\chi(P'_i) \subseteq \chi(P_i)$ and $\chi(P'_i)$ is minimal with respect to containment (*i.e.*, there does not exist a path P''_i in $G-w$ between the endpoints of P_i satisfying $\chi(P''_i) \subsetneq \chi(P'_i)$). Since P contains a vertex that contains an external color to w in $G - w$, there exists an $i \in [q]$ such that P'_i contains a vertex that contains an external color to w in $G-w$; otherwise, by concatenating (in the right sequence) the P_i's that do not contain an external color to w (in $G - w$) with the P'_i's, instead of P_i, for the P_i's that contain an external color to w (in $G - w$) we would obtain a u-v path P' in $G - w$ satisfying $\chi(P') \subsetneq \chi(P)$ (since $\chi(P') \subseteq \chi(P)$ and P contains an external color to w and P' does not), thus contradicting the minimality of $\mathcal{M}$. Pick any $i \in [q]$ satisfying that P'_i contains a vertex that contains an external color to w in $G - w$, associate P with P'_i, and assign P'_i to the set of paths $\mathcal{P}_{jj'}$ such that s_j and $s_{j'}$ are the endpoints of P'_i. Since each P'_i contains an external color that appears on P and the paths in $\mathcal{M}$ are pairwise-color disjoint, it follows that the map that maps each P to its P'_i is a bijection.

Therefore, to upper bound the number of paths in group (4), it suffices to upper bound the number of paths assigned to the sets $\mathcal{P}_{jj'}$, where $j, j' \in [r], j < j'$. Fix a set $\mathcal{P}_{jj'}$. The paths in $\mathcal{P}_{jj'}$ have $s_j, s_{j'}$ as endpoints, and are pairwise color-disjoint. Moreover, each path in $\mathcal{P}_{jj'}$ contains a vertex that contains an external color to w in $G - w$. It follows from the previous statements that $\mathcal{P}_{jj'}$ satisfies properties (i) and (ii) of Definition 4.4 with respect to G and w. Moreover, from the definition of each path in $\mathcal{P}_{jj'}$, $\mathcal{P}_{jj'}$ satisfies properties (iii) of Definition 4.4 as well. Finally, observe that each path $P'_i \in \mathcal{P}_{jj'}$ was constructed based on a subpath P_i of a path P in group 4, and satisfying that P_i has endpoints $s_j, s_{j'}$ and no internal vertex on P_i is in A. Since P is a u-v path in $\mathcal{M}$ and S is a vertex-separator of M, $V(P_i)$ is either contained in $V(M_u) \cup S$ or in $V(M_v) \cup S$. Since P is in group (4), P contains at most $k-1$ colors from each of M_u and M_v. Since the vertices in S are empty, we deduce that P_i contains at most $k - 1$ colors. Since $\chi(P'_i) \subseteq \chi(P_i)$, P'_i contains at most $k - 1$ colors as well, and hence, every path in $\mathcal{P}_{jj'}$ contains at most $k - 1$ colors. It follows that $\mathcal{P}_{jj'}$ is a minimal set of $(k - 1)$-valid s_j-$s_{j'}$ paths in G with respect to w. By the inductive hypothesis, we have $|\mathcal{P}_{jj'}| \leq g(k - 1)$. Since the number of sets $\mathcal{P}_{jj'}$ is at most $\binom{2k+5}{2}$, the number of paths in group (4) is $\mathcal{O}(k^2) \cdot g(k - 1)$.

It follows from the above that $|\mathcal{M}| \leq g(k)$, where $g(k)$ satisfies the recurrence relation $g(k) \leq 3 + (2k + 3) + 2(4k + 6) + \mathcal{O}(k^2) \cdot g(k - 1) = \mathcal{O}(k^2) \cdot g(k - 1)$, where 3 acounts for the 3 paths we removed from $\mathcal{M}$ at the beginning of the proof to get a small u-v vertex separator. Solving the aforementioned recurrence relation we get $g(k) = \mathcal{O}(c^k k^{2k})$, where $c > 1$ is a constant. Adding 1 to $g(k)$ to account for the single path in $\mathcal{M}$ containing only internal colors w.r.t. w in M yields the same asymptotic upper bound. □

Theorem 4.14. *Let G be a plane color-connected graph, let $u, v, w \in V(G)$, and let $\mathcal{P}$ be a set of minimal k-valid u-v paths w.r.t. w in $G - w$. Then $|\mathcal{P}| \leq h(k)$, where $h(k) = \mathcal{O}(c^{k^2} k^{2k^2+k})$, for some constant $c > 1$.*

Proof. The proof is by induction on k. If $k = 1$, then by minimality of $\mathcal{P}$, we have $\mathcal{P} = \mathcal{M}$. Lemma 4.13 gives an upper bound of $\mathcal{O}(c^k k^{2k}) = \mathcal{O}(c^{k^2} k^{2k^2+k})$ on $|\mathcal{P}|$.

Assume by the inductive hypothesis that the statement of the lemma is true for $1 \leq i < k$. Let $\mathcal{M}$ be a maximal set of pairwise color-disjoint paths in $\mathcal{P}$. By Lemma 4.13, $|\mathcal{M}| \leq g(k) = \mathcal{O}(c^k k^{2k})$. The number of colors contained in vertices of $\mathcal{M}$ is at most $r \leq k \cdot g(k)$. We group the paths in $\mathcal{P}$ into r groups $\mathcal{P}_1, \ldots, \mathcal{P}_r$, such that all the paths in $\mathcal{P}_i$, $i \in [r]$, share the same color c_i, where $i \in [r]$, that is distinct from each color c_j shared by the paths $\mathcal{P}_j$, for $j \neq i$. We upper bound the number of paths in each $\mathcal{P}_i$, $i \in [r]$, to obtain an upper bound on $|\mathcal{P}|$.

Let G_i be the graph obtained by removing color c_i from each vertex in G that c appears on, and let $\mathcal{P}'_i$ be the set of paths obtained from $\mathcal{P}_i$ by removing color c_i from each vertex in $\mathcal{P}_i$ that c appears on. Clearly, every path in $\mathcal{P}'_i$ is a $(k-1)$-valid u-v path. Moreover, it is easy to verify that $\mathcal{P}'_i$ satisfies properties (i)-(iii) in Definition 4.4, and hence, $\mathcal{P}'_i$ is minimal w.r.t. w in $G_i - w$. By the inductive hypothesis, we have $|\mathcal{P}'_i| \leq h(k-1)$. It follows that the total number of paths in $\mathcal{P}$ is at most $h(k)$, where $h(k)$ satisfies the recurrence relation $h(k) \leq r \cdot h(k-1) \leq k \cdot g(k) \cdot h(k-1)$. Solving the aforementioned recurrence relations yields $h(k) = \mathcal{O}((k \cdot g(k))^k) = \mathcal{O}(c^{k^2} k^{2k^2+k})$. □

The result of Theorem 4.14 will be employed in the next section in the form presented in the following corollary:

Corollary 4.15. *Let G be a plane color-connected graph, and let $w \in V(G)$. Let G' be a subgraph of $G - w$, and let $u, v \in V(G')$. Every set $\mathcal{P}$ of minimal k-valid u-v paths in G' w.r.t. w satisfies $|\mathcal{P}| \leq h(k)$, where $h(k) = \mathcal{O}(c^{k^2} k^{2k^2+k})$, for some constant $c > 1$.*

Proof. Contract every connected component of $(G-w)-G'$ into a single vertex containing the union of the color-sets of the vertices in the component, and add $k+1$ new distinct colors to the resulting vertex. Denote the resulting graph by G''. Observe that the resulting graph is color-connected, and that every k-valid u-v path in G' w.r.t. w is a k-valid u-v path in G'' w.r.t. w, and vice versa. Therefore, every set $\mathcal{P}$ of minimal k-valid u-v paths in G' w.r.t. w is also a set of minimal k-valid u-v paths in G'' w.r.t. w. For any set $\mathcal{P}$ of minimal k-valid u-v paths w.r.t. w in G', by applying Theorem 4.14 to $\mathcal{P}$ in $G'' - w$, the corollary follows. □

5 The Algorithm

In this section, we present an FPT algorithm for Connected Obstacle Removal, parameterized by both k and the treewidth of the input graph. The main obstacle that faces a standard dynamic programming approach based on tree decomposition is that there can be too many (*i.e.*, more than FPT-many) subsets of colors that appear in a bag, and hence, that the algorithm may need to store/remember. To overcome this obstacle, we show how to extend the notion of a minimal set of k-valid u-v paths w.r.t. a vertex w—from the previous section—to a "representative set" of paths w.r.t. a specific bag and a specific enumerated configuration for the bag. This allows us to upper bound the size of the table, in the dynamic programming algorithm, stored at a bag by a function of both k and the treewidth of the input graph.

Let (G, C, χ, s, t, k) be an instance of Connected Obstacle Removal. The algorithm is a dynamic programming algorithm based on a tree decomposition of G. Let $(\mathcal{V}, \mathcal{T})$ be a nice tree decomposition of G. By Assumption 2.2, we can assume that s and t are nonadjacent empty vertices. We add s and t to every bag in $\mathcal{T}$, and from now on, we assume that $\{s, t\} \subseteq X_i$, for every bag $X_i \in \mathcal{T}$. For a bag X_i, we say that $v \in X_i$ is *useful* if $|\chi(v)| \leq k$. Let U_i be the set of all useful vertices in X_i and let $\overline{U_i} = X_i \setminus U_i$. We denote by V_i the set of vertices in the bags of the subtree of $\mathcal{T}$ rooted at X_i.

Let X_i be a bag. For any two vertices $u, v \in X_i$, let $G^i_{uv} = G[(V_i \setminus X_i) \cup \{u, v\}]$. We extend the notion of a minimal set of k-valid u-v paths with respect to a vertex, developed in the previous section, to the set of vertices in a bag of $\mathcal{T}$.

Definition 5.1. *A set of k-valid u-v paths $\mathcal{P}_{uv}$ in G^i_{uv} is* minimal *w.r.t. X_i if it satisfies the following properties:*

(i) There does not exist two paths $P_1, P_2 \in \mathcal{P}_{uv}$ such that $\chi(P_1) \cap \chi(X_i) = \chi(P_2) \cap \chi(X_i)$;

(ii) there does not exist two paths $P_1, P_2 \in \mathcal{P}_{uv}$ such that $\chi(P_1) \subseteq \chi(P_2)$; and

(iii) for any $P \in \mathcal{P}_{uv}$ there does not exist a u-v path P' in G^i_{uv} such that $\chi(P') \subsetneq \chi(P)$.

The following lemma uses the upper bound on the cardinality of a minimal set of k-valid u-v paths w.r.t. a vertex, derived in Corollary 4.15 in the previous section, to obtain an upper bound on the cardinality of a minimal set of k-valid u-v paths with respect to a bag of $\mathcal{T}$:

Lemma 5.2. *Let X_i be bag, $u, v \in X_i$, and $\mathcal{P}_{uv}$ a set of k-valid u-v paths in G^i_{uv} that is minimal w.r.t. X_i. Then the number of paths in $\mathcal{P}_{uv}$ is at most $h(k)^{|X_i|}$, where $h(k) = \mathcal{O}(c^{k^2} k^{2k^2+k})$, for some constant $c > 1$.*

Proof. Let $X_i \setminus \{u, v\} = \{w_1, \ldots, w_r\}$, where $r = |X_i| - 2$. For each $w_j \in X_i$, $j \in [r]$, let $\mathcal{P}_j$ be a minimal set of k-valid u-v paths w.r.t. w_j in G^i_{uv}. Without loss of generality, we can pick $\mathcal{P}_j$ such that there is no k-valid u-v path P in G^i_{uv} such that $\mathcal{P}_j \cup \{P\}$ is minimal. From Corollary 4.15, we have $|\mathcal{P}_j| \leq h(k) = \mathcal{O}(c^{k^2} k^{2k^2+k})$, for some constant $c > 1$. For each $P \in \mathcal{P}_{uv}$, and each $j \in [r]$, define $C_j = \chi(P) \cap \chi(w_j)$. Define the *signature* of P (w.r.t. the colors of $w_1, \ldots, w_r$) to be the tuple $(C_1, \ldots, C_r)$. Observe that no two (distinct) paths $P_1, P_2 \in \mathcal{P}_{uv}$ have the same signature; otherwise, since u and v appear on both P_1, P_2, $\chi(P_1) \cap \chi(X_i) = \chi(P_2) \cap \chi(X_i)$, which contradicts condition (i) of the minimality of $\mathcal{P}_{uv}$. For each $P \in \mathcal{P}_{uv}$, and each $j \in [r]$, there is a path $P' \in \mathcal{P}_j$ such that $\chi(P') \cap \chi(w_j) = C_j$. If this were not true, then P would have been added to $\mathcal{P}_j$ for the following reasons. Clearly, P does not contradict conditions (i) and (iii) of the minimality of $\mathcal{P}_j$. It cannot contradict (ii) either, because otherwise, and since P does not contradict (i), there would be a path $P'' \in \mathcal{P}_j$ such that $\chi(P'') \subsetneq \chi(P)$, contradicting the minimality of $\mathcal{P}_{uv}$. It follows that the number of signatures of paths in $\mathcal{P}_{uv}$ is at most $\prod_{j=1}^{r} |\mathcal{P}_j| \leq h(k)^{|X_i|}$. Since no two distinct paths in $\mathcal{P}_{uv}$ have the same signature, it follows that $|\mathcal{P}_{uv}| \leq h(k)^{|X_i|}$. □

Definition 5.3. Let X_i be a bag in $\mathcal{T}$. A *pattern* π for X_i is a sequence $(v_1 = s, \sigma_1, v_2, \sigma_2, \ldots, \sigma_{r-1}, v_r = t)$, where $\sigma_i \in \{0, 1\}$ and $v_i \in U_i$. For a bag X_i, and a pattern $(v_1 = s, \sigma_1, v_2, \sigma_2, \ldots, \sigma_{r-1}, v_r = t)$ for X_i, we say that a sequence of paths $\mathcal{S} = (P_1, \ldots, P_{r-1})$ *conforms* to (X_i, π) if:

- for each $j \in [r-1]$, $\sigma_j = 1$ implies that P_j is an induced path from v_j to v_{j+1} whose internal vertices are contained in $V_i \setminus X_i$ and P_j is empty otherwise; and
- $|\chi(\mathcal{S})| = |\bigcup_{j \in [r-1]} \chi(P_j)| \leq k$.

Definition 5.4. Let X_i be a bag, π a pattern for X_i, and $\mathcal{S}_1, \mathcal{S}_2$ two sequences of paths that conform to (X_i, π). We write $\mathcal{S}_1 \preceq_i \mathcal{S}_2$ if $|\chi(\mathcal{S}_1) \cup (\chi(\mathcal{S}_2) \cap \chi(X_i))| \leq |\chi(\mathcal{S}_2)|$.

We note that at a certain point during the dynamic programming algorithm, we will have to deal for a short while with sequences of walks instead of sequences of paths (until we refine them), but the definition of a sequence of paths conforming to a bag and a pattern, and the relation $\preceq$, extend seamlessly to sequences of walks.

Lemma 5.5. *Let X_i be a bag and π a pattern for X_i. The relation $\preceq_i$ is a transitive relation on the set of all sequences of paths that conform to (X_i, π).*

Proof. Let $\mathcal{S}_1, \mathcal{S}_2, \mathcal{S}_3$ be three sequences that conform to (X_i, π). Suppose that $\mathcal{S}_1 \preceq_i \mathcal{S}_2$ and $\mathcal{S}_2 \preceq_i \mathcal{S}_3$. We need to show that $\mathcal{S}_1 \preceq_i \mathcal{S}_3$. To simplify the notation in the proof, let $A = \chi(S_1), B = \chi(S_2), C = \chi(S_3), X = X_i$. Since $\mathcal{S}_1 \preceq_i \mathcal{S}_2$, we have

$$\begin{aligned} |A \cup B \cap X| &\leq |B| \\ |A| + |B \cap X| - |A \cap B \cap X| &\leq |B|, \end{aligned} \tag{1}$$

and since $\mathcal{S}_2 \preceq_i \mathcal{S}_3$ we have:

$$\begin{aligned} |B \cup C \cap X| &\leq |C| \\ |B| + |C \cap X| - |B \cap C \cap X| &\leq |C|. \end{aligned} \tag{2}$$

From Inequalities (1) and (2) we get:

$$\begin{aligned} |A| + |C \cap X| + |B \cap X| - |A \cap B \cap X| - |B \cap C \cap X| &\leq |C| \\ |A| + |C \cap X| + |B \cap X| - (|A \cap B \cap X| + |B \cap C \cap X| - |A \cap B \cap C \cap X| + |A \cap B \cap C \cap X|) &\leq |C| \\ |A| + |C \cap X| + |B \cap X| - (|A \cap B \cap X \cup B \cap C \cap X| + |A \cap B \cap C \cap X|) &\leq |C| \\ |A| + |C \cap X| + |B \cap X| - (|(A \cup C) \cap (B \cap X)| + |A \cap B \cap C \cap X|) &\leq |C| \\ |A| + |C \cap X| + |B \cap X| - (|B \cap X| + |A \cap B \cap C \cap X|) &\leq |C| \\ |A| + |C \cap X| - |A \cap B \cap C \cap X|) &\leq |C| \\ |A| + |C \cap X| - |A \cap C \cap X|) &\leq |C|. \end{aligned}$$

The last inequality proves that $\mathcal{S}_1 \preceq_i \mathcal{S}_3$.

□

Using the relation $\preceq_i$ on the set of sequences that conform to (X_i, π), we are now ready to define the key notion that makes the dynamic programming approach work:

Definition 5.6. Let X_i be a bag and $\pi = (v_1, \sigma_1, v_2 \ldots, \sigma_{r-1}, v_r)$ a pattern for X_i. A set $\mathcal{R}_\pi$ of sequences that conform to (X_i, π) is a *representative* set for (X_i, π) if:

(i) For every sequence $\mathcal{S}_1 \in \mathcal{R}_\pi$, and for every sequence $\mathcal{S}_2 \neq \mathcal{S}_1$ that conforms to (X_i, π), if $\mathcal{S}_1 \preceq_i \mathcal{S}_2$ then $\mathcal{S}_2 \notin \mathcal{R}_\pi$;

(ii) for every sequence $\mathcal{S} \in \mathcal{R}_\pi$, and for every path $P \in \mathcal{S}$ between v_j and v_{j+1}, $j \in [r-1]$, there does not exist a v_j-v_{j+1} path P' in $G^i_{v_j v_{j+1}}$ such that $\chi(P') \subsetneq \chi(P)$; and

(iii) for every sequence $\mathcal{S} \notin \mathcal{R}_\pi$ that conforms to (X_i, π) and satisfies that no two paths in $\mathcal{S}$ share a vertex that is not in X_i, there is a sequence $\mathcal{W} \in \mathcal{R}_\pi$ such that $\mathcal{W} \preceq_i \mathcal{S}$.

Observation 5.7. *Let X_i and X_j be two bags such that $X_i \subseteq X_j$, let π be a pattern for both X_i and X_j, and let $\mathcal{S}, \mathcal{S}'$ be two sequences that conform to both (X_i, π) and (X_j, π). If $\mathcal{S} \preceq_j \mathcal{S}'$ then $\mathcal{S} \preceq_i \mathcal{S}'$.*

Proof. Since $X_i \subseteq X_j$, we have $|\chi(\mathcal{S}) \cup \chi(\mathcal{S}') \cap \chi(X_i)| \leq |\chi(\mathcal{S}) \cup \chi(\mathcal{S}') \cap \chi(X_j)|$. □

Lemma 5.8. *Let X_i be a bag, π a pattern for X_i, and $\mathcal{S}_1, \mathcal{S}_1', \mathcal{S}_2, \mathcal{S}_2', \mathcal{S}, \mathcal{S}'$ sequences that conform to (X_i, π) and that satisfy the following: $\mathcal{S}_1' \preceq_i \mathcal{S}_1$, $\mathcal{S}_2' \preceq_i \mathcal{S}_2$, $\chi(\mathcal{S}_1) \cup \chi(\mathcal{S}_2) = \chi(\mathcal{S})$, $\chi(\mathcal{S}_1') \cup \chi(\mathcal{S}_2') = \chi(\mathcal{S}')$, and $\chi(\mathcal{S}_1) \cap \chi(\mathcal{S}_2) \subseteq \chi(X_i)$. Then $\mathcal{S}' \preceq_i \mathcal{S}$.*

Proof. Let $A = \chi(\mathcal{S}_1), B = \chi(\mathcal{S}_2), C = \chi(\mathcal{S}), A' = \chi(\mathcal{S}_1'), B' = \chi(\mathcal{S}_2'), C' = \chi(\mathcal{S}')$, and $X = \chi(X_i)$. Since $\mathcal{S}_1' \preceq_i \mathcal{S}_1$ we have:

$$\begin{aligned} |A' \cup A \cap X| &\leq |A| \\ |A'| + |A \cap X| - |A' \cap A \cap X| &\leq |A|. \end{aligned} \tag{3}$$

Since $\mathcal{S}_2' \preceq_i \mathcal{S}_2$ we have:

$$\begin{aligned} |B' \cup B \cap X| &\leq |B| \\ |B'| + |B \cap X| - |B' \cap B \cap X| &\leq |B|. \end{aligned} \tag{4}$$

Adding Inequality (3) to (4) and subtracting $|A \cap B|$ from each side of the resulting inequality, we obtain:

$$|A'| + |B'| + |A \cap X| + |B \cap X| - |A' \cap A \cap X| - |B' \cap B \cap X| - |A \cap B| \leq |A \cup B|. \tag{5}$$

Replacing in the last Inequality (5) $|A'| + |B'|$ by $|A' \cup B'| + |A' \cap B'|$, and $|A \cap X| + |B \cap X|$ with $|(A \cup B) \cap X| + |A \cap B \cap X|$, observing that $A \cap B \cap X = A \cap B$ (because $A \cap B \subseteq X$), and simplifying, we get:

$$|A' \cup B'| + |(A \cup B) \cap X| + |A' \cap B'| - |A' \cap A \cap X| - |B' \cap B \cap X| \leq |A \cup B|$$
$$|(A' \cup B') \cup (A \cup B) \cap X| + |(A' \cup B') \cap (A \cup B) \cap X| + |A' \cap B'| - |A' \cap A \cap X| - |B' \cap B \cap X| \leq |A \cup B|.$$

Replacing $-|A' \cap A \cap X| - |B' \cap B \cap X|$ in the last inequality with $-(|A' \cap A \cup B' \cap B) \cap X| + |A' \cap A \cap B' \cap B \cap X|) = -(|A' \cap A \cup B' \cap B) \cap X| + |A' \cap A \cap B' \cap B|)$ (because $A \cap B \subseteq X$), and observing that $|(A' \cap A \cup B' \cap B) \cap X| \leq |(A' \cup B') \cap (A \cup B) \cap X|$, and $|A' \cap A \cap B' \cap B| \leq |A' \cap B'|$, we conclude that:

$$|(A' \cup B') \cup (A \cup B) \cap X| \leq |A \cup B|. \tag{6}$$

Inequality (6) establishes that $\mathcal{S}' \preceq_i \mathcal{S}$. ☐

Lemma 5.9. *Let X_i be bag, π a pattern for X_i, and $\mathcal{R}_\pi$ be a representative set for (X_i, π). Then the number of sequences in $\mathcal{R}_\pi$ is at most $h(k)^{|X_i|^2}$, where $h(k) = \mathcal{O}(c^{k^2} k^{2k^2+k})$, for some constant $c > 1$.*

Proof. Let $\pi = (v_1 = s, \sigma_1, v_2, \sigma_2, \ldots, \sigma_{r-1}, v_r = t)$ and let v_j and v_{j+1} be two consecutive vertices in π such that $\sigma_j = 1$. For each $j \in [r-1]$ such that $\sigma_j = 1$, let $\mathcal{P}_j$ be a minimal set of k-valid v_j-v_{j+1} paths w.r.t. X_i. Without loss of generality, we can pick $\mathcal{P}_j$ such that there is no k-valid u-v path P in $G^i_{v_j v_{j+1}}$ such that $\mathcal{P}_j \cup \{P\}$ is minimal w.r.t. X_i. From Lemma 5.2 it follows that $|\mathcal{P}_j| \leq h(k)^{|X_i|}$, where $h(k) = \mathcal{O}(c^{k^2} k^{2k^2+k})$, for some constant $c > 1$. For a sequence $\mathcal{S} = (P_1, \ldots, P_{r-1})$ in $\mathcal{R}_\pi$ we define the *signature* of $\mathcal{S}$ (w.r.t. X_i) to be the tuple $(\chi(P_1) \cap \chi(X_i), \ldots, \chi(P_{r-1}) \cap \chi(X_i))$. Observe that if $\mathcal{S}_1$ and $\mathcal{S}_2$ have the same signature w.r.t. X_i, then $\chi(\mathcal{S}_1) \cup (\chi(\mathcal{S}_2) \cap \chi(X_i)) = \chi(\mathcal{S}_1)$ and $\chi(\mathcal{S}_2) \cup (\chi(\mathcal{S}_1) \cap \chi(X_i)) = \chi(\mathcal{S}_2)$; hence, either $\mathcal{S}_1 \preceq_i \mathcal{S}_2$ or $\mathcal{S}_2 \preceq_i \mathcal{S}_1$. It follows from property (i) of representative sets that no two sequences in $\mathcal{R}_\pi$ have the same signature w.r.t. X_i. Now let $\mathcal{S} = (P_1, \ldots, P_{r-1})$ be a sequence in $\mathcal{R}_\pi$ with a signature $(C_1, \ldots, C_{r-1})$. Note that if $C_j \neq \emptyset$, then P_j is not the empty path, and hence $\sigma_j = 1$. We show that for each $j \in [r-1]$ such that $C_j \neq \emptyset$, there is a path $P \in \mathcal{P}_j$ such that $\chi(P) \cap \chi(X_i) = C_j$. Suppose, for a contradiction, that this is not the case. Then for some $j \in [r-1]$ such that $C_j \neq \emptyset$, there is no path $P \in \mathcal{P}_j$ such that $\chi(P) \cap \chi(X_i) = C_j$. Clearly, $P_j \notin \mathcal{P}_j$, and therefore, by our choice of $\mathcal{P}_j$, the set $\mathcal{P}_j \cup \{P_j\}$ is not a minimal set w.r.t. X_i. By assumption, $\mathcal{P}_j \cup \{P_j\}$ does not contradict property (i) in the definition of minimal set of paths w.r.t. X_i. Moreover, since $\mathcal{S} \in \mathcal{R}_\pi$, it follows from property (ii) of representative sets that P_j, and hence $\mathcal{P}_j \cup \{P_j\}$, satisfies property (iii) of minimal set of paths w.r.t. X_i. Therefore, $\mathcal{P}_j \cup \{P_j\}$ has to contradict property (ii) in the definition of minimal set of paths w.r.t. X_i, and there are two paths $Q_1, Q_2 \in \mathcal{P}_j \cup \{P_j\}$ such that $\chi(Q_1) \subseteq \chi(Q_2)$. However, if $\chi(Q_1) = \chi(Q_2)$, then Q_1 and Q_2 contradict property (i) of a minimal set of paths w.r.t. X_i, and if $\chi(Q_1) \subsetneq \chi(Q_2)$, then Q_2 contradicts property (iii), and we already established that $\mathcal{P}_j \cup \{P_j\}$ satisfies properties (i) and (iii). Therefore, $\mathcal{P}_j \cup \{P_j\}$ is a set of minimal paths w.r.t. X_i, which is a contradiction. We conclude that, for each $j \in [r-1]$ such that $C_j \neq \emptyset$, there is a path $P \in \mathcal{P}_j$ such that $\chi(P) \cap \chi(X_i) = C_j$. It follows that the number of signatures of paths in $\mathcal{P}_{uv}$ is at most $\prod_{j=1}^{r-1} |\mathcal{P}_j| \leq h(k)^{|X_i|^2}$. Since no two distinct sequences in $\mathcal{R}_\pi$ have the same signature, it follows that $|\mathcal{R}_\pi| \leq h(k)^{|X_i|^2}$. □

For each bag X_i, we maintain a table Γ_i that contains, for each pattern for X_i, a representative set of sequences $\mathcal{R}_\pi$ for (X_i, π). For two vertices vertices $u, v \in X_i$ and two u-v paths P, P' in G^i_{uv}, we say that P' *refines* P if $\chi(P') \subseteq \chi(P)$. For two sequences $\mathcal{S} = (P_1, \ldots, P_{r-1})$ and $\mathcal{S}' = (P'_1, \ldots, P'_{r-1})$ that conform to (X_i, π), we say that $\mathcal{S}'$ *refines* $\mathcal{S}$ if each path P'_j refines P_j, for $j \in [r-1]$.

Lemma 5.10. *Let X_i be a bag, $\pi = (v_1 = s, \sigma_1, v_2, \sigma_2, \ldots, \sigma_{r-1}, v_r = t)$ a pattern for X_i, and $\mathcal{W} = (W_1, \ldots, W_{r-1})$ a sequence of walks, where each W_j is a walk between vertices v_j and v_{j+1} in $G^i_{v_j v_{j+1}}$ satisfying $\chi(W_j) \leq k$. Then in time $\mathcal{O}^*(2^k)$ we can compute a sequence $\mathcal{S} = (P_1, \ldots, P_{r-1})$ of induced paths, where each P_j is an induced path between vertices v_j and v_{j+1} in $G^i_{v_j v_{j+1}}$ such that $\chi(P_j) \subseteq \chi(W_j)$, for $j \in [r-1]$, and such that $\mathcal{S}$ satisfies property (ii) of representative sets.*

Proof. For each walk W_j, $j \in [r-1]$, we do the following. For each subset $C' \subseteq \chi(W)$ considered in a nondecreasing order of cardinality, we form the subgraph G' from $G^i_{v_j v_{j+1}}$ by removing every vertex x in $G^i_{v_j v_{j+1}}$ that does not satisfy $\chi(x) \subseteq C'$. We then check if there is a v_j-v_{j+1} induced path in G', and set P_j to this path if it exists. It is clear that the path P_j satisfies $\chi(P_j) \subseteq \chi(W_j)$ and that the sequence $\mathcal{S}' = (P_1, \ldots, P_{r-1})$ conforms to π w.r.t. X_i and satisfies property (ii) of representative sets. Since each W_j satisfies $\chi(W_j) \leq k$, we can enumerate all subsets of $\chi(W_j)$ in time $\mathcal{O}^*(2^k)$. Since checking if there is an induced v_j-v_{j+1} path in G' takes polynomial time, it follows that computing P_j from W_j takes $\mathcal{O}^*(2^k)$, and so does the computation of $\mathcal{S}$. □

For a bag X_i, pattern π for X_i, and a set of sequences $\mathcal{R}$ that conform to (X_i, π), we define the procedure **Refine()** that takes the set $\mathcal{R}$ and outputs a set $\mathcal{R}'$ of sequences that conform to (X_i, π), and does not violate properties (i) and (ii). For each sequence $\mathcal{S}$ in $\mathcal{R}$, we compute a sequence $\mathcal{S}'$ that refines $\mathcal{S}$ and satisfies property (ii), and replace $\mathcal{S}$ with $\mathcal{S}'$ in $\mathcal{R}$. Afterwards, we initialize $\mathcal{R}' = \emptyset$, and order the sequences in $\mathcal{R}$ arbitrarily. We iterate through the sequences in $\mathcal{R}$ in order, and add a sequence $\mathcal{S}_p$ to $\mathcal{R}'$ if there is no sequence $\mathcal{S}$ already in $\mathcal{R}'$ such that $\mathcal{S} \preceq_i \mathcal{S}_p$, and there is no sequence $\mathcal{S}_q \in \mathcal{R}$, $q > p$ (*i.e.*, $\mathcal{S}_q$ comes after $\mathcal{S}_p$ in the order), such that $\mathcal{S}_q \preceq \mathcal{S}_p$.

Lemma 5.11. *Let X_i be a bag, $\pi = (v_1 = s, \sigma_1, v_2, \sigma_2, \ldots, \sigma_{r-1}, v_r = t)$ a pattern for X_i, and $\mathcal{W} = (W_1, \ldots, W_{r-1})$ a sequence of walks, where each W_j is a walk between vertices v_j and v_{j+1} in $G^i_{v_j v_{j+1}}$ satisfying $\chi(W_j) \leq k$. The procedure* **Refine()** *on input $\mathcal{W}$ produces a set of sequences $\mathcal{R}'$ that* *conforms to (X_i, π) satisfying properties (i) and (ii), and such that for each sequence $\mathcal{S} \in \mathcal{W}$, there is a sequence $\mathcal{S}' \in \mathcal{R}'$ satisfying $\mathcal{S}' \preceq_i \mathcal{S}$. Moreover, the procedure runs in time $\mathcal{O}^*(2^k|\mathcal{W}| + |\mathcal{W}|^2)$.*

Proof. By Lemma 5.10, refining a sequence in $\mathcal{W}$ takes $\mathcal{O}^*(2^k)$ time, and hence, refining all sequences in $\mathcal{W}$ takes $\mathcal{O}^*(2^k|\mathcal{W}|)$ time. After refining $\mathcal{W}$, we initialize $\mathcal{R}'$ to the empty set, and iterate through the sequences in $\mathcal{W}$, adding a sequence $\mathcal{S}_p \in \mathcal{W}$ to $\mathcal{R}'$ if there is no sequence $\mathcal{S}$ already in $\mathcal{R}'$ such that $\mathcal{S} \preceq_i \mathcal{S}_p$; clearly this takes $\mathcal{O}^*(|\mathcal{W}|^2)$ time, and the lemma follows. □

If a bag X_i is a leaf in $\mathcal{T}$, then $X_i = V_i = \{s, t\}$, and there are only two patterns $(s, 0, t)$ and $(s, 1, t)$ for X_i. Clearly, the only sequence that conforms to $(s, 0, t)$ is the sequence $(())$ containing exactly one empty path. Moreover, there is no edge $st \in E(G)$. Therefore, there is no sequence that conforms to $(s, 1, t)$, and the following claim holds:

Claim 2. *If a bag X_i is a leaf in $\mathcal{T}$, then $\Gamma_i = \{((s, 0, t), \{(())\}), ((s, 1, t), \emptyset)\}$ contains, for each pattern for X_i, a representative set for (X_i, π).*

We describe next how to update the table stored at a bag X_i, based on the tables stored at its children in $\mathcal{T}$. We distinguish the following cases based on the type of bag X_i.

Case 1. X_i is an introduce node with child X_j. Let $X_i = X_j \cup \{v\}$.

Clearly, for every pattern π for X_i that does not contain v, we can set $\Gamma_i[\pi] = \Gamma_j[\pi]$. $\Gamma_j[\pi]$ is a representative set for (X_i, π) for the following reasons: (i) follows because every color in $\chi(X_i)\backslash\chi(X_j)$ does not appear in V_j, since X_i is a vertex separator in G separating v and V_j and colors are connected. Hence, if two sequences in $\Gamma_j[\pi]$ that conform to (X_i, π) contradict (i), they contradict (i) w.r.t. (X_j, π) as well, but $\Gamma_j[\pi]$ is a representative set for (X_j, π). For properties (ii) and (iii), it is easy to observe that v does not appear on any path between two vertices in π having internal vertices in $V_i \setminus X_i$, and hence, these properties are inherited from the child node X_j.

Now let $\pi = (v_1 = s, \sigma_1, v_2, \sigma_2, \ldots, \sigma_{r-1}, v_r = t)$ be a pattern such that $v_q = v$, $q \in [2..r-1]$, and let $\pi' = (v_1, \sigma_1, \ldots v_{q-1}, 0, v_{q+1}, \sigma_{q+1}, \ldots, \sigma_{r-1}, v_r)$. Note that since X_j is a separator between v and V_j, the only possibility for a path from v to a different vertex in X_i to have all internal vertices in $V_i \setminus X_i$ is if it is a direct edge. Therefore, if $\sigma_{q-1} = 1$ (resp. $\sigma_q = 1$) then $v_{q-1}v$ (resp. $v_q v$) is an edge in G. Otherwise, there is no sequence conforms to (X_i, π).

We obtain $\Gamma_i[\pi]$ from $\Gamma_j[\pi']$ as follows. For every $\mathcal{S}' = (P'_1, P'_2, \ldots, P'_{r-2}) \in \Gamma_j[\pi']$, we replace the empty path corresponding to 0 between v_{q-1} and v_{q+1} in π' by two paths P_{q-1}, P_q such that $P_{q-1} = ()$ (resp. $P_q = ()$) if $\sigma_{q-1} = 0$ (resp. $\sigma_q = 0$) and $P_{q-1} = (v_{q-1}, v)$ (resp. $P_{q-1} = (v, v_q)$) otherwise and we obtain $\mathcal{S} = (P'_1, \ldots, P'_{q-2}, P_{q-1}, P_q, P'_q, \ldots, P'_{r-2})$. Denote by $\mathcal{R}_\pi$ the set of all formed sequences $\mathcal{S}$. Finally, we set $\Gamma_i[\pi] = \textbf{Refine}(\mathcal{R}_\pi)$. We claim that $\Gamma_i[\pi]$ is a representative set for (X_i, π).

Claim 3. *If X_i is an introduce node with child X_j, and Γ_j contains for each pattern π' for X_j a representative set for (X_j, π'), then $\Gamma_i[\pi]$ defined above is a representative set for (X_i, π).*

Proof. It is clear that from the application of **Refine()**, $\Gamma_i[\pi]$ does not contradict properties (i)-(ii) of the definition of representative sets. Assume now that there exists a sequence $\mathcal{S} \notin \Gamma_i[\pi]$ that conforms to (X_i, π) such that $\mathcal{S}$ violates property (iii). We define the sequence $\mathcal{S}'$ that conforms to π', and is the same as $\mathcal{S}$ on all paths that π and π' share. Since no two paths in $\mathcal{S}$ share a vertex that is not in X_i (since $\mathcal{S}$ violates (iii)), and all paths in $\mathcal{S}'$ are also in $\mathcal{S}$, it follows that no two paths in $\mathcal{S}'$ share a vertex that is not in X_j. Since $\Gamma_j[\pi']$ is a representative set for (X_j, π'), it follows that there exists $\mathcal{S}'_1 \in \Gamma_j[\pi']$ such that $\mathcal{S}'_1 \preceq_j \mathcal{S}'$. Let $\mathcal{S}_1$ be the sequence obtained from $\mathcal{S}'_1$ and conforming to (X_i, π). Then $\mathcal{S}_1 \in \mathcal{R}_\pi$, and hence by Lemma 5.11, there is a sequence $\mathcal{S}_2 \in \Gamma_i[\pi]$ such that $\mathcal{S}_2 \preceq_i \mathcal{S}_1$. Since either both $\mathcal{S}_1$ and $\mathcal{S}$ contain v or none of them does, we have $\mathcal{S}_1 \preceq_i \mathcal{S}$. By transitivity of $\preceq_i$ (Lemma 5.5), it follows that $\mathcal{S}_2 \preceq_i \mathcal{S}$. This contradicts the assumption that $\mathcal{S}$ violates property (iii). ☐

Case 2. X_i is a forget node with child X_j. Let $X_i = X_j \setminus \{v\}$.

Let $\pi = (s = v_1, \sigma_1, \ldots, \sigma_{r-1}, v_r = t)$ be a pattern for the vertices in X_i. For $q \in [r-1]$, such that $\sigma_q = 1$, we define $\pi^q = (s = v'_1, \sigma'_1, \ldots, \sigma'_r, v'_{r+1} = t)$ to be the pattern obtained from π by inserting v between v_q and v_{q+1} and setting $\sigma'_q = \sigma'_{q+1} = 1$. More precisely, we set $v'_p = v_p$ and $\sigma'_p = \sigma_p$ for $1 \le p \le q$, $v'_{q+1} = v$ and $\sigma'_{q+1} = 1$, and finally $v'_p = v_{p-1}$ and $\sigma'_p = \sigma_{p-1}$ for $q+2 \le p \le r$. We define $\mathcal{R}_\pi$ as follows:

$$\mathcal{R}_\pi = \Gamma_j[\pi] \cup \{\mathcal{S} = (P_1, \ldots, P_{q-1}, P_q \circ P_{q+1}, P_{q+2}, \ldots, P_r) \mid (P_1, \ldots, P_r) \in \Gamma_j[\pi^q], q \in [r-1] \wedge \sigma_q = 1\}.$$

Finally, we set $\Gamma_i[\pi] = \mathbf{Refine}(\mathcal{R}_\pi)$ and we claim that $\Gamma_i[\pi]$ is a representative set for (X_i, π).

Claim 4. *If X_i is a forget node with child X_j, and Γ_j contains for each pattern π' for X_j a representative set for (X_j, π'), then $\Gamma_i[\pi]$ defined above is a representative set for (X_i, π).*

Proof. It is straightforward to see that $\Gamma_i[\pi]$ satisfies properties (i) and (ii) due to the way procedure **Refine()** works. Assume for a contradiction that there exists a sequence $\mathcal{S}$ that violates property (iii). We distinguish two cases.

First, suppose that no path in $\mathcal{S}$ contains the vertex v. Then this path conforms to the pattern π in X_j. Since no two paths in $\mathcal{S}$ share a vertex that is not in X_i, and since $\Gamma_j[\pi]$ is a representative set, there exists $\mathcal{S}_1 \in \Gamma_j[\pi]$ such that $\mathcal{S}_1 \preceq_j \mathcal{S}$. Then $\mathcal{S}_1 \in \mathcal{R}_\pi$, and hence by Lemma 5.11, $\Gamma_i[\pi]$ contains a sequence $\mathcal{S}_2$ such that $\mathcal{S}_2 \preceq_i \mathcal{S}_1$. Since $\mathcal{S}_1 \preceq_j \mathcal{S}$ and $X_i \subsetneq X_j$, it follows from Observation 5.7 that $\mathcal{S}_1 \preceq_i \mathcal{S}$. By transitivity of $\preceq_i$, it follows that $\mathcal{S}_2 \preceq_i \mathcal{S}$, which is a contradiction to the assumption that $\mathcal{S}$ violates property (iii).

Second, suppose that there is a path P_q in $\mathcal{S}$ that contains v on a path between v_q and v_{q+1}. We form a sequence $\mathcal{S}'$ from $\mathcal{S}$ by keeping every path $P \neq P_q$ in $\mathcal{S}$, and replacing P_q in the sequence by the two subpaths of P_q, $P'_q = (v_q, \ldots, v)$ and $P'_{q+1} = (v, \ldots, v_{q+1})$. The sequence $\mathcal{S}'$ conforms to (X_j, π^q), and since no two paths in $\mathcal{S}$ share a vertex that is not in X_i, no two paths in $\mathcal{S}'$ share a vertex that is not in X_j. Since $\Gamma_j[\pi^q]$ is a representative set for (X_j, π^q), it follows that there exists a sequence $\mathcal{S}'_1 \in \Gamma_j[\pi^q]$ such that $\mathcal{S}'_1 \preceq_j \mathcal{S}'$. Let $\mathcal{S}_1$ be the sequence conforming to (X_i, π) obtained from $\mathcal{S}'_1$ by applying the operation $\circ$ to the two paths in $\mathcal{S}'_1$ that share v. Then $\mathcal{S}_1 \in \mathcal{R}_\pi$. Therefore, by Lemma 5.11, $\Gamma_i[\pi]$ contains a sequence $\mathcal{S}_2$ such that $\mathcal{S}_2 \preceq_i \mathcal{S}_1$. Since $\mathcal{S}'_1 \preceq_j \mathcal{S}'$, $\chi(\mathcal{S}') = \chi(\mathcal{S})$, $\chi(\mathcal{S}'_1) = \chi(\mathcal{S}_1)$, and $X_i \subsetneq X_j$, it follows that $\mathcal{S}_1 \preceq_i \mathcal{S}$. By transitivity of $\preceq_i$, it follows that $\mathcal{S}_2 \preceq_i \mathcal{S}$, which is a contradiction to the assumption that $\mathcal{S}$ violates property (iii). ☐

Case 3. X_i is a join node with children X_j, $X_{j'}$.

Let $\pi = (s = v_1, \sigma_1, \ldots, \sigma_{r-1}, v_r = t)$ be a pattern for X_i. Initialize $\mathcal{R}_\pi = \emptyset$. For every two patterns $\pi_1 = (s = v_1, \tau_1, \ldots, \tau_{r-1}, v_r = t)$ and $\pi_2 = (s = v_1, \mu_1, \ldots, \mu_{r-1}, v_r = t)$ such that $\sigma_q = \tau_q + \mu_q$, and for every two sequences $\mathcal{S}_1 = (P_1^1, \ldots, P_1^{r-1}) \in \Gamma_j[\pi_1]$ and $\mathcal{S}_2 = (P_2^1, \ldots, P_2^{r-1}) \in \Gamma_{j'}[\pi_2]$, we add the sequence $\mathcal{S} = (P_1, \ldots, P_{r-1})$ to $\mathcal{R}_\pi$, where $P_q = P_1^q$ if P_2^q is the empty path, otherwise, $P_q = P_2^q$, for $q \in [r-1]$. We set $\Gamma_i[\pi] = \mathbf{Refine}(\mathcal{R}_\pi)$, and we claim that $\Gamma_i[\pi]$ is a representative set for (X_i, π).

Claim 5. *If X_i is a join node with children X_j, $X_{j'}$, and Γ_j (resp. $\Gamma_{j'}$) contains for each pattern π' for $X_j = X_{j'} = X_i$ a representative set for (X_j, π') (resp.$(\pi', X_{j'})$), then $\Gamma_i[\pi]$ defined above is a representative set for (X_i, π).*

Proof. Clearly $\Gamma_i[\pi]$ satisfies properties (i) and (ii) due to the application of the procedure **Refine()**. To argue that $\Gamma_i[\pi]$ satisfies properties (iii), suppose not, and let $\mathcal{S} = (P_1, \ldots, P_{r-1})$ be a sequence that violates property (iii). Notice that every path P_q, $q \in [r-1]$ is either an edge between two vertices in X_i, or is a path between two vertices in X_i such that its internal vertices are either all in $V_j \setminus X_i$ or in $V_{j'} \setminus X_i$; this is true because X_i is a vertex separator separating $V_j \setminus X_i$ from $V_{j'} \setminus X_i$ in G. Define the two sequences $\mathcal{S}_1 = (P_1^1, \ldots, P_1^{r-1})$ and $\mathcal{S}_2 = (P_2^1, \ldots, P_2^{r-1})$ as follows. For $q \in [r-1]$, if P_q is empty then set both P_1^q and P_2^q to the empty path; if P_q is an edge then set $P_1^q = P_q$ and P_2^q to the empty path. Otherwise, P_q is either a path in $G[V_j]$ or in $G[V_{j'}]$; in the former case set $P_1^q = P_q$ and P_2^q to the empty path, and in the latter case set $P_2^q = P_q$ and P_1^q to the empty path. Since no two paths in $\mathcal{S}$ share a vertex that is not in X_i, and $X_i = X_j = X_{j'}$, no two paths in $\mathcal{S}_1$ (resp. $\mathcal{S}_2$) share a vertex that is not in X_j (resp. $X_{j'}$). Let $\pi_1 = (s = v_1, \tau_1, \ldots, \tau_{r-1}, v_r = t)$ and $\pi_2 = (s = v_1, \mu_1, \ldots, \mu_{r-1}, v_r = t)$ be the two patterns that $\mathcal{S}_1$ and $\mathcal{S}_2$ conform to, respectively, and observe that, for every $q \in [r-1]$, we have $\sigma_q = \tau_q + \mu_q$. Since $\Gamma_j[\pi_1]$ and $\Gamma_{j'}[\pi_2]$ are representative sets, it follows that there exist $\mathcal{S}_1' = (Y_1', \ldots, Y_{r-1}')$ in $\Gamma_j[\pi_1]$ and $\mathcal{S}_2' = (Z_1', \ldots, Z_{r-1}')$ in $\Gamma_{j'}[\pi_2]$ such that $\mathcal{S}_1' \preceq_j \mathcal{S}_1$ and $\mathcal{S}_2' \preceq_{j'} \mathcal{S}_2$. Let $\mathcal{S}' = (P_1', \ldots, P_{r-1}')$, where $P_q' = Y_q'$ if Z_q' is the empty path, otherwise, $P_q = Z_q'$, for $q \in [r-1]$. The sequence $\mathcal{S}'$ conforms to π and is in $\mathcal{R}_\pi$. By Lemma 5.11, $\Gamma_i[\pi]$ contains a sequence $\mathcal{S}''$ such that $\mathcal{S}'' \preceq_i \mathcal{S}'$. From Observation 5.7, since $X_i = X_j = X_{j'}$, from $\mathcal{S}_1' \preceq_j \mathcal{S}_1$ and $\mathcal{S}_2' \preceq_{j'} \mathcal{S}_2$ it follows that $\mathcal{S}_1' \preceq_i \mathcal{S}_1$ and $\mathcal{S}_2' \preceq_i \mathcal{S}_2$. Since $\chi(\mathcal{S}_1) \cup \chi(\mathcal{S}_2) = \chi(\mathcal{S})$ and $\chi(\mathcal{S}_1') \cup \chi(\mathcal{S}_2') = \chi(\mathcal{S}')$, and since $\chi(\mathcal{S}_1) \cap \chi(\mathcal{S}_2) \subseteq \chi(X_i)$, by Lemma 5.8, it follows that $\mathcal{S}' \preceq_i \mathcal{S}$. Since $\mathcal{S}'' \preceq_i \mathcal{S}'$, by transitivity of $\preceq_i$, it follows that $\mathcal{S}'' \preceq_i \mathcal{S}$, which concludes the proof. ☐

We can now conclude with the following theorem:

Theorem 5.12. *There is an algorithm that on input (G, C, χ, s, t, k) of* Connected Obstacle Removal, *either outputs a k-valid s-t path in G or decides that no such path exists, in time $\mathcal{O}^\star(f(k)^{6\omega^2})$, where ω is the treewidth of G and $f(k) = \mathcal{O}(c^{k^2} k^{2k^2+k})$, for some constant $c > 1$. Therefore,* Connected Obstacle Removal *parameterized by both k and the treewidth of the input graph is in* FPT.

Proof. First, in time $\mathcal{O}(|V(G)|^4)$, we can compute a branch decomposition of G, and hence a tree decomposition, of width at most $3\omega/2$, where ω is the treewidth of G [16, 22, 23]. From this tree decomposition, in polynomial time we can compute a nice tree decomposition $(\mathcal{V}, \mathcal{T})$ of G whose width is at most $3\omega/2$ and satisfying $|\mathcal{V}| = \mathcal{O}(|V(G)|)$ [19]. The algorithm starts by removing the colors of s and t from G, and decrements k by $|\chi(s) \cup \chi(t)|$ (see Assumption 2.2). Afterwards, if $k < 0$, the algorithm concludes that there is no k-valid s-t path in G. If $st \in E(G)$ and $k \geq 0$, the algorithm outputs the path (s, t). Now we know that s and t are not adjacent, and that $\chi(s) = \chi(t) = \emptyset$. The algorithm then adds s and t to every bag in $\mathcal{T}$, and executes the dynamic programming algorithm based on $(\mathcal{V}, \mathcal{T})$, described in this section, to compute a table Γ_i that contains, for each bag X_i in $\mathcal{T}$ and each pattern π for X_i, a representative set $\mathcal{R}_\pi$ for (X_i, π).

From Claims 2, 3, 4, 5, it follows, by induction on the height of the tree-decomposition $(\mathcal{V}, \mathcal{T})$ (the base case corresponds to the leaves), that the root node X_r contains a representative set $\Gamma_r[\pi]$ for the sequence $\pi = (s, 1, t)$. If $\Gamma_r[\pi]$ is empty, the algorithm concludes that there is no k-valid s-t path in G. Otherwise, noting that there is only one sequence $\mathcal{S}$ in the representative set $\Gamma_r[\pi]$ since $X_r = \{s, t\}$ and s and t are empty, the algorithm outputs the k-valid s-t path P formed by $\mathcal{S}$. The correctness follows from the following argument, which shows that if there is a k-valid s-t path in G, then the algorithm outputs such a path. Suppose that P' is a k-valid induced s-t path such that there does not exist an s-t path P'' in G satisfying $\chi(P'') \subsetneq \chi(P')$, and let $\mathcal{S}' = (P')$. Since $G^r_{st} = G$, it follows that $\mathcal{S}'$ conforms to (X_r, π). Since $\mathcal{S}'$ contains exactly one path that is induced, no two paths in $\mathcal{S}'$ share a vertex. Therefore, by property (iii) of representative sets, there exists a sequence $\mathcal{S}$ in $\Gamma_r[\pi]$ satisfying $\mathcal{S} \preceq_r \mathcal{S}'$. Noting that a sequence in $\Gamma_r[\pi]$ must consist of a single k-valid s-t path, it follows that the algorithm correctly outputs such a path.

Next, we analyze the running time of the algorithm. We observe that among the three types of bags in $\mathcal{T}$, the worst running time is for a join bag. Therefore, it suffices to upper bound the running time for a join bag, and since $|\mathcal{V}| = \mathcal{O}(n)$, the upper bound on the overall running time would follow.

Consider a join bag X_i with children $X_j, X_{j'}$. Let ω' be the width of $\mathcal{T}$ plus 1, which serves as an upper bound on the bag size in $\mathcal{T}$, and note that $\omega' \leq 3\omega/2 + 3$, where the (additional) plus 2 is to account for the vertices s and t that were added to each bag. The algorithm starts by enumerating each pattern π for X_i. The number of such patterns is at most $2^{\omega'} \cdot \omega' \cdot \omega'! = \mathcal{O}^*(2^{\omega'} \cdot \omega'!)$, where $\omega' \cdot \omega'!$ is an upper bound on the number of ordered selections of a subset of vertices from the bag, and $2^{\omega'}$ is an upper bound on the number of combinations for the σ_i's in the selected pattern. Fix a pattern π for X_i. To compute $\Gamma_i[\pi]$, the algorithm enumerates all ways of partitioning π into pairs of patterns π_1, π_2 for the children bags; there are $2^{\omega'}$ ways of partitioning π into such pairs, because for each $\sigma_i = 1$ in π, the path between v_i and v_{i+1} is either reflected in π_1 or in π_2. For a fixed pair π_1, π_2, the algorithm iterates through all pairs of sequences in the two tables $\Gamma_j[\pi_1]$ and $\Gamma_{j'}[\pi_2]$. Since each table contains a representative set, by Lemma 5.9, the size of each table is $\mathcal{O}(h_1(k)^{\omega'^2})$, where $h_1(k) = \mathcal{O}(c_1^{k^2} k^{2k^2+k})$, for some constant $c_1 > 1$, and hence iterating over all pairs of sequences in the two tables can be done in $\mathcal{O}(h_1(k)^{2\omega'^2})$ time. From the above, it follows that the set $\mathcal{R}_\pi$ can be computed in time $2^{\omega'} \cdot \mathcal{O}(h_1(k)^{2\omega'^2}) = \mathcal{O}(h_2(k)^{2\omega'^2})$, where $h_2(k) = \mathcal{O}(c_2^{k^2} k^{2k^2+k})$, for some constant $c_2 > 1$, which is also an upper bound on the size of $\mathcal{R}_\pi$. By Lemma 5.11, applying **Refine()** to $\mathcal{R}_\pi$ takes time $\mathcal{O}^*(2^k h_2(k)^{2\omega'^2} + h_2(k)^{4\omega'^2}) = \mathcal{O}^*(h_3(k)^{4\omega'^2})$, where $h_3(k) = \mathcal{O}(c_3^{k^2} k^{2k^2+k})$, for some constant $c_3 > 1$. It follows from all the above that the running time taken by the algorithm to compute Γ_i is $\mathcal{O}^*(h_3(k)^{4\omega'^2} \cdot 2^{\omega'} \cdot \omega'!) = \mathcal{O}^*(h_4(k)^{4\omega'^2})$, where $h_4(k) = \mathcal{O}(c_4^{k^2} k^{2k^2+k})$, for some constant $c_4 > 1$, and hence the running time of the algorithm is $\mathcal{O}^\star(f(k)^{6\omega^2})$, where $f(k) = \mathcal{O}(c^{k^2} k^{2k^2+k})$, for some constant $c > 1$. ☐

6 Extensions and Applications

In this section, we extend the FPT results for Connected Obstacle Removal w.r.t. the combined parameters k and ω—the treewidth of the input graph, to show that the Connected Obstacle Removal problem parameterized by both k and the length ℓ of the sought path is FPT. We also show some applications of these FPT results. We formally define the problem Bounded-length Connected Obstacle Removal:

Bounded-length Connected Obstacle Removal
Given: A planar graph G; a set of colors C; $\chi : V \longrightarrow 2^C$; and two designated vertices $s, t \in V(G)$
Parameter: k, ℓ
Question: Does there exist a k-valid s-t-path of length at most ℓ in G?

We start with the following lemma that enables us to upper bound the treewidth of the input graph by a function of the parameter ℓ:

Lemma 6.1. *Let $(G, C, \chi, s, t, k, \ell)$ be an instance of* Bounded-length Connected Obstacle Removal, *and let v be a vertex in G such that $d_G(s, v) > \ell + 1$. Let G' be the graph obtained* *from G by contracting any edge uv that is incident to v, and let $\chi'(x) = \chi(u) \cup \chi(v)$, where x is the new vertex resulting from contracting uv, and $\chi'(w) = \chi(w)$ for any $w \in V(G) \setminus \{u, v\}$. Then $(G', C, \chi', s, t, k, \ell)$ is a yes-instance of* BOUNDED-LENGTH CONNECTED OBSTACLE REMOVAL*if and only if $(G, C, \chi, s, t, k, \ell)$ is.*

Proof. Since G is color-connected and $\chi(x) = \chi(u) \cup \chi(v)$, it is easy to see that G' is color-connected as well. Because $d_G(s, v) > \ell + 1$, any solution to $(G, C, \chi, s, t, k, \ell)$ does not contain any of u, v, and hence, is a solution to $(G', C, \chi', s, t, k, \ell)$. Conversely, because $d_G(s, v) \geq \ell + 1$ any solution to $(G', C, \chi', s, t, k, \ell)$ does not contain x, and hence is a solution to $(G, C, \chi, s, t, k, \ell)$. □

By Lemma 6.1, we may assume, without loss of generality, that in an instance $(G, C, \chi, s, t, k, \ell)$ of BOUNDED-LENGTH CONNECTED OBSTACLE REMOVAL, every vertex $v \in V(G)$ satisfies $d_G(s, v) \leq \ell + 1$. Therefore, we may assume that G has radius at most $\ell + 1$, and hence G has treewidth at most $3 \cdot (\ell + 1) + 1 = 3\ell + 4$ [22].

At this point we draw the following observation. Although the treewidth of G is bounded by a function of ℓ, we cannot use the FPT algorithm for OBSTACLE REMOVAL, parameterized by k and the treewidth of G, to solve BOUNDED-LENGTH CONNECTED OBSTACLE REMOVAL because the k-valid path returned by the algorithm for OBSTACLE REMOVAL may have length exceeding the desired upper bound ℓ. In fact, extending the FPT results for OBSTACLE REMOVAL to BOUNDED-LENGTH CONNECTED OBSTACLE REMOVAL turns out to be a nontrivial task, that necessitates a nontrivial extension of the structural results in Section 4, as well as the dynamic programming algorithm in Section 5. In particular, the color contraction operation, on which the structural results developed in Section 4 hinge, is no longer applicable since contracting an edge may decrease the distance between s and t in the resulting instance, and hence, may not result in an equivalent instance of the problem. However, we will show in the next section that we can extend the notion of a minimal set of k-valid paths between two vertices to incorporate the length of these paths, while still being able to upper bound the size of such a set by a function of both k and the length of these paths.

6.1 Extended Structural Results

We start with the following definition:

Definition 6.2. Let $u, v, w \in V(G)$, and let $\lambda \in [\ell]$. Let $\mathcal{P}$ be a set of k-valid u-v paths in $G - w$, each of length λ. The set $\mathcal{P}$ is said to be λ-*minimal* with respect to w if there does not exist two paths $P_1, P_2 \in \mathcal{P}$ such that $\chi(P_1) \cap \chi(w) = \chi(P_2) \cap \chi(w)$.

Let $u, v, w \in V(G)$, $\lambda \in [\ell]$, and let $\mathcal{P}$ be a set of λ-minimal k-valid u-v paths in $G - w$. Let $\mathcal{M}$ be a set of λ-minimal k-valid color-disjoint u-v paths in $G - w$. Let H be the subgraph of $G - w$ induced by the edges of the paths in $\mathcal{P}$, and let M be that induced by the edges of the paths in $\mathcal{M}$.

Lemma 6.3. *M has a u-v vertex-separator of cardinality at most $2(\lambda + 1)$.*

Proof. We proceed by contradiction, and assume that M does not have a u-v vertex-separator of cardinality at most $2(\lambda + 1)$. By Menger's theorem [5], there exists a set $\mathcal{D} = \{P_1, \ldots, P_r\}$, where $r \geq 2\lambda+3$, of vertex-disjoint u-v paths in M. Let $u_1, \ldots, u_r$ be the neighbors of u in counterclockwise order such that P_i contains u_i, $i \in [r]$, and let Q_i be a path in $\mathcal{M}$ containing u_i.

Since all paths in $\mathcal{M}$ have the same length λ, Definition 6.2 implies that Observation 4.5 holds. Therefore, at most one path in $\mathcal{M}$ contains only internal colors with respect to w in M. By Observation 4.7, any vertex on a path in $\mathcal{M}$ such that the vertex contains an external color w.r.t. w

in M must be incident to the external face to w in M. Choose $r' \in [r]$ such that $|r' - \lfloor \frac{r}{2} \rfloor|$ is minimum and $Q_{r'}$ contains an external color w.r.t. w in M. Since $Q_{r'}$ contains an external color w.r.t. w in M, $Q_{r'}$ contains a vertex incident to the external face to w in M. Since all paths in $\mathcal{D}$ are u-v vertex-disjoint paths, $Q_{r'}$ contains vertices other than u and v from at least $\lfloor r/2 \rfloor - 1$ distinct paths (including itself) in $\mathcal{D}$. Since the paths in $\mathcal{D}$ are all vertex disjoint, it follows that $|Q_{r'}| \geq r/2 - 1$, and hence $\lambda \geq r/2 - 1$, which implies that $r \leq 2(\lambda + 1)$. This contradicts our assumption that $r \geq 2\lambda + 3$. □

Lemma 6.4. *$|\mathcal{M}| \leq g(\lambda)$, where $g(\lambda) = \mathcal{O}(c^{\lambda}\lambda^{3\lambda})$, for some constant $c > 1$.*

Proof. As in the proof of Lemma 6.3, Definition 6.2 implies that Observation 4.5 holds, and hence, at most one path in $\mathcal{M}$ contains only internal colors w.r.t. w in $G - w$. Therefore, we upper bound the number of paths in $\mathcal{M}$ that each contains at least one external color to w in $G - w$, and add 1 to $g(\lambda)$ at the end. Henceforth, we shall assume that every path in $\mathcal{M}$ contains a color that is external to w in M.

The proof is by induction on λ, over every color-connected plane graph G, every triplet of vertices u, v, w in G, and every λ-minimal set $\mathcal{M}$ w.r.t. w in $G - w$ of k-valid pairwise color-disjoint u-v paths. If $\lambda = 1$, then $|\mathcal{M}| \leq 1 \leq g(1)$, if we choose $g(1)$ to be at least 1.

Suppose, by the inductive hypothesis, that for any $1 \leq i < \lambda$, we have $|\mathcal{M}| \leq g(i)$. By Lemma 6.3, M has a u-v vertex-separator S satisfying $|S| \leq 2\lambda + 2$. S separates M into two subgraphs M_u and M_v such that $u \in V(M_u)$, $v \in V(M_v)$, and there is no edge between M_u and M_v. We partition $\mathcal{M}$ into two groups: (1) The set of paths in $\mathcal{M}$ that each contains a nonempty vertex in S; and (2) the set of remaining paths $\mathcal{M}_{\emptyset}$, which contains each path in $\mathcal{M}$ whose intersection with S consists of only empty vertices. Since the paths in $\mathcal{M}$ are pairwise color-disjoint, no nonempty vertex in S can appear on two distinct paths from group (1). Therefore, the number of paths in group (1) is at most $|S| \leq 2\lambda + 2$.

To upper bound the number of paths in group (2), suppose that $S = \{s_2, \ldots, s_{r-1}\}$, where $r \leq 2\lambda + 4$, and extend S by adding the two vertices $s_1 = u$ and $s_r = v$ to form the set $A = \{s_1, s_2, \ldots, s_r\}$. For every two (distinct) vertices $s_j, s_{j'} \in A$, $j, j' \in [r], j < j'$, we define a set of paths $\mathcal{P}_{jj'}$ in $G - w$ whose endpoints are s_j and $s_{j'}$ as follows. For each path P in group (2), partition P into subpaths $P_1, \ldots, P_q$ satisfying the property that the endpoints of each P_i, $i \in [q]$, are in A, and no internal vertex to P_i is in A. Since P contains a vertex that contains an external color to w in $G - w$, there exists an $i \in [q]$ such that P_i contains a vertex that contains an external color to w in $G - w$; pick any such $i \in [q]$, and assign P_i to the set of paths $\mathcal{P}_{jj'}$ such that s_j and $s_{j'}$ are the endpoints of P_i. Since each P_i contains an external color that appears on P and the paths in $\mathcal{M}$ are pairwise-color disjoint, it follows that the map that maps each P to its P_i is a bijection. Moreover, since each path P must intersect $S \setminus \{u, v\}$, the length of each path in $\mathcal{P}_{jj'}$, for $j, j' \in [r], j < j'$, is strictly smaller than λ.

To upper bound the number of paths in group (2), fix a set $\mathcal{P}_{jj'}$. For any fixed length $i' \in [\lambda - 1]$, the subset of paths in $\mathcal{P}_{jj'}$ of length i', $\mathcal{P}^{i'}_{jj'}$, have $s_j, s_{j'}$ as endpoints, and are pairwise color-disjoint. Moreover, each path in $\mathcal{P}^{i'}_{jj'}$ contains a vertex that contains an external color to w in $G - w$. It follows from the previous statements that $\mathcal{P}^{i'}_{jj'}$ satisfies Definition 6.2 with respect to G and w, and hence $\mathcal{P}^{i'}_{jj'}$, $i' \in [\lambda - 1]$, is an i'-minimal set of k-valid s_j-$s_{j'}$ paths in G with respect to w. By the inductive hypothesis, we have $|\mathcal{P}^{i'}_{jj'}| \leq g(i')$. Since the number of sets $\mathcal{P}_{jj'}$ is at most $\binom{2\lambda+4}{2}$, $i' \leq \lambda - 1$, and noting that g is an increasing function, the number of paths in group (2) is $\mathcal{O}(\lambda^2) \cdot (\lambda - 1) \cdot g(\lambda - 1) = \mathcal{O}(\lambda^3) \cdot g(\lambda - 1)$.

It follows from the above that $|\mathcal{M}| \leq g(\lambda)$, where $g(\lambda)$ satisfies the recurrence relation $g(\lambda) \leq (2\lambda + 2) + \mathcal{O}(\lambda^3) \cdot g(\lambda - 1) = \mathcal{O}(\lambda^3) \cdot g(\lambda - 1)$. Solving the aforementioned recurrence relation gives

$g(\lambda) = \mathcal{O}(c^\lambda \lambda^{3\lambda})$, where $c > 1$ is a constant. Adding 1 to $g(\lambda)$ to account for the single path in $\mathcal{M}$ containing only internal colors w.r.t. w in M yields the same asymptotic upper bound. □

Theorem 6.5. *Let G be a plane color-connected graph, let $u, v, w \in V(G)$, let $\lambda \in [\ell]$, and let $\mathcal{P}$ be a set of λ-minimal k-valid u-v paths w.r.t. w in $G - w$. Then $|\mathcal{P}| \leq h(k, \lambda)$, where $h(k, \lambda) = \mathcal{O}(c^{\lambda k} \cdot k^k \cdot \lambda^{3\lambda k})$, for some constant $c > 1$.*

Proof. The proof is by induction on k. If $k = 0$, then by minimality of $\mathcal{P}$, there can be at most one path in $\mathcal{P}$, namely the path consisting of empty vertices. If $k = 1$, then by minimality of $\mathcal{P}$, we have $\mathcal{P} = \mathcal{M}$, and by Lemma 6.4, $|\mathcal{P}| = \mathcal{O}(c^\lambda \lambda^{3\lambda}) = \mathcal{O}(c^{\lambda k} \cdot k^k \cdot \lambda^{3\lambda k})$.

Assume by the inductive hypothesis that the statement of the lemma is true for $1 \leq i < k$. Let $\mathcal{M}$ be a maximal set of pairwise color-disjoint paths in $\mathcal{P}$. By Lemma 4.13, $|\mathcal{M}| \leq g(\lambda) = \mathcal{O}(c^\lambda \lambda^{3\lambda})$. The number of colors contained in vertices of $\mathcal{M}$ is at most $r \leq k \cdot g(\lambda)$. We group the paths in $\mathcal{P}$ into r groups $\mathcal{P}_1, \ldots, \mathcal{P}_r$, such that all the paths in $\mathcal{P}_i$, $i \in [r]$, share the same color c_i, where $i \in [r]$, that is distinct from each color c_j shared by the paths $\mathcal{P}_j$, for $j \neq i$. We upper bound the number of paths in each $\mathcal{P}_i$, $i \in [r]$, to obtain an upper bound on $|\mathcal{P}|$.

Let G_i be the graph obtained by removing color c_i from each vertex in G that c appears on, and let $\mathcal{P}'_i$ be the set of paths obtained from $\mathcal{P}_i$ by removing color c_i from each vertex in $\mathcal{P}_i$ that c appears on. Clearly, every path in $\mathcal{P}'_i$ is a $(k-1)$-valid u-v path of length λ. Moreover, it is easy to verify that $\mathcal{P}'_i$ satisfies Definition 6.2, and hence, $\mathcal{P}'_i$ is λ-minimal w.r.t. w in $G_i - w$. By the inductive hypothesis, we have $|\mathcal{P}'_i| \leq h(k-1, \lambda)$. It follows that the total number of paths in $\mathcal{P}$ is at most $h(k, \lambda)$, where $h(k, \lambda)$ satisfies the recurrence relation $h(k, \lambda) \leq r \cdot h(k-1, \lambda) \leq k \cdot g(\lambda) \cdot h(k-1, \lambda)$. Solving the aforementioned recurrence relations yields $h(k, \lambda) = \mathcal{O}((k \cdot g(\lambda))^k) = \mathcal{O}(c^{\lambda k} \cdot k^k \cdot \lambda^{3\lambda k})$. □

The result of Theorem 6.5 will be employed in the next section in the form presented in the following corollary:

Corollary 6.6. *Let G be a plane color-connected graph, let $w \in V(G)$, and let $\lambda \in [\ell]$. Let G' be a subgraph of $G - w$, and let $u, v \in V(G')$. Every set $\mathcal{P}$ of λ-minimal k-valid u-v paths in G' w.r.t. w satisfies $|\mathcal{P}| \leq h(k, \lambda)$, where $h(k) = \mathcal{O}(c^{\lambda k} \cdot k^k \cdot \lambda^{3\lambda k})$, for some constant $c > 1$.*

Proof. Contract every connected component of $(G-w)-G'$ into a single vertex containing the union of the color-sets of the vertices in the component, and add $k+1$ new distinct colors to the resulting vertex. Denote the resulting graph by G''. Observe that the resulting graph is color-connected, and that every k-valid u-v path of length λ in G' w.r.t. w is a k-valid u-v path of length λ in G'' w.r.t. w, and vice versa. Therefore, every set $\mathcal{P}$ of λ-minimal k-valid u-v paths in G' w.r.t. w is also a set of λ-minimal k-valid u-v paths in G'' w.r.t. w. For any set $\mathcal{P}$ of λ-minimal k-valid u-v paths w.r.t. w in G', by applying Theorem 6.5 to $\mathcal{P}$ in $G'' - w$, the corollary follows. □

6.2 The Extended Algorithm

Let $(G, C, \chi, s, t, k, \ell)$ be an instance of Bounded-length Connected Obstacle Removal. The algorithm is a dynamic programming algorithm based on a tree decomposition of G. Let $(\mathcal{V}, \mathcal{T})$ be a nice tree decomposition of G. By Assumption 2.2, we can assume that s and t are nonadjacent empty vertices. We add s and t to every bag in $\mathcal{T}$, and from now on, we assume that $\{s, t\} \subseteq X_i$, for every bag $X_i \in \mathcal{T}$. For a bag X_i, we say that $v \in X_i$ is *useful* if $|\chi(v)| \leq k$. Let U_i be the set of all useful vertices in X_i and let $\overline{U_i} = X_i \setminus U_i$. We denote by V_i the set of vertices in the bags of the subtree of $\mathcal{T}$ rooted at X_i.

Let X_i be a bag. For any two vertices $u, v \in X_i$, let $G^i_{uv} = G[(V_i \setminus X_i) \cup \{u, v\}]$. We extend the notion of a λ-minimal set of k-valid u-v paths with respect to a vertex, developed in the previous section, to the set of vertices in a bag of $\mathcal{T}$.

Definition 6.7. *Let $\lambda \in [\ell]$. A set of k-valid u-v paths $\mathcal{P}_{uv}$ in G^i_{uv} is* λ-minimal *w.r.t. X_i if each path in $\mathcal{P}_{uv}$ has length exactly λ and there does not exist two paths $P_1, P_2 \in \mathcal{P}_{uv}$ such that $\chi(P_1) \cap \chi(X_i) = \chi(P_2) \cap \chi(X_i)$.*

Lemma 6.8. *Let X_i be bag, $u, v \in X_i$, $\lambda \in [\ell]$ and $\mathcal{P}_{uv}$ a λ-minimal set of k-valid u-v paths w.r.t. X_i in G^i_{uv}. Then the number of paths in $\mathcal{P}_{uv}$ is at most $h(k, \lambda)^{|X_i|}$, where $h(k, \lambda) = \mathcal{O}(c^{\lambda k} \cdot k^k \cdot \lambda^{3\lambda k})$, for some constant $c > 1$.*

Proof. Let $X_i \setminus \{u, v\} = \{w_1, \ldots, w_r\}$, where $r = |X_i| - 2$. For each $w_j \in X_i$, $j \in [r]$, let $\mathcal{P}_j$ be a λ-minimal set of k-valid u-v paths w.r.t. w_j in G^i_{uv}. Without loss of generality, we can pick $\mathcal{P}_j$ such that there is no k-valid u-v path P in G^i_{uv} such that $\mathcal{P}_j \cup \{P\}$ is λ-minimal. From Corollary 6.6, we have $|\mathcal{P}_j| \leq h(k, \lambda) = \mathcal{O}(c^{\lambda k} \cdot k^k \cdot \lambda^{3\lambda k})$, for some constant $c > 1$. For each $P \in \mathcal{P}_{uv}$, and each $j \in [r]$, define $C_j = \chi(P) \cap \chi(w_j)$. Define the *signature* of P (w.r.t. the colors of $w_1, \ldots, w_r$) to be the tuple $(C_1, \ldots, C_r)$. Observe that no two (distinct) paths $P_1, P_2 \in \mathcal{P}_{uv}$ have the same signature; otherwise, since u and v appear on both P_1, P_2, $\chi(P_1) \cap \chi(X_i) = \chi(P_2) \cap \chi(X_i)$, which contradicts the definition of the λ-minimality of $\mathcal{P}_{uv}$. For each $P \in \mathcal{P}_{uv}$, and each $j \in [r]$, there is a path $P' \in \mathcal{P}_j$ such that $\chi(P') \cap \chi(w_j) = C_j$. Otherwise, $\mathcal{P}_j \cup \{P\}$ satisfy Definition 6.2, which contradicts our assumption that there is no k-valid u-v path P in G^i_{uv} such that $\mathcal{P}_j \cup \{P\}$ is λ-minimal. It follows that the number of signatures of paths in $\mathcal{P}_{uv}$ is at most $\prod_{j=1}^{r} |\mathcal{P}_j| \leq h(k)^{|X_i|}$. Since no two distinct paths in $\mathcal{P}_{uv}$ have the same signature, it follows that $|\mathcal{P}_{uv}| \leq h(k)^{|X_i|}$. $\square$

We define the length of a sequence of paths (walks) $\mathcal{S}$, denoted by $|\mathcal{S}|$, to be the sum of the lengths of the paths in $\mathcal{S}$.

Definition 6.9. Let X_i be a bag and $\pi = (v_1, \sigma_1, v_2 \ldots, \sigma_{r-1}, v_r)$ a pattern for X_i. A set $\mathcal{R}_\pi$ of sequences of length at most ℓ that conform to (X_i, π) is a *representative* set for (X_i, π) if:

(i) For every sequence $\mathcal{S}_1 \in \mathcal{R}_\pi$, and for every sequence $\mathcal{S}_2 \neq \mathcal{S}_1$ that conforms to (X_i, π), if $\mathcal{S}_1 \preceq_i \mathcal{S}_2$ and $|\mathcal{S}_1| \leq |\mathcal{S}_2|$ then $\mathcal{S}_2 \notin \mathcal{R}_\pi$; and

(ii) for every sequence $\mathcal{S} \notin \mathcal{R}_\pi$, $|\mathcal{S}| \leq \ell$, that conforms to (X_i, π) and satisfies that no two paths in $\mathcal{S}$ share a vertex that is not in X_i, there is a sequence $\mathcal{W} \in \mathcal{R}_\pi$ such that $\mathcal{W} \preceq_i \mathcal{S}$ and $|\mathcal{W}| \leq |\mathcal{S}|$.

We mention that Lemma 5.5 and Observation 5.7 extend as they are to the current setting.

Lemma 6.10. *Let X_i be bag, π a pattern for X_i, and $\mathcal{R}_\pi$ a representative set for (X_i, π). Then the number of sequences in $\mathcal{R}_\pi$ is at most $h(k, \ell)^{|X_i|^2}$, where $h(k, \ell) = \mathcal{O}(c^{\ell k} \cdot k^k \cdot \ell^{3\ell k})$, for some constant $c > 1$.*

Proof. Let $\pi = (v_1 = s, \sigma_1, v_2, \sigma_2, \ldots, \sigma_{r-1}, v_r = t)$, and let $\Lambda = (\lambda_1, \ldots, \lambda_{r-1})$ be such that, for each $j \in [r-1]$: (1) $\lambda_j = 0$ if $\sigma_j = 0$ and $\lambda_j \in [\ell]$ otherwise, and (2) $\sum_{j=1}^{r-1} \lambda_j \leq \ell$. For each $\lambda \in [\ell]$, the number of tuples $\Lambda = (\lambda_1, \ldots, \lambda_{r-1})$ satisfying $\sum_{j=1}^{r-1} \lambda_j = \lambda$ is the number of *weak compositions* of λ into $r-1$ parts, which is $\binom{\lambda+r-2}{r-2}$. It follows that the number of tuples $\Lambda = (\lambda_1, \ldots, \lambda_{r-1})$ satisfying $\sum_{j=1}^{r-1} \lambda_j \leq \ell$ is upper bounded by $\binom{\ell+r-1}{r-2} \leq \binom{|X_i|+\ell}{r-2} \leq 2^{|X_i|+\ell}$. Therefore, if we upper bound the number of sequences in $\mathcal{R}_\pi$ corresponding to some fixed tuple Λ by $h_1(k, \ell)^{|X_i|^2} = \mathcal{O}(c_1^{\ell k} \cdot k^k \cdot \ell^{3\ell k})$ for some constant $c_1 > 1$, then we obtain $\mathcal{R}_\pi \leq 2^{|X_i|+\ell} \cdot h_1(k, \ell)^{|X_i|^2} \leq h(k, \ell)^{|X_i|^2}$, where $h(k, \ell) = \mathcal{O}(c^{\ell k} \cdot k^k \cdot \ell^{3\ell k})$, for some constant $c > 1$. Therefore, for the rest of the proof, we fix $\Lambda = (\lambda_1, \ldots, \lambda_{r-1})$ and we let $\mathcal{R}^\Lambda_\pi$ be the subset of $\mathcal{R}_\pi$ such that for each sequence $\mathcal{S} = (P_1, \ldots, P_{r-1})$ in $\mathcal{R}^\Lambda_\pi$ it holds that the length of P_j is λ_j for each $j \in [r-1]$ such that $\sigma_j = 1$.

For each $j \in [r-1]$ such that $\sigma_j = 1$, let $\mathcal{P}_j$ be a λ_j-minimal set of k-valid v_j-v_{j+1} paths w.r.t. X_i. Without loss of generality, we can pick $\mathcal{P}_j$ such that there is no k-valid u-v path P of length λ_j in $G^i_{v_j v_{j+1}}$ such that $\mathcal{P}_j \cup \{P\}$ is λ_j-minimal w.r.t. X_i. From Lemma 6.8 it follows that $|\mathcal{P}_j| \le h(k, \lambda_j)^{|X_i|}$, where $h_1(k, \lambda) = \mathcal{O}(c_1^{\lambda k} \cdot k^k \cdot \lambda^{3\lambda k})$, for some constant $c_1 > 1$.

For a sequence $\mathcal{S} = (P_1, \ldots, P_{r-1})$ in $\mathcal{R}_\pi$ we define the *signature* of $\mathcal{S}$ (w.r.t. X_i) to be the tuple $(\chi(P_1) \cap \chi(X_i), \ldots, \chi(P_{r-1}) \cap \chi(X_i))$. Observe that if $\mathcal{S}_1$ and $\mathcal{S}_2$ have the same signature w.r.t. X_i, then $\chi(\mathcal{S}_1) \cup (\chi(\mathcal{S}_2) \cap \chi(X_i)) = \chi(\mathcal{S}_1)$ and $\chi(\mathcal{S}_2) \cup (\chi(\mathcal{S}_1) \cap \chi(X_i)) = \chi(\mathcal{S}_2)$; hence, either $\mathcal{S}_1 \preceq_i \mathcal{S}_2$ or $\mathcal{S}_2 \preceq_i \mathcal{S}_1$. Since all sequences in $\mathcal{R}^\Lambda_\pi$ have the same length, it follows from property (i) of representative sets that no two sequences in $\mathcal{R}_\pi$ have the same signature w.r.t. X_i. Now let $\mathcal{S} = (P_1, \ldots, P_{r-1})$ be a sequence in $\mathcal{R}_\pi$ with a signature $(C_1, \ldots, C_{r-1})$. Note that if $C_j \neq \emptyset$, then P_j is not the empty path, and hence $\sigma_j = 1$ and the length of P_j is λ_j. For each $j \in [r-1]$ such that $C_j \neq \emptyset$, there is a path $P \in \mathcal{P}_j$ such that $\chi(P) \cap \chi(X_i) = C_j$; otherwise, since $\chi(P_j) \cap \chi(X_i) = C_j$ and the length of P_j is λ_j, $\mathcal{P}_j \cup \{P_j\}$ would also be a λ_j-minimal set of paths w.r.t. X_i, which contradicts our choice of $\mathcal{P}_j$. It follows that the number of signatures of sequences in $\mathcal{R}^\Lambda_\pi$ is at most $\prod_{j=1}^{r-1} |\mathcal{P}_j| \le \prod_{j=1}^{r-1} h(k, \lambda_j) \le h_1(k, \ell)^{|X_i|^2}$. Since no two distinct sequences in $\mathcal{R}^\Lambda_\pi$ have the same signature, it follows that $|\mathcal{R}^\Lambda_\pi| \le h_1(k, \ell)^{|X_i|^2}$ and $\mathcal{R}_\pi \le 2^{|X_i|+\ell} \cdot h_1(k, \ell)^{|X_i|^2} \le h(k, \ell)^{|X_i|^2}$. □

For each bag X_i, we maintain a table Γ_i that contains, for each pattern for X_i, a representative set of sequences $\mathcal{R}_\pi$ for (X_i, π). For two vertices $u, v \in X_i$ and two u-v paths P, P' in G^i_{uv}, we say that P' *refines* P if $\chi(P') \subseteq \chi(P)$. For two sequences $\mathcal{S} = (P_1, \ldots, P_{r-1})$ and $\mathcal{S}' = (P'_1, \ldots, P'_{r-1})$ that conform to (X_i, π), we say that $\mathcal{S}'$ *refines* $\mathcal{S}$ if each path P'_j refines P_j, for $j \in [r-1]$.

Lemma 6.11. *Let X_i be a bag, $\pi = (v_1 = s, \sigma_1, v_2, \sigma_2, \ldots, \sigma_{r-1}, v_r = t)$ a pattern for X_i, and $\mathcal{W} = (W_1, \ldots, W_{r-1})$ a sequence such that each W_j is a walk between vertices v_j and v_{j+1} in $G^i_{v_j v_{j+1}}$ satisfying $\chi(W_j) \le k$. Then in time $\mathcal{O}(r \cdot (|V(G)| + |V(E)|))$ we can compute a sequence $\mathcal{S} = (P_1, \ldots, P_{r-1})$, where for each $j \in [r-1]$, P_j is an induced path between v_j and v_{j+1} in $G^i_{v_j v_{j+1}}$ such that $\chi(P_j) \subseteq \chi(W_j)$ and the length of P_j is at most the length of W_j.*

Proof. For each walk W_j, $j \in [r-1]$, we do the following. We form the subgraph G' from $G^i_{v_j v_{j+1}}$ by removing every vertex x in $G^i_{v_j v_{j+1}}$ that does not satisfy $\chi(x) \subseteq \chi(W_j)$. Clearly, W_j is a subgraph of G', and hence there exists a v_j-v_{j+1} path of length at most the length of W_j in G'. We find a shortest v_j-v_{j+1} path in G' in time $\mathcal{O}(|V(G)| + |E(G)|)$ and set P_j to this path. Clearly, the computation of $\mathcal{S}$ takes time $\mathcal{O}(r \cdot (|V(G)| + |E(G)|))$. □

For a bag X_i, pattern π for X_i, and a set of sequences (of walks) $\mathcal{R}$ that conform to (X_i, π), we define the procedure **Refine()** that takes the set $\mathcal{R}$ and outputs a set $\mathcal{R}'$ of sequences of length at most ℓ that conform to (X_i, π), and does not violate property (i) of Definition 6.9. First, for each sequence $\mathcal{S}$ in $\mathcal{R}$, we compute a sequence $\mathcal{S}'$ that refines $\mathcal{S}$ and has length at most the length of $\mathcal{S}$, and replace $\mathcal{S}$ with $\mathcal{S}'$ in $\mathcal{R}$. Afterwards, we initialize $\mathcal{R}' = \emptyset$, and order the sequences in $\mathcal{R}$ arbitrarily. We iterate through the sequences in $\mathcal{R}$ in order, and add a sequence $\mathcal{S}_p$ to $\mathcal{R}'$ if $|\mathcal{S}_p| \le \ell$, there is no sequence $\mathcal{S}$ already in $\mathcal{R}'$ such that $\mathcal{S} \preceq_i \mathcal{S}_p$ and $|\mathcal{S}| \le |\mathcal{S}_p|$, and there is no sequence $\mathcal{S}_q \in \mathcal{R}$, $q > p$ (*i.e.*, $\mathcal{S}_q$ comes after $\mathcal{S}_p$ in the order), such that $\mathcal{S}_q \preceq \mathcal{S}_p$ and $|\mathcal{S}_q| \le |\mathcal{S}_p|$.

Lemma 6.12. *Let X_i be a bag, π a pattern for X_i, and $\mathcal{W}$ be a set of sequences of walks that conforms to (X_i, π). The procedure* **Refine()***, on input $\mathcal{W}$, produces a set of sequences of induced paths $\mathcal{R}'$ that conform to (X_i, π) and satisfy property (i) of Definition 6.9, and such that for each sequence $\mathcal{S} \in \mathcal{W}$ with $|\mathcal{S}| \le \ell$, there is a sequence $\mathcal{S}' \in \mathcal{R}'$ satisfying $\mathcal{S}' \preceq_i \mathcal{S}$ and $|\mathcal{S}'| \le |\mathcal{S}|$. Moreover, the procedure runs in time $\mathcal{O}^*(|\mathcal{W}|^2)$.*

Proof. By Lemma 6.11, refining a sequence in $\mathcal{W}$ takes $\mathcal{O}(|V(G)|+|E(G)|)$ time, and hence, refining all sequences in $\mathcal{W}$ takes $\mathcal{O}^*(|\mathcal{W}|)$ time. After refining $\mathcal{W}$, we initialize $\mathcal{R}'$ to the empty set, and iterate through the sequences in $\mathcal{W}$, adding a sequence $\mathcal{S}_p \in \mathcal{W}$ to $\mathcal{R}'$ if: $|\mathcal{S}_p| \leq \ell$, there is no sequence $\mathcal{S}$ already in $\mathcal{R}'$ such that $\mathcal{S} \preceq_i \mathcal{S}_p$, and $|\mathcal{S}| \leq |\mathcal{S}_p|$. Clearly, this takes $\mathcal{O}^*(|\mathcal{W}|^2)$ time. Moreover, for a sequence $\mathcal{S} \in \mathcal{W}$ with $|\mathcal{S}| \leq \ell$, the refined sequence $\mathcal{S}'$ we obtained from the application of Lemma 6.11 to $\mathcal{S}$ satisfies $|\mathcal{S}'| \leq \ell$. The lemma follows. □

If a bag X_i is a leaf in $\mathcal{T}$, then $X_i = V_i = \{s, t\}$, and there are only two patterns $(s, 0, t)$ and $(s, 1, t)$ for X_i. Clearly, the only sequence that conforms to $(s, 0, t)$ is the sequence $(())$ containing exactly one empty path. Moreover, there is no edge $st \in E(G)$. Therefore, there is no sequence that conforms to $(s, 1, t)$, and the following claim holds:

Claim 6. *If a bag X_i is a leaf in $\mathcal{T}$, then $\Gamma_i = \{((s,0,t), \{(())\}), ((s,1,t), \emptyset)\}$ contains, for each pattern for X_i, a representative set for (X_i, π).*

We describe next how to update the table stored at a bag X_i, based on the tables stored at its children in $\mathcal{T}$. We distinguish the following cases based on the type of bag X_i.

Case 1. X_i is an introduce node with child X_j. Let $X_i = X_j \cup \{v\}$.

Clearly, for every pattern π for X_i that does not contain v, we can set $\Gamma_i[\pi] = \Gamma_j[\pi]$. $\Gamma_j[\pi]$ is a representative set for (X_i, π) for the following reasons: (i) follows because every color in $\chi(X_i)\setminus\chi(X_j)$ does not appear in V_j, since X_i is a vertex separator in G separating v and V_j and colors are connected. Hence, if two sequences in $\Gamma_j[\pi]$ that conform to (X_i, π) contradict (i), they contradict (i) w.r.t. (X_j, π) as well, thus contradicting that $\Gamma_j[\pi]$ is a representative set for (X_j, π). For property (ii), it is easy to observe that v does not appear on any path between two vertices in π having internal vertices in $V_i \setminus X_i$, and hence, this property is inherited from the child node X_j.

Now let $\pi = (v_1 = s, \sigma_1, v_2, \sigma_2, \ldots, \sigma_{r-1}, v_r = t)$ be a pattern such that $v_q = v$, $q \in [2..r-1]$, and let $\pi' = (v_1, \sigma_1, \ldots v_{q-1}, 0, v_{q+1}, \sigma_{q+1}, \ldots, \sigma_{r-1}, v_r)$. Note that since X_j is a separator between v and V_j, the only possibility for a path from v to a different vertex in X_i to have all internal vertices in $V_i \setminus X_i$ is if it is a direct edge. Therefore, if $\sigma_{q-1} = 1$ (resp. $\sigma_q = 1$) then $v_{q-1}v$ (resp. $v_q v$) is an edge in G. Otherwise, there is no sequence that conforms to (X_i, π).

We obtain $\Gamma_i[\pi]$ from $\Gamma_j[\pi']$ as follows. For every $\mathcal{S}' = (P'_1, P'_2, \ldots, P'_{r-2}) \in \Gamma_j[\pi']$, we replace the empty path corresponding to 0 between v_{q-1} and v_{q+1} in π' by two paths P_{q-1}, P_q such that $P_{q-1} = ()$ (resp. $P_q = ()$) if $\sigma_{q-1} = 0$ (resp. $\sigma_q = 0$) and $P_{q-1} = (v_{q-1}, v)$ (resp. $P_{q-1} = (v, v_q)$) otherwise and we obtain $\mathcal{S} = (P'_1, \ldots, P'_{q-2}, P_{q-1}, P_q, P'_q, \ldots, P'_{r-2})$. Denote by $\mathcal{R}_\pi$ the set of all formed sequences $\mathcal{S}$. Finally, we set $\Gamma_i[\pi] = \mathbf{Refine}(\mathcal{R}_\pi)$. We claim that $\Gamma_i[\pi]$ is a representative set for (X_i, π).

Claim 7. *If X_i is an introduce node with child X_j, and Γ_j contains for each pattern π' for X_j a representative set for (X_j, π'), then $\Gamma_i[\pi]$ defined above is a representative set for (X_i, π).*

Proof. From the application of **Refine()**, it is clear that $\Gamma_i[\pi]$ does not violate property (i) of the definition of representative sets. Assume now that there exists a sequence $\mathcal{S} \notin \Gamma_i[\pi]$ of length at most ℓ that conforms to (X_i, π) and violates property (ii) of Definition 6.9. We define the sequence $\mathcal{S}'$ that conforms to π', and is the same as $\mathcal{S}$ on all paths that π and π' share. Since no two paths in $\mathcal{S}$ share a vertex that is not in X_i (since $\mathcal{S}$ violates (ii)), and all paths in $\mathcal{S}'$ are also in $\mathcal{S}$, it follows that no two paths in $\mathcal{S}'$ share a vertex that is not in X_j. Moreover, $|\mathcal{S}'| \leq |\mathcal{S}| \leq \ell$. Since $\Gamma_j[\pi']$ is a representative set for (X_j, π'), it follows that there exists $\mathcal{S}'_1 \in \Gamma_j[\pi']$ such that $\mathcal{S}'_1 \preceq_j \mathcal{S}'$ and $|\mathcal{S}'_1| \leq |\mathcal{S}'|$. Let $\mathcal{S}_1$ be the sequence obtained from $\mathcal{S}'_1$ and conforming to (X_i, π). Then $\mathcal{S}_1 \in \mathcal{R}_\pi$

and it is easy to verify that $|\mathcal{S}_1| \leq \ell$. Hence by Lemma 6.12, there is a sequence $\mathcal{S}_2 \in \Gamma_i[\pi]$ such that $\mathcal{S}_2 \preceq_i \mathcal{S}_1$ and $|\mathcal{S}_2| \leq |\mathcal{S}_1|$. Since either both $\mathcal{S}_1$ and $\mathcal{S}$ contain v or none of them does, we have $\mathcal{S}_1 \preceq_i \mathcal{S}$ and $|\mathcal{S}_1| \leq |\mathcal{S}|$. By transitivity of $\preceq_i$ (Lemma 5.5) and of $\leq$, it follows that $\mathcal{S}_2 \preceq_i \mathcal{S}$ and $|\mathcal{S}_2| \leq |\mathcal{S}|$. This contradicts the assumption that $\mathcal{S}$ violates property (ii). □

Case 2. X_i is a forget node with child X_j. Let $X_i = X_j \setminus \{v\}$.

Let $\pi = (s = v_1, \sigma_1, \ldots, \sigma_{r-1}, v_r = t)$ be a pattern for X_i. For $q \in [r-1]$ such that $\sigma_q = 1$, we define $\pi^q = (s = v'_1, \sigma'_1, \ldots, \sigma'_r, v'_{r+1} = t)$ to be the pattern obtained from π by inserting v between v_q and v_{q+1} and setting $\sigma'_q = \sigma'_{q+1} = 1$. More precisely, we set $v'_p = v_p$ and $\sigma'_p = \sigma_p$ for $1 \leq p \leq q$, $v'_{q+1} = v$ and $\sigma'_{q+1} = 1$, and finally $v'_p = v_{p-1}$ and $\sigma'_p = \sigma_{p-1}$ for $q+2 \leq p \leq r$. We define $\mathcal{R}_\pi$ as follows:

$$\mathcal{R}_\pi = \Gamma_j[\pi] \cup \{\mathcal{S} = (P_1, \ldots, P_{q-1}, P_q \circ P_{q+1}, P_{q+2}, \ldots, P_r) \mid (P_1, \ldots, P_r) \in \Gamma_j[\pi^q], q \in [r-1] \wedge \sigma_q = 1\}.$$

Finally, we set $\Gamma_i[\pi] = \mathbf{Refine}(\mathcal{R}_\pi)$ and we claim that $\Gamma_i[\pi]$ is a representative set for (X_i, π).

Claim 8. *If X_i is a forget node with child X_j, and Γ_j contains for each pattern π' for X_j a representative set for (X_j, π'), then $\Gamma_i[\pi]$ defined above is a representative set for (X_i, π).*

Proof. It is straightforward to see that $\Gamma_i[\pi]$ satisfies property (i) due to the way procedure **Refine()** works. Assume for a contradiction that there exists a sequence $\mathcal{S}$ that violates property (ii). We distinguish two cases.

First, suppose that no path in $\mathcal{S}$ contains v. Then $\mathcal{S}$ conforms to (X_j, π). Since no two paths in $\mathcal{S}$ share a vertex that is not in X_i, and since $\Gamma_j[\pi]$ is a representative set, there exists $\mathcal{S}_1 \in \Gamma_j[\pi]$ such that $\mathcal{S}_1 \preceq_j \mathcal{S}$ and $|\mathcal{S}_1| \leq |\mathcal{S}| \leq \ell$. Then $\mathcal{S}_1 \in \mathcal{R}_\pi$, and hence by Lemma 6.12, $\Gamma_i[\pi]$ contains a sequence $\mathcal{S}_2$ such that $\mathcal{S}_2 \preceq_i \mathcal{S}_1$ and $|\mathcal{S}_2| \leq |\mathcal{S}_1|$. Since $\mathcal{S}_1 \preceq_j \mathcal{S}$ and $X_i \subsetneq X_j$, it follows from Observation 5.7 that $\mathcal{S}_1 \preceq_i \mathcal{S}$. By transitivity of $\preceq_i$, it follows that $\mathcal{S}_2 \preceq_i \mathcal{S}$. Moreover, $|\mathcal{S}_2| \leq |\mathcal{S}|$, which is a contradiction to the assumption that $\mathcal{S}$ violates property (ii).

Second, suppose that there is a path P_q in $\mathcal{S}$ between v_q and v_{q+1} that contains v. We form a sequence $\mathcal{S}'$ from $\mathcal{S}$ by keeping every path $P \neq P_q$ in $\mathcal{S}$, and replacing P_q in the sequence by the two subpaths of P_q, $P'_q = (v_q, \ldots, v)$ and $P'_{q+1} = (v, \ldots, v_{q+1})$. The sequence $\mathcal{S}'$ conforms to (X_j, π^q), and since no two paths in $\mathcal{S}$ share a vertex that is not in X_i, no two paths in $\mathcal{S}'$ share a vertex that is not in X_j. Moreover, it is straightforward that $|\mathcal{S}'| = |\mathcal{S}|$. Since $\Gamma_j[\pi^q]$ is a representative set for (X_j, π^q), it follows that there exists a sequence $\mathcal{S}'_1 \in \Gamma_j[\pi^q]$ such that $\mathcal{S}'_1 \preceq_j \mathcal{S}'$ and $|\mathcal{S}'_1| \leq |\mathcal{S}'|$. Let $\mathcal{S}_1$ be the sequence conforming to (X_i, π) obtained from $\mathcal{S}'_1$ by applying the operation $\circ$ to the two paths in $\mathcal{S}'_1$ that share v. Then $\mathcal{S}_1 \in \mathcal{R}_\pi$ and $|\mathcal{S}_1| = |\mathcal{S}'_1| \leq \ell$. Therefore, by Lemma 5.11, $\Gamma_i[\pi]$ contains a sequence $\mathcal{S}_2$ such that $\mathcal{S}_2 \preceq_i \mathcal{S}_1$. Since $\mathcal{S}'_1 \preceq_j \mathcal{S}'$, $\chi(\mathcal{S}') = \chi(\mathcal{S})$, $\chi(\mathcal{S}'_1) = \chi(\mathcal{S}_1)$, and $X_i \subsetneq X_j$, it follows that $\mathcal{S}_1 \preceq_i \mathcal{S}$. By transitivity of $\preceq_i$, it follows that $\mathcal{S}_2 \preceq_i \mathcal{S}$. Moreover $|\mathcal{S}_2| \leq |\mathcal{S}|$, which is a contradiction to the assumption that $\mathcal{S}$ violates property (ii). □

Case 3. X_i is a join node with children X_j, $X_{j'}$.

Let $\pi = (s = v_1, \sigma_1, \ldots, \sigma_{r-1}, v_r = t)$ be a pattern for X_i. Initialize $\mathcal{R}_\pi = \emptyset$. For every two patterns $\pi_1 = (s = v_1, \tau_1, \ldots, \tau_{r-1}, v_r = t)$ and $\pi_2 = (s = v_1, \mu_1, \ldots, \mu_{r-1}, v_r = t)$ such that $\sigma_q = \tau_q + \mu_q$, and for every two sequences $\mathcal{S}_1 = (P_1^1, \ldots, P_1^{r-1}) \in \Gamma_j[\pi_1]$ and $\mathcal{S}_2 = (P_2^1, \ldots, P_2^{r-1}) \in \Gamma_{j'}[\pi_2]$, we add the sequence $\mathcal{S} = (P_1, \ldots, P_{r-1})$ to $\mathcal{R}_\pi$, where $P_q = P_1^q$ if P_2^q is the empty path, otherwise, $P_q = P_2^q$, for $q \in [r-1]$. We set $\Gamma_i[\pi] = \mathbf{Refine}(\mathcal{R}_\pi)$, and we claim that $\Gamma_i[\pi]$ is a representative set for (X_i, π).

Claim 9. *If X_i is a join node with children X_j, $X_{j'}$, and Γ_j (resp. $\Gamma_{j'}$) contains for each pattern π' for $X_j = X_{j'} = X_i$ a representative set for (X_j, π') (resp. $(\pi', X_{j'})$), then $\Gamma_i[\pi]$ defined above is a representative set for (X_i, π).*

Proof. Clearly $\Gamma_i[\pi]$ satisfies property (i) due to the application of the procedure **Refine()**. To argue that $\Gamma_i[\pi]$ satisfies properties (ii), suppose not, and let $\mathcal{S} = (P_1, \ldots, P_{r-1})$ be a sequence that violates property (ii). Notice that every path P_q, $q \in [r-1]$ is either an edge between two vertices in X_i, or is a path between two vertices in X_i such that its internal vertices are either all in $V_j \setminus X_i$ or in $V_{j'} \setminus X_i$; this is true because X_i is a vertex separator separating $V_j \setminus X_i$ from $V_{j'} \setminus X_i$ in G. Define the two sequences $\mathcal{S}_1 = (P_1^1, \ldots, P_1^{r-1})$ and $\mathcal{S}_2 = (P_2^1, \ldots, P_2^{r-1})$ as follows. For $q \in [r-1]$, if P_q is empty then set both P_1^q and P_2^q to the empty path; if P_q is an edge then set $P_1^q = P_q$ and P_2^q to the empty path. Otherwise, P_q is either a path in $G[V_j]$ or in $G[V_{j'}]$; in the former case set $P_1^q = P_q$ and P_2^q to the empty path, and in the latter case set $P_2^q = P_q$ and P_1^q to the empty path. Since no two paths in $\mathcal{S}$ share a vertex that is not in X_i, and $X_i = X_j = X_{j'}$, no two paths in $\mathcal{S}_1$ (resp. $\mathcal{S}_2$) share a vertex that is not in X_j (resp. $X_{j'}$). Moreover, it is easy to see that $|\mathcal{S}_1| + |\mathcal{S}_2| = |\mathcal{S}| \leq \ell$. Let $\pi_1 = (s = v_1, \tau_1, \ldots, \tau_{r-1}, v_r = t)$ and $\pi_2 = (s = v_1, \mu_1, \ldots, \mu_{r-1}, v_r = t)$ be the two patterns that $\mathcal{S}_1$ and $\mathcal{S}_2$ conform to, respectively, and observe that, for every $q \in [r-1]$, we have $\sigma_q = \tau_q + \mu_q$. Since $\Gamma_j[\pi_1]$ and $\Gamma_{j'}[\pi_2]$ are representative sets, it follows that there exist $\mathcal{S}_1' = (Y_1', \ldots, Y_{r-1}')$ in $\Gamma_j[\pi_1]$ and $\mathcal{S}_2' = (Z_1', \ldots, Z_{r-1}')$ in $\Gamma_{j'}[\pi_2]$ such that $\mathcal{S}_1' \preceq_j \mathcal{S}_1$, $|\mathcal{S}_1'| \leq |\mathcal{S}_1|$ and $\mathcal{S}_2' \preceq_{j'} \mathcal{S}_2$, $|\mathcal{S}_2'| \leq |\mathcal{S}_2|$. Let $\mathcal{S}' = (P_1', \ldots, P_{r-1}')$, where $P_q' = Y_q'$ if Z_q' is the empty path, otherwise, $P_q = Z_q'$, for $q \in [r-1]$. The sequence $\mathcal{S}'$ conforms to π, is in $\mathcal{R}_\pi$, and $|\mathcal{S}'| = |\mathcal{S}_1'| + |\mathcal{S}_2'| \leq |\mathcal{S}|$. By Lemma 6.12, $\Gamma_i[\pi]$ contains a sequence $\mathcal{S}''$ such that $\mathcal{S}'' \preceq_i \mathcal{S}'$ and $|\mathcal{S}''| \leq |\mathcal{S}'|$. From Observation 5.7, since $X_i = X_j = X_{j'}$, from $\mathcal{S}_1' \preceq_j \mathcal{S}_1$ and $\mathcal{S}_2' \preceq_{j'} \mathcal{S}_2$ it follows that $\mathcal{S}_1' \preceq_i \mathcal{S}_1$ and $\mathcal{S}_2' \preceq_i \mathcal{S}_2$. Since $\chi(\mathcal{S}_1) \cup \chi(\mathcal{S}_2) = \chi(\mathcal{S})$ and $\chi(\mathcal{S}_1') \cup \chi(\mathcal{S}_2') = \chi(\mathcal{S}')$, and since $\chi(\mathcal{S}_1) \cap \chi(\mathcal{S}_2) \subseteq \chi(X_i)$, by Lemma 5.8, it follows that $\mathcal{S}' \preceq_i \mathcal{S}$. Since $\mathcal{S}'' \preceq_i \mathcal{S}'$, by transitivity of $\preceq_i$, it follows that $\mathcal{S}'' \preceq_i \mathcal{S}$. Moreover $|\mathcal{S}''| \leq |\mathcal{S}|$, which concludes the proof. □

We can now conclude with the following theorem:

Theorem 6.13. *There is an algorithm that on input $(G, C, \chi, s, t, k, \ell)$ of* Bounded-length Connected Obstacle Removal, *either outputs a k-valid s-t path in G or decides that no such path exists, in time $\mathcal{O}^\star(f(k,\ell)^{37\ell^2})$, where $f(k,\ell) = \mathcal{O}(c^{\ell k} \cdot k^k \cdot \ell^{3\ell k})$, for some constant $c > 1$. Therefore,* Bounded-length Connected Obstacle Removal *parameterized by both k and the length of the path is in* FPT.

Proof. If $d_G(s,t) > \ell$, then, by definition, there is no s-t path of length at most ℓ. Hence, we assume that $d_G(s,t) \leq \ell$. By Lemma 6.1, if there exists a vertex v such that $d_G(s,v) > \ell + 1$, we can contract any edge incident to v and obtain an equivalent instance. The contraction of an single edge can be done in time polynomial in the size of the instance and after applying Lemma 6.1 $|E|$ times, we would get a trivial instance. Moreover, from the proof of Lemma 6.1 it follows that we can obtain a solution in the original instance from a solution in contracted instance in polynomial time. Therefore, we can assume for the rest of the proof that we applied Lemma 6.1 exhaustively, and hence G has radius at most $\ell + 1$ and treewidth ω that is at most $3\ell + 4$ [22]. Moreover, a tree decomposition of G of width ω can be computed in (polynomial) time $\mathcal{O}(\ell \cdot n)$ [18]. From such a tree decomposition, in polynomial time we can compute a nice tree decomposition $(\mathcal{V}, \mathcal{T})$ of G whose width is at most $\omega \leq 3\ell + 4$ and satisfying $|\mathcal{V}| = \mathcal{O}(|V(G)|)$ [19].

The algorithm starts by removing the colors of s and t from G, and decrements k by $|\chi(s) \cup \chi(t)|$ (see Assumption 2.2). Afterwards, if $k < 0$, the algorithm concludes that there is no k-valid s-t path in G. If $st \in E(G)$ and $k \geq 0$, the algorithm outputs the path (s,t). Now we know that s

and t are not adjacent, and that $\chi(s) = \chi(t) = \emptyset$. The algorithm then adds s and t to every bag in $\mathcal{T}$, and executes the dynamic programming algorithm based on $(\mathcal{V}, \mathcal{T})$, described in this section, to compute a table Γ_i that contains, for each bag X_i in $\mathcal{T}$ and each pattern π for X_i, a representative set $\mathcal{R}_\pi$ for (X_i, π).

From Claims 6, 7, 8, 9, it follows, by induction on the height of the tree-decomposition $(\mathcal{V}, \mathcal{T})$ (the base case corresponds to the leaves), that the root node X_r contains a representative set $\Gamma_r[\pi]$ for the sequence $\pi = (s, 1, t)$. If $\Gamma_r[\pi]$ is empty, the algorithm concludes that there is no k-valid s-t path of length at most ℓ in G. Otherwise, noting that there is only one sequence $\mathcal{S}$ in the representative set $\Gamma_r[\pi]$ since $X_r = \{s, t\}$ and s and t are empty, the algorithm outputs the k-valid s-t path P formed by $\mathcal{S}$. The correctness follows from the following argument, which shows that if there is a k-valid s-t path of length at most ℓ in G, then the algorithm outputs such a path. Suppose that P' is a k-valid induced s-t path of length at most ℓ and let $\mathcal{S}' = (P')$. Since $G^r_{st} = G$, it follows that $\mathcal{S}'$ conforms to (X_r, π). Moreover $|\mathcal{S}'| \leq \ell$ and $\mathcal{S}'$ contains exactly one path that is induced, hence no two paths in $\mathcal{S}'$ share a vertex. Therefore, by property (ii) of representative sets, there exists a sequence $\mathcal{S}$ in $\Gamma_r[\pi]$ satisfying $\mathcal{S} \preceq_r \mathcal{S}'$ and $|\mathcal{S}| \leq |\mathcal{S}'|$. Noting that a sequence in $\Gamma_r[\pi]$ must consist of a single k-valid s-t path of length at most ℓ, it follows that the algorithm correctly outputs such a path.

Next, we analyze the running time of the algorithm. We observe that among the three types of bags in $\mathcal{T}$, the worst running time is for a join bag. Therefore, it suffices to upper bound the running time for a join bag, and since $|\mathcal{V}| = \mathcal{O}(n)$, the upper bound on the overall running time would follow.

Consider a join bag X_i with children $X_j, X_{j'}$. Let ω' be the width of $\mathcal{T}$ plus 1, which serves as an upper bound on the bag size in $\mathcal{T}$. Therefore, we have $\omega' \leq 3\ell+7$, where the (additional) plus 2 is to account for the vertices s and t that were added to each bag. The algorithm starts by enumerating each pattern π for X_i. The number of such patterns is at most $2^{\omega'} \cdot \omega' \cdot \omega'! = \mathcal{O}^*(2^{\omega'} \cdot \omega'!)$, where $\omega' \cdot \omega'!$ is an upper bound on the number of ordered selections of a subset of vertices from the bag, and $2^{\omega'}$ is an upper bound on the number of combinations for the σ_i's in the selected pattern. Fix a pattern π for X_i. To compute $\Gamma_i[\pi]$, the algorithm enumerates all ways of partitioning π into pairs of patterns π_1, π_2 for the children bags; there are $2^{\omega'}$ ways of partitioning π into such pairs, because for each $\sigma_i = 1$ in π, the path between v_i and v_{i+1} is either reflected in π_1 or in π_2. For a fixed pair π_1, π_2, the algorithm iterates through all pairs of sequences in the two tables $\Gamma_j[\pi_1]$ and $\Gamma_{j'}[\pi_2]$. Since each table contains a representative set, by Lemma 6.10, the size of each table is $h_1(k, \ell)^{\omega'^2}$, where $h_1(k, \ell) = \mathcal{O}(c_1^{\ell k} \cdot k^k \cdot \ell^{3\ell k})$ for some constant $c_1 > 1$, and hence iterating over all pairs of sequences in the two tables can be done in $\mathcal{O}(h_1(k, \ell)^{2\omega'^2})$ time. From the above, it follows that the set $\mathcal{R}_\pi$ can be computed in time $2^{\omega'} \cdot \mathcal{O}(h_1(k, \ell)^{2\omega'^2}) = \mathcal{O}(h_2(k, \ell)^{2\omega'^2})$, where $h_2(k, \ell) = \mathcal{O}(c_2^{\ell k} \cdot k^k \cdot \ell^{3\ell k})$, for some constant $c_2 > 1$, which is also an upper bound on the size of $\mathcal{R}_\pi$. By Lemma 6.12, applying **Refine()** to $\mathcal{R}_\pi$ takes time $\mathcal{O}^*(h_2(k, \ell)^{4\omega'^2})$. It follows from all the above that the running time taken by the algorithm to compute Γ_i is $\mathcal{O}^*(h_2(k, \ell)^{4\omega'^2} \cdot 2^{\omega'} \cdot \omega'!) = \mathcal{O}^*(h_3(k, \ell)^{4\omega'^2})$, where $h_3(k, \ell) = \mathcal{O}(c_3^{\ell k} \cdot k^k \cdot \ell^{3\ell k})$, for some constant $c_3 > 1$, and hence the running time of the algorithm is $\mathcal{O}^\star(f(k, \ell)^{37\ell^2})$, where $f(k, \ell) = \mathcal{O}(c^{\ell k} \cdot k^k \cdot \ell^{3\ell k})$, for some constant $c > 1$. □

6.3 Applications

In this subsection, we describe some applications of Theorem 6.13. The first result is a direct consequence of Theorem 6.13. We still mention it as a theorem due its practical applications, as one naturally seeks a path that is not very long, and in particular, whose length is not much larger than the number of obstacles intersected by the path:

Theorem 6.14. *For any computable function h, the restriction of* Connected Obstacle Removal *to instances in which the length of the sought path is at most $h(k)$ is* FPT *parameterized by k only.*

We note that the above restriction of Connected Obstacle Removal is NP-hard, as a consequence of (the proof of) Corollary 3.3.

The second application we describe is related to an open question posed in [8]. For an instance $I = (G, C, \chi, s, t, k)$ of Connected Obstacle Removal, and a color $c \in C$, define the *intersection number* of c, denoted $\iota(c)$, to be the number of vertices in G on which c appears. Define the *intersection number* of G, $\iota(G)$, as $\max\{\iota(c) \mid c \in C\}$. Consider the following problem:
Bounded-intersection Connected Obstacle Removal
Given: A planar graph G such that $\iota(G) \leq i$; a set of colors C; $\chi : V \longrightarrow 2^C$; and two designated vertices $s, t \in V(G)$
Parameter: k, i
Question: Does there exist a k-valid s-t-path in G?

Again, the above problem is NP-hard, as a consequence of (the proof of) Corollary 3.3.

Theorem 6.15. Bounded-intersection Connected Obstacle Removal *is* FPT*.*

Proof. Since the number of vertices in G on which any color $c \in C$ appears is at most $\iota(G)$, the length of any k-valid s-t path is $\mathcal{O}(k \cdot i)$. The result now follows from Theorem 6.13. □

The following corollary is a direct consequence of Theorem 6.15:

Corollary 6.16. *For any computable function h,* Bounded-intersection Connected Obstacle Removal *restricted to instances (G, C, χ, s, t, k) satisfying $\iota(G) \leq h(k)$ is* FPT *parameterized by k only.*

Corollary 6.15 has applications pertaining to geometric instances of the connected obstacle removal problem whose auxiliary graph is an instance of Bounded-intersection Connected Obstacle Removal. In particular, an interesting case that was studied in the literature corresponds to the case in which the obstacles are convex polygons, each intersecting at most a constant number of other polygons. The complexity of this problem was left as an open question in [8], and remains unresolved. The result in Corollary 6.16 subsumes this case, and even the more general case in which the obstacles are arbitrary connected convex regions satisfying that the number of regions intersected by any region is a constant, as it is easy to see that the auxiliary graph of such instances will have a constant intersection number[1]. In fact, we can even allow the intersection number to be any function of the parameter:

Theorem 6.17. *Let h be a computable function. The restriction of* Geomertic Connected Obstacle Removal *to any set of connected convex obstacles in the plane satisfying that each obstacle intersects at most $h(k)$ other obstacles, is* FPT *parameterized by k.*

Whereas the complexity of the problem in the above theorem is open, the theorem settles its parameterized complexity by showing it to be in FPT.

[1]Note that convexity is essential here, as otherwise, the intersection number of the auxiliary graph may be unbounded.

References

[1] J. Canny. A new algebraic method for robot motion planning and real geometry. In *Proceedings of the 28th Annual Symposium on Foundations of Computer Science*, pages 39–48. IEEE Computer Society, 1987.

[2] D. Chen and H. Wang. Computing shortest paths among curved obstacles in the plane. *ACM Transanctions on Algorithms*, 11(4):26:1–26:46, 2015.

[3] Y. Chen, J. Flum, and M. Grohe. Machine-based methods in parameterized complexity theory. *Theoretical Computer Science*, 339(2-3):167–199, 2005.

[4] Hubert de Fraysseix, János Pach, and Richard Pollack. Small sets supporting fary embeddings of planar graphs. In *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*, STOC '88, pages 426–433, New York, NY, USA, 1988. ACM.

[5] R. Diestel. *Graph Theory, 4th Edition*. Springer, 2012.

[6] R. Downey and M. Fellows. *Fundamentals of Parameterized Complexity*. Texts in Computer Science. Springer, Berlin, Heidelberg, 2013.

[7] E. Eiben, J. Gemmell, I. Kanj, and A. Youngdahl. Improved results for minimum constraint removal, 2017. Under submission to a double-blind reviewed conference.

[8] L. Erickson and S. LaValle. A simple, but NP-hard, motion planning problem. In *Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence*, 2013.

[9] István Fáry. On straight line representation of planar graphs. *Acta Univ. Szeged. Sect. Sci. Math.*, 11:229–233, 1948.

[10] M.R. Garey and D.S. Johnson. *Computers and Intractability*. W.H. Freeman, 1979.

[11] A. Gorbenko and V. Popov. The discrete minimum constraint removal motion planning problem. In *Proceedings of the American Institute of Physics*, volume 1648. AIP Press, 2015.

[12] S. Hartung and R. Niedermeier. Incremental list coloring of graphs, parameterized by conservation. *Theoretical Computer Science*, 494:86–98, 2013.

[13] K. Hauser. The minimum constraint removal problem with three robotics applications. *International Journal of Robotics Research*, 33(1):5–17, 2014.

[14] J. Hershberger, N. Kumar, and S. Suri. Shortest Paths in the Plane with Obstacle Violations. In *Proceedings of ESA*, volume 87 of *LIPIcs*, pages 49:1–49:14, 2017.

[15] J. Hershberger and S. Suri. An optimal algorithm for Euclidean shortest paths in the plane. *SIAM Journal on Computing*, 28(6):2215–2256, 1999.

[16] I. Hicks. Planar branch decompositions I: The ratcatcher. *INFORMS Journal on Computing*, 17(4):402–412, 2005.

[17] D. Johnson and M. Szegedy. What are the least tractable instances of Max independent set? In *Proceedings of the Tenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 927–928, 1999.

[18] I. Katsikarelis. Computing bounded-width tree and branch decompositions of k-outerplanar graphs. *CoRR*, abs/1301.5896, 2013.

[19] T. Kloks. *Treewidth, Computations and Approximations*, volume 842 of *Lecture Notes in Computer Science*. Springer, 1994.

[20] S. LaValle. *Planning Algorithms*. Cambridge University Press, New York, NY, USA, 2006.

[21] Z. McCarthy, T. Bretl, and S. Hutchinson. Proving path non-existence using sampling and alpha shapes. In *IEEE International Conference on Robotics and Automation*, pages 2563–2569. IEEE, 2012.

[22] N. Robertson and P. Seymour. Graph minors. III. planar tree-width. *Journal of Combinatorial Theory, Series B*, 36(1):49 – 64, 1984.

[23] P. Seymour and R. Thomas. Call routing and the ratcatcher. *Combinatorica*, 14(2):217–241, 1994.