Non-monochromatic and conflict-free coloring on tree spaces and planar network spaces

Citation for published version (APA):
Aronov, B., de Berg, M. T., Markovic, A., & Woeginger, G. J. (2018). Non-monochromatic and conflict-free coloring on tree spaces and planar network spaces. *arXiv*, *2018*, [1805.02538vl]. https://doi.org/10.48550/arXiv.1805.02538

DOI:
10.48550/arXiv.1805.02538

Document status and date:
Published: 07/05/2018

Document Version:
Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Non-Monochromatic and Conflict-Free Coloring on Tree Spaces and Planar Network Spaces

Boris Aronov* Mark de Berg† Aleksandar Markovic† Gerhard Woeginger‡

September 10, 2018

Abstract

It is well known that any set of n intervals in $\mathbb{R}^1$ admits a non-monochromatic coloring with two colors and a conflict-free coloring with three colors. We investigate generalizations of this result to colorings of objects in more complex 1-dimensional spaces, namely so-called tree spaces and planar network spaces.

1 Introduction

Conflict-free colorings, or CF-colorings for short, were introduced by Even *et al.* [4] and Smorodinsky [8] to model frequency assignment to base stations in wireless networks. In the basic setting one is given a set S of objects in the plane—often disks are considered—and the goal is to assign a color to each object such that the following holds: for any point p in the plane such that the set $S_p := \{D \in S \mid p \in D\}$ of objects containing p is non-empty, S_p must contain an object whose color is different from the colors of the other objects in S_p. Even *et al.* proved, among other things, that any set of disks admits a CF-coloring with $O(\log n)$ colors. This bound is tight in the worst case. Since then many different geometric variants of CF-colorings have been studied. For example, Har-Peled and Smorodinsky [5] generalized the result to objects with near-linear union complexity, while Even *et al.* [4] considered the dual version of the problem. See the survey by Smorodinsky [10] for an overview. A restricted type of a CF-coloring is a *unique-maximum* (*UM*) *coloring*, in which the colors are identified with integers, and the maximum color in the set S_p is required to be unique. Another type of coloring, often used as an intermediate step to obtain a CF-coloring, is *non-monochromatic* (*NM*). In an NM-coloring—sometimes called *a proper coloring*—we only require that, for any point p in the plane, if the set S_p contains at least two elements, not all of them have the same color. Smorodinsky [9] showed that if an NM-coloring of k elements using $\beta(k)$ colors exists for every k, one can CF-color n elements with $O(\beta(n) \log n)$ colors.

CF- or NM-coloring objects in $\mathbb{R}^1$ is significantly easier than in the planar case. In $\mathbb{R}^1$ the objects become intervals, assuming we require the objects to be connected, and a folklore result states that any set of intervals in $\mathbb{R}^1$ can be CF-colored with three colors and NM-colored with two colors. (This is achieved by the *chain methods*, which we describe below.) Thus, unlike in the planar case, the number of colors for a CF- or NM-coloring of intervals in $\mathbb{R}^1$ does not depend on the number of intervals to be colored.

We are interested in generalizations of this result to 1-dimensional spaces that have a more complex topology than $\mathbb{R}^1$. To this end we consider *network spaces*: 1-dimensional spaces with

[*] New York University, USA. BA has been partially supported by NSF Grants CCF-11-17336, CCF-12-18791, and CCF-15-40656, and by BSF grant 2014/170.

[†] TU Eindhoven, the Netherlands. MdB and AM are supported by the Netherlands' Organisation for Scientific Research (NWO) under project no. 024.002.003.

[‡] RWTH Aachen, Germany.

the topology of an arbitrary graph. It is convenient to view a network space $\mathcal{N}$ as being embedded in $\mathbb{R}^2$, although the embedding is actually immaterial. In this view the *nodes* of $\mathcal{N}$ are points in $\mathbb{R}^2$, and the *edges* are simple curves connecting pairs of nodes and otherwise disjoint. We let $d\colon \mathcal{N}^2 \to \mathbb{R}_+$ denote the geodesic distance on $\mathcal{N}$. In other words, for two points $p, q \in \mathcal{N}$—these points may lie in the interior of an edge—we let $d(p, q)$ denote the minimum Euclidean length of any path connecting p to q in $\mathcal{N}$. We consider two special types of network spaces, *tree spaces* and *planar network spaces*, whose topology is that of a tree and a planar graph, respectively.

The objective of our paper is to investigate the number of colors needed to CF- or NM-color a set $\mathcal{A}$ of n objects in a network space, where we consider various classes of connected objects. (Here CF- and NM-colorings are defined as above: in a CF-coloring, for any point $p \in \mathcal{N}$ the set $S_p := \{o \in \mathcal{A} \mid p \in o\}$ of objects containing p should have an object with a unique color when it is non-empty, and in an NM-coloring the set S_p should not be monochromatic when it consists of at least two objects.) In particular, we consider balls on $\mathcal{N}$—the *ball centered at* $p \in \mathcal{N}$ *of radius* r is defined as $B(p, r) := \{q \in \mathcal{N} \mid d(p, q) \leqslant r\}$— and, for tree spaces, we also consider arbitrary connected subsets as objects. Note that, if the given network space is a single curve, then our setting, both for balls and for connected subspaces, reduces to coloring intervals in $\mathbb{R}^1$. The main question we want to answer is: How does the maximum number of colors needed to NM- or CF-color a set $\mathcal{A}$ of objects in a network space depend on the complexity of the network space and of the objects to be colored?

Our results. We assume without loss of generality that the nodes in our network space either have degree 1 or degree at least 3—there are no nodes of degree 2. Nodes of degree 1 are also called *leaves*, and nodes of degree at least 3 are also called *internal nodes*.

We start by considering colorings on a tree space, which we denote by $\mathcal{T}$. Let $\mathcal{A}$ be the set of n objects that we wish to color, where each object $T \in \mathcal{A}$ is a connected subset of $\mathcal{T}$. Note that each such object is itself also a tree. From now on we refer to the objects in $\mathcal{A}$ as "trees," and always use "tree space" when talking about $\mathcal{T}$. Observe that internal nodes of a tree are necessarily internal nodes of $\mathcal{T}$, but a tree leaf may lie in the interior of an edge of $\mathcal{T}$. We will investigate CF- and NM-chromatic number of trees on tree space as a function of the following parameters:

- k, the number of leaves of the tree space $\mathcal{T}$;
- ℓ, the maximum number of leaves of any tree in $\mathcal{A}$;
- n, the number of objects in $\mathcal{A}$.

We define the CF-chromatic number $X_{\mathrm{cf}}^{\mathrm{tree,trees}}(k, \ell; n)$ as the minimum number of colors sufficient to CF-color any set $\mathcal{A}$ of n trees of at most ℓ leaves each, in a tree space of at most k leaves. The NM-chromatic number $X_{\mathrm{nm}}^{\mathrm{tree,trees}}(k, \ell; n)$ is defined similarly. Rows 3 and 4 in Table 1 give our bounds on these chromatic numbers. Notice that the upper bounds do not depend on n. In other words, any set of trees in a tree space can be colored with a number of colors that depends only on the complexity of the tree space $\mathcal{T}$ and of the trees in $\mathcal{A}$. (Obviously the number of objects, n, is an upper bound on these chromatic numbers as well. To avoid cluttering the statements, we usually omit this trivial bound.) We also study balls in tree spaces. Here it turns out to be more convenient to not use k (the number of leaves) as the complexity measure of $\mathcal{T}$, but

- t, the number of internal nodes of $\mathcal{T}$.

We are interested in the chromatic numbers $X_{\mathrm{cf}}^{\mathrm{tree,balls}}(t; n)$ and $X_{\mathrm{nm}}^{\mathrm{tree,balls}}(t; n)$. Rows 5 and 6 of Table 1 state our bounds for these chromatic numbers.

After studying balls in tree spaces, we turn our attention to balls in planar network spaces. Our bounds on the corresponding chromatic numbers $X_{\mathrm{cf}}^{\mathrm{planar,balls}}(t; n)$ and $X_{\mathrm{nm}}^{\mathrm{planar,balls}}(t; n)$ are contained in row 7 and 8 of Table 1.

Related results. Above we considered CF- and NM-colorings in a geometric setting, but they can also be defined more abstractly. A CF-coloring on a hypergraph $\mathcal{H} = (V, E)$ is a coloring of

Space	Objects	Coloring	Upper Bound	Lower Bound	Reference
Line	Intervals	NM	2	2	Folklore
Line	Intervals	CF	3	3	Folklore
Tree	Trees	NM	$\min\left(\ell+1, 2\sqrt{6k}\right)$	$\min\left(\ell+1, \left\lfloor \frac{1+\sqrt{1+8k}}{2} \right\rfloor\right)$	Section 2
Tree	Trees	CF	$O(\ell \log k)$	$\lfloor \log_2 \min(k, n) \rfloor$	Section 2
Tree	Balls	NM	2	2	Section 3.1
Tree	Balls	CF	$\lceil \log t \rceil + 3$	$\lceil \log(t+1) \rceil$	Section 3.1
Planar	Balls	NM	4	4	Section 3.3
Planar	Balls	CF	$\lceil \log_{4/3} t \rceil + 3$	$\lceil \log(t+1) \rceil$	Section 3.3

Table 1: Overview of our results. The folklore result for intervals on the line (that is, in $\mathbb{R}^1$) is explained below.

the vertex set V such that, for every (non-empty) hyperedge $e \in E$, there is a vertex in e whose color is different from that of the other vertices in e. In a NM-coloring any hyperedge with at least two vertices should not be monochromatic. Smorodinsky's survey [10] also gives an overview of results on CF-colorings in this abstract setting.

The basic geometric version mentioned above—coloring objects in $\mathbb{R}^2$ with respect to points—can be phrased in terms of hypergraphs by letting the objects be the node set V and, for each point p in the plane, creating a hyperedge $e := S_p$. Another avenue for constructing a hypergraph $\mathcal{H}$ to be colored is to start with a graph $\mathcal{N}$, let the vertices of $\mathcal{H}$ be the nodes of $\mathcal{N}$ and create hyperedges for (the sets of vertices of) certain subgraphs of $\mathcal{N}$. For example, Pach and Tardos [7] considered the case where hyperedges are all the node neighborhoods. For this case, Abel *et al.* [1] recently showed that a planar graph can always be CF-colored with only three colors, if we allow some nodes to be uncolored. (Otherwise, we can use a dummy color, increasing the number of colors to four.) As another example, we let the hyperedges be induced by all the paths in the graph. This setting is equivalent to an older notion of *node ranking* [2], or *ordered coloring* [6]. Note that in the above results the goal is to color the nodes of a graph. We, on the other hand, do not want to color nodes, but objects (connected subsets) in a network space (which has a graph topology, but is a geometric object).

Preliminaries: the chain methods. We start by describing a folklore technique, called the *chain method*, to color intervals in $\mathbb{R}^1$ in a non-monochromatic fashion using at most two colors. We order the intervals left-to-right by their left endpoints (in case of ties, we take the longest interval first) and color them in this order using the so-called *active color* which is defined as follows. We start with blue as the active color. We color the first interval, then change the active color to red. We then use the following procedure: we color the next interval I in the ordering using the active color, then if the right endpoint of I is not contained in any other already colored interval, we change the active color from red to blue or blue to red.

To obtain a CF-coloring the chain method proceeds as follows. First, the interval with the leftmost left endpoint—in case of ties, the longest such interval—is colored blue. Next, the following procedure is repeated until we get stuck: Let I be the interval colored last. Among all intervals whose left endpoint lies in I and that are not contained in it, color the one extending farthest to the right red (if I is blue) or blue (if I is red). This creates a chain of alternating blue and red intervals. Each remaining interval is now either completely covered by the already colored intervals, or it lies completely to the right of them. The former intervals are given a dummy color (grey), the latter intervals are colored by applying the above procedure again.

Lemma 1.1. *There is a NM-coloring of intervals on a line using two colors, and a CF-coloring*

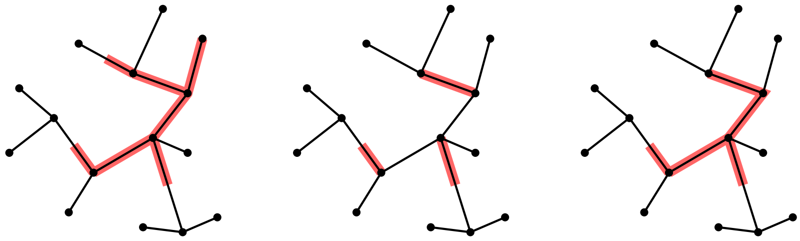

Figure 1: The original tree T (left), the set $\bigcup_{e\in E(T)} e \cap T$ (middle), and the new tree T' (right).

using three colors.

Proof. We prove the latter coloring is conflict-free; the proof for the NM-coloring is similar. Consider a point p contained in an interval. It is clear that p is contained in either a red or a blue interval. We suppose without loss of generality it is contained in a red interval $I_0 = [a_0, b_0]$. We show it is not contained in another red interval. Let us suppose by contradiction that it is contained in another red interval $I_1 = [a_1, b_1]$ with $a_1 \geqslant a_0$. Then p must also be contained in a blue interval $I_2 = [a_2, b_2]$, with $a_1 \geqslant a_2 \geqslant a_0$. Moreover, we have that $b_2 < b_1$. Thus, I_2 starts in I_0 and extends further than I_1, hence should have been chosen to be colored blue, which is a contradiction. Therefore, p is always contained in at most one red interval, and similarly, in at most one blue interval, and is always contained in a blue or in a red interval. Thus the coloring is conflict-free. □

2 Trees on Tree Spaces

2.1 The upper bound

Overview of the coloring procedure. Let $\mathcal{T}$ be a tree space with k leaves and let $\mathcal{A}$ be a set of n trees in $\mathcal{T}$, each with at most ℓ leaves. We describe an algorithm that NM-colors $\mathcal{A}$ in two phases: first, we select a subset $\mathcal{C} \subseteq \mathcal{A}$ of size at most $6k-12$ and color it with at most $\min\left(\ell+1, 2\sqrt{6k}\right)$ colors. In the second phase we extend this coloring to the whole set $\mathcal{A}$ without using new colors.

An edge e of $\mathcal{T}$ is a *leaf edge* if it is incident to a leaf; the remaining edges are *internal.* We define $\mathcal{C} \subseteq \mathcal{A}$ as the set of at most $6k-12$ trees selected as follows. For every pair (e, v), where e is an edge of $\mathcal{T}$ and v is an endpoint of e that is not a leaf of $\mathcal{T}$, we choose two trees containing v and extending the furthest into e (if they exist), that is, trees T of $\mathcal{A}$ containing v for which length$(T \cap e)$ is maximal, and place them in $\mathcal{A}(e, v)$. If two or more trees of $\mathcal{A}$ fully contain e, then $\mathcal{A}(e, v)$ contains two of them, chosen arbitrarily. If a tree contains an internal edge e fully, it may be chosen by both endpoints. We now define $\mathcal{A}(e) := \mathcal{A}(e, u) \cup \mathcal{A}(e, v)$ for each internal edge $e = uv$, $\mathcal{A}(e) := \mathcal{A}(e, v)$ for each leaf edge $e = uv$ with non-leaf endpoint v, and $\mathcal{C} := \bigcup \mathcal{A}(e)$, with the union taken over all edges e of $\mathcal{T}$. Then $\mathcal{A}(e)$ contains at most four trees for any internal edge e and at most two trees for any leaf edge e. If $\mathcal{T}$ has at most k leaves, it has at most k leaf edges and at most $k-3$ internal edges; recall that $\mathcal{T}$ has no degree-two nodes. Thus $|\mathcal{C}| \leqslant 6k-12$, as claimed. We first explain how to color $\mathcal{C}$.

Coloring $\mathcal{C}$. We color $\mathcal{C}$ in two steps. Let $T \in \mathcal{C}$ be a tree. We define $E(T)$ to be the set of edges e of $\mathcal{T}$ with $T \in \mathcal{A}(e)$. Firstly, if $\ell > 2\sqrt{6k}$ we select all subtrees T with $|E(T)| \geqslant \sqrt{6k}$, and give each of them a unique color. Since $\sum_e |\mathcal{A}(e)| \leqslant 6k-12$, there are at most $\sqrt{6k}-1$ such trees, so we use at most $\sqrt{6k}-1$ colors. For each uncolored $T \in \mathcal{C}$, we create a new tree T', defined as the smallest tree containing $\bigcup_{e\in E(T)} e \cap T$; see Fig. 1. T' has at most $\ell' := \min(\ell, \sqrt{6k})$ leaves because $|E(T)| < \sqrt{6k}$. Define $\mathcal{C}' := \{T' \mid T \in \mathcal{C}\}$.

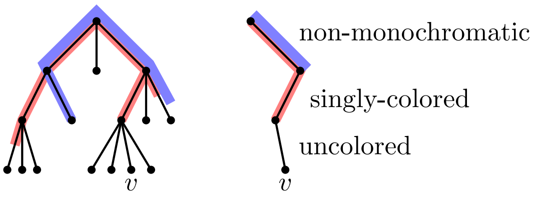

Figure 2: A coloring of trees (left) and an illustration of the invariant for v (right).

The second step is to color $\mathcal{C}'$. We need the following lemma, which shows that an NM-coloring of $\mathcal{C}'$ carries over to $\mathcal{C}$.

Lemma 2.1. *Any NM-coloring of $\mathcal{C}'$ corresponds to an NM-coloring of $\mathcal{C}$, that is, if we give each tree $T \in \mathcal{C}$ the color of the corresponding tree $T' \in \mathcal{C}'$ then we obtain an NM-coloring.*

Proof. Let q be a point on an edge e of $\mathcal{T}$ contained in at least two trees of $\mathcal{C}$ (if no such trees exists, the coloring is trivially non-monochromatic at q). Since q is contained in at least two trees of $\mathcal{C}$, it is also contained in two trees of $\mathcal{A}(e)$. Call these trees T_1 and T_2. Note that T_1 either receives a color in the first coloring step—namely, when $\ell > 2\sqrt{6k}$ and $|E(T_1)| \geqslant \sqrt{k}$—or $T_1' \in \mathcal{C}'$ contains q, since $e \in E(T_1)$. A similar statement holds for T_2. Since the colors used in the first step are unique and $\mathcal{C}'$ is NM-colored, this implies that T_1 and T_2 have different colors. Hence, $\mathcal{C}$ is NM-colored. □

Next we show how to NM-color $\mathcal{C}'$. Fix an arbitrary internal node r of $\mathcal{T}$ and treat $\mathcal{T}$ as rooted at r.. Our coloring procedure for $\mathcal{C}'$ maintains the following invariant: any path from r to a leaf v of $\mathcal{T}$ consists of three disjoint consecutive subpaths (some possibly empty), in this order, as illustrated in Fig. 2:

- a *non-monochromatic* subpath containing the root on which at least two trees are colored with at least two different colors,
- a *singly-colored* subpath covered by exactly one colored tree, and
- an *uncolored* subpath containing the leaf on which no tree is colored.

Observation 2.2. *Any set of trees containing r and satisfying the invariant described above is NM-colored if we disregard uncolored trees.*

We color the trees $T \in \mathcal{C}'$ that contain r in an arbitrary order, using $\ell' + 1$ colors, as follows: for each leaf v of T, we follow the path from v to the root r to find a singly-colored part. Note that if we find a singly-colored part—by the invariant there is at most one such part on the path from v to r—we cannot use that color for T. Since T has at most ℓ' leaves, this eliminates at most ℓ' colors. Hence, at least one color remains for T.

Lemma 2.3. *The procedure described above maintains the invariant and colors all trees of $\mathcal{C}'$ containing r with at most $\ell' + 1$ colors.*

Proof. Suppose the invariant holds before the coloring of T. Then we need to make sure the invariant still holds after T has been colored. Let w be a leaf of $\mathcal{T}$ and π_w the path from w to the root. Let v be the closest point to w in $\pi_w \cap T$. Note that v always exists as $r \in \pi_w \cap T$. Now let $\pi_v \subseteq \pi_w$ be the path from v to r. It is obvious that $\pi_w \cap T = \pi_v$. Then the part of π_v that was uncolored (if it was non-empty) now is singly-colored. The part that was singly-colored now becomes non-monochromatic, as we eliminated that color for T. And the part that was already non-monochromatic stays so. Therefore the invariant is indeed maintained for π_w, concluding the proof. □

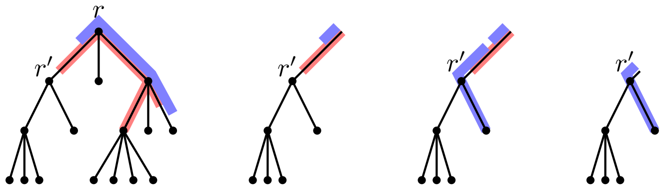

Figure 3: When recursing on the subspace rooted at r' (leftmost), the invariant does not hold anymore (middle left), as the parts are switched on the edge between r and r'. To remedy this, we first color the tree extending the farthest into that edge (middle right), starting from r'. We then trim the tree to fix the invariant (rightmost).

Once all the trees containing r are colored we delete r from $\mathcal{T}$, that is, we consider the space $\mathcal{T} \setminus \{r\}$, and we take the closures of the resulting connected components. This creates a number of subspaces such that each uncolored tree in $\mathcal{C}'$ is contained in exactly one of them. Consider such a subspace $\mathcal{T}'$ and let r' be the neighbor of r in $\mathcal{T}'$. We now want to recursively color the uncolored trees in $\mathcal{T}'$, taking r' as the root of $\mathcal{T}'$. However, the invariant might not hold on the edge e from r' to the old root r: Since now r is considered a child of r', the order of the three parts might switch on e—see Fig. 3. Suppose this is the case, and let c_e be the color of the singly-colored part on the edge e. (If the singly-colored part is empty, we can cut the tree between the non-monochromatic and the uncolored part and recurse immediately, which maintains the invariant.) Note also that, for the order to switch, the non-monochromatic part needs to end on e, and therefore the only color used in any singly-colored part of the tree rooted at r' is c_e. We overcome this problem by carefully choosing the order in which we color the trees containing r'. Namely, we fist color the tree T extending the farthest into e. In this case, there is only one color forbidden, namely c_e. We can therefore easily color T. We can then trim the treespace $\mathcal{T}'$ to remove any non-monochromatic and singly-colored part and hence restore the invariant and continue with the coloring.

Lemma 2.4. *$\mathcal{C}$ admits an NM-coloring with $\min(\ell + 1, 2\sqrt{6k})$ colors.*

Proof. The fact that the procedure above produces an NM-coloring follows from Lemmas 2.1 and 2.3. When $\ell > 2\sqrt{6k}$ we use $\sqrt{6k} - 1$ colors to deal with trees T with $|E(T)| \geqslant \sqrt{6k}$ and $\ell' + 1 \leqslant \min(\ell, 2\sqrt{6k}) + 1 \leqslant \sqrt{6k} + 1$ colors for the other trees, giving $2\sqrt{6k}$ colors in total. When $\ell \leqslant 2\sqrt{6k}$ we do not treat the trees with $|E(T)| \geqslant \sqrt{6k}$ separately, so we just use $\ell' + 1 \leqslant \min(\ell, \sqrt{6k}) + 1 \leqslant \ell + 1$ colors. □

Extending the coloring from $\mathcal{C}$ to $\mathcal{A}$. Let $c \colon \mathcal{C} \to \mathbb{N}$ be an NM-coloring on $\mathcal{C}$. We extend the coloring to $\mathcal{A}$ as follows. We start by coloring all trees in $\mathcal{A} \setminus \mathcal{C}$ containing an internal node of $\mathcal{T}$ using an arbitrary color already used. We then treat all edges in an abritrary order, coloring all trees contained in the edge as explained now.

Let $e = rr'$ be an arbitrary edge of $\mathcal{T}$ and $\mathcal{A}^*(e)$ be the set of uncolored trees contained in e. We color $\mathcal{A}^*(e)$ as follows. We first color the set of uncolored trees contained in e naively using the chain method. For this we use two new colors, which are used for all chains—we can re-use the same two colors for the chains, since trivially the chains in any two edges e, e' do not interact. However, we can avoid using two extra colors and re-use the colors from $\mathcal{C}$ as explained next.

First, if c uses fewer than two colors, then each node of $\mathcal{T}$ is contained in at most one tree. We then forget the trivial coloring c and use the chain method from scratch on $\mathcal{A}$. We start at a arbitrarily fixed leaf u of $\mathcal{T}$, and for any other leaf u', we consider the path between u and u' and use the chain method on the trees restricted to this path. Since for any node v, at most one tree contains v, no tree receives two different colors on two different paths. Moreover, the coloring is conflict-free, since any point in $\mathcal{T}$ is contained in a path from u to a certain leaf u'.

We may now suppose that c uses at least two colors. Let $T_r \in \mathcal{A}(e, r)$ and $T_{r'} \in \mathcal{A}(e, r')$, be

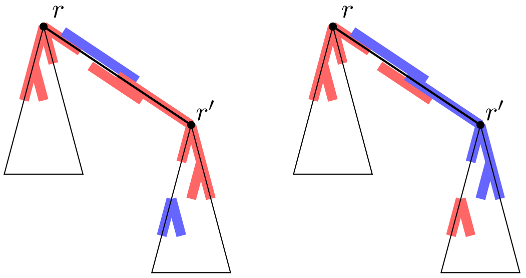

Figure 4: If the color of $T_{r'}$ changes with the chain method, we swap the labels of the old and new colors of $T_{r'}$ in the subspace rooted at r'.

the trees extending the farthest into e (arbitrarily chosen in case of a tie). Note that these trees might not exist. Also note that T_r and $T_{r'}$ are not in $\mathcal{A}^*(e)$. We define the following colors.

- Let c_r be the color of T_r, if T_r exists, and an arbitrary color otherwise.
- Let $c_{r'}$ be the color of $T_{r'}$, if $T_{r'}$ exists, and $c(T_{r'}) \neq c(T_r)$ (if T_r does not exist, we assume this is always true), and an arbitrary color different from c_r otherwise.

We then do the following.

(a) If T_r fully contains e, we color all trees in $\mathcal{A}^*(e)$ using $c_{r'}$.

(b) If $T_{r'}$ fully contains e, we color all trees in $\mathcal{A}^*(e)$ using c_r.

(c) Otherwise, we use the chain method for NM-colorings using c_r and $c_{r'}$ on $\mathcal{A}^*(e) \cup \{T_r\} \cup \{T_{r'}\}$. We start from r with color c_r so that T_r is the first tree colored and keep its color. We then check if the color of $T_{r'}$ changed. If so, let $\mathcal{C}_{r'} \subseteq \mathcal{C}$ be the subset of trees contained in the subspace rooted at r' (including e but not r) and excluding $T_{r'}$. We exchange c_r and $c_{r'}$ in $\mathcal{C}_{r'}$; see Fig. 4.

The following lemma proves the extended coloring is non-monochromatic.

Lemma 2.5. *Any NM-coloring c on $\mathcal{C}$ can be extended to $\mathcal{A}$ without using any extra color if c uses two colors or more, and with two colors otherwise.*

Proof. Let $\mathcal{A}_1$ be the subset of trees in $\mathcal{A} \setminus \mathcal{C}$ that contain an internal node of $\mathcal{T}$, and let $\mathcal{A}_2$ be the remaining trees in $\mathcal{A} \setminus \mathcal{C}$. By Lemma 2.4, we have an NM-coloring on $\mathcal{C}$. To prove that the method described above gives us an NM-coloring on $\mathcal{C} \cup \mathcal{A}_2$, we show that the following invariant holds each time an edge is colored: the coloring on $\mathcal{C} \cup \mathcal{A}_2$ is non-monochromatic when restricted to colored trees. It is clear that before the first edge is colored, the coloring is non-monochromatic as at this point the only trees colored are exactly those in $\mathcal{C}$. We hence only have to show the invariant still holds after coloring an edge $e = \{r, r'\}$. If we are in cases (a) or (b), the invariant trivially holds. It remains to consider the third case.

In the case (c) we use the chain method on $\mathcal{A}^*(e) \cup \{T_r\} \cup \{T_{r'}\}$, which immediately implies the coloring is non-monochromatic on e. To prove it is also non-monochromatic elsewhere, let $p \notin e$ be a point contained in at least two trees. Then we only have to show that the label swap we did on one side of e keeps the coloring non-monochromatic. The point p cannot be contained in one tree containing r and one tree containing r' at the same time, because no tree contains e fully. Therefore, p is contained in at least two trees from either side of e, hence two trees of different color.

Furthermore, the trees in $\mathcal{A}_1$ received an arbitrary color already used. To prove that this gives an NM-coloring for $\mathcal{A} = \mathcal{C} \cup \mathcal{A}_1 \cup \mathcal{A}_2$, it suffices to prove that each tree $T \in \mathcal{A}_1$ is *doubly-covered* by $\mathcal{C}$, that is, any point $q \in T$ is contained in at least two trees in $\mathcal{C}$. To this end, let e be an edge

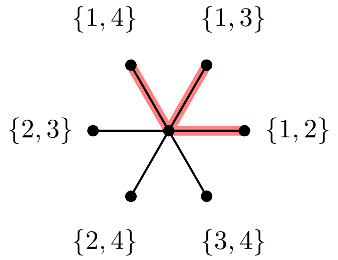

Figure 5: An example of the non-monochromatic lower bound for $k = 6$, $\ell = 3$, and $n = 4$. The tree T_1 is drawn in red.

such that $q \in e$. Then, since $T \notin \mathcal{C}$ and T contains an endpoint v of e, the two trees in $\mathcal{A}(e, v)$ contain q. Hence, T is doubly-covered by $\mathcal{C}$, as claimed. □

Theorem 2.6.

1. $X_{\text{nm}}^{\text{tree,trees}}(k, \ell; n) \leqslant \min\left(\ell + 1, 2\sqrt{6k}\right)$.

2. $X_{\text{cf}}^{\text{tree,trees}}(k, \ell; n) = O(\ell \log k)$.

Proof. For the NM-coloring part of the theorem, we use Lemmas 2.4 and 2.5. For the second part, if $\ell > 2\sqrt{6k}$ we again reduce $\mathcal{C}$ to $\mathcal{C}'$ using at most $\sqrt{6k} - 1$ colors. Then use the result by Smorodinsky [9] on the NM-coloring on $\mathcal{C}'$ provided by Lemma 2.3. Since this coloring uses at most $\ell' + 1$ colors and $|\mathcal{C}'| \leqslant 6k - 12$, the CF-coloring uses $O(\ell \log k)$ colors. We then extend the coloring to $\mathcal{A}$ using similar techniques as for the NM-coloring. This coloring uses $O(\sqrt{k} \log k)$ colors if $\ell > 2\sqrt{6k}$, which is in $O(\ell \log k)$, and directly $O(\ell \log k)$ colors otherwise. Note that a direct application of the result of Smorodinsty [9] would give a $O(\ell \log n)$ bound instead. □

2.2 The lower bound

We show a lower bound for the number of colors[1] needed to NM-color a set of trees in a tree space.

Theorem 2.7. *For all n, k, and ℓ, there exist a tree space $\mathcal{T}$ with k leaves and a set $\mathcal{A}$ at most n trees on $\mathcal{T}$, each with at most ℓ leaves, such that any non-monochromatic coloring of $\mathcal{A}$ uses at least* $\min\left(\ell + 1, \left\lfloor \frac{1+\sqrt{1+8k}}{2} \right\rfloor, n\right)$ *colors. In other words,*

$$X_{\text{nm}}^{\text{tree,trees}}(k, \ell; n) \geqslant \min\left(\ell + 1, \left\lfloor \frac{1+\sqrt{1+8k}}{2} \right\rfloor, n\right).$$

Proof. Let $\mathcal{T}$ be a star with k leaves. We construct the set $\mathcal{A}$ of m trees such that, for each pair of trees $T, T' \in \mathcal{A}$, there is a leaf of $\mathcal{T}$ contained in T and T', and no other tree from $\mathcal{A}$. Consequently, each tree in $\mathcal{A}$ must be assigned a distinct color. To this end, we define $m := \min(\ell + 1, m', n)$, where $m' := \lfloor (1+\sqrt{1+8k})/2 \rfloor$ is the largest integer such that $\binom{m'}{2} \leqslant k$. Then, for every pair $\{i, j\}$ with $1 \leqslant i < j \leqslant m$, we choose a distinct leaf of $\mathcal{T}$ and associate it with $\{i, j\}$. The total number of such pairs is $\binom{m}{2} \leqslant \binom{m'}{2} \leqslant k$, hence we can indeed associate a distinct leaf to each pair.

Let now $\mathcal{A} := \{T_1, \ldots, T_m\}$ be the set of trees defined as follows: for each $i = 1, \ldots, m$, the tree T_i is defined as the tree containing all the leaves associated with pairs $\{i, j\}$ for some $j \neq i$, i.e., T_i is the union, for all $j \neq i$, of edges from the root to a leaf associated with $\{i, j\}$. Fig. 5 shows an example. We now have to prove that the construction is possible within the parameters. Recall that $m \leqslant n$ so we have indeed at most n trees in $\mathcal{A}$, and that $m \leqslant m'$ where m' is chosen

[1] From now on, we either identify colors with integers or we use actual colors (red, blue, etc.) in our descriptions, whichever is more convenient.

to ensure k leaves are enough. We therefore only have to show that no tree $T_i, \dots, T_m$ has more than ℓ leaves. However, the number of leaves of each tree T_i is at most $m-1$, as we only create at most one leaf for T_i for each T_j with $j \neq i$. Hence, since $m \leqslant \ell+1$, each tree has at most ℓ leaves. Thus, the construction does not violate the parameters.

Finally, each tree needs a distinct color, and since there are m trees, the number of colors needed is $m = \min(\ell+1, \lfloor \frac{1+\sqrt{1+8k}}{2} \rfloor, n)$. □

Since any CF-coloring is also an NM-coloring, the lower bound in Theorem 2.7 holds for CF-coloring as well. The next theorem gives a stronger lower bound for CF-coloring in the case $\ell = 2$, that is, when the objects are paths.

Theorem 2.8. *For all n and k, there exist a tree space $\mathcal{T}$ with k leaves and a set $\mathcal{A}$ of at most n paths in $\mathcal{T}$ such that any conflict-free coloring of $\mathcal{A}$ uses at least $\lfloor \log_2 \min(k,n) \rfloor$ colors. In other words,*

$$X_{\text{cf}}^{\text{tree,paths}}(k;n) \geqslant \lfloor \log_2 \min(k,n) \rfloor.$$

Proof. Let $\mathcal{T}$ be a rooted complete binary tree of height $h = \lfloor \log_2 \min(k,n) \rfloor$. Note that $\mathcal{T}$ has at most $\min(k,n)$ leaves. For each leaf v of $\mathcal{T}$, we define π_v to be the path from v to the root r of $\mathcal{T}$. Our set $\mathcal{A}$ of objects is now defined as $\mathcal{A} := \{\pi_v \mid v \text{ a leaf of } \mathcal{T}\}$. (Trivially, $|\mathcal{A}| \leqslant n$.)

Let $c \colon \mathcal{A} \to \mathbb{N}$ be a conflict-free coloring of $\mathcal{A}$. We prove that c uses at least $h = \lfloor \log_2 \min(k,n) \rfloor$ colors by induction on the height h of $\mathcal{T}$. If $h = 1$, then there is only one degenerate path and the claim trivially holds. Suppose now that the claim holds for a tree of height h, and suppose the height of $\mathcal{T}$ is $h+1$. Since c is a conflict-free coloring, among the paths containing the root $r_1 := r$ of $\mathcal{T}$, there must be a path π_1 of unique color. Since by construction all paths in $\mathcal{A}$ contain the root, the color of π_1 is unique among all paths. Let r_2 be the child of r_1 not contained in π_1. We now use the induction hypothesis on the subtree rooted at r_2 with paths containing r_2 cut above it. Among these paths, there are h that use distinct colors. Moreover, none of these path can use $c(\pi_1)$, as this color is unique among all paths. Hence, we have indeed $h+1$ paths using distinct colors. This concludes the proof. □

The following theorem is a direct consequence of the previous two.

Theorem 2.9. *For all n, k, and ℓ, there exist a tree space $\mathcal{T}$ with k leaves and a set $\mathcal{A}$ at most n trees in $\mathcal{T}$ with at most ℓ leaves each such that any conflict-free coloring of $\mathcal{A}$ uses at least $\min\left(\ell+1, \left\lfloor \frac{1+\sqrt{1+8k}}{2} \right\rfloor, \lfloor \log_2 \min(k,n) \rfloor\right)$ colors. In other words,*

$$X_{\text{cf}}^{\text{tree,trees}}(k,\ell;n) \geqslant \max \begin{cases} \min\left(\ell+1, \left\lfloor \frac{1+\sqrt{1+8k}}{2} \right\rfloor\right) \\ \lfloor \log_2 \min(k,n) \rfloor. \end{cases}$$

3 Balls in Tree Spaces and on Planar Network Spaces

In this section we restrict the objects to balls. Let $\mathcal{N}$ be a network space, $d \colon \mathcal{N}^2 \to \mathbb{R}$ a distance function on $\mathcal{N}$, and let $\mathcal{A}$ be a set of balls on $\mathcal{N}$. We define the coverage $cov_x(B)$ of a node x by a ball $B = B(p,r)$ containing x as $cov_x(B) := r - d(p,x)$. Given a node x contained in at least one ball from $\mathcal{A}$, we define B_x as the ball maximizing the coverage of x, where we break ties using an arbitrary but fixed ordering on the balls. We say that B_x is *assigned* to x. Note that B_x does not exist if no ball contains x, and that a ball can be assigned to multiple nodes. We will regularly use the following lemma regarding the assigned balls.

Lemma 3.1. *Let x be an internal node of $\mathcal{N}$.*

(i) Suppose $\mathcal{N}$ is a tree space, and let $\mathcal{T}_1, \dots, \mathcal{T}_{\deg(x)}$ denote the subtrees resulting from removing x from $\mathcal{N}$ or, more precisely, the closures of the connected components of $\mathcal{T} \setminus \{x\}$. Let p be a point in some subtree $\mathcal{T}_i$ and suppose p is contained in a ball $B \in \mathcal{A}$ whose center lies in $\mathcal{T}_j$ with $j \neq i$. Then $p \in B_x$.

(ii) Suppose x is contained in at least one ball in $\mathcal{A}$. Let π be a shortest path from x to the center of B_x, and let y be a node on the path π. Then B_x is also assigned to y, that is, $B_x = B_y$.

Proof. Part (i) follows immediately from the definition of B_x. To prove part (ii), suppose for a contradiction that $B_y \neq B_x$ for some $y \in \pi$. Thus, $cov_y(B_y) \geqslant cov_y(B_x)$. Because π is a shortest path from x to the center of B_x, we have that $cov_x(B_x) = cov_y(B_x) - d(x,y)$. Moreover, $cov_y(B_y) - d(x,y) \leqslant cov_x(B_y)$ because of the triangle inequality. Hence, $cov_x(B_x) \geqslant cov_x(B_y) \geqslant cov_y(B_y) - d(x,y) \geqslant cov_y(B_x) - d(x,y) = cov_x(B_x)$. Thus $cov_x(B_x) = cov_x(B_y)$ and $cov_y(B_x) = cov_y(B_y)$. However, this is a contradiction as in case of a tie, we use the fixed ordering to choose which ball to assign to a node. □

3.1 Tree spaces: the upper bound

For balls on a tree space $\mathcal{T}$, the upper bounds from Theorem 2.6 with $\ell = k$ apply. Below we improve upon these bounds using the special structures of balls. Let $\mathcal{T}$ be a tree with t internal nodes. We present algorithms to NM-color balls on trees using two colors, and CF-color them with $\log t + 3$ colors.

Let $\mathcal{A}$ be a set of n balls on $\mathcal{T}$. Let also $\mathcal{C} := \{B = B(c,r) \mid \exists x : B = B_x\}$ be the set of balls assigned to at least one internal node. Recall that an internal node x is assigned the ball maximizing the coverage of x.

NM-coloring. We first explain how to NM-color $\mathcal{A}$. We use a divide-and-conquer approach. If $t = 0$, that is $\mathcal{T}$ consists of a single node or a single edge, we use the chain method for NM-coloring with colors blue and red. If $t > 0$, then we proceed as follows. Let $e = uv$ be an edge of $\mathcal{T}$. Let $\mathcal{T}_u$, respectively $\mathcal{T}_v$, be the connected component of $\mathcal{T} \setminus e$ containing u, respectively v. Recall that B_u and B_v are the balls assigned to u and v, respectively. Note that we may assume that both B_u or B_v exist, for otherwise recursion is trivial. Also observe that B_u and B_v may coincide. We define

$$\mathcal{A}(u) := \{\text{balls } B \in \mathcal{A} \text{ whose center lies in } \mathcal{T}_u\} \cup \{B_u\},$$

We define $\mathcal{A}(v)$ similarly. We recursively color $\mathcal{A}(u)$ in $\mathcal{T}_u$ and $\mathcal{A}(v)$ in $\mathcal{T}_v$, obtaining colorings of $\mathcal{A}(u)$ and $\mathcal{A}(v)$ with colors blue and red. In the recursive calls on $\mathcal{A}(u)$, and similarly for $\mathcal{A}(v)$, we "clip" the balls to within $\mathcal{T}_u$. Note that the clipped balls are still balls in the space $\mathcal{T}_u$. This is clear for the balls whose center lies in $\mathcal{T}_u$. The center of B_u may not lie in $\mathcal{T}_u$, but in that case it behaves within $\mathcal{T}_u$ as a ball with center u and radius $cov_u(B_u)$.

Let $\mathcal{A}(e) := \mathcal{A} \setminus (\mathcal{A}(u) \cup \mathcal{A}(v))$ be the set of the remaining balls. In other words, $\mathcal{A}(e)$ contains the balls whose center is contained in e, except for B_u and B_v. We color $\mathcal{A}(e)$, possibly swapping colors in $\mathcal{A}(u)$ or $\mathcal{A}(v)$, as follows.

- If $B_u = B_v$, we first ensure that it gets the same color in both $\mathcal{A}(u)$ and $\mathcal{A}(v)$ by swapping colors in one of the two subsets if necessary. We then color all balls in $\mathcal{A}(e)$ blue if B_u is red, and red if B_u is blue.
- If $B_u \neq B_v$, let π be a longest simple path containing u and v. We color $\mathcal{A}(e) \cup \{B_u, B_v\}$ restricted to π using the non-monochromatic chain method. We then possibly swap colors in $\mathcal{A}(u)$ and $\mathcal{A}(v)$ so that B_u and B_v match the colors they were given by the chain method.

Both cases are illustrated in Fig. 6.

Theorem 3.2. $X^{\text{balls,trees}}_{\text{nm}}(t; n) = 2.$

Proof. The coloring obviously uses two colors. It remains to show it is non-monochromatic. We use induction on t. If $t = 0$, the coloring is non-monochromatic since it uses the chain method.

Suppose now that $t \geqslant 1$ and that the claim holds for any tree space with fewer than t internal nodes. Let p be a point contained in at least two balls.

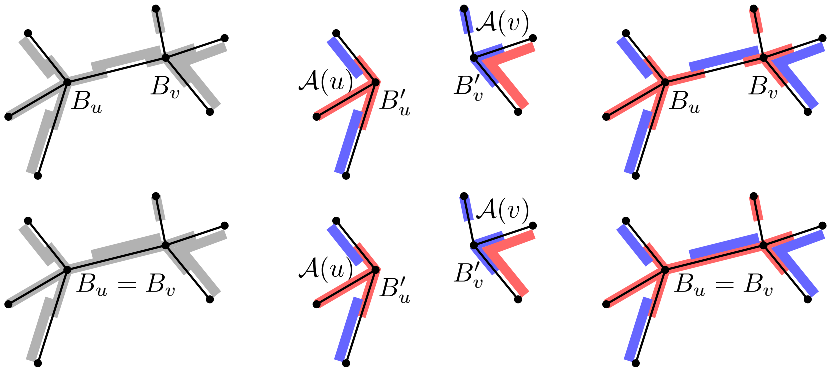

Figure 6: On the left, we have the two different initial cases, i.e., on the top, $B_u \neq B_v$, on the bottom, $B_u = B_v$. In the middle, the recursive call is made. On the right, we use the two recursive colorings and swap colors if needed.

If p is contained in balls only of $\mathcal{A}(v)$, only of $\mathcal{A}(u)$, or only of $\mathcal{A}(e)$, it is contained in at least two balls of different colors. Indeed, the colorings of $\mathcal{A}(v)$ and $\mathcal{A}(u)$ are non-monochromatic since they use the method on a tree with fewer than t internal nodes and we can use the induction hypothesis. Moreover $\mathcal{A}(e)$ is non-monochromatic due to the chain method.

It remains to consider the case where p is contained in balls from at least two of the sets $\mathcal{A}(u)$, $\mathcal{A}(v)$, and $\mathcal{A}(e)$. We distinguish two cases: p is contained in a ball of $\mathcal{A}(e)$ and p is not contained in a ball of $\mathcal{A}(e)$.

If p is contained in a ball B of $\mathcal{A}(e)$, we can assume without loss of generality that p is also contained in a ball of $\mathcal{A}(v)$. By Lemma 3.1(i), we have that $p \in B_v$.

If $B_u = B_v$ then all balls in $\mathcal{A}(e)$ are given a different color than B_v hence p is contained in two balls of different color. If $B_u \neq B_v$ then we use the chain method on π. Hence if $p \in \pi$, it is contained in two balls of different color. To show that if $p \notin \pi$ then p is still contained in two balls of different colors, it suffices to notice that for any subset of balls of $\mathcal{A}(e)$ in which p is contained, the point $p' \in \pi$ at distance $d(u,p)$ from u is contained in the same set of balls from $\mathcal{A}(e)$ as π is the longest path containing e.

On the other hand, if p is not contained in a ball of $\mathcal{A}(e)$, then it is contained in at least one ball from $\mathcal{A}(u)$ and one from $\mathcal{A}(v)$. By Lemma 3.1 we have that $p \in B_u \cap B_v$.

We then have two cases. If $B_u = B_v$, then p is contained in another ball of $\mathcal{A}(u)$ or $\mathcal{A}(v)$, and then the coloring is non-monochromatic by the induction hypothesis. Otherwise B_u and B_v are part of the chain $\mathcal{A}(e) \cup \{B_u, B_v\}$, and hence p is contained in at least two balls of different color. □

CF-coloring. The second algorithm CF-colors $\mathcal{A}$ using $\lceil \log t \rceil + 3$ colors. As before, define $\mathcal{C} := \{B = B(c,r) \mid \exists x : B = B_x\}$. We explain how to color $\mathcal{C}$ and then extend the coloring to $\mathcal{A}$. Let r be a node whose removal results in subtrees each of at most $t/2$ internal nodes. We color B_r (if it exists) with color 1. Let $\mathcal{T}_1, \ldots, \mathcal{T}_{deg(r)}$ be subtrees resulting from removing r, that is, the closures of the connected components of $\mathcal{T} \setminus \{r\}$. For each $i = 1, \ldots, deg(r)$, we recurse on $\mathcal{T}_i$ with the balls from $\mathcal{C}$ whose centers lie in $\mathcal{T}_i$. In such a recursive call, we consider a node to be an internal node when it was an internal node in the original space $\mathcal{T}$ and when it has not yet been selected as a splitting node in a previous call. Hence, when $t = 0$ in a recursive call on a subtree $\mathcal{T}' \subset \mathcal{T}$, then $\mathcal{T}'$ must be a single edge both of whose endpoints have already been treated.

The recursion stops when there are no more balls left (which must be the case when we have a recursive call with $t = 0$). Note that the internal nodes are fixed from the beginning, hence at some point of the recursion, a leaf node might still be considered internal for the purposes of the recursion.

Lemma 3.3. *The above algorithm CF-colors $\mathcal{C}$ using $\lceil \log t \rceil$ colors.*

Proof. The number of colors used comes immediately from the splitting of $\mathcal{T}$ into trees of at most $\frac{t}{2}$ internal nodes. We now show the coloring is indeed confict-free by showing that it is a unimin coloring: for any point p the minimum color among the colors of the balls containing p is unique. Let $p \in \mathcal{T}$ be a point contained in two balls $B_1 = B(p_1, r_1)$ and $B_2 = B(p_2, r_2)$ both of color i. We show that this implies the existence of a ball of higher color containing p. Let v_1 be the node B_1 is assigned to, and v_2 the node B_2 is assigned to. Since B_1 and B_2 have the color i, they were contained in different trees when they were colored in the recursive process. Let v_0 be the node that disconnected v_1 and v_2 and let B_0 be the ball assigned to v_0. Note that $c(B_0) < i$.

We prove that $p \in B_0$. Let π be the unique simple path between p and v_0. It cannot be the case that both $p_1 \in \pi$ and $p_2 \in \pi$. Suppose without loss of generality that $p_2 \notin \pi$. Let d be the distance between p and v_0. Since $p \in B_2$, we have that $cov_{v_0}(B_2) \geqslant d$. And since $cov_{v_0}(B_0) \geqslant cov_{v_0}(B_2)$, we have that $p \in B_0$, concluding the proof. □

We now wish to extend the coloring to balls in $\mathcal{A} \setminus \mathcal{C}$. To this end, define $\mathcal{T}' := \mathcal{T} \setminus (\bigcup \mathcal{C})$ to be the part of $\mathcal{T}$ that remains after removing all points covered by the balls in $\mathcal{C}$.

We finish the coloring with three more colors (using the chain method for CF-colorings) as explained next, resulting in $\lceil \log t \rceil + 3$ colors. We use the following lemma to show that the remaining balls can be reduced to intervals on disjoint lines. Note that it does not use tree spaces and can hence be applied also for planar network spaces.

Lemma 3.4. *For any ball $B \notin \mathcal{C}$, we have $\{p \in B \mid p \notin \cup\mathcal{C}\} \subseteq e$, where e is the edge containing the center of B.*

Proof. Suppose for a contradiction that there is a point $p \notin e$ contained in B but not in $\cup\mathcal{C}$. Consider the endpoint v of e belonging to the geodesic from the center of B to p. We claim that $cov_v(B) > cov_v(B_v)$, contradicting the definition of B_v. Indeed, $cov_v(B) > d(v, p)$ (since v lies on the geodesic from B's center to p) and $cov_v(B_v) < d(v, p)$ (since $p \notin \mathcal{C}$ and, hence, $p \notin B_v$). □

Theorem 3.5. $X_{\mathrm{cf}}^{\mathrm{tree,balls}}(t; n) \leqslant \lceil \log t \rceil + 3$.

3.2 Tree spaces: the lower bound

Lemma 3.6. $X_{\mathrm{cf}}^{\mathrm{tree,balls}}(t; n) \geqslant \lceil \log(t + 1) \rceil$.

Proof. Let $\mathcal{T}$ be as follows. We take $t + 2$ points $p_1, \ldots, p_{t+2}$ in the plane, with $p_i = (i, 0)$ for each $i = 1, \ldots, t + 2$, and we link consecutive points with a unit distance segment. We then take $t+2$ additional points $p'_1, \ldots, p'_{t+2}$, with $p'_i = (i, t+2)$, and for each $i = 1, \ldots, t+2$ we link p_i and p'_i with a segment of length $t + 2$. Note that p_1 and p_{t+2} do not count as internal nodes as their degree is two. Finally, we place $t + 1$ balls $B_1 = B(c_1, t + 2), \ldots, B_{t+1} = B(c_{t+1}, t + 2)$, for all $i = 1, \ldots, t + 1$, with $c_i = (i + \frac{2}{3}, 0)$, see Fig. 7. Consider the hypergraph $\mathcal{H}$ whose nodes are the balls B_i, and whose hyperedges are the subsets of balls such that there is a point $p \in \mathcal{T}$ contained in exactly that subset (and no other balls). We claim (and will prove below) that the set of hyperedges is exactly the set $\{\{B_i, B_{i+1}, \ldots, B_j\} \mid i \geqslant j\}$. In other words, there is a hyperedge for a subset of balls if and only if there is an interval on the x-axis containing exactly the centers of these balls. Hence, we can apply the $\lceil \log(t + 1) \rceil$ lower bound for CF-coloring points with respect to intervals [4].

To prove the claim, note that if p_i is the ball center nearest to p then $d(p, p_1) > d(p, p_2) > \cdots > d(p, p_i)$ and $d(p, p_{i+1}) > \cdots > d(p, p_{t+2})$, which implies that any hyperedge is of the form $\{B_i, B_{i+1}, \ldots, B_j\}$. On the other hand, the point $(\lfloor (j + i)/2 \rfloor, t + 2 - (j - i)/2)$ is contained in exactly the balls $B_i, B_{i+1}, \ldots, B_j$. □

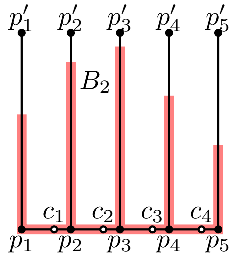

Figure 7: Example of the lower bound construction with $t = 3$. For clarity purposes, only B_2 is displayed, in red.

3.3 Planar network spaces

NM-coloring. We first explain how to NM-color balls on a planar network space $\mathcal{N}$. Let again $\mathcal{C}$ be the set $\{B = B(c, r) \mid \exists x : B = B_x\}$. We create a graph $\mathcal{G}_\mathcal{C}$ whose node set is $\mathcal{C}$ and whose edge set is defined as follows: there is an edge between B and B' if and only if there is an edge vv' in $\mathcal{T}$ with $B_v = B$ and $B_{v'} = B'$. It follows from Lemma 3.1 that for any ball B, the set of nodes of $\mathcal{N}$ to which B is assigned, together with the edges between these nodes, is a connected set. Therefore, $\mathcal{G}_\mathcal{C}$ is planar as well since its nodes correspond to disjoint connected subspaces in the planar space $\mathcal{N}$. We now use the Four Color Theorem to color $\mathcal{G}_\mathcal{C}$ and we give each ball in $\mathcal{C}$ the same color as the corresponding node in $\mathcal{G}_\mathcal{C}$.

Lemma 3.7. *The coloring on $\mathcal{C}$ is non-monochromatic and uses at most four colors.*

Proof. It is clear that the coloring uses at most four colors. Now let p be a point contained in two balls B_1 and B_2 of the same color. Let v_1 and v_2 be nodes of $\mathcal{N}$ with $B_1 = B_{v_1}$ and $B_2 = B_{v_2}$. Let π_1 and π_2 be two shortest paths between p and v_1, v_2, respectively. If all the nodes in $\pi_1 \cup \pi_2$ are either assigned B_1 or B_2, then there is an edge between B_1 and B_2 in $\mathcal{G}_\mathcal{C}$ and hence B_1 and B_2 are given different colors. Therefore there must be a node v in $\pi_1 \cup \pi_2$ (we assume without loss of generality that $v \in \pi_1$) with $B_v \notin \{B_1, B_2\}$ and $c(B_v) = c(B_1)$. Note that if $c(B_v) = c(B_1)$ for all $v \in \pi_1$, then there must be an edge between two balls of the same color in $\mathcal{G}_\mathcal{C}$ which is a contradiction, hence there must be a vertex $v \in \pi_1$ with $c(B_v) \neq c(B_1)$. Since π_1 is a shortest path between v_1 and p, and since $v \in \pi_1$, we have that π_1 contains a shortest path between v and p. Moreover, $cov_v(B_v) \geqslant cov_v(B_1) \geqslant d(v, p)$, which implies that $p \in B_v$ and concludes the proof. □

We now wish to extend the coloring to balls in $\mathcal{A} \setminus \mathcal{C}$. To this end, define $\mathcal{N}' := \mathcal{N} \setminus (\bigcup \mathcal{C})$ to be the part of $\mathcal{N}$ that remains after removing all points covered by the balls in $\mathcal{C}$. The proof of the following lemma is similar to the proof of Lemma 3.4.

Lemma 3.8. *Consider a ball $B \in \mathcal{A} \setminus \mathcal{C}$, and let $B' := B \cap \mathcal{N}'$. Then B' is contained in a single edge of $\mathcal{N}'$.*

For each edge e of $\mathcal{N}'$, let $\mathcal{A}(e)$ denote the set of balls contained in e. Let u and v denote the endpoints of the edge in $\mathcal{N}$ containing e. We color the uncolored balls in e using the chain method with two colors not equal to $c(B_u)$ and $c(B_v)$. We have now colored the balls in $\mathcal{C}$ as well as the balls in $\mathcal{A} \setminus \mathcal{C}$ that lie at least partially in $\mathcal{N}'$. Next we explain how to color the remaining balls, which are fully covered by the balls in $\mathcal{C}$.

Lemma 3.9. *Any uncolored ball is contained in the union of at most three balls.*

Proof. Any uncolored ball B is contained in $\cup\mathcal{C}$. If B is fully contained in a single edge e of $\mathcal{N}$, it must be covered by the two balls from $\mathcal{C}$ extending the farthest into e, starting from each of the two endpoints. If not, let v be a node contained in B. Now $B \setminus B_v$ is contained in a single edge e

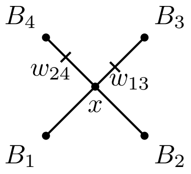

Figure 8: We suppose for a contradiction that the edges B_1B_3 and B_2B_4 cross. The crossing point is a node x of $\mathcal{N}$. Let w_{13} be the witness of the edge B_1B_3 and w_{24} the winess of B_2B_4.

of $\mathcal{N}$ and so $B \setminus B_v$ can be covered by two balls (as just explained), which implies that B can be covered by three balls. □

Using this lemma, we can easily finish the NM-coloring.

Theorem 3.10. $X_{\text{nm}}^{\text{planar,balls}}(t;n) = 4.$

Proof. The coloring obviously uses four colors at most. Moreover, it is easy to see the coloring is non-monochromatic. It remains to show that there is an instance requiring at least four colors. To that purpose, let $\mathcal{N}$ be an embedding of K_4 where all edges have length one. Then, for each node v of $\mathcal{N}$, we create the ball $B(v, 2/3)$. Since no two balls can have the same color, we need at least four colors. □

CF-coloring. We now explain how to CF-color balls on a planar network. As before, define $\mathcal{C} := \{B = B(c,r) \mid \exists x : B = B_x\}$. We first CF-color $\mathcal{C}$ using the following recursive algorithm introduced by Smorodinsky [9]: we select a maximum independent set in $C_1 := \mathcal{C}$, we give it color 1, place all uncolored balls in C_2, and recurse. We claim that for all i, the Delauney graph $D_i := (C_i, E_i)$ on the balls in C_i is planar, where $E_i := \{\{B_1, B_2\} \mid \exists p \in \mathcal{N} : p \in B_1 \cap B_2 \text{ and } \forall B \notin \{B_1, B_2\} : p \notin B\}$.

Lemma 3.11. *D_i is planar.*

Proof. We draw D_i using the drawing of $\mathcal{N}$ as follows: each ball is represented by its center. Then, for every edge in D_i, we find a witness, that is a point contained in the intersection of the two balls and not in any other ball. We finally draw the edge as two geodesics on $\mathcal{N}$: one from one endpoint to the witness point, and the other from the witness point to the other endpoint.

We claim that this drawing is plane. Suppose by contradiction that it is not the case and there is a crossing between the two edges B_1B_3 and B_2B_4. Suppose also that the endpoints of the two edges are distinct: the argument when an endpoint is shared is similar. Since we based our drawing on $\mathcal{N}$, a planar graph, the point where the two edges cross must be a node x in $\mathcal{N}$. Let w_{13} be the witness of the edge B_1B_3 and w_{24} the winess of B_2B_4. Fig. 8 shows the two crossing edges, with the crossing node x in the middle, and the two witnesses w_{13} and w_{24} used to draw the geodesics.

Suppose, without loss of generality, that the distance from x to w_{24} is greater than or equal to the distance from x to w_{13}. Thus, the distance from the center of B_1 to w_{24} is greater than or equal to w_{13}. Hence, w_{13} is also contained in the ball B_1, which contradicts the definition of a witness. Thus, the drawing is plane. □

Using this lemma and the Four Color Theorem, we get a coloring on $\mathcal{C}$ using $\lceil \log_{4/3} t \rceil$ colors. Note that this method does not give an efficient algorithm because of the use of the Four Color Theorem. For a fast algorithm, we can use a linear-time algorithm [3] to find an independent set of size at least $n/5$, leading to $\lceil \log_{5/4} t \rceil$ colors.

We then color the balls in $\mathcal{A} \setminus \mathcal{C}$. Using Lemma 3.4, we have that for any such ball B, the set of points contained in B but not in any ball in $\mathcal{C}$ is contained in one edge of $\mathcal{N}$. Therefore, if we cut $\cup\mathcal{C}$ out of $\mathcal{N}$, the remaining space is a union of disjoint segments, and any object that is not colored is contained in at most one segment. We can therefore use the chain coloring on each segment with the two additional colors and the dummy one.

Finally, any point in $\cup\mathcal{C}$ is contained in a ball in $\mathcal{C}$ of unique color, and any point not in $\cup\mathcal{C}$, is contained in at most one ball of each of the two additional colors. Therefore, the coloring is conflict-free. This yields the following theorem.

Theorem 3.12. $X_{\mathrm{cf}}^{\mathrm{planar,balls}}(t;n) \leqslant \lceil \log_{4/3} t \rceil + 3$.

4 Concluding Remarks

We studied NM- and CF-colorings on network spaces, where the objects to be colored are connected regions of the network space. We showed that the number of colors can be bounded as a function of the complexity (which depends on the type of space and of objects) of the network space and the objects, rather than on the number of objects. All our bounds are tight up to some constants, except for $X_{\mathrm{cf}}^{\mathrm{tree,trees}}(k,\ell;n)$ where the upper bound is a factor ℓ away from the lower bound. Closing this gap remains an open problem. It would also be interesting to find bounds on general connected objects on any network space, or other settings where the number of colors depends on the complexity of the space and objects rather the number of objects.

References

[1] Zachary Abel, Victor Alvarez, Erik D. Demaine, Sándor P. Fekete, Aman Gour, Adam Hesterberg, Phillip Keldenich, and Christian Scheffer. Three colors suffice: Conflict-free coloring of planar graphs. In *Proceedings of the 28th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1951–1963, 2017.

[2] Hans L. Bodlaender, Jitender S. Deogun, Klaus Jansen, Ton Kloks, Dieter Kratsch, Haiko Müller, and Zsolt Tuza. Ranking of graphs. In *Graph-Theoretic Concepts in Computer Science, 20th International Workshop, WG '94, Herrsching, Germany, June 16-18, 1994, Proceedings*, pages 292–304, 1994.

[3] Norishige Chiba, Takao Nishizeki, and Nobuji Saito. A linear 5-coloring algorithm of planar graphs. *J. Algorithms*, 2(4):317–327, 1981.

[4] Guy Even, Zvi Lotker, Dana Ron, and Shakhar Smorodinsky. Conflict-free colorings of simple geometric regions with applications to frequency assignment in cellular networks. *SIAM Journal on Computing*, 33(1):94–136, 2003.

[5] Sariel Har-Peled and Shakhar Smorodinsky. Conflict-free coloring of points and simple regions in the plane. *Discrete & Computational Geometry*, 34(1):47–70, 2005.

[6] Meir Katchalski, William McCuaig, and Suzanne M. Seager. Ordered colourings. *Discrete Mathematics*, 142(1-3):141–154, 1995.

[7] János Pach and Gábor Tardos. Conflict-free colourings of graphs and hypergraphs. *Combinatorics, Probability & Computing*, 18(5):819–834, 2009.

[8] Shakhar Smorodinsky. *Combinatorial Problems in Computational Geometry. PhD thesis.* PhD thesis, Tel-Aviv University, 2003.

[9] Shakhar Smorodinsky. On the chromatic number of some geometric hypergraphs. In *Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2006, Miami, Florida, USA, January 22-26, 2006*, pages 316–323, 2006.

[10] Shakhar Smorodinsky. Conflict-free coloring and its applications. *CoRR*, abs/1005.3616, 2010.