Linear Transformations Between Colorings in Chordal Graphs

Nicolas Bousquet
Univ. Grenoble Alpes, CNRS, Grenoble INP, G-SCOP, France
nicolas.bousquet@grenoble-inp.fr

Valentin Bartier
Univ. Grenoble Alpes, CNRS, Grenoble INP, G-SCOP, France
valentin.bartier@grenoble-inp.fr

Abstract

Let k and d be such that $k \geq d+2$. Consider two k-colorings of a d-degenerate graph G. Can we transform one into the other by recoloring one vertex at each step while maintaining a proper coloring at any step? Cereceda et al. answered that question in the affirmative, and exhibited a recolouring sequence of exponential length.

If $k = d+2$, we know that there exists graphs for which a quadratic number of recolorings is needed. And when $k = 2d+2$, there always exists a linear transformation. In this paper, we prove that, as long as $k \geq d+4$, there exists a transformation of length at most $f(\Delta) \cdot n$ between any pair of k-colorings of chordal graphs (where Δ denotes the maximum degree of the graph). The proof is constructive and provides a linear time algorithm that, given two k-colorings c_1, c_2 computes a linear transformation between c_1 and c_2.

2012 ACM Subject Classification Mathematics of computing → Graph theory

Keywords and phrases graph recoloring, chordal graphs

Digital Object Identifier 10.4230/LIPIcs.ESA.2019.24

Related Version A full version of the paper is available at `http://arxiv.org/abs/1907.01863`.

Funding This work was supported by ANR project GrR (ANR-18-CE40-0032).

1 Introduction

Reconfiguration problems consist in finding step-by-step transformations between two feasible solutions of a problem such that all intermediate states are also feasible. Such problems model dynamic situations where a given solution already in place has to be modified for a more desirable one while maintaining some properties throughout the transformation. Reconfiguration problems have been studied in various fields such as discrete geometry [6], optimization [1] or statistical physics [22] in order to transform, generate, or count solutions. In the last few years, graph reconfiguration received a considerable attention, e.g. reconfiguration of independent sets [3, 21], matchings [7, 20], dominating sets [24] or fixed-degree sequence graphs [9]. For a complete overview of the reconfiguration field, the reader is referred to the two recent surveys on the topic [23, 25].

Two main questions are at the core of combinatorial reconfiguration. (i) Is it possible to transform any solution into any other? (ii) If yes, how many steps are needed to perform this transformation? These two questions and their algorithmic counterparts received considerable attention.

27th Annual European Symposium on Algorithms (ESA 2019).
Editors: Michael A. Bender, Ola Svensson, and Grzegorz Herman; Article No. 24; pp. 24:1–24:15
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Graph recoloring

Throughout the paper, $G = (V, E)$ denotes a graph, $n = |V|$, Δ denotes the maximum degree of G, and k is an integer. For standard definitions and notations on graphs, we refer the reader to [15]. A *(proper) k-coloring* of G is a function $f : V(G) \to \{1, \ldots, k\}$ such that, for every edge $xy \in E$, we have $f(x) \neq f(y)$. Since we will only consider proper colorings, we will then omit the proper for brevity. The *chromatic number* $\chi(G)$ of a graph G is the smallest k such that G admits a k-coloring. Two k-colorings are *adjacent* if they differ on exactly one vertex. The *k-reconfiguration graph of G*, denoted by $\mathcal{G}(G, k)$ and defined for any $k \geq \chi(G)$, is the graph whose vertices are k-colorings of G, with the adjacency condition defined above. Cereceda, van den Heuvel and Johnson provided an algorithm to decide whether, given two 3-colorings, one can transform one into the other in polynomial time and characterized graphs for which $\mathcal{G}(G, 3)$ is connected [12, 13]. Given any two k-colorings of a graph, it is **PSPACE**-complete to decide whether one can be transformed into the other for $k \geq 4$ [5].

The *k-recoloring diameter* of a graph G is the diameter of $\mathcal{G}(G, k)$ if $\mathcal{G}(G, k)$ is connected and is $+\infty$ otherwise. In other words, it is the minimum D for which any k-coloring can be transformed into any other one through a sequence of at most D recolorings. Bonsma and Cereceda [5] proved that there exists a class $\mathcal{C}$ of graphs and an integer k such that, for every graph $G \in \mathcal{C}$, there exist two k-colorings whose distance in the k-reconfiguration graph is finite and super-polynomial in n.

A graph G is *d-degenerate* if any subgraph of G admits a vertex of degree at most d. In other words, there exists an ordering $v_1, \ldots, v_n$ of the vertices such that for every i, v_i has at most d neighbors in $v_{i+1}, \ldots, v_n$. It was shown independently by Dyer et al [16] and by Cereceda et al. [12] that for any d-degenerate graph G and every $k \geq d + 2$, $\mathcal{G}(G, k)$ is connected. Note that the bound on k is the best possible since the $\mathcal{G}(K_n, n)$ is not connected. Cereceda [11] conjectured the following:

▶ **Conjecture 1** (Cereceda [11]). *For every d, every $k \geq d + 2$, and every d-degenerate graph G, the diameter of $\mathcal{G}(G, k)$ is at most $C_d \cdot n^2$.*

If true, the quadratic function is the best possible, even for paths, as shown in [2]. Bousquet and Heinrich [8] recently proved that the diameter of $\mathcal{G}(G, k)$ is $O(n^{d+1})$. In the general case, Cereceda's conjecture is only known to be true for $d = 1$ (trees) [2] and $d = 2$ and $\Delta \leq 3$ [18]. The diameter of $\mathcal{G}(G, k)$ is $O(n^2)$ when $k \geq \frac{3}{2}(d + 1)$ as shown in [8]. Even if Cereceda's conjecture is widely open for general graphs, it has been proved for a few graph classes, e.g. chordal graphs [2], bounded treewidth graphs [4], and bipartite planar graphs [8].

Jerrum conjectured that if $k \geq \Delta + 2$, the mixing time (time needed to approach the stationary distribution) of the Markov chain of graph colorings[1] is $O(n \log n)$. So far, the conjecture has only been proved if $k \geq (\frac{11}{6} - \epsilon)\Delta$ [14]. Since the diameter of the reconfiguration graph is a lower bound of the mixing time, a lower bound on the diameter is of interest to study the mixing time of the Markov chain. In order to obtain such a mixing time, we need an (almost) linear diameter.

The diameter of $\mathcal{G}(G, k)$ is linear if $k \geq 2d + 2$ [10] or if k is at least the grundy number of G plus 1 [4]. When $k = d + 2$, the diameter of $\mathcal{G}(G, k)$ may be quadratic, even for paths [2]. But it might be true that the diameter of $\mathcal{G}(G, k)$ is linear whenever $k \geq d + 3$. In this paper, we investigate the following question, raised for instance in [8]: when does the k-recoloring diameter of d-degenerate graphs become linear?

[1] A random walk on $\mathcal{G}(G, k)$. Fore more details on Markov chains on graph colorings, the reader is for instance referred to [14].

Our results

A graph is *chordal* if it does not contain any induced cycle of length at least 4. Chordal graphs admit a *perfect elimination ordering*, i.e. there exists an ordering $v_1, \dots, v_n$ of V such that, for every i, $N[v_i] \cap \{v_{i+1}, \dots, v_n\}$ is a clique. Chordal graphs are $(\omega(G)-1)$-degenerate where $\omega(G)$ is the size of a maximum clique of G. Our main result is the following:

▶ **Theorem 2.** *Let Δ be a fixed integer. Let G be a d-degenerate chordal graph of maximum degree Δ. For every $k \geq d+4$, the diameter of $\mathcal{G}(G,k)$ is at most $O_\Delta(n)$. Moreover, given two colorings c_1, c_2 of G, a transformation of length at most $O_\Delta(n)$ can be found in linear time.*

Theorem 2 improves the best existing upper bound on the diameter of $\mathcal{G}(G,k)$ (where G is chordal) which was quadratic up to $k = 2d+1$ [10].

Note that the bound on k is almost the best possible since we know that this result cannot hold for $k \leq d+2$ [2]. So there is only one remaining case which is the case $k = d+3$.

▶ **Question 3.** *Is the diameter of $\mathcal{G}(G,d+3)$ at most $f(\Delta(G)) \cdot n$ for any d-degenerate graph G?*

In some very restricted cases (such as power of paths), our proof technique can be extended to $k = d+3$, but this is mainly due to the very strong structure of these graphs. For chordal graphs (or even interval graphs), we need at least $d+4$ colors at several steps of the proof and decreasing k to $d+3$ seems to be a challenging problem.

We also ask the following question: is it possible to remove the dependency on Δ to only obtain a dependency on the degeneracy? More formally:

▶ **Question 4.** *Is the diameter of $\mathcal{G}(G,d+3)$ at most $f(d) \cdot n$ for any d-degenerate chordal graph G?*

The best known result related to that question is the following: $\mathcal{G}(G,k)$ has linear diameter if $k \geq 2d+2$ (and f is a constant function) [10].

▶ **Question 5.** *Is the diameter of $\mathcal{G}(G,d+3)$ at most $f(\Delta(G)) \cdot n$ for any bounded treewidth graph G?*

Our proof cannot be immediately adapted for bounded treewidth graphs since we use the fact that all the vertices in a bag have distinct colors. Feghali [17] proposed a method to "chordalize" bounded treewidth graphs for recoloring problems. However his proof technique does not work here since it may increase the maximum degree of the graph. We nevertheless think that our proof technique can be adapted in order to study many well-structured graph classes.

Proof outline

In order to prove Theorem 2, we introduce a new proof technique to obtain linear diameters for recoloring graphs. The existing results (e.g. [10]) ensuring that $\mathcal{G}(G,k)$ has linear diameter are based on inductive proofs that completely fail when k is close to d. On the other hand, in the proofs giving quadratic diameters (e.g. [2]), the technique usually consists in finding two vertices that can be "identified" and then applying induction on the reduced graph. In that case, even if we can identify two vertices after a constant number of single vertex recolorings, we only obtain a quadratic diameter (since each vertex might "represent" a linear number of initial vertices). Both approaches are difficult to adapt to obtain linear transformations since they do not use or "forget" the original structure of the graph.

Let us roughly describe the idea of our method on interval graphs. A buffer $\mathcal{B}$ is a set of vertices contained in $f(\omega)$ consecutive cliques of the clique path. We assume that at the left of the buffer, the coloring of the graph already matches the target coloring. We moreover assume that the coloring of $\mathcal{B}$ is special in the sense that, for every vertex v in $\mathcal{B}$, at most $d+2$ colors appear in the neighborhood of v [2]. Note that in order to satisfy this property (and several others detailed in Section 2), the buffer has to be "long enough". The main technical part of the proof consists in showing that if the buffer is "long enough", then we can modify the colors of vertices of the buffer in such a way that the same assumptions hold for the buffer starting one clique to the right of the starting clique of $\mathcal{B}$. We simply have to repeat at most n times this operation to guarantee that the coloring of the whole graph is the target coloring. Since a vertex is recolored only if it is in the buffer and a vertex is in the buffer a constant (assuming Δ constant) number of times, every vertex is recolored a constant number of times.

The structure of this special coloring of the buffer, which is the main novelty of this paper, is described in Sections 2.2 to 2.4. We actually show that this graph recoloring problem can be rephrased into a "vectorial recoloring problem" (Section 2.5) which is easier to manipulate. And we finally prove that this vectorial recoloring problem can be solved by recoloring every element (and then every vertex of the graph) at most a constant number of times in Section 3.

2 Buffer and vectorial coloring

Throughout this section, $G = (V, E)$ is a chordal graph on n vertices of maximum clique number ω and maximum degree Δ. Let $k \geq \omega+3$ be the number of colors denoted by $1, \ldots, k$. Given two integers $x \leq y$, $[\![x, y]\!]$ is the set $\{x, x+1, \ldots, y\}$. The *closed neighbourhood* of a set $S \subseteq V$ is $N[S] := S \cup (\cup_{v \in S} N(v))$.

2.1 Chordal graphs and clique trees

Vertex ordering and canonical coloring

Let $v_1, v_2, \ldots v_n$ be a perfect elimination ordering of V. A greedy coloring of $v_n, v_{n-1}, \ldots,\ v_1$ gives an optimal coloring c_0 of G using only ω colors. The coloring c_0 is called the *canonical coloring of G*. The colors $c \in 1, 2 \ldots, \omega$ are called the *canonical colors* and the colors $c > \omega$ are called the *non-canonical colors*. Note that the independent sets $X_i := \{v \in V$ such that $c_0(v) = i\}$ for $i \leq \omega$, called the *classes* of G, partition the vertex set V.

Clique tree

Let $G = (V, E)$ be a chordal graph. A *clique tree* of G is a pair (T, B) where $T = (W, E')$ is a tree and B is a function that associates to each node U of T a subset of vertices B_U of V (called *bag*) such that: (i) every bag induces a clique, (ii) for every $x \in V$, the subset of nodes whose bags contain x forms a non-empty subtree in T, and (iii) for every edge $(U, W) \in T$, $B_U \setminus B_W$ and $B_W \setminus B_U$ are non empty. Note that the size of every bag is at most $\omega(G)$. A clique-tree of G can be found in linear time [19]. Throughout this section, $T = (V_T, E_T)$ is a clique tree of G. We assume that T is rooted on an arbitrary node. Given a rooted tree T and a node C of T, the *height* of C denoted by $h(C)$ is the length of the path from the root to C.

[2] Our condition is actually even more restrictive.

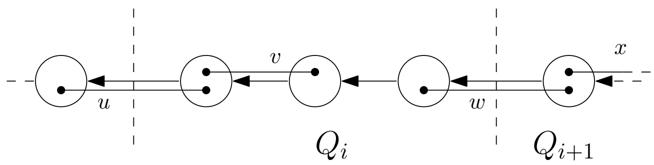

Figure 1 The nodes represent cliques of G. The vertices v and w belong to Q_i. The vertex u does not belong to Q_i even if it intersects cliques of Q_i, and the vertex x belongs to Q_{i+1}.

▶ **Observation 6.** *Let G be a chordal graph of maximum degree Δ and T be a clique tree of G rooted in an arbitrary node. Let x be a vertex of G and C_i, C_j be two bags of T that contain x. Then $h(C_j) - h(C_i) \leq \Delta$.*

Proof. We can assume without loss of generality (free to replace the one with the smallest height by the first common ancestor of C_i and C_j) that C_i is an ancestor of C_j (indeed, this operation can only increase the difference of height). Let P be the path of T between C_i and C_j and (U, W) be an edge of P with $h(U) < h(W)$. By assumption on the clique tree, there is a vertex y that appears in B_W and that does not appear in B_U. Since this property is true for every edge of T and since all the bags of P induce cliques and contain x, the vertex x has at least $|P|$ neighbors. ◀

2.2 Buffer, blocks and regions

Let C_e be a clique of T. We denote by T_{C_e} the subtree of T rooted in C_e and by $h_{C_e}(C)$ the height of the clique $C \in T_{C_e}$. Given a vertex $v \in T_{C_e}$, we say that v *starts* at height h if the maximum height of a clique of T_{C_e} containing v is h (in T_{C_e}).

Let $s := 3\binom{\omega}{2} + 2$ and $N = s + k - \omega + 1$ where k is the number of colors. The *buffer $\mathcal{B}$ rooted in C_e* is the set of vertices of G that start at height at most $3\Delta N - 1$ in T_{C_e}. For every $0 \leq i \leq 3N - 1$, the *block* Q_{3N-i} of $\mathcal{B}$ is the set of vertices of G that start at height h with $i\Delta \leq h \leq (i+1)\Delta - 1$. Finally, for $0 \leq i \leq N - 1$, the *region* R_i of $\mathcal{B}$ is the set of blocks $Q_{3i+1}, Q_{3i+2}, Q_{3(i+1)}$. Unless stated otherwise, we will always denote the three blocks of R_i by A_i, B_i and C_i, and the regions of a buffer $\mathcal{B}$ by $R_1, \ldots, R_N$. Given a color class X_p and $S \subseteq V$, we denote by $N[S, p]$ the set $N[S \cap X_p]$. By definition of a block and Observation 6 we have:

▶ **Observation 7.** *Let C_e be a clique of T and $\mathcal{B}$ be the buffer rooted in T_{C_e}. Let Q_{i-1}, Q_i, Q_{i+1} be three consecutive blocks of $\mathcal{B}$. Then $N[Q_i] \subseteq Q_{i-1} \cup Q_i \cup Q_{i+1}$. In particular for each region $R_i = (A_i, B_i, C_i)$ of $\mathcal{B}$, $N[B_i] \subseteq R_i$.*

Proof. Let v be a vertex of Q_i. By definition of Q_i, v starts at height h with $(3N - i)\Delta \leq h \leq (3N - i + 1)\Delta - 1$. Let u be a neighbour of v. By Observation 6, u starts at height h' with $h - \Delta \leq h' \leq h + \Delta$, thus we have $(3N - i - 1)\Delta \leq h' \leq (3N - i + 2)\Delta - 1$. It follows that u belongs either to Q_{i-1}, Q_i, or Q_{i+1}. ◀

We refer to this property as the *separation property*. It implies that when recoloring a vertex of Q_i, one only has to show that the coloring induced on Q_{i-1}, Q_i, Q_{i+1} remains proper.

2.3 Vectorial coloring

Let $\mathcal{B}$ be a buffer. We denote the set of vertices of class p that belong to the sequence of blocks $Q_i, \ldots, Q_j$ of $\mathcal{B}$ by $(Q_i, \ldots, Q_j, p)$. A *color vector* ν is a vector of size ω such that $\nu(p) \in [\![1, k]\!]$ for every $p \in [\![1, \omega]\!]$, and $\nu(p) \neq \nu(q)$ for every $p \neq q \leq \omega$. A block Q

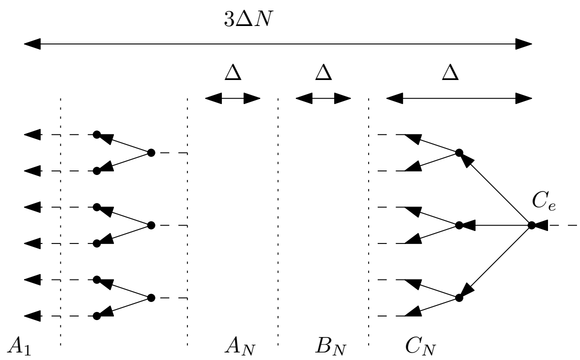

Figure 2 The buffer $\mathcal{B}$ rooted at C_e. The dots represent cliques of T and the dashed lines separates the blocks of $\mathcal{B}$.

is *well-colored for a color vector* ν_Q if all the vertices of (Q,p) are colored with $\nu_Q(p)$. It does not imply that all the colors are $\leq \omega$ but just that all the vertices of a same class have the same color (and vertices of different classes have different colors). For brevity, we say that (Q,ν_Q) is *well-colored* and when ν_Q is clear from context we just say that Q is *well-colored*. In particular, a well-colored block is properly colored. Since the set (Q,p) may be empty, a block may be well-colored for different vectors. However, a color vector defines a unique coloring of the vertices of a block (if (Q,ν_Q) is well-colored then every vertex of (Q,p) has to be colored with $\nu_Q(p)$). The color vector ν is *canonical* if $\nu(p)=p$ for every $p \leq \omega$. A sequence of blocks $Q_1,\dots,Q_r$ is *well-colored for* $(\nu_1,\dots,\nu_r)$ if (Q_i,ν_i) is well-colored for every $i \leq r$.

▶ **Definition 8** (Waiting region). *A region R well-colored for vectors ν_A,ν_B,ν_C is a waiting region if $\nu_A=\nu_B=\nu_C$.*

▶ **Definition 9** (Color region). *A region R well-colored for vectors ν_A,ν_B,ν_C is a color region if there exist a canonical color c_1, a non-canonical color z and a class p such that:*
1. *$\nu_A(m)=\nu_B(m)=\nu_C(m)\notin\{c_1,z\}$ for every $m\neq p$.*
2. *$\nu_A(p)=c_1$ and $\nu_B(p)=\nu_C(p)=z$.*

In other words, the color of exactly one class is modified from a canonical color to a non-canonical color between blocks A and C. We say that the color c_1 *disappears* in R and that the color z *appears* in R. For brevity we say that R is a color region for the class X_p and colors c_1,z.

▶ **Definition 10** (Transposition region). *A region R well-colored for vectors ν_A,ν_B,ν_C is a transposition region if there exist two canonical colors $c_1\neq c_2$, two non-canonical colors $z\neq z'$ and two distinct classes p,q such that:*
1. *$\nu_A(m)=\nu_B(m)=\nu_C(m)\notin\{c_1,c_2,z,z'\}$ and is canonical for every $m\notin\{p,q\}$.*
2. *$\nu_A(p)=c_1$, $\nu_B(p)=z$, $\nu_C(p)=c_2$.*
3. *$\nu_A(q)=c_2$, $\nu_B(q)=z'$, $\nu_C(q)=c_1$.*

Note that ν_A and ν_C only differ on the coordinates p and q which have been permuted. The colors z and z' are called the *temporary colors* of R. Note that the coloring induced on R is proper since the separation property ensures that $N[A\cap R]\subseteq A\cup B$, $N[C\cap R]\subseteq B\cap C$ and no class in B is colored with c_1 nor c_2.

Let ν be a color vector. The color vector ν' is obtained from ν by *swapping the coordinates* $p,\ell\leq\omega$ if for every $m\notin\{p,l\}$, $\nu'(m)=\nu(m)$, $\nu'(p)=\nu(\ell)$, and $\nu'(\ell)=\nu(p)$. In other words, ν' is the vector obtained from ν by permuting the coordinates p and ℓ.

Waiting region

	A_i	B_i	C_i
X_1	1	1	1
X_2	3	3	3
X_3	4	4	4
X_4	2	2	2

Color region

1	1	1
3	5	5
4	4	4
2	2	2

Transposition region

1	$\underline{5}$	3
3	$\underline{6}$	1
4	4	4
2	2	2

Figure 3 Example of waiting, color, and transposition regions with $\omega = 4$. Each square represents a region, the dotted lines separate the blocks and the dashed lines separate the classes. The colors 1 to 4 are the canonical colors and the colors 5 and 6 are non-canonical colors. The underlined colors in the transposition region indicate the temporary colors.

Swapping the coordinates p and ℓ in a region R well-colored for (ν_A, ν_B, ν_C) means that for every block $Q \in \{A, B, C\}$, we replace ν_Q by the color vector ν'_Q obtained by swapping the coordinates p and ℓ of ν_Q. It does not refer to a reconfiguration operation but just to an operation on the vectors.

▶ **Observation 11.** *Swapping two coordinates in a waiting (resp. color, resp. transposition) region leaves a waiting (resp. color, resp transposition) region.*

Using the following lemma, we can assume that all the transposition regions use the same temporary colors.

▶ **Lemma 12.** *Let R be a transposition region with temporary colors z, z'. Let $z'' \notin \{z, z'\}$ be a non-canonical color. By recoloring the vertices of R at most once, we can assume the temporary colors are z, z''.*

Proof. Let p and q be the coordinates which are permuted in R. By definition of transposition regions, no vertex of R is colored with z''. As any class is an independent set and by the separation property, we can recolor (B, q) with z'' to obtain the desired coloring of R. ◀

2.4 Valid and almost valid buffers

In what follows, a bold symbol $\boldsymbol{\nu}$ always denote a tuple of vectors and a normal symbol ν always denotes a vector. Let $\mathcal{B} = R_1, R_2, \ldots, R_N$ be a buffer such that all the regions $R_i = A_i, B_i, C_i$ are well-colored for the vectors $\nu_{A_i}, \nu_{B_i}, \nu_{C_i}$. So $\mathcal{B}$ is well-colored for $\boldsymbol{\nu} = (\nu_{A_1}, \nu_{B_1}, \nu_{C_1}, \nu_{A_2}, \ldots, \nu_{C_N})$. The buffer $(\mathcal{B}, \boldsymbol{\nu})$ is *valid* if:

1. [*Continuity property*] For every $i \in 1, 2, ..., N-1$, $\nu_{C_i} = \nu_{A_{i+1}}$.
2. The vectors ν_{A_1}, ν_{B_1} and ν_{C_1} are canonical (and then R_1 is a waiting region).
3. The regions $R_2, \ldots, R_{s-1}$ define a *transposition buffer*, that is a sequence of consecutive regions that are either waiting regions or transposition regions using the same temporary colors.
4. The regions $R_{s+1}, \ldots, R_{N-1}$ define a *color buffer*, that is a sequence of consecutive regions that are either color regions or waiting regions.
5. The regions R_s and R_N are waiting regions.

Note that Property 1 along with the definition of well-colored regions enforce "continuity" in the coloring of the buffer: the coloring of the last block of R_i and the first block of R_{i+1} in a valid buffer have to be the same.

An *almost valid buffer* $(\mathcal{B}, \boldsymbol{\nu})$ is a buffer that satisfies Properties 1 to 4 of a valid buffer and for which Property 5 is relaxed as follows:

5'. The region R_s is a transposition region or a waiting region. R_N is a waiting region.

Let us make a few observations.

▶ **Observation 13.** *Let $(\mathcal{B}, \boldsymbol{\nu})$ be an almost valid buffer. For every $i \leq s$, the color vectors ν_{A_i} and ν_{C_i} are permutations of the canonical colors.*

Proof. By induction on i. By Property 2 of almost valid buffers, it is true for every $i \leq r$. Suppose now that the property is verified for R_i with $2 < i < s$. By assumption ν_{C_i} is a permutation of $[\![1, \omega]\!]$ and the continuity property (Property 1 of almost valid buffer) ensures that $\nu_{A_{i+1}} = \nu_{C_i}$. By Property 3 we only have two cases to consider, either R_{i+1} is a waiting region and by definition $\nu_{c_{i+1}} = \nu_{A_{i+1}}$, or R_{i+1} is a transposition region. In the later case, by definition of a transposition region, $\nu_{C_{i+1}}$ is equal to $\nu_{A_{i+1}}$ up to a transposition of some classes k, ℓ and thus is a permutation of the canonical colors. ◀

▶ **Observation 14.** *Let $(\mathcal{B}, \boldsymbol{\nu})$ be an almost valid buffer and c be a non-canonical color. There exists a unique class $p \leq \omega$ such that $\nu_{C_s}(p) = c$. Furthermore, either the class p is colored with c on all the blocks of $R_{s+1}, \ldots, R_N$, or the color c disappears in a color region for the class p.*

Proof. By Observation 13, ν_{C_s} is a permutation of the canonical colors. Thus there exists a unique class $p \leq \omega$ such that $\nu_{C_s}(p) = c$. Furthermore, by Property 4 of almost valid buffer, the regions $R_{s+1}, \ldots, R_N$ are either waiting or color regions. The continuity property then ensures that either the class p is colored with c on $R_{s+1}, \ldots, R_N$ or that the color c disappears in a color region if there exists a color region for the class p. ◀

Since only non-canonical colors can appear in a color region, we have the following observation:

▶ **Observation 15.** *Let $(\mathcal{B}, \boldsymbol{\nu})$ be an almost valid buffer and z be a non-canonical color. Either no vertex of $R_{s+1}, \ldots, R_N$ is colored with z, or there exists a color region R_i with $s < i < N$ for the class p in which z appears. In the latter case, the vertices of the color buffer of $\mathcal{B}$ colored with z are exactly the vertices of $(B_i, C_i, \ldots, C_N, p)$.*

Finally, since the number of regions in the color buffer is the number of non-canonical colors, we have:

▶ **Observation 16.** *Let $(\mathcal{B}, \boldsymbol{\nu})$ be an almost valid buffer. There exists a waiting region in $R_{s+1}, \ldots, R_{N-1}$ if and only if there exists a non-canonical color that does not appear in $R_{s+1}, \ldots, R_{N-1}$.*

2.5 Vectorial recoloring

Let $(\mathcal{B}, \boldsymbol{\nu})$ be a buffer. The tuple of color vectors $\boldsymbol{\nu} = (\nu_{Q_1}, \ldots, \nu_{Q_{3\Delta N}})$ is a *(proper) vectorial coloring* of $\mathcal{B}$ if for every color c and every $i \leq 3\Delta N - 1$ such that c is in both ν_{Q_i} and $\nu_{Q_{i+1}}$, then there exists a unique class $p \leq \omega$ such that $\nu_{Q_i}(p) = \nu_{Q_{i+1}}(p) = c$.

▶ **Observation 17.** *Any proper vectorial coloring $(\mathcal{B}, \boldsymbol{\nu})$ induces a proper coloring of $G[\mathcal{B}]$.*

Proof. Indeed, two different classes in two consecutive blocks cannot have the same color in a proper vectorial coloring. Since for any block Q_i of $\mathcal{B}$ and for any class p, $N[Q_i, p] \subseteq Q_{i-1} \cup Q_i \cup Q_{i+1}$, the coloring induced on $G[\mathcal{B}]$ is proper. ◀

Note that if $(\mathcal{B}, \boldsymbol{\nu})$ is an almost valid buffer then $\boldsymbol{\nu}$ is a proper vectorial coloring of $\mathcal{B}$ by the continuity property and the definition of waiting, color, and transposition regions. Since we will only consider proper vectorial colorings, we will omit the term proper for brevity.

Let ν_Q be a color vector. A color vector ν'_Q is *adjacent to* ν_Q if there exist a class p and a color $c \notin \nu_Q$ such that $\nu'_Q(p) = c$ and $\nu'_Q(m) = \nu_Q(m)$ for every $m \neq p$.

▶ **Observation 18.** *Let Q be a block well-colored for ν_Q and let ν'_Q be a color vector adjacent to ν_Q such that $\nu'_Q(p) = c \neq \nu_Q(p)$. Then recoloring the vertices of (Q,p) one by one is a proper sequence of recolorings of $G[Q]$ after which Q is well-colored for ν'_Q.*

Let $(\boldsymbol{\nu}, \boldsymbol{\nu}')$ be two vectorial colorings of a buffer $\mathcal{B}$. The coloring $\boldsymbol{\nu}'$ is a *vectorial recoloring* of $\boldsymbol{\nu}$ if there exists a unique $i \in [\![1, 3\Delta N]\!]$ such that ν'_{Q_i} is adjacent to ν_{Q_i} and $\nu'_{Q_j} = \nu_{Q_j}$ for $j \neq i$. By Observation 18, we have:

▶ **Observation 19.** *Let $t \geq 1$ and $(\boldsymbol{\nu}^1, \boldsymbol{\nu}^t)$ be two (proper) vectorial colorings of a buffer $\mathcal{B}$. If there exists a sequence of adjacent (proper) vectorial recolorings $\boldsymbol{\nu}^1, \boldsymbol{\nu}^2, \ldots \boldsymbol{\nu}^t$, then there exists a sequence of (proper) single vertex recolorings of $G[\mathcal{B}]$ after which the coloring of $\mathcal{B}$ is well-colored for $\boldsymbol{\nu}^t$.*

Given a sequence of vectorial recolorings $\boldsymbol{\nu}^1, \boldsymbol{\nu}^2, \ldots \boldsymbol{\nu}^t$, we say that each coordinate is recolored at most ℓ times if for every coordinate $p \leq \omega$ and every $r \in [\![1, 3\Delta N]\!]$, there exist at most ℓ indices $t_1, \ldots, t_\ell$ such that the unique difference between $\boldsymbol{\nu}^{t_i}$ and $\boldsymbol{\nu}^{t_{i+1}}$ is the p-th coordinate of the r-th vector of the tuples.

3 Algorithm outline

Let G be a chordal graph of maximum degree Δ and maximum clique size ω, T be a clique tree of G, and ϕ be any k-coloring of G. We propose an iterative algorithm that recolors the vertices of the bags of T from the leaves to the root until we obtain the canonical coloring defined in Section 2.1. Let S be a clique of T. A coloring α of G is *treated up to S* if:

1. Vertices starting at height more than $3\Delta N$ in T_S are colored canonically, and
2. The buffer rooted at S is valid.

Let C be a clique of T. We associate a vector ν_C of length ω to the clique C as follows. We set $\nu_C(\ell) = \alpha(v)$ if there exists $v \in X_\ell \cap C$. Then we arbitrarily complete ν_C in such a way all the coordinates of ν_C are distinct (which is possible since $|\nu_C| < k$).

Given two vectors ν and ν' the *difference* $D(\nu, \nu')$ between ν and ν' is $|\{p : \nu(p) \neq \nu'(p)\}|$, i.e. the number of coordinates on which ν and ν' differ. Given an almost valid buffer $(\mathcal{B}, \boldsymbol{\nu})$ and a vector ν_C, the *border error* $D_{\mathcal{B}}(\nu_C, \boldsymbol{\nu})$ is $D(\nu_{C_N}, \nu_C)$.

Let $\mathcal{B}$ be a buffer. The class $p \leq \omega$ is *internal* to $\mathcal{B}$ if $N[R_N, p] \subseteq R_{N-1} \cup R_N$.

We first state the main technical lemmas of the paper with their proof outlines and finally explain how we can use them to derive Theorem 2. The complete proofs of the lemmas annotated with $(*)$ are included in the full version of the article (see related version).

▶ **Lemma 20 (*).** *Let C be a clique associated with ν_C. Let S be a child of C, $\mathcal{B}$ be the buffer rooted at S and $\boldsymbol{\nu}$ be a tuple of vectors such that $(\mathcal{B}, \boldsymbol{\nu})$ is valid. If $D_{\mathcal{B}}(\nu_C, \boldsymbol{\nu}) > 0$, then there exists a recoloring sequence of $\cup_{i=s}^N R_i$ such that the resulting coloring $\boldsymbol{\nu}'$ satisfies $D_{\mathcal{B}}(\nu_C, \boldsymbol{\nu}') < D_{\mathcal{B}}(\nu_C, \boldsymbol{\nu})$, and $(\mathcal{B}, \boldsymbol{\nu}')$ is almost valid.*
Moreover, every coordinate of $\cup_{i=s}^N R_i$ is recolored at most 3 times and only internal classes are recolored.

Outline of the proof. Let ℓ be a class on which ν_C and ν_{C_N} are distinct. Then, in particular, no vertex of X_ℓ is in $C_N \cap C$ thus the class ℓ is internal. Given an internal class ℓ, if we modify $\nu_{C_N}(\ell)$ and maintain a proper vectorial coloring of the buffer $\mathcal{B}$, then the corresponding recoloring of the graph is proper. So, if we only recolor internal classes of R_N, then we simply have to check that the vectorial coloring of $\mathcal{B}$ remains proper. The proof is then based on a case study depending on whether $\nu_C(\ell)$ is canonical or not. A more complete sketch is given in Section 3.1 ◀

▶ **Lemma 21 (*).** *Let $(\mathcal{B}, \boldsymbol{\nu})$ be an almost valid buffer. There exists a recoloring sequence of $\cup_{i=2}^{s} R_i$ such that every coordinate is recolored at most 6 times and the resulting coloring $\boldsymbol{\nu}'$ is such that $(\mathcal{B}, \boldsymbol{\nu}')$ is valid.*

Outline of the proof. The proof distinguishes two cases: either there exists a waiting region in the transposition buffer or not. In the first case, we show that we can "slide" the waiting regions to the right of the transposition buffer and then ensure that R_s is a waiting region. Otherwise, because of the size of the transposition buffer, then some pair of colors has to be permuted twice. In this case, we show that these two transpositions can be replaced by waiting regions (and we can apply the first case). A more complete sketch is given in Section 4. ◀

Note that given a clique C and its associated vector ν_C, applying Lemma 21 to an almost valid buffer $(\mathcal{B}, \boldsymbol{\nu})$ rooted at a child S of C does not modify $D_{\mathcal{B}}(\nu_C, \boldsymbol{\nu})$ since the region R_N is not recolored.

Let C be a clique and S_1, S_2 be two children of C. For every $i \leq 2$, let $\mathcal{B}_i$ be the buffer of S_i and assume that $\mathcal{B}_i$ is valid for $\boldsymbol{\nu}^i$. We say that $\mathcal{B}_1$ and $\mathcal{B}_2$ have *the same coloring* if $\boldsymbol{\nu}^1 = \boldsymbol{\nu}^2$.

▶ **Lemma 22 (*).** *Let C be a clique associated with ν_C. Let $S_1, S_2, \ldots S_e$ be the children of C, and for every $i \leq e$, $\mathcal{B}_i$ be the buffer rooted at S_i. Let $\boldsymbol{\nu}^i$ be a vectorial coloring such that $(\mathcal{B}_i, \boldsymbol{\nu}^i)$ is valid. If $D_{\mathcal{B}_i}(\nu_C, \boldsymbol{\nu}^i) = 0$ for every $i \leq e$, then there exists a recoloring sequence of $\cup_{j=2}^{N-1} R_j^i$ such that every coordinate is recolored $O(\omega^2)$ times, the final coloring of all the $\mathcal{B}_i$s is the same coloring $\boldsymbol{\nu}'$, $D_{\mathcal{B}_i}(\nu_c, \boldsymbol{\nu}') = 0$, and $(\mathcal{B}_i, \boldsymbol{\nu}')$ is valid for every $i \leq e$.*

Outline of the proof. First, we prove that it is possible to transform the coloring of $\mathcal{B}_i$ in such a way that all the color buffers have the same coloring, and that $\nu_s^1 = \nu_s^i$ for $i \in [\![2, e]\!]$. We then have to ensure that the vectors of the transposition buffers are the same, which is more complicated. Indeed, even if we know that the vectors ν_s^1 and ν_s^i are the same, we are not sure that we use the same sequence of transpositions in the transposition buffers of $\mathcal{B}_1$ and $\mathcal{B}_i$ to obtain it. Let $\tau_1, \ldots, \tau_r$ be the set of transpositions of $\mathcal{B}_1$. The proof consists in showing that we can add to $\mathcal{B}_i$ the transpositions $\tau_1, \ldots, \tau_r, \tau_r^{-1}, \ldots, \tau_1^{-1}$ at the beginning of the transposition buffer. It does not modify ν_{A_s} since this sequence of transpositions gives the identity. Finally, we prove that $\tau_r^{-1}, \ldots, \tau_1^{-1}$ can be cancelled with the already existing transpositions of $\mathcal{B}_i$. And then the transposition buffer of $\mathcal{B}_i$ only consists of $\tau_1, \ldots, \tau_r$. ◀

▶ **Lemma 23 (*).** *Let C be a clique of T with children $S_1, S_2, \ldots S_e$ and let α be a k-coloring of G treated up to S_i for every $i \in [\![1, e]\!]$. Let ν_C be a vector associated with C and $\mathcal{B}_i = R_1^i, \ldots, R_N^i$ denote the buffer rooted at S_i. Assume that there exists $\boldsymbol{\nu}$ such that $(\mathcal{B}_i, \boldsymbol{\nu})$ is valid and satisfies $D_{\mathcal{B}_i}(\nu_C, \boldsymbol{\nu}) = 0$ for every $i \leq e$. Then there exists a recoloring sequence of $\cup_{j=2}^{N-1} R_j^i$ such that, for every $i \leq e$, every vertex of $\mathcal{B}_i$ is recolored at most one time and such that the resulting coloring of G is treated up to C.*

Outline of the proof. This proof "only" consists in shifting the buffer of one level. We simply recolor the vertices that now start in another region (of the buffer rooted at C) with their new color. We prove that the recoloring algorithm cannot create any conflict. ◀

Given Lemmas 20, 21, 22 and 23 we can prove our main result:

▶ **Theorem 24.** *Let Δ be a fixed constant. Let $G(V,E)$ be a d-degenerate chordal graph of maximum degree Δ and ϕ be any k-coloring of G with $k \geq d+4$. Then we can recolor ϕ into the canonical coloring c_0 in at most $O(d^4\Delta \cdot n)$ steps. Moreover the recoloring algorithm runs in linear time.*

Proof. Let c_0 be the canonical coloring of G as defined in Section 2.1, and T be a clique tree of G. Let us first show that given a clique $C \in T$ with children $S_1, \dots, S_e$ and a coloring α treated up to S_i for every $i \leq e$, we can obtain a coloring of G treated up to C. Let ν_C be a vector associated with C. For every $i \leq e$, let $\mathcal{B}_i$ be the buffer rooted in S_i and $\boldsymbol{\nu}_i$ be a vectorial coloring of $\mathcal{B}_i$ such that $(\mathcal{B}_i, \boldsymbol{\nu}^i)$ is valid. For every $i \leq e$, by applying Lemmas 20 and 21 at most $D_{\mathcal{B}_i}(\nu_C, \boldsymbol{\nu}^i)$ times to $(\mathcal{B}_i, \boldsymbol{\nu}^i)$, we obtain a vectorial coloring $\boldsymbol{\nu}^i$ such that $(\mathcal{B}_i, \boldsymbol{\nu}^i)$ is valid and $D_{\mathcal{B}_i}(\nu_c, \boldsymbol{\nu}^i) = 0$. By Lemma 22, we can recolor each $\boldsymbol{\nu}^i$ into $\boldsymbol{\nu}'$ such that for every i, $(\mathcal{B}_i, \boldsymbol{\nu}')$ is valid and $D_{\mathcal{B}_i}(\nu_c, \boldsymbol{\nu}') = 0$. Then we can apply Lemma 23 to obtain a coloring of G such that the buffer $(\mathcal{B}, \boldsymbol{\nu})$ rooted in C is valid. Since no vertex starting in cliques $W \in T_C$ with $h_C(W) > 3\Delta N$ is recolored, these vertices remain canonically colored and the resulting coloring of G is treated up to C. Note that only vertices of T_C that start in cliques of height at most $3\Delta N$ are recolored at most $O(\omega^2)$ times to obtain a coloring treated up to C.

Let us now describe the recoloring algorithm and analyze its running time. We root T at an arbitrary node C_r and orient the tree from the root to the leaves. We then do a breadth-first-search starting at C_r and store the height of each node in a table h such that $h[i]$ contains all the nodes of T of height i. Let i_h be the height of T. We apply Lemmas 20 to 23 to every $C \in h[i]$ for i from i_h to 0. Let us show that after step i, the coloring of G is treated up to C for every $C \in h[i]$. It is true for $i = i_h$ since for any $C \in h[i_h]$ the sub-tree T_C of T only contains C. Suppose it is true for some $i > 0$ and let $C \in h[i-1]$. Let $S_1, \dots, S_e$ be the children of C. For all $j \in 1, \dots, e$, $S_j \in h[i]$ and by assumption the current coloring is treated up to S_j after step i. Thus we can apply Lemmas 20 to 23 to C. After iteration i_h we obtain a coloring of G that is treated up to C_r. Up to adding "artificial" vertices to G, we can assume that C_r is the only clique of T adjacent to a clique path of length $3\Delta N$ (in fact we only need a tuple of $3N$ color vectors) in T and apply Lemmas 20 to 23 until we obtain a coloring such that C_r is canonically colored, and the algorithm terminates. A clique tree of G can be computed in linear time [19], as well as building the table h via a breadth-first-search. Given a clique C, we can access to the cliques of the buffer rooted at T_C in constant time by computing their height and using the table h. Furthermore, a vertex of height i is recolored during the iterations $i+1, \dots, i+3\Delta N$ only. As each vertex is recolored at most $O(\omega^2)$ times at each iteration, it follows that the algorithm runs in linear time. Finally, as $N = 3\binom{\omega}{2} + k - \omega + 3$, each vertex is recolored at most $O(\omega^4\Delta)$ times, and thus the algorithm recolors ϕ to c_0 in at most $O(\omega^4\Delta \cdot n)$ steps. ◀

The proof of Theorem 2 immediately follows:

Proof of Theorem 2. Let ϕ and ψ be two k-colorings of G with $k \geq d+4$ and let c_0 be the canonical coloring of G defined in Section 2.1. By Theorem 24, there exists a recoloring sequence from ϕ (resp. ψ) to c_0 of length $O((d+1)^4\Delta \cdot n)$. Thus there exists a sequence of length $O_\Delta(n)$ that recolors ϕ to ψ. Furthermore the recoloring sequences from ϕ to c_0 and from ψ to c_0 can be found in linear time by Theorem 24, which concludes the proof. ◀

3.1 Proof of Lemma 20

Recall that in a buffer $\mathcal{B}$, R_s is the region that sits between the transposition buffer and the color buffer. Before proving Lemma 20, let us first start with some observations.

▶ **Observation 25.** *Let R be a well-colored region that does not contain color c. If we set $\nu_Q(p) = c$ for every block Q of R, then we obtain a waiting region if R was a waiting region or a color region for the class p, and we obtain a color region if R was a color region for a class $q \neq p$.*

▶ **Lemma 26 (*).** *Let $(\mathcal{B}, \boldsymbol{\nu})$ be an almost valid buffer and R_i, R_j be two regions of $\mathcal{B}$ with $1 \leq i \leq j$. Let X be a block in $\{B_i, C_i\}$ and let $Y = B_j$ or $Y = C_N$. Then recoloring $(X, \ldots, Y, p)$ with c preserves the continuity property.*

Given a color c, a class p, and a sequence of consecutive blocks $(Q_i, \ldots, Q_j)$, we say that $(Q_i, \ldots, Q_j, p)$ is *c-free* if no vertex of $N[\cup_{t=i}^{j} Q_t \cap X_p]$ is colored with c. Note that a sequence of (proper) vectorial recolorings of a buffer $(\mathcal{B}, \boldsymbol{\nu})$ that does not recolor A_1 and such that all the classes recolored on C_N are internal to $\mathcal{B}$ yields a (proper) sequence of single vertex recolorings of G by Observation 19. With the definition of clique-tree we can make the following observation:

▶ **Observation 27.** *Let C be a clique associated with ν_C, S be a child of C, and $(\mathcal{B}, \boldsymbol{\nu})$ be the buffer rooted at S. If $\nu_{C_N}(\ell) \neq \nu_C(\ell)$ for some $\ell \leq \omega$, then the class ℓ is internal to $\mathcal{B}$.*

Proof. Suppose that the class $\ell \leq \omega$ is not internal to $\mathcal{B}$. Then there exists a vertex $u \in X_\ell \cap R_N$ which has a neighbor v that does not belong to $R_{N-1} \cup R_N$. Thus u and v must be contained in C or in a clique C' that is an ancestor of C. In the latter case, Property (ii) of clique tree ensures that u must also be contained in C. Then, by definition of ν_C, it must be that $\nu_C(\ell) = \nu_{C_n}(\ell)$. ◀

We also need the following technical lemma:

▶ **Lemma 28.** *Let $(\mathcal{B}, \boldsymbol{\nu})$ be an almost valid buffer. Let $s < i < N$, c be a color and p be an internal class. If one of the following holds:*

1. *R_i is a waiting region, c is a non-canonical color that does not appear in $R_s, \ldots, R_N$ and the class p is not involved in a color region, or*
2. *R_i is a color region for the class p. Moreover c is non-canonical and does not appear in $R_s, \ldots, R_N$, or*
3. *R_i is a color region for the class p where c is the canonical color that disappears.*

Then changing the color of $(B_i, \ldots, C_N, p)$ by c also gives an almost valid buffer and a proper coloring of G.

We can now give the flavour of Lemma 20. For a complete proof, the reader is referred to the full version of the article.

Proof of Lemma 20. Let C be the clique associated with vector ν_C and S be a child of C. Let $(\mathcal{B}, \boldsymbol{\nu})$ be the valid buffer rooted at S. Assume that $D_{\mathcal{B}}(\nu_c, \boldsymbol{\nu}) > 0$. Then there exists $p \leq \omega$ such that $\nu_{C_N}(p) \neq \nu_C(p) := c$ and by Observation 27 the class p is internal to $\mathcal{B}$. The following sequences of recolorings only recolor blocks of $R_s, \ldots, R_N$ and all the recolorings fit in the framework of Lemma 26. Thus Properties 1, 2 and 3 of almost valid buffers are always satisfied. We then only have to check Properties 4 and 5' to conclude the proof. The proof is then based on a case distinction. Let us give a couple of simple cases to give an idea of the general proof:

Case 1. No class is colored with c on $R_s, \dots, R_N$ in $\boldsymbol{\nu}$.

Then c is not canonical since ν_{A_s} is a permutation of the canonical colors by Observation 13. Suppose first that there does not exist a color region for the class p in $\boldsymbol{\nu}$. Since c does not appear in the color buffer, Observation 16 ensures that there exists a waiting region R_i with $s < i < N$. Then by Lemma 28.1, we can recolor $(B_i, \dots, C_N, p)$ with c and obtain an almost valid buffer. Suppose otherwise that there exists a color region R_j for class p in $\boldsymbol{\nu}$. By Lemma 28.2, we can recolor $(B_j, \dots, C_N, p)$ with c and obtain an almost valid buffer. In both cases, the border error decreases.

Case 2. A class $\ell \neq p$ is colored with c, c is canonical, and c disappears in the color region R_j for class ℓ where the non-canonical color z appears (see Figure 4 for an illustration). Let $z' \neq z$ be a non-canonical color and let $c_1 = \nu_{A_s}(p)$. We apply the following recolorings:

(1) Recolor $(B_s, \dots, A_j, \ell)$ with z,

(2) Recolor (B_s, p) with z',

(3) Recolor $(C_s, \dots, C_N, p)$ with c,

(4) Recolor $(C_s, \dots, A_j, \ell)$ with c_1.

Recoloring 1 is proper since Observation 15 ensures that the only class colored with z in $R_s, \dots, R_N$ is the class ℓ on $(B_j, \dots, C_N)$. By the separation property, $(B_s, \dots, A_j, \ell)$ is z-free in $\boldsymbol{\nu}$. Recoloring 2 is proper as the only non-canonical color in R_s after recoloring 1 is $z \neq z'$. Recoloring 3 is proper as the class p is internal and thus after recoloring 1, $(C_s, \dots, C_N, p)$ is c-free by the separation property. Finally, recoloring 4 is proper as after recolorings 2 and 3, $(C_s, \dots, A_j, \ell)$ is c_1-free by the separation property.

Let us show that the resulting coloring defines an almost valid buffer. First note that regions $R_{j+1}, \dots, R_N$ are only modified by recoloring 3 and Observation 25 ensures they remain either waiting or color regions. In particular R_N remains a waiting region. Note that after recolorings 3 and 4, $\boldsymbol{\nu}'$ on regions $R_{s+1}, \dots, R_{j-1}$ is obtained from $\boldsymbol{\nu}$ by swapping coordinates p and ℓ (on these regions). Thus the nature of these regions is maintained by Observation 11. Since R_j was a color region for the class ℓ and colors c, z in $\boldsymbol{\nu}$, B_j, C_j are not modified and since $\nu'_{A_j}(\ell) = c_1$, R_j is a color region for class ℓ and colors c_1, z in $\boldsymbol{\nu}'$. So the regions $R_{s+1}, \dots, R_N$ remain either waiting or color regions. Furthermore no new color region is created and colors c_1, z are involved in exactly one color region thus Property 4 is satisfied. Finally, R_s is indeed a transposition region in $\boldsymbol{\nu}'$ since $\boldsymbol{\nu}$ is a valid buffer and z and z' are non-canonical colors, thus Property 5' holds. Furthermore, $\nu'_{C_N}(p) = c$ thus the border error has decreased. ◀

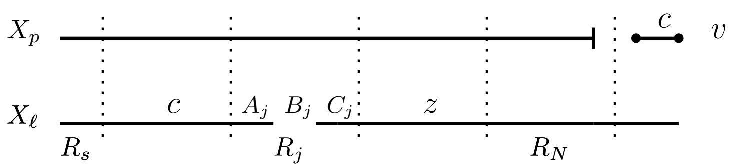

Figure 4 The initial coloring $\boldsymbol{\nu}$ for case 2 in the proof of Lemma 20. The rows represent the classes. A blank indicates a color region for the class. The vertical segment at the end of the buffer indicates that the class is internal. The dotted vertical lines separate the different regions.

4 Proof of Lemma 21

The proof distinguishes two cases:

Case 1. there is a region of the transposition buffer of $\mathcal{B}$ that is a waiting region.

The core of the proof consists in iteratively applying the following lemma:

▶ **Lemma 29 (*).** *Let $(\mathcal{B}, \boldsymbol{\nu})$ be an almost valid buffer and R_i, R_{i+1} be two consecutive regions with $1 < i < s$ such that R_i is a waiting region and R_{i+1} is a transposition region. Then there exists a recoloring sequence of $R_i \cup R_{i+1}$ such that, in the resulting coloring $\boldsymbol{\nu}'$, R_i is a transposition region, R_{i+1} is a waiting region, and $(\mathcal{B}, \boldsymbol{\nu}')$ is almost valid. Moreover only coordinates of $R_i \cup R_{i+1}$ are recolored at most twice.*

Case 2. All the regions of the transposition buffer of $\mathcal{B}$ are transposition regions.

As there are $3\binom{\omega}{2}$ regions in the transposition buffer and only $\binom{\omega}{2}$ distinct transpositions of $[\![1, \omega]\!]$, there must exist two distinct regions R_i and R_j with $1 < i < j < s$ for which the same pair of colors is permuted (note that the colors might be associated to different classes in R_i and R_j but it does not matter). The proof consists in applying the following lemma and then applying Case 1.

▶ **Lemma 30 (*).** *Let $(\mathcal{B}, \boldsymbol{\nu})$ be an almost valid buffer and c_1, c_2 be two canonical colors. If there exist two transposition regions R_i and R_j where colors c_1 and c_2 are transposed, then there exists a sequence of recolorings of $\cup_{t=i}^{j} R_t$ such that each coordinate is recolored at most twice, R_i and R_j are waiting regions in the resulting coloring $\boldsymbol{\nu}'$, and $(\mathcal{B}, \boldsymbol{\nu}')$ is almost valid.*

References

1 Hans-Joachim Böckenhauer, Elisabet Burjons, Martin Raszyk, and Peter Rossmanith. Reoptimization of Parameterized Problems. *arXiv e-prints*, 2018. `arXiv:1809.10578`.

2 M. Bonamy, M. Johnson, I. Lignos, V. Patel, and D. Paulusma. Reconfiguration graphs for vertex colourings of chordal and chordal bipartite graphs. *Journal of Combinatorial Optimization*, pages 1–12, 2012. `doi:10.1007/s10878-012-9490-y`.

3 Marthe Bonamy and Nicolas Bousquet. Token Sliding on Chordal Graphs. In *Graph-Theoretic Concepts in Computer Science - 43rd International Workshop, WG 2017*, pages 127–139, 2017.

4 Marthe Bonamy and Nicolas Bousquet. Recoloring graphs via tree decompositions. *Eur. J. Comb.*, 69:200–213, 2018.

5 P. Bonsma and L. Cereceda. Finding Paths Between Graph Colourings: PSPACE-Completeness and Superpolynomial Distances. In *MFCS*, volume 4708 of *Lecture Notes in Computer Science*, pages 738–749, 2007.

6 Prosenjit Bose, Anna Lubiw, Vinayak Pathak, and Sander Verdonschot. Flipping edge-labelled triangulations. *Comput. Geom.*, 68:309–326, 2018.

7 Nicolas Bousquet, Tatsuhiko Hatanaka, Takehiro Ito, and Moritz Mühlenthaler. Shortest Reconfiguration of Matchings. *CoRR*, abs/1812.05419, 2018. `arXiv:1812.05419`.

8 Nicolas Bousquet and Marc Heinrich. A polynomial version of Cereceda's conjecture. *arXiv e-prints*, 2019. `arXiv:1903.05619`.

9 Nicolas Bousquet and Arnaud Mary. Reconfiguration of Graphs with Connectivity Constraints. In *Approximation and Online Algorithms - 16th International Workshop, WAOA 2018*, pages 295–309, 2018. `doi:10.1007/978-3-030-04693-4_18`.

10 Nicolas Bousquet and Guillem Perarnau. Fast recoloring of sparse graphs. *Eur. J. Comb.*, 52:1–11, 2016.

11 L. Cereceda. *Mixing Graph Colourings*. PhD thesis, London School of Economics and Political Science, 2007.

12 L. Cereceda, J. van den Heuvel, and M. Johnson. Mixing 3-colourings in bipartite graphs. *Eur. J. Comb.*, 30(7):1593–1606, 2009. doi:10.1016/j.ejc.2009.03.011.

13 L. Cereceda, J. van den Heuvel, and M. Johnson. Finding paths between 3-colorings. *Journal of Graph Theory*, 67(1):69–82, 2011. doi:10.1002/jgt.20514.

14 Sitan Chen, Michelle Delcourt, Ankur Moitra, Guillem Perarnau, and Luke Postle. Improved Bounds for Randomly Sampling Colorings via Linear Programming. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019*, pages 2216–2234, 2019. doi:10.1137/1.9781611975482.134.

15 R. Diestel. *Graph Theory*, volume 173 of *Graduate Texts in Mathematics*. Springer-Verlag, Heidelberg, third edition, 2005.

16 M. Dyer, A. D. Flaxman, A. M Frieze, and E. Vigoda. Randomly coloring sparse random graphs with fewer colors than the maximum degree. *Random Structures & Algorithms*, 29(4):450–465, 2006.

17 Carl Feghali. Paths between colourings of graphs with bounded tree-width. *Information Processing Letters*, 144, December 2018. doi:10.1016/j.ipl.2018.12.006.

18 Carl Feghali, Matthew Johnson, and Daniël Paulusma. A Reconfigurations Analogue of Brooks' Theorem and Its Consequences. *Journal of Graph Theory*, 83(4):340–358, 2016.

19 Philippe Galinier, Michel Habib, and Christophe Paul. Chordal graphs and their clique graphs. In Manfred Nagl, editor, *Graph-Theoretic Concepts in Computer Science*, pages 358–371, Berlin, Heidelberg, 1995. Springer Berlin Heidelberg.

20 Takehiro Ito, Naonori Kakimura, Naoyuki Kamiyama, Yusuke Kobayashi, and Yoshio Okamoto. Reconfiguration of Maximum-Weight b-Matchings in a Graph. In *Computing and Combinatorics – 23rd International Conference, COCOON 2017, Hong Kong, China, August 3-5, 2017, Proceedings*, pages 287–296, 2017.

21 Daniel Lokshtanov and Amer E. Mouawad. The complexity of independent set reconfiguration on bipartite graphs. In *Proceedings of the Symposium on Discrete Algorithms, SODA 2018*, pages 185–195, 2018.

22 Bojan Mohar and Jesús Salas. On the non-ergodicity of the Swendsen–Wang–Kotecký algorithm on the Kagomé lattice. *Journal of Statistical Mechanics: Theory and Experiment*, 2010(05):P05016, 2010.

23 N. Nishimura. Introduction to reconfiguration. *preprint*, 2017.

24 A. Suzuki, A. Mouawad, and N. Nishimura. Reconfiguration of Dominating Sets. *CoRR*, 1401.5714, 2014. arXiv:1401.5714.

25 J. van den Heuvel. *The Complexity of change*, page 409. Part of London Mathematical Society Lecture Note Series. Cambridge University Press, S. R. Blackburn, S. Gerke, and M. Wildon edition, 2013.