Innov. Graph Theory **2**, 2025, pp. 119–156
https://doi.org/10.5802/igt.8

INNOVATIONS
IN GRAPH THEORY

SHORT AND LOCAL TRANSFORMATIONS BETWEEN $(\Delta + 1)$-COLORINGS

by Nicolas BOUSQUET, Laurent FEUILLOLEY,
Marc HEINRICH & Mikaël RABIE (*)

Abstract. — Recoloring a graph is about finding a sequence of proper colorings of this graph from an initial coloring σ to a target coloring η. Adding the constraint that each pair of consecutive colorings must differ on exactly one vertex, one asks: Is there a sequence of colorings from σ to η? If yes, how short can it be?

In this paper, we focus on $(\Delta + 1)$-colorings of graphs of maximum degree Δ. Feghali, Johnson and Paulusma proved that, if both colorings are unfrozen (i.e. if we can change the color of at least one vertex), then a recoloring sequence of length at most quadratic in the size of the graph always exists. We improve their result by proving that there actually exists a linear transformation (assuming that Δ is a constant).

In addition, we prove that the core of our algorithm can be performed locally. Informally, this means that after some preprocessing, the color changes that a given vertex has to perform only depend on the colors of the vertices in a constant size neighborhood. We make this precise by designing of an efficient recoloring algorithm in the LOCAL model of distributed computing.

1. Introduction

1.1. Graph Recoloring and configuration graph

A (proper) coloring of a graph is an assignment of colors to the vertices such that no two neighbors have the same color. Given two colorings of a given graph (referred to as the *source* and *target* colorings), we want to recolor one into the other, that is, start from the source coloring and change the color of one vertex at a time, in order to reach the target coloring, with the guarantee that the coloring is proper at all steps.

For a given graph G, and an integer k, the two classic recoloring questions are:

Keywords: Graph reconfiguration, recoloring, linear transformation, unfrozen colorings, distributed algorithms.
(*) This work was supported by ANR project GrR (ANR-18-CE40-0032).

Question 1.1. — *Is it possible to find a recoloring between any pair of k-colorings of G?*

Question 1.2. — *When it is possible, how many steps are needed?*

We can restate these questions from an alternative point of view using the *configuration graph*. For a given graph G and an integer k, the *k-configuration graph* $\mathcal{G}(G, k)$ has a vertex for each proper k-coloring of G, and an edge between every pair of colorings that differ on exactly one vertex of G. Finding a recoloring sequence between two colorings is then the same as finding a path in the configuration graph. From this point of view, Question 1.1 becomes: Is the configuration graph of k-colorings connected? and Question 1.2 becomes: What is the diameter of this configuration graph?

Typical behavior when the number of colors varies. For any given graph, the answer to the two questions above varies with the total number k of colors (the *palette size*). The typical behavior one can expect can be described by a series of regimes:

(1) With too few colors, proper colorings simply do not exist, hence we cannot discuss recoloring.
(2) With few colors, proper colorings exist, but for most pairs of colorings, recoloring is impossible. (That is, the configuration graph has many small connected components.)
(3) With a larger palette, we reach a regime where recoloring is feasible in general, but it can take many steps. (That is, the configuration graph is connected, or not far from being connected, but it has large diameter.)
(4) By increasing the number of colors, we make the recoloring sequences shorter and shorter.
(5) On the extreme, with a large enough number of colors, a recoloring process exists where every vertex has its color changed only a constant number of times.

1.2. Palette size as a function of the maximum degree

Many works are devoted to understand precisely when the changes of regimes occur for specific classes of graphs, or for different values of some graph parameters. The best-studied parameters are the degeneracy of the graph and the maximum degree. In this paper, we focus on graphs of maximum degree Δ, where the palette size is a function of Δ. This setting has attracted a lot of interest, not only in the graph theory community, but also in the random sampling and statistical physics communities. (We will mention some results from this perspective now, but more can be found in the Related Work section.)

Previous work has established that the most important change of regime happens between the palette sizes $\Delta + 1$ and $\Delta + 2$. For every $k \geqslant \Delta + 2$, the k-configuration graph is connected, that is, any coloring can be reached from any other by vertex recolorings. Moreover, the diameter of the configuration graph is at most $2n$ [13]. Note that the diameter of the configuration graph is always at least linear in n, since if we consider two colorings where colors are permuted, all the vertices have to be recolored at least once. An important conjecture in the area of random sampling is that the mixing time of the Markov chain of the $(\Delta + 2)$-colorings of any graph is $O(n \log n)$. In other words, given any $(\Delta + 2)$-coloring of a graph, if we perform a (lazy) random walk on the set of proper $(\Delta + 2)$-colorings, we should sample (almost) at random a coloring after $O(n \log n)$ steps.(1)

In contrast, for $(\Delta + 1)$ colors, the configuration graph is disconnected, in general. At first sight, this is a huge step for just one color less: we go from a case where we could navigate between colorings in the fastest way, even at random somehow, to a case where we cannot even reach some colorings from some others. The landscape is actually more subtle. Indeed, Feghali, Johnson, and Paulusma [23] proved that the configuration graph of the $(\Delta + 1)$-colorings of a graph consists of a set of isolated vertices plus a unique component containing all the other colorings. Note that isolated vertices in the configuration graph correspond to colorings that are *frozen*, in the sense that no change of colors can be performed. In addition, Bonamy, Bousquet, and Perarnau [5] proved that, if G is connected, then the proportion of frozen $(\Delta+1)$-colorings of G is exponentially smaller than the total number of colorings. Therefore, the configuration graph consists of a giant component of small diameter plus isolated vertices, and a long enough random walk can visit almost all $(\Delta + 1)$-colorings.

Now, the key question is: What is the diameter of the giant component? If it is small as a function of n, then we are in a situation close to the one of $\Delta + 2$ colors, with just some special cases which are frozen colorings. If it is large, then there is a real change of regime: not only are there some

(1) This question is still widely open, and the best known upper bound on the number of colors to obtain a polynomial mixing time is $(\frac{11}{6} - \epsilon)\Delta$ [18], slightly improving a classical result of Vigoda [32].

isolated vertices, but it is hard to navigate between the colorings of the giant component.

In their influential paper, Feghali, Johnson, and Paulusma [23] proved an upper bound of $O(n^2)$ on the diameter of the unique non-trivial component. That is, the configuration graph does not have a very large diameter, but we do not know whether it is as small as in the case of $\Delta+2$ colors. Actually, we know that $\Theta(n^2)$ is the best possible for paths, thanks to a specific lower bound construction for 3-colorings of paths [6]. But what about graphs that are not paths? For general graphs, the only known lower bound is the trivial $\Omega(n)$, leaving open whether there is something special about $\Delta = 2$ or whether the lower bound could be generalized. Our first theorem establishes that we are in the first situation: as soon as Δ is at least 3, the diameter of the non-trivial component drops to $\Theta(n)$ (for constant Δ).

Theorem 1.3. — *Let G be a connected graph with $\Delta \geqslant 3$ and σ, η be two unfrozen k-colorings of G with $k \geqslant \Delta + 1$. Then we can transform σ into η with a sequence of at most $O(\Delta^{c\Delta} n)$ single vertex recolorings, where c is a constant.*

In other words, we lower the upper bound on the diameter of the non-trivial component from $O(n^2)$ to $f(\Delta) \cdot n$. This brings $(\Delta + 1)$-colorings in the category of colorings for which there exists linear transformations, a topic that has received considerable attention in recent years (see Section 1.3).

An interesting direction for future work is to determine whether we can reduce the dependency in terms of Δ. We actually have no lower bound that ensures that a dependency on Δ is necessary. In other words, we leave the following question open:

Question 1.4. — *Given α, β two unfrozen $(\Delta + 1)$-colorings, is it possible to transform α into β in $O(n)$ steps independent of Δ?*

Finally, note that at some steps of the proof, we can reduce the exponential dependency on Δ into a polynomial one by adapting a result of Bousquet and Heinrich [11], but we did not succeed to do it at every step. We thus decided to keep the proof as simple as possible.

1.3. Configuration graphs of linear diameter

An active line of research consists in determining which number of colors ensures that the diameter of the configuration graph is linear, in various settings beyond the bounded degree case. In addition to the optimality, the focus on this regime is motivated by the fact that having a linear diameter is a necessary condition to get an almost linear mixing time for the underlying Markov chain.

Theorem 1.3 is a contribution in this line of research since we prove that the configuration graph consists of isolated vertices plus, possibly, a component of linear diameter. This is, as far as we know, the first result which provides a linear diameter on the components of the configuration graph while the configuration graph itself is not necessarily connected.

An important result in this perspective is by Bousquet and Perarnau [12] who proved that the diameter of the k-configuration graph is $O(dn)$ as long as $k \geqslant 2d + 2$, where d denotes the degeneracy of the graph. The proof of this result proceeds by induction, which is a classic approach in reconfiguration, but unusual for linear diameters, where one is often forced to use other techniques, such as discharging proofs [3, 11], Thomassen-like approaches [20], or buffer sliding [9].

Our proof of Theorem 1.3 introduces a new proof technique to ensure that the reconfiguration graph admits a linear diameter, which is of independent interest and could probably be used for other problems. The technique is related to the notions of parallelization and locality for reconfiguration, that we introduce in the next subsections. These have been studied recently by the distributed computing community, but as far as we know, had not been used in the more classic (sequential) reconfiguration world.

1.4. Parallel recoloring

Recoloring in parallel and dependencies between recoloring steps. Consider a recoloring instance where the source and the target colorings differ only on an independent set. In this instance, any sequence created by iteratively assigning its target color to a vertex that does not have it already, is a valid one. The order can be chosen arbitrarily, because there is no dependency between the color changes: a vertex does not need one of its neighbors to first change its color in order to be able to change its own.

A way to capture this absence of dependency is to note that we can parallelize the recoloring: we can just take all the vertices that do not have their target colors and recolor them in parallel. Note that we need to be careful with the notion of parallel recoloring: after all, for any recoloring task, we could just say “recolor all vertices in parallel”, but this would not make sense, since we simultaneously recolor adjacent vertices. To make it meaningful, the standard definition consists in allowing parallel recoloring only for vertices that are not adjacent.(2)

In other words, at any given step of a parallel recoloring schedule, the vertices that change color form an independent set. From such a parallel schedule, it is easy to derive a sequential recoloring sequence: decompose any parallel step by performing all the individual vertex recolorings one after the other.

Now, one might wonder if, in general, allowing parallel recoloring dramatically reduces the number of steps or not. Let us consider two examples with very different behaviors.

Paths with $\Delta + 1$ colors. Consider the case of paths with 3 colors (note that $\Delta = 2$ for paths). In particular, consider a source coloring of the form 1,2,3,1,2,3... and a target coloring of the form 2,3,1,2,3,1... In this case, it is easy to see that at step i, only the vertices at distance at most i from an endpoint can change color, even if we allow parallelization. Therefore, the recoloring must be very sequential, and we will use at least $\Omega(n)$ parallel steps. In other words, there are strong dependencies between color changes(3).

General graphs with $2\Delta+2$ colors. Now for $2\Delta+2$ colors, the behavior is completely different. We will illustrate this by designing an algorithm producing a very short parallel schedule. The algorithm is based on the idea of partitioning the palette of colors into two sub-palettes of size $\Delta+1$ each, *palette* A (with colors $a_1, ..., a_{\Delta+1}$) and *palette* B (with colors $b_1, ..., b_{\Delta+1}$). The key observation is: in any proper coloring, for any vertex with a color from palette A (resp. B), we can find a new non-conflicting color in palette B (resp. A), because the sub-palettes are large enough. See Algorithm 1.

Note that the color changes happening at the same step are performed by sets of vertices that are independent, since they are color classes either in the source or the target coloring.

This algorithm produces a parallel schedule that uses $4(\Delta+1)$ recoloring steps, which is very small when compared with the (at least) n steps that are necessary for sequential recoloring in general, and for the parallel recoloring of paths in the previous paragraph.

This naturally leads to the following third question.

---

(2) We will review the literature on distributed recoloring later in the paper.

(3) Actually in that case, one can prove that a recoloring sequence needs $\Theta(n^2)$ single vertex recolorings and that we can recolor it with $\Theta(n)$ parallel steps.

Algorithm 1 Generating a parallel schedule for $2\Delta+2$ colors.

for i from 1 to $\Delta+1$ **do**
At Step i: every vertex with source color b_i takes a new non-conflicting color in palette A.
for i from 1 to $\Delta+1$ **do**
At Step $i+\Delta+1$: every vertex with target color b_i takes its target color.
for i from 1 to $\Delta+1$ **do**
At Step $i+2\Delta+2$: every vertex with target color a_i takes a new non-conflicting color in palette B.
for i from 1 to $\Delta+1$ **do**
At Step $i+3\Delta+3$: every vertex with target color a_i takes its target color.

QUESTION 1.5. — *When recoloring is possible, how many* parallel steps *are needed?*

Parallel recoloring with $(\Delta+1)$ **colors.** When expressed as a parallel schedule, the sequence of Feghali, Johnson, and Paulusma [23] takes $O(n)$ parallel steps, and this cannot be improved. Indeed, consider the power of a path (which can be chosen of arbitrarily large degree): we can have an almost frozen coloring except on the boundaries (as in the case of paths with 3 colors), therefore, in order to recolor a vertex in the middle, we first have to recolor a long chain of vertices starting on the border of the graph.

We prove that we can have a parallel recoloring schedule with a very specific form: first a sequential schedule of linear length and then a very short schedule using parallelism.

THEOREM 1.6 (A more precise version of Theorem 1.3). — *There exists a function f, such that for any connected graph of maximum degree $\Delta \geqslant 3$ and $k \geqslant \Delta+1$, we can transform any unfrozen k-coloring into any other with a sequence of:*

- *at most $O(n)$ single vertex recolorings followed by,*
- *a parallel schedule of length at most $O(f(\Delta))$.*

Actually our result is even better since we can ensure that, if, in the initial and target colorings, for each vertex there is an unfrozen vertex close enough, we can simply remove the first part and only recolor with a parallel schedule of length at most $O(f(\Delta))$.

1.5. Distributed algorithms and locality

From parallel recoloring to locality. When studying Algorithm 1, it appears that not only does it produce a short parallel schedule, but it is also very *local*. Let us explain what we mean by this. Let the recoloring schedule of a given vertex be the series of color changes it has to take, each along with the appropriate time step. We claim that the recoloring schedule of any fixed vertex is independent of the vertices that are outside a ball of radius $4(\Delta+1)$ around it. That is, changing the source/target colors of vertices far away, or even the topology of the graph far from the vertex would not change anything from the viewpoint of the vertex.

Let us quickly prove this claim. We prove by induction that: for every vertex v, for every $j \in [1, 4(\Delta+1)]$, the possible color change of v at step j only depends on the ball of radius i around v (including source and target colors in this ball). At step $j = 1$, vertices with source color b_1 change to a non-conflicting color in palette A. This only depends on the source colors of the neighbors of v, that is, on the ball of radius 1 around v. Assume now that the hypothesis is true for some $j < 4(\Delta+1)$, and consider a change of color at step $j+1$ on a vertex v. If $j+1$ is in $[\Delta+2, 2\Delta+2]$ or $[3\Delta+4, 4\Delta+4]$, the hypothesis holds, because the color change only depends on the target color of v. Otherwise, the color change depends on the current colors of the neighbors of v (which do not change at step $j+1$, by construction). Hence, the color change of v depends on the vertices that are in the balls of radius j around its neighbors, that is, in the ball of radius $j+1$ around itself. This proves the claim.

Locality and LOCAL model. Notions of locality with the flavor described above have been studied for a long time in the theory of distributed computing, in what is called the *LOCAL model*. There, a typical question is: if we let every vertex know its neighborhood at distance ℓ, its initial and target colors, can it choose an output such that the collection of individual outputs makes sense globally? In our setting, this translates to: if every vertex knows its neighborhood at distance ℓ, can it produce a recoloring schedule for itself, such that we get a consistent parallel schedule when considering all the vertices together? This distance ℓ corresponds to the *number of rounds* in the LOCAL model, and can be called the *locality* of the task. Hence, we have the following question:

QUESTION 1.7. — *What is the locality of recoloring?*

Our example of $2\Delta+2$ colors was useful to introduce both the notions of parallel schedule length and of locality, but it can be misleading because the two are basically equal in that case. This is because, at each step, a vertex can check the colors of its neighbors and update its own, performing both the computation of the schedule and the application of it. In general, there is no such equality. It can be that the locality is larger, because the vertices need to look far to be able to produce a proper schedule (in particular, with no two adjacent vertices changing color at the same time). It can also be that the schedule is larger, for example it can be larger than n (when the recoloring sequence is super polynomial for instance) whereas the locality can never be larger than n.

We will come back to this when defining the LOCAL model properly in Section 2.

Back to Δ+1 colors. In general, when we consider two unfrozen $(\Delta+1)$-colorings σ and η of a graph G, possibly there exists no short parallel schedule recoloring σ to η. For example, if σ has a unique unfrozen vertex v whereas all vertices are unfrozen in η, the "non-frozenness" has to be propagated edge by edge to the rest of the graph, similarly as for 3-colored paths. Moreover, to compute its own schedule, a vertex would need to know its distance to the unfrozen vertex in σ, inducing a linear number of rounds for the furthest vertex from v. We prove that this is basically the only case where the transformations between $(\Delta+1)$-colorings have to be global. That is, if unfrozen vertices are well-spread, then we can actually compute a recoloring sequence locally. Our second main theorem is the following:

Theorem 1.8. — *Let $k, \Delta, r \in \mathbb{N}$ such that $k > \Delta \geqslant 3$. There exists three constants c, c', c'' such that, for any graph G of bounded degree Δ, and σ, η two k-colorings of G which are r-locally unfrozen*[(4)]*, we can transform σ into η with a parallel schedule of length at most $O(k^{c\Delta}+\Delta^{c'}r)$. Moreover, this schedule can be computed in:*

- *$O(\Delta^{c''}+\log^* n+r)$ rounds if $k \geqslant \Delta+2$.*
- *$O(\Delta^{c''}+\log^2 n\cdot\log^2\Delta+r)$ rounds if $k=\Delta+1$.*

Informally, the number of rounds we need in the LOCAL model to provide a distributed recoloring sequence measures how well we need to understand the graph globally to provide a recoloring sequence.

The $\log^* n$ (or $\log^2 n$) term in the number of rounds arises from computing a maximal independent set at distance $\Omega(1)$ (or a $\Delta+1$ coloring). If we are given such colorings and independent sets, then the number of rounds is independent of n.

(4) For a formal definition of LOCAL model and of r-locally unfrozen colorings, the reader is referred to Definition 2.1.

Impact and open questions for distributed recoloring. Our results are also interesting from the viewpoint of distributed computing, since they improve on the state of the art of distributed recoloring in several ways. Theorem 1.8 directly improves some results of [7] on distributed recoloring. One problem studied in [7] consists in recoloring 3-colored graphs of maximum degree 3 with the help of an extra color. The authors provide an algorithm that finds a parallel schedule of length $O(\log n)$ in a polylogarithmic number of rounds in the LOCAL model. Theorem 1.8 implies that a constant length schedule can be found in $O(\log^* n)$ rounds (and it holds even if we start from an arbitrary locally unfrozen 4-colorings instead of 3-colorings plus an additional color). Theorem 1.8 also directly solves two open questions from [7]:

- The first question is about the complexity of finding a schedule to recolor a Δ-coloring with an extra color. Since these colorings can be seen as unfrozen $(\Delta+1)$-colorings, Theorem 1.8 gives an algorithm that finds a parallel schedule of length $f(\Delta)$ in $O(F(\Delta)\log^* n)$ communication rounds, where f and F are functions with no hidden dependencies in n.
- The second question deals with the case of 4-colored toroidal grids with an extra color. We provide an algorithm with a constant length schedule after $O(\log^* n)$ rounds.

We leave as an open problem whether a schedule can be found even more quickly. In particular, we conjecture that, in the case of toroidal grids, such a schedule could be found in $O(1)$ communication rounds, by using the input and target colorings as symmetry-breaking tools. More generally, we were not able to answer that question:

Question 1.9. — *Is it the case that computing a recoloring schedule in the LOCAL model between any pair of* 28*-locally unfrozen* $(\Delta+1)$*-colorings requires* $\omega(1)$ *communication rounds?*

Note that a lower bound result of this flavor can be found in [15] for the problem of maximal independent set reconfiguration, but we did not manage to adapt it to our setting.

1.6. Related work

In this section, we focus on recoloring literature. For references about the larger field of reconfiguration, the reader is referred to the two recent surveys on the topic [28, 27].

Markov chain motivation. A major motivation to study the configuration graph of colorings is the importance of this object for random sampling. The diameter of the configuration graph is a straightforward lower bound on the mixing time of the underlying Markov chain, which corresponds to sampling colorings by local changes. Since proper colorings correspond to states of the anti-ferromagnetic Potts model at zero temperature, Markov chains related to graph colorings have received considerable attention in statistical physics and many questions related to the ergodicity or the mixing time of these chains remain widely open (see e.g. [18, 24]).

Recoloring graphs with other bounded parameters. So far we have considered graphs where the degree is bounded, since it is the setting of our results. Let us quickly mention results in classes where other parameters are bounded. Bonsma and Cereceda [8] proved that there exists a family $\mathcal{G}$ of graphs and an integer k such that, for every graph $G \in \mathcal{G}$, there exist two k-colorings whose distance in the k-configuration graph is finite and super-polynomial in n. Cereceda conjectured that the situation is different for degenerate graphs. A graph G is *d-degenerate* if any subgraph of G contains 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 $i \leqslant n$, the vertex v_i has at most d neighbors in $v_{i+1}, \ldots, v_n$. It was shown independently in [21] and [17] that for any d-degenerate graph G and every $k \geqslant d+2$, the graph $\mathcal{G}(G,k)$ is connected. However, the (upper) bound on the k-recoloring diameter given by these constructive proofs is $O(c^n)$ (where c is a constant). Cereceda [16] conjectured that the diameter of $\mathcal{G}(G,k)$ is $\mathcal{O}(n^2)$, as long as $k \geqslant d+2$. If correct, the quadratic function is tight, even for paths or chordal graphs as proved in [6]. The best known upper bound here is due to Bousquet and Heinrich [11], who proved that the diameter of $\mathcal{G}(G,k)$ is in $O(n^{d+1})$. The conjecture is known to be true for a few graph classes, such as chordal graphs [6] and bounded treewidth graphs [4, 22].

Distributed recoloring. Distributed recoloring in the LOCAL model was introduced in [7], and implicitly studied before in [29]. In [7], the authors focus on recoloring 3-colored trees, subcubic graphs and toroidal grids, and in [29], the focus is on transforming a $(\Delta + 1)$-coloring into a Δ-coloring. More recently, [10] designed efficient distributed algorithms for recoloring chordal and interval graphs. A few reconfiguration problems different from coloring have been studied in the distributed setting, including vertex cover [14], maximal independent sets [15], and spanning trees [26].

1.7. Organization of the paper

This introduction described the motivation and big picture. In Section 2, we give the definitions needed in the rest of the paper. In Section 3, we sketch the proof techniques. Sections 4 and 5 provide the full proofs of the results.

2. Preliminaries

Classic graph definitions. All along the paper $G = (V, E)$ denotes a graph, n is the number of vertices (i.e. $n = |V|$), and k is a positive integer. For standard definitions and notations on graphs, we refer the reader to [19]. Let G be a graph and v be a vertex of G. We denote by $N(v)$ the set of *neighbors* of v, that is the set of vertices adjacent to v. The *degree* of a vertex is the number of neighbors it has, and $\Delta(G)$ is the maximum degree of G (often denoted Δ for short). The set $N[v]$, called the *closed neighborhood of* v, denotes the set $N(v) \cup \{v\}$. Given a set X, we denote by $N(X)$, the set $(\cup_{v \in X} N(v)) \setminus X$. The *distance* between u and v in G is the length of a shortest path from u to v in G (by convention, it is $+\infty$ if no such path exists), and it is denoted by $d(u, v)$. For any integer $r \in \mathbb{N}$, we denote by $B(v, r)$ the *ball* of center v and radius r, which is the set of vertices at distance at most r from v. A vertex w belongs to the *boundary* of $B(v, r)$ if the distance between v and w is exactly r. The *interior* of a ball B is the ball minus its boundary (*i.e.* $B(v, r-1)$ for a ball $B(v, r)$, with $r > 0$). An *independent set at distance* d is a set of vertices at pairwise distance at least d.

Recoloring definitions. Let c be a coloring of G. A vertex v is *frozen* in c if all the colors appear in $N[v]$. The coloring c is *frozen* if all the vertices are frozen. Note that a frozen coloring is an isolated vertex of the configuration graph.

Let α be a coloring of G, and X be a subset of vertices. We denote by $G[X]$ the subgraph of G induced by X, and by α_X the coloring α restricted to the vertices of X. We say that two colorings α and β *agree on* X if $\alpha_X = \beta_X$.

A *recoloring step* consists in changing the color of an unfrozen vertex to one that does not appear in its neighborhood. In a *recoloring by independent sets*, instead of changing the color of one vertex at each step, we are allowed to change the colors of an independent set of unfrozen vertices (while keeping a proper coloring).

We introduce two new definitions: *r-locally unfrozen colorings* and *ladders*.

DEFINITION 2.1. — *A coloring is* r-locally unfrozen *if, for every vertex* v*, there exists an unfrozen vertex at distance at most* r *from* v*.*

The last definition we introduce is motivated by the following remark.

Remark 2.2. — Consider an unfrozen vertex u in a $(\Delta+1)$-coloring of a graph. If we change the color of u, then all its frozen neighbors become unfrozen.

Indeed, before the change, for any frozen neighbor v of u, all the colors appear exactly once in $N[v]$ (because we consider $\Delta+1$ colors). Thus, after the change, the old color c of u does not appear anymore in $N[v]$, and v has two possible colors: its current color and c. Now, let us go one step further. Suppose that v had another neighbor z, not adjacent to u, that was also frozen at the beginning. The recoloring of u keeps z frozen, but then the recoloring of v with color c unfreezes it. By iterating this process, we get what we call a *ladder*.

DEFINITION 2.3. — *Given an induced path* P *where the first vertex in the path is unfrozen, and all the other vertices are frozen,* a ladder *is a portion of a recoloring sequence that recolors all the vertices of* P *one by one.*

Let u and w be the two endpoints of the path, u being the unfrozen vertex. Note that at the end of the sequence, vertex w has changed color, and it is unfrozen. Moreover, for every consecutive pair of vertices $v_i v_{i+1}$ in the path, where v_i appears first between u and w, the final color of v_{i+1} is the initial color of v_i.

LOCAL model. The LOCAL model is a classic model of distributed computing (see the books and surveys [1, 30, 31]).

DEFINITION 2.4. — *A* distributed algorithmic problem *in a graph class* $\mathcal{C}$*, consists, for every graph of* $G \in \mathcal{C}$*, in a list of correct input-output configurations: a pair of functions that assign a bit string to any vertex.*

For example, the task of distributed k-coloring in general graphs consists in the list of all the graphs, along with the input-output configurations, where the first function assigns an empty string to every vertex (because there is no input), and the second function corresponds to a proper k-coloring of the graph.

Given a distributed algorithmic problem, a *legal input configuration* is a function f, for which there exists a function g, such that (f, g) is a correct input-output configuration.

In order to avoid symmetry issues in distributed algorithms, we will assume that the vertices have names. More formally:

Definition 2.5. — *A graph on n vertices is equipped with* unique identifiers *if every vertex holds a distinct integer in* $[1, n^2]$.

In the paper, this will only appear implicitly, since we will use known distributed algorithms as black box, only those will require unique identifiers.

Definition 2.6. — *An algorithmic problem can be solved in $r(n)$ rounds in the LOCAL model in graphs equipped with unique identifiers if the following two equivalent conditions hold. On any graph of the relevant class with a legal input configuration:*

(1) (Computational definition) If we suppose that the vertices start with only the knowledge of their identifier, and can send messages to their neighbors in synchronous rounds, then after $r(n)$ rounds, every vertex can output a bit string such that the input-output configuration is correct with respect to the problem specification.

(2) (Locality definition) There exists a function ℓ that maps every ball of radius $r(n)$ (including identifiers) to a bit string, such that if every vertex outputs the result of ℓ applied to its ball of radius $r(n)$, the input-output configuration is correct with respect to the problem specification.

See [30] for the equivalence of the two definitions. Note that in n rounds any problem can be solved; this is because in that number of rounds, every vertex gets the full knowledge of the graph, and then can run a centralized algorithm.

Distributed recoloring. In order to define a distributed version of recoloring, we introduce the notion of parallel recoloring step, and parallel schedule.

Definition 2.7. — *A* parallel step*, in recoloring, consists in changing the color of an independent set of vertices between two proper colorings of the graph. A* parallel schedule *is such as each recoloring step is parallel.*

Distributed recoloring in the LOCAL model is defined as follows. Each vertex v is given as input its initial color c_0 and its target color c_{end}. It outputs a schedule $c_0, c_1, \ldots, c_\ell = c_{end}$ of length ℓ, which is the list of colors taken by v all along the transformation. The output is correct if this schedule is achieved by recoloring by independent sets. In one communication round, each vertex can check that the schedule is consistent by checking that at each step: (i) its color differs from its neighbors', and (ii) if its color changes at some step $i > 0$ (i.e. $c_{i-1} \neq c_i$), then none of its neighbors have their colors modified at that same step. The later condition ensures that an independent set of vertices is recolored at each step, guaranteeing that we have a parallel schedule if each vertex agrees with its neighbors.

When we handle r-locally unfrozen colorings in the distributed setting, a vertex is given as input its distance to a closest unfrozen vertex in both the initial and target colorings. The input validity can be checked in one round, as each vertex just needs to check that (i) both colorings are locally proper (around its vertex), and that (ii) it is unfrozen if the integer assigned to it is 0, otherwise among all its neighbors, the minimum distance is one less than its own distance.

3. Outline of the proofs

The proofs of both our Theorems 1.3 and 1.8 are in two steps. The first step is slightly different, but the second step is the same for both results.

First step. The first step consists in reaching a coloring where the vertices of a fixed set I are all unfrozen. For Theorem 1.3 (centralized recoloring), this step corresponds to the following proposition, where we start from an unfrozen coloring.

PROPOSITION 3.1. — *Let G be a connected graph of maximum degree $\Delta \geqslant 3$. In this graph, let I be a maximal independent set at distance $d \geqslant 15$, and σ be an unfrozen coloring. It is possible to transform σ into a coloring μ where I is unfrozen, with $O(n)$ single vertex recolorings.*

For Theorem 1.8 (distributed recoloring), the first step corresponds to the following proposition, where we start from an r-locally unfrozen coloring.

PROPOSITION 3.2. — *Let G be a connected graph of maximum degree $\Delta \geqslant 3$. In this graph, let I be a maximal independent set at distance $d \geqslant 15$, and σ be an r-locally unfrozen coloring. It is possible to transform σ into a coloring μ where I is unfrozen, with a parallel schedule of length $O((r + d)d\Delta^{6d+10})$. Moreover, this schedule can be computed in $O(d\Delta^{4d+10} + d \log^* n + r)$ rounds.*

Actually, the proofs of both Proposition 3.1 and 3.2 will use as an essential building block the following theorem, which is of independent interest.

Theorem 3.3. — *For every $r \geqslant 7$, every graph G of maximum degree $\Delta \geqslant 3$, and every $\Delta + 1$-coloring of G, the following holds. For every unfrozen vertex v, and every vertex w at distance r from v, there exists a recoloring sequence, such that:*

(1) At the end of the sequence, both v and w are unfrozen.
(2) All the other vertices that are recolored in the sequence are in the interior of $B(v, r)$.

Moreover, this recoloring sequence recolors each vertex at most twice, and recolors at most $2r$ vertices in total.

Informally speaking, the result ensures that, in $(\Delta+1)$-colorings, we can locally "duplicate" unfrozen vertices. This would not be possible without the condition $\Delta \geqslant 3$, as illustrated by the following example. Consider a path with the following coloring:

$$...1, 2, 3, 1, 2, 3, 1, 2, \mathbf{3{,}2}, 3, 1, 2, 3, 1, 2, 3, 1, 2, 3, ...$$

The only unfrozen vertices are the bolded ones in the middle, with color 3 and 2. Now we can either turn 3 into 1, or 2 into 1. In both cases, we are in a similar situation with a pair of unfrozen adjacent vertices. Therefore, it is impossible to duplicate unfrozen vertices.

Second step. The second step, which is common to both theorems, consists in reaching a fixed coloring γ, and it is achieved by the following proposition.

Proposition 3.4. — *Let G be a graph with $\Delta \geqslant 3$ and I be an independent set at distance 28. Let $k, k' \in \mathbb{N}$ such that $k' < k$, $k \geqslant \Delta + 1$. Let μ, γ be two colorings, using respectively at most k and k' colors, that are both unfrozen on I. There is a parallel schedule from μ to γ of length at most $(k')^{O(\Delta)}$. Moreover, such a recoloring schedule can be computed in $O(\Delta)$ rounds in the LOCAL model.*

Note that even if $k = \Delta + 1$, a k'-coloring with $k' = \Delta$ exists, by Brook's theorem. Indeed, since we have an unfrozen coloring and $\Delta \geqslant 3$, G is neither a clique nor an odd cycle.

We now have all the ingredients to establish our main results. Let us restate and prove our two main theorems.

Theorem 3.5 (Restatement of Theorem 1.3). — *Let G be a connected graph with $\Delta \geqslant 3$ and σ, η be two unfrozen k-colorings of G with $k \geqslant \Delta+1$.*

Then we can transform σ into η with a sequence of at most $O(\Delta^{c\Delta}n)$ single vertex recolorings, where c is a constant.

Proof. — Let I be an maximal independent set at distance 28. By Proposition 3.1, we can transform σ (resp. η) into a coloring μ (resp. μ') which is unfrozen on I by $O(n)$ vertex recolorings.

Let γ be an arbitrary Δ-coloring of G (note that this coloring is unfrozen on all the vertices, as we are allowed at least $\Delta+1$ colors). In order to build the recoloring sequence from μ to μ', we will build one from μ to γ, and one from γ to μ'. By Proposition 3.4, there is a recoloring schedule from μ (resp. μ') to γ recoloring at most $\Delta^{O(\Delta)}$ independent sets. This sequence recolors at most $\Delta^{O(\Delta)}$ times each vertex, which completes the proof. □

The second theorem is about local reconfiguration, and we assume that the colorings are r-locally unfrozen.

THEOREM 3.6 (Restatement of Theorem 1.8). — *Let $k, \Delta, r \in \mathbb{N}$ such that $k > \Delta \geqslant 3$. There exists three constants c, c', c'' such that, for any graph G of bounded degree Δ, and σ, η two k-colorings of G which are r-locally unfrozen*(5)*, we can transform σ into η with a parallel schedule of length at most $O(k^{c\Delta} + \Delta^{c'}r)$. Moreover, this schedule can be computed in:*

- $O(\Delta^{c''} + \log^* n + r)$ *rounds if* $k \geqslant \Delta + 2$.
- $O(\Delta^{c''} + \log^2 n \cdot \log^2 \Delta + r)$ *rounds if* $k = \Delta + 1$.

Proof. — We first compute an independent set at distance $d = 28$ in a distributed manner in time $O(\Delta^{28} + \log^* n)$ rounds in the LOCAL model by [2]. Then by applying Proposition 3.2 with $d = 28$, we can transform σ (resp. η) into a coloring μ (resp. μ') such that all the vertices of I are unfrozen with a recoloring schedule of length $O(r\Delta^{178})$ in $O(\Delta^{122} + \log^* n + r)$ rounds.

Assume first that $k \geqslant \Delta + 2$. It is easy to transform μ into a coloring γ with $k - 1$ colors in one round: for every vertex that has color $\Delta + 2$, recolor it with a color of smaller index. Such a color must exist, and the transformation takes only one round. Now by Proposition 3.4, we can transform μ' into γ efficiently, and finish this proof.

Now, if $k = \Delta + 1$, we first compute an arbitrary Δ-coloring, in time $O(\log^2 n \log^2 \Delta)$, using the algorithm of [25], and then use Proposition 3.4 twice (between μ and γ, and between μ' and γ). □

---

(5) For a formal definition of LOCAL model and of r-locally unfrozen colorings, the reader is referred to Definition 2.1.

4. Local warming and consequences

4.1. Maximum degree at least 3 ensures local warming

The goal of this section is to prove the following theorem, that ensures that, in any large enough ball centered at an unfrozen vertex, we can unfreeze at a vertex on the border while keeping its center unfrozen.

THEOREM 4.1 (Restatement of Theorem 3.3). — *For every $r \geqslant 7$, every graph G of maximum degree $\Delta \geqslant 3$, and every $\Delta + 1$-coloring of G, the following holds. For every unfrozen vertex v, and every vertex w at distance r from v, there exists a recoloring sequence, such that:*

(1) At the end of the sequence, both v and w are unfrozen.
(2) All the other vertices that are recolored in the sequence are in the interior of $B(v, r)$.

Moreover, this recoloring sequence recolors each vertex at most twice, and recolors at most $2r$ vertices in total.

Consider a graph G of maximum degree $\Delta \geqslant 3$. In this section, we consider σ to be an unfrozen $(\Delta+1)$-coloring of G and v to be an unfrozen vertex of σ. Let $B = B(v, r)$. Note that if the boundary of B is empty (that is, the whole graph is contained in $B(v, r-1)$) then the theorem holds. For the rest of the section, we will assume that this is not the case.

Let w be a vertex of the boundary of B. Our goal is to prove that there exists a recoloring sequence of the vertices of the interior of B plus w, which recolors w, and such that at the end of the sequence, v is still unfrozen. In the following, such a recoloring sequence will be called a *nice* sequence. The existence of a nice recoloring sequence implies Theorem 3.3. Let us first give some conditions which ensure the existence of a nice recoloring sequence.

LEMMA 4.2. — *Let P be a shortest path from v to w. Assume that P contains an unfrozen vertex not in $N[v]$. Then there is a nice recoloring sequence.*

Proof. — Let z be the unfrozen vertex of P closest to w. By assumption, we know that z is not adjacent to v. Let P' be the subpath from z to w. We can recolor w by recoloring a ladder along this path P'. Let us check that this is a nice recoloring sequence. All the vertices of P', except w, are in the interior of B, because P is a shortest path from the center of the ball B to w. Moreover, after this transformation v is still unfrozen since none of its neighbors were recolored. Finally, every vertex is recolored at most once. See Figure 4.1 for an illustration. □

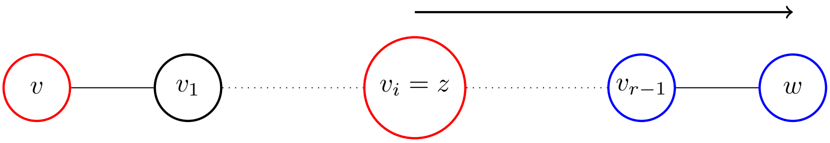

Figure 4.1. Illustration of Lemma 4.2. Blue vertices are frozen, red vertices are unfrozen. The $\rightarrow$ arrow means that we perform a ladder from z to w.

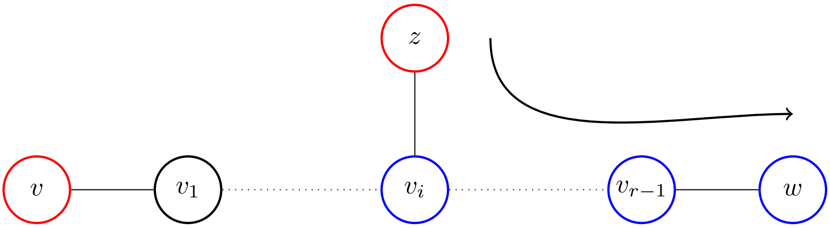

Figure 4.2. Illustration of Lemma 4.3. We perform a ladder from z to w.

We can extend this property to the vertices at distance 1 from the path P.

LEMMA 4.3. — *Let P be a shortest path from v to w. Assume that there is an unfrozen vertex z adjacent to P, such that $3 \leqslant d(v, z) \leqslant r - 1$. Then there is a nice recoloring sequence.*

Proof. — The argument is similar to the one of Lemma 4.2. Let z be a vertex satisfying the conditions of the lemma, that is the closest to w. Note that z is in the interior of B, since $d(v, z) \leqslant r - 1$. Let z' be the neighbor of z in P which is the closest to w, then z' is at distance at least 2 from v, in particular, it is not a neighbor of v. Then, we can again recolor along a ladder that starts with z, z', and then continues along P towards w. This allows us to recolor w while leaving the neighbors of v and the boundary of B untouched. Each vertex is recolored at most once, which implies that this is a nice recoloring sequence. See Figure 4.2 for an illustration. ☐

LEMMA 4.4. — *Let $P = v_0, \ldots, v_r$ be a shortest path from $v = v_0$ to $w = v_r$. If there is an index $2 \leqslant i \leqslant r - 3$, such that $\sigma(v_i) \neq \sigma(v_{i+3})$, then there is a nice recoloring sequence.*

Proof. — By Lemma 4.2, we can assume that all the vertices of P, except for $v = v_0$ and its neighbor v_1, are frozen. Let us denote by η the coloring obtained by recoloring the ladder along P, starting either from v, if v_1 is frozen, or v_1, if it is unfrozen, and ending in w. In η, we have recolored w, but now v might be frozen. If v is unfrozen, we are done. If v_1 is unfrozen, then again we are done, since we can make a ladder with just v_1 and v. Thus, let us assume that both v_1 and v are frozen in η.

Amongst the indices $2 \leqslant i \leqslant r-3$ such that $\sigma(v_i) \neq \sigma(v_{i+3})$, let i be the minimum one. We have the following claim:

Claim 4.5. — *The vertex v_{i+2} is unfrozen in the coloring η.*

Proof. — Let $c = \sigma(v_{i+3})$. Let us make a few remarks:

(1) $\sigma(v_{i+2}) \neq c$, because σ is a proper coloring,
(2) $\sigma(v_{i+1}) \neq c$, because v_{i+2} is frozen in σ. More generally, none of the neighbors of v_{i+2} except v_{i+3} has color c.
(3) $\sigma(v_i) \neq c$, because $\sigma(v_i) \neq \sigma(v_{i+3})$ by assumption.

Now, by construction and by the properties of ladders, we have $\eta(v_{j+1}) = \sigma(v_j)$, for every vertex v_j of the ladder, except $v_r = w$. Transposing the remarks above about σ to η we get that:

(1) $\eta(v_{i+3}) \neq c$,
(2) $\eta(v_{i+2}) \neq c$, and more generally, no neighbor of v_{i+2} has color c.
(3) $\eta(v_{i+1}) \neq c$.

Consequently, c does not appear in the closed neighborhood of v_{i+2} in η, which implies that v_{i+2} is unfrozen in η, as claimed. □

By Claim 4.5, v_{i+2} is unfrozen in η. We can make a new ladder in η along the path P from v_{i+2} to v. The vertex w is not recolored by this ladder, and at the end v is unfrozen. Since every vertex is recolored at most twice, we get a nice recoloring sequence. □

We now have all the tools to prove that a nice recoloring sequence always exists. Let us assume that we do not fall into one of the previous cases. Let $P = v_0, \ldots, v_r$ be a shortest path from v to w. By Lemma 4.2, all the vertices in P but the first two are frozen. By Lemma 4.3, all the neighbors of P that are at distance at least three from v are frozen. Since we are free to rename colors, Lemma 4.4 ensures that $\sigma(v_i) \in \{0, 1, 2\}$ and $\sigma(v_i) = i \mod 3$ for every $i \geqslant 2$. We denote by η the coloring obtained by recoloring the ladder along P starting either from v, if v_1 is frozen, or from v_1 otherwise. As before, at that point we are done, unless both v and v_1 are frozen in η. Note that, for $i \geqslant 3$, $\eta(v_i) = (i-1) \mod 3$, because of the color shift of the ladder.

Let us consider the vertex v_5. It cannot have degree 2, because it is frozen in σ, and no degree-2 vertex can be frozen in a $\Delta + 1$-coloring, with $\Delta \geqslant 3$.

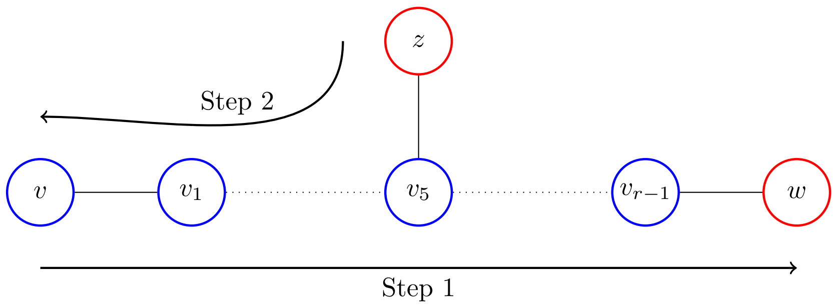

Figure 4.3. Illustration of Claim 4.6. We first perform a ladder from v to w, then a ladder from z to v.

Hence, we can assume that v_5 has a neighbor z outside P. And because P is a shortest path, z is at distance at least 4 from v. Also note that, since we assume that $r \geqslant 7$, $d(v, z) \leqslant d(v, v_5) + 1 \leqslant r - 1$. Therefore, by Lemma 4.3, z is frozen in σ. We will use the following claim:

Claim 4.6. — *If z is unfrozen in η, then a nice recoloring exists.*

Proof. — Indeed, from η, we can recolor along a ladder from z to v. After this operation, no other vertex of the boundary is recolored, v is unfrozen, and each vertex has been recolored at most twice. Hence, this defines a nice recoloring sequence. See Figure 4.3 for an illustration. □

We make a case analysis depending on the number of neighbors of z in P. Since P is a shortest path, z has at most three neighbors in P.

Case 1: z has exactly one neighbor in P. Since z is frozen in σ, v_5 is its only neighbor colored with $\sigma(v_5)$. In η, v_5 is recolored with a different color, which implies that z is no longer frozen in η. By Claim 4.6, the conclusion follows.

Case 2: z has exactly two neighbors in P. Let c_1 and c_2 be the colors of these two neighbors in σ. Since z is frozen in σ, it does not have two neighbors colored with the same color. Moreover, in η, the two neighbors of z in P have color $c_1' = c_1 - 1 \mod 3$ and $c_2' = c_2 - 1 \mod 3$ by Lemma 4.4 (since z is incident to v_5, the other neighbor is at least v_3). Then we have $\{c_1, c_2\} \neq \{c_1', c_2'\}$. It follows that z is unfrozen in η, and the result follows from Claim 4.6.

Case 3: z has exactly three neighbors in P. Since P is a shortest path, these neighbors are consecutive in P. Let $3 \leqslant i \leqslant 5$ such that v_i, v_{i+1}, v_{i+2} are the neighbors of z in P. Since z is adjacent to v_{i+1}, we have $\sigma(z) \neq$

$\sigma(v_{i+1}) = (i+1) \mod 3$. Let P' be the path obtained from P by replacing v_{i+1} by z. (Note that z is not in the boundary of B, and then $z \neq w$.) Then P' is a shortest path from v to w, and since $\sigma(z) \neq (i+1) \mod 3$, we can apply Lemma 4.4 on P' to conclude. More precisely, if $i = 3$, then $(i+1)+3 \leqslant r$ because $r \geqslant 7$, and if $i = 4, 5$, then $(i+1) - 3 \geqslant 2$, thus in both cases Lemma 4.4 applies.

To sum up, the only remaining case was the one where none of the previous lemmas apply, in which case we have just proved that considering v_5 and its neighbor z, we can find a nice recoloring. Hence, we can always find a nice recoloring, which implies Theorem 3.3.

4.2. Using local warming to unfreeze an independent set

The next lemma ensures that, in the centralized setting, we can obtain a 28-locally unfrozen coloring.

Proposition 4.7 (Restatement of Proposition 3.1). — *Let G be a connected graph of maximum degree $\Delta \geqslant 3$. In this graph, let I be a maximal independent set at distance $d \geqslant 15$, and σ be an unfrozen coloring. It is possible to transform σ into a coloring μ where I is unfrozen, with $O(n)$ single vertex recolorings.*

We note that in the proposition above the $O()$ notation does not have a dependency in Δ.

Proof. — See Figure 4.4 for an illustration of the proof. We start by unfreezing a vertex of I. Consider a vertex v in I that minimizes its distance to an unfrozen vertex. If v is unfrozen, we are done. Otherwise, we take a shortest path from v to the closest unfrozen vertex, and build a ladder along this path to unfreeze v.

We construct an auxiliary graph H, where $V(H) = I$, and we put an edge (u, u') in H if there exists a path of length at most $2d$ from a vertex of $B(u, 7)$ to a vertex of $B(u', 7)$ in G, which does not contain any vertex in $B(u'', 7)$ for any $u'' \in I, u'' \neq u, u'$. Note that for any pair $a, b \in I$, $B(a, 7)$ and $B(b, 7)$ are disjoint, since $d \geqslant 15$.

Claim 4.8. — *The graph H is connected.*

Proof. — Suppose the claim does not hold. Let A be a (connected) component of H. Let $u \in A$ and $w \in I \setminus A$, such that $d_G(u, w)$ is minimum among the such pairs. If $d(u, w) \geqslant 2d + 2$, then the vertex in the middle of a shortest path between u and w is at distance at least $d+1$ from any vertex in I which contradicts the assumption that I is a maximal independent set at distance d. Therefore, there exists $u \in A$ and $w \in I \setminus A$ such that $d_G(u, w) \leqslant 2d + 1$. Let us choose again those vertices to minimize $d_G(u, w)$ and let P be a shortest path from u to w. Let x be the last vertex of P in $B(u, 7)$ and y the first vertex of P in $B(w, 7)$. We necessarily have $d(x, y) \leqslant 2d$. Since u and w are not connected in H, P must intersect $B(y, 7)$, for some vertex $y \in I, y \neq u, w$ (by definition of H). As y is either in A or $I \setminus A$, this contradicts the choice of u and w that minimizes their distance (in the first case, w and y was a better option, u and y was a better option in the second case). □

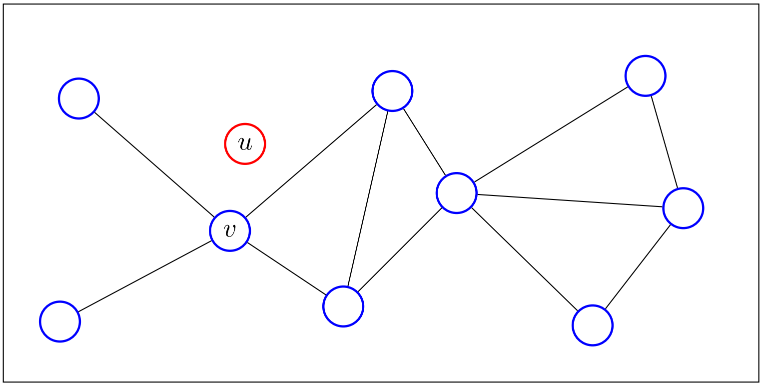

Figure 4.4. Illustration for Section 4.2. Building the graph H over the set of independent vertices I (vertices in blue). Vertex u is not frozen, and its closest neighbor in I is v. If v is unfrozen, we perform a ladder from u to v.

Now, let us denote by T a spanning tree of H rooted in v. Let τ be a breadth first search (BFS) ordering of T. The *index* of a vertex of H is its position of appearance in the BFS. Let u be the first vertex of τ that is frozen. (If no such vertex exists, we are done.) Note that u cannot be the root of the tree, since v is unfrozen.

CLAIM 4.9. — *By recoloring a constant number of vertices, we can unfreeze u, and this operation leaves the vertices of smaller index unfrozen.*

Proof. — Let u' be the parent of u in T. By definition, u' is unfrozen. Let P be a path from u to u' in G corresponding to the edge (u, u') in H. By definition, P has length at most $2d$ and does not intersect $B(u'', 7)$ for

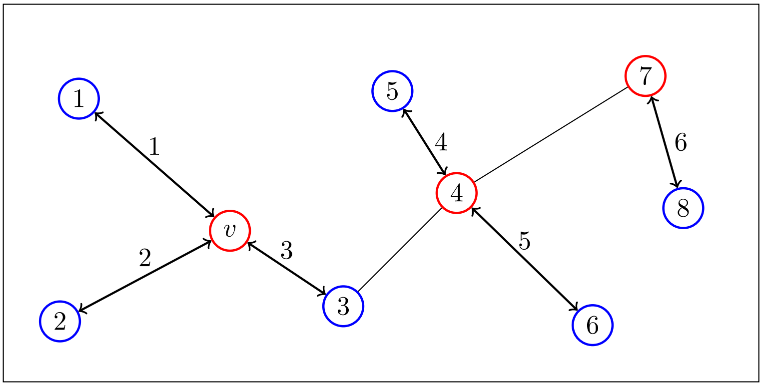

Figure 4.5. Illustration of Claim 4.9. We build a BFS tree from v. For vertices visited by the BFS from first to last, if this vertex is frozen, use its parent in the tree to unfreeze it (symbolized with the $\leftrightarrow$ arrow). The label on each arrow tells in which order (blue) frozen vertices get unfrozen.

any $u'' \in I, u'' \neq u, u'$. Also, we can assume that P is an induced path, since otherwise we can take a path on a subset of vertices of P, satisfying the same properties. If there is a vertex y in $P \setminus B(u', 7)$ that is unfrozen, we simply recolor a ladder from y to u, to unfreeze u. Otherwise, let x be the last vertex of P in $B(u', 7)$. By Theorem 3.3, by recoloring at most 14 vertices, we can unfreeze x, leave u unfrozen and while recoloring only vertices in $B(u', 6)$ (and x). We can recolor a ladder from x to u to get the conclusion. In both cases, the recoloring sequence has length at most $2d + 14$, and the unfrozen vertices of I are kept unfrozen. □

We iterate this construction to get all of I unfrozen. See Figure 4.5 for an illustration. This requires at most $(2d+14)|I| \leqslant (2d+14)\cdot n$ recoloring steps. This proves Proposition 3.1. Note that since every vertex v has at most Δ^{2d} other vertices of I at distance at most $2d$, and that only a constant number of ladders for each such vertex can recolor v, every vertex is recolored at most $O(\Delta^{2d})$ times during the whole process. □

We now prove a local analogue of the previous proposition. Intuitively, it says that if we have a well-spread set of unfrozen vertices, we can move it to another well-spread set locally.

Proposition 4.10 (Restatement of Proposition 3.2). — *Let G be a connected graph of maximum degree $\Delta \geqslant 3$. In this graph, let I be a* *maximal independent set at distance $d \geqslant 15$, and σ be an r-locally unfrozen coloring. It is possible to transform σ into a coloring μ where I is unfrozen, with a parallel schedule of length $O((r+d)d\Delta^{6d+10})$. Moreover, this schedule can be computed in $O(d\Delta^{4d+10} + d\log^* n + r)$ rounds.*

Note that r could be large and depend on n, in which case Proposition 3.2 not only moves the set of well-spread unfrozen vertices around, but also makes it denser.

Proof. — Let σ be an r-locally unfrozen coloring. We proceed in two steps: first, we show that we can somehow replace the set of unfrozen vertices with a subset of I, and then we show how to unfreeze all the vertices of I. For both steps, we will use an auxiliary coloring of the vertices of I. Note that this auxiliary coloring is just a tool, and is independent of the coloring we are modifying. Let p be an integer. Consider a graph H, whose vertex set is I and whose edges are the pairs $(a,b) \in I^2$, such that $d_G(a,b) \leqslant p$. The graph H has maximum degree $\Delta_H = O(\Delta^p)$, thus we can compute a (Δ_H+1)-coloring α of H in $O(\Delta_H + \log^*(|H|))$ rounds in H [2]. Since any computation round in H can be simulated in p rounds in G (since each edge in H is a path of length at most p in G), we can compute the auxiliary coloring α of I in G in $O(p\Delta^p + p\log^* n)$ rounds (in G).

CLAIM 4.11. — *From σ, we can reach a coloring η in which any vertex of I is at distance at most $r+d$ from an unfrozen vertex of I, with a parallel schedule of length $O(d\Delta^{2d+4})$ computed in $O(d\Delta^{2d+2} + d\log^* n)$ rounds.*

Proof. — Let N be the set of unfrozen vertices at the beginning of the algorithm. Consider an auxiliary (Δ_H+1)-coloring α of H, with $p = 2d+2$. Let M_i be the set of vertices of I that have color i in α. We will go through the sets M_i, in successive phases. At phase i, for every $u \in M_i$ that is frozen, if $B(u,d)$ contains a vertex v of N that is still unfrozen, we recolor a ladder from v to u (where we take v to be the closest unfrozen vertex). Since, $p = 2d+2$, the balls $B(u,d+1)$ with $u \in M_i$ are all disjoint by construction of the M_i. Therefore, we can perform these transformations for each vertex of M_i with a unfrozen node at distance at most d in parallel without coordination. Now, we want the additional property that a vertex u of I that has been unfrozen cannot be refrozen. This could happen if there is an unfrozen vertex in the neighborhood of u that is the start of a ladder (thus at distance exactly d from another vertex of I). We add a twist to the algorithm: if this situation occurs, we do not build the ladder.

To prove that the claim holds at the end of this process, consider a vertex w of I. By assumption, at the beginning w was at distance at most r from an unfrozen vertex x of N. Consider a vertex u of I in $B(x, d)$ (such a vertex exists by maximality). If this vertex u is unfrozen, then the claim holds for w. If u is frozen, the only possibility is that we did not build a ladder from x to u because of the twist in the algorithm. But in this case there exists a vertex $u' \in I$ in the neighborhood of x which is necessarily unfrozen (since there is no obstruction to building a ladder from x to u').

The round complexity is dominated by the computation of the auxiliary coloring, and the schedule length can be bounded by the maximum size of a ladder inside a ball $O(d)$, times the number of color classes $O(\Delta^{2d+4})$. See Figure 4.6 for an illustration. □

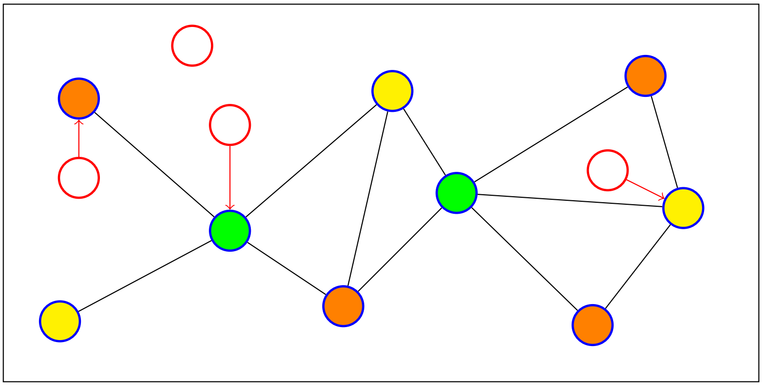

Figure 4.6. Illustration for Claim 4.11. The vertices represented as red circles are unfrozen vertices (i.e. N). The other vertices are in I, with the edges of H. The colors of the vertices of I correspond to the auxiliary coloring. An edge from a vertex in N to a vertex v in I corresponds to a ladder to unfreeze v. This algorithm ensures that for all vertices in I there is an unfrozen vertex in I at distance at most $r + d$ in G.

We have ensured that, for each vertex in I, there is an unfrozen vertex not too far. We will now show how to ensure that each vertex in I gets unfrozen efficiently:

Claim 4.12. — *Consider a coloring of G where some vertex in I is frozen. We can reach a new coloring, where each frozen vertex in I has a strictly smaller distance to an unfrozen vertex, with a parallel schedule of length $O(\Delta^{6d+14})$ in $O(d\Delta^{4d+10} + d \log^* n)$ rounds.*

Proof. — Again, consider an auxiliary $(\Delta_H + 1)$-coloring α of H, with this parameter $p = 4d + 10$. We will consider the color classes M_i, one after another. For every $u \in M_i$, let X_u be the ball $B(u, 2d + 4)$ plus the vertices of I at distance exactly $2d + 5$ from u in G. Note that no vertex of I in $V \setminus X_u$ is adjacent to X_u. If u is unfrozen, then we can unfreeze all the vertices of $I \cap X_u$: since $d \geqslant 15$, we can proceed exactly like in the proof of Proposition 3.1. Note that, similarly to the previous proof, because of our definition of the sets X_u (for vertices u), these recolorings can be performed in parallel, and no vertex of I that was unfrozen can be refrozen. See Figure 4.7 for an illustration.

We claim that, at the end of this recoloring, the minimum distance from any vertex u of I to the closest unfrozen vertex of I has decreased. Indeed, let v be the closest unfrozen vertex of I from u at the beginning. If $d(u, v) \leqslant 2d+4$, u is unfrozen at the end of the algorithm by construction. Otherwise, let x be the $(d+1)$-th vertex of a shortest path from v to u. Note that x must be at distance at most d from a vertex v' of I. Thus v' is in $B(v, 2d + 1)$. So v' is unfrozen at the end of the algorithm. And since the distance from u to v' is strictly smaller than the one from u to v, we get the condition of the claim. The computation of the schedule length and number of rounds are similar to the ones of the previous claim, except the unfreezing of each $X_u \cap I$ uses $O(\Delta^{2d+4})$ recoloring steps. □

From the r-locally unfrozen coloring, by using the algorithm of Claim 4.11, we ensure that vertices in I have unfrozen vertices at distance at most $r + d$. We then iterate the algorithm of Claim 4.12, to finally unfreeze all of I. As each iteration decreases the maximal distance by 1, after $r + d$ iterations, we get that all vertices in I are unfrozen.

The number of iteration of Claim 4.12 is at most $r + d$ by Claim 4.11, thus the total schedule length is in $O((r+d)d\Delta^{6d+10})$. The total number of rounds is $O(d\Delta^{4d+10} + d \log^* n + r)$ since we can reuse the same auxiliary coloring for all the iterations.

□

5. Recoloring locally unfrozen colorings

The goal of this section is to prove Proposition 3.4, which roughly says that we can perform an efficient distributed recoloring between two colorings that are unfrozen on the same independent set S. (The exact statement will be reminded a bit later.) Figure 5.1 illustrates the general strategy.

Let us first prove a few lemmas.

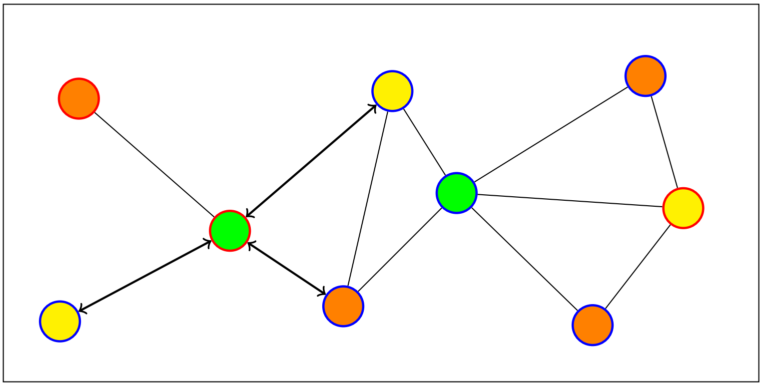

Figure 4.7. Illustration for Claim 4.12. For each color in α, if a vertex has this color and is unfrozen, unfreeze vertices in I that are in its distance-$(2d+4)$ neighborhood, using the process from Claim 4.9. Here, for color green, the left vertex that is unfrozen will unfreeze its frozen neighbors. The other green vertex does nothing as it is frozen, however it gets a closer unfrozen vertex in I after this step.

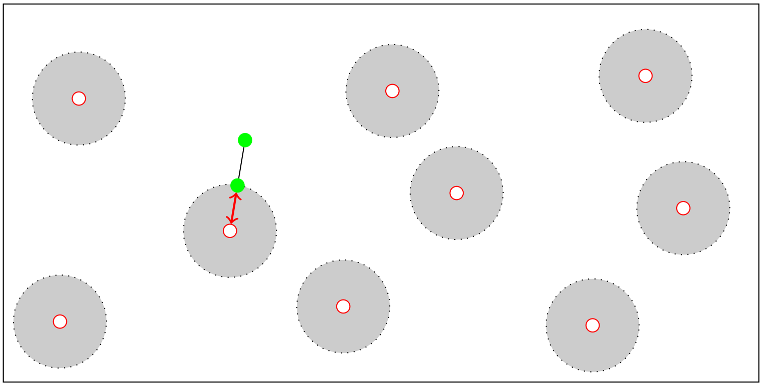

Figure 5.1. To compute the recoloring schedule, we first consider the subgraph where, for each node x in I, we remove $B(x,r)$ (the gray balls in the figure). This holed graph has a specific structure, that allows us to compute a recoloring schedule efficiently. Then we extend this schedule to the full graph. Each time a vertex needs to take a color that is blocked by a vertex from one of the balls B, we use Theorem 3.3 to unfreeze it, allowing it to free the color needed.

5.1. Degeneracy ordering lemma

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 $v_1, \dots, v_n$ of the vertices such that for every $i \leqslant n$, the vertex v_i has at most d neighbors in $v_{i+1}, \dots, v_n$. In the following, we group vertices in independent sets $V_1, \dots, V_q$, such as all vertices in V_i have at most d neighbors in $V_{i+1} \cup \dots \cup V_q$ (note that vertices in V_i do not have neighbors in V_i, since the V_i's are independent sets).

LEMMA 5.1. — *Let G be a connected r-locally unfrozen graph which is k-colorable, and let S be a maximal independent set at distance at least $2r + 2$. Let B_S be the set of vertices at distance at most r from S, and $G' = G \setminus B_S$.*

Then there exists a $(\Delta - 1)$-degeneracy ordering of G', consisting of $O(r \cdot k)$ independent sets. Moreover, if we are given a k-coloring c of G', such an ordering can be found in $O(r)$ rounds in the LOCAL model.

Proof. — The graph G' is $(\Delta - 1)$-degenerate because we have removed at least one vertex from a connected graph of maximum degree Δ. The degeneracy ordering of G' will be built by first splitting G' into layers such that each vertex v in layer i has at most $\Delta - 1$ neighbors in layers $j \geqslant i$. Then we will split each layer into independent sets using the coloring c.

We define layer i, noted L_i, of G' as the set of vertices at distance exactly i from B_S. Since S is a maximal independent set at distance $2r + 2$, all the vertices of G' belong to a layer i with $i \leqslant r + 2$. All the vertices in layer 1 have a neighbor in B_S and, for every $i \geqslant 2$, all the vertices in layer i have at least one neighbor in layer $(i - 1)$. So the graph induced by the layers $\cup_{j \geqslant i} L_j$ is $(\Delta - 1)$-degenerate (and all the vertices of L_i have degree at most $\Delta - 1$ in $\cup_{j \geqslant i} L_j$). We now split each layer into k independent sets using the color classes of a k-coloring c. We can order the vertices in the layers by color, and get a $(\Delta - 1)$-degeneracy ordering of G' composed of $O(r \cdot k)$ consecutive independent sets.

Note that in the LOCAL model, if S is given, computing this partition can be done in $O(r)$ rounds. Indeed, after computing its distance to S, each vertex knows if it is in B_S or in which layer it is. As their color in c is given as input, they do not need more information. ☐

5.2. List-coloring lemma

The following lemma is a list-coloring adaptation of a proof of Dyer, Flaxman, Frieze, and Vigoda [21], that ensures that one can transform any $(d+2)$-coloring of a d-degenerate graph into any other. Let G be a graph in which, for every vertex u, we are given a list L_u of colors. A coloring c of G is *compatible* with the lists L_u, if the coloring is proper and for every vertex u, $c(u) \in L_u$. Let τ be an ordering of $V(G)$. We denote by $d^+_\tau(u)$ (or $d^+(u)$, when τ is clear from context) the number of neighbors of u that appear after u in τ. We say that a set of lists is *safe for* τ if, for every vertex u, $|L_u| \geqslant d^+_u + 2$.

We will consider particular schedules in the LOCAL model such that, at each step, all the recolored vertices are recolored from the same color a to the same a color b (in particular, the recolored vertices form an independent set). We call such a reconfiguration step an *$a \to b$ step.* A recoloring schedule where all the steps are $a \to b$ steps is called a *restricted schedule.* Note that any schedule can be transformed into a restricted schedule by multiplying the length of the schedule by $O(k^2)$ (where k is the total number of colors). Indeed, we simply have to split each step s of the initial schedule into $k(k-1)$ different $a \to b$ steps $s_{a,b}$ for every pair of colors a, b. At step $s_{a,b}$, we recolor from a to b all the vertices recolored from a to b at step s. Note that since at step s, the set of recolored vertices is an independent set, all the intermediate colorings obtained after $s_{a,b}$ are proper.

Lemma 5.2. — *Let G be a graph, τ be an ordering of G composed of t consecutive independent sets, and $d = \max_{v\in V} d^+_\tau(v)$. Consider a set of lists $(L_v)_{v\in V}$ that are safe for τ. Let σ, η be two k-colorings of G compatible with $(L_v)_{v\in V}$.*

There exists a parallel schedule from σ to η with a restricted schedule of length at most k^{t+1} where $k = |\cup_{v\in V} L_v|$. Moreover, this schedule can be found in $O(t)$ rounds, if τ is given.

Proof. — Let $I_1, \ldots, I_t$ be the independent sets of the ordering τ. For every $i \leqslant t$, we denote by G_i the graph $G[\cup_{j\leqslant i} I_j]$.

Let us prove by induction on i that we can recolor G_i from σ_{G_i} to η_{G_i} with a restricted parallel schedule of length at most k^{i+1}. Since G_1 induces an independent set, a restricted schedule of length $k\cdot(k-1) \leqslant k^2$ exists: for every pair $a \neq b$, we create an $a \to b$ step where we recolor the vertices of I_1 colored a in σ and b in η, from color a to color b. After all these steps, the coloring is η_{G_1}. Since I_1 is an independent set, we indeed recolor an independent set at any step.

In order to extend the transformation of G_{i-1} into a transformation of G_i (with $i \geqslant 2$) we perform as follows. For each step s of the transformation of G_{i-1}, we will add $(k-2)$ new steps before s. Since the transformation is a restricted schedule, there exists a, b such that s is an $a \to b$ step. For every $c \neq a, b$, we add a $b \to c$ step, denoted $s_{b,c}$, between s and the step before in the transformation of G_{i-1}. Let I be the set of vertices recolored at step s, and N_I be the set of vertices at distance exactly 1 from a vertex of I. In $s_{b,c}$, we recolor all the vertices of $G_i \cap N_I$ colored b with the color c, if it is possible (i.e. if c is in their lists, and they do not have any neighbor already colored c) . Note that every vertex v of I colored b can indeed be recolored with some color c, distinct from a, since the size of the list of v is at least the degree of v plus two in G_i. So after these new steps, we can safely apply the $a \to b$ step without creating monochromatic edges in G_i.

Finally, at the end of the reconfiguration sequence of G_{i-1}, we add $k \cdot (k-1)$ steps in order to recolor the vertices of I_i with their target colors (after G_{i-1} has reached its target coloring) as we did for I_1. This provides a restricted schedule of length $(k-2) \cdot k^i + k \cdot (k-1) \leqslant k^{i+1}$ from σ_{G_i} to η_{G_i}. This proves the induction.

In order for a vertex to compute its own schedule, it simulates the induction above. For a given node, changing its own color implies the change of colors of some neighbors of smaller level. Hence, in the LOCAL model, it suffices to know the neighborhood at distance t. □

As an immediate corollary, we obtain the following, where the lists are just the same k colors for every vertex:

LEMMA 5.3. — *Let G be a d-degenerate graph and σ, η be two k-colorings of G with $k \geqslant d+2$. Assume that G has a degeneracy ordering composed of t consecutive independent sets. Then there exists a parallel schedule from σ to η with a restricted schedule of size at most k^{t+1}, that can be computed n $O(t)$ rounds in the LOCAL model.*

5.3. Recoloring outside the balls

Let us now prove that we can obtain coloring, where the target colors have been reached, for the vertices of $V \setminus B_S$. Then we will explain how we can transform such a coloring into the target coloring by recoloring (almost) only vertices of B_S.

Lemma 5.4. — *Let $k \geqslant \Delta+1$ and $r \geqslant 7$. Let G be a graph of maximum degree $\Delta \geqslant 3$, and let σ, η be two r-locally unfrozen k-colorings of G. Let S be a maximal independent set at distance $r' \geqslant 2r+2$. Let $G' = G[V \setminus B_S]$ where $B_S = \cup_{x \in S} B(x,r)$.*

Then, there exists a coloring η' such that $\eta'_{G'} = \eta_{G'}$ and a parallel schedule of length $k^{O(r'k)}$ from σ to η'.

Proof. — The first part of the recoloring sequence is a pre-processing step to ensure that every vertex $v \in S$ is unfrozen. Since σ is r-locally unfrozen, for every v in S, there is a vertex u in $B(v,r)$ such that u is unfrozen. By recoloring a ladder along a shortest path from u to v, v is unfrozen. Since $B(v,r)$ does not share an edge with $B(v',r)$ for any $v, v' \in S$, we can repeat this argument for every $v \in S$ and then assume that S is unfrozen. In the LOCAL model, all these recolorings pre-processing steps can be performed in parallel. So, from now on, we can assume that, in σ, every vertex of S is unfrozen (and we will keep this property all along the schedule).

By Lemma 5.1, we have a degeneracy order, and Lemma 5.3 uses it to provide a restricted recoloring schedule $\mathcal{R}$ in G' from $\sigma_{G'}$ to $\eta_{G'}$ in at most $k^{O(r'k)}$ steps.

Let us now explain how we can extend the restricted schedule $\mathcal{R}$ of G' to G, that is, avoid the conflicts between vertices in G' and their neighbors in G that are in B_S. Let X be the set of vertices which are recolored during an $a \to b$ step of $\mathcal{R}$. Denote by Y the set of vertices of B_S such that Y is adjacent to a vertex of X. We will recolor these vertices, before they create any conflict.

For each ball of radius r centered in $u \in S$, we first identify the vertices of $Y_u = Y \cap B(u,r)$ that are colored b. Note that Y_u is an independent set. By Theorem 3.3, we can recolor each vertex of Y_u in at most $2r$ steps with a different color, leaving u unfrozen, and without modifying the color of any other vertex in Y_u. Since Y_u contains at most Δ^r vertices, we can change the color of all the vertices of Y_u with a schedule of length at most $2r \cdot \Delta^r \leqslant 2r'k^{r'}$. Since all the balls of radius r centered in S are disjoint and do not share an edge, we can perform these schedules in parallel for each ball of radius r centered in S.

Since the restricted schedule $\mathcal{R}$ has length at most $k^{O(r'k)}$, the new schedule has length at most $k^{O(r'k)} \cdot 2r'k^{r'} = k^{O(r'k)}$, which completes the proof. $\square$

The previous lemma ensures that, from any locally unfrozen coloring, we can obtain a locally unfrozen coloring where all the vertices but the vertices of B_S are colored with the target coloring. Before completing the proof of Proposition 3.4, we need one more lemma.

5.4. Recoloring inside the balls (easy case)

LEMMA 5.5. — *Let $k \geqslant \Delta+1$. Let σ and η be two k-colorings of a graph G which only differ on $X \subseteq V$. Assume that, in each connected component C of $G[X]$, there exists a vertex that has degree at most $k-2$ or has two neighbors in $V \setminus X$ colored the same. Then there is a parallel schedule from σ to η of length at most $k^{O(\mathrm{diam}(X)k)}$.*

Proof. — Let C be a component of X. For every vertex v of $G[C]$, let Z_v be the set of colors in σ that appear on neighbors outside X, that is on $N(v) \cap (V \setminus X)$. We assign to every vertex v of $G[C]$ the list of colors $[k] \setminus Z_v$. Note that since the total number of colors is $k \geqslant \Delta+1$, every vertex $x \in C$ has a list of size at least $deg_{G[X]}(x)+1$. Moreover, if a vertex x has degree at most $k-2$ in G, or two neighbors of x are colored the same in $V \setminus X$, its list has size at least $deg_{G[X]}(x)+2$. We claim that we can build a degeneracy ordering of C for which the lists of C are safe, and that consists of $\mathrm{diam}(C)k$ consecutive independent sets. Indeed, similarly to earlier in the paper, we can take the vertices of C by layers, corresponding to the distance from $V \setminus X$, and then split these layers into independent sets using the colors of σ.

Finally, by Lemma 5.3, there exists a recoloring sequence of $G[C]$ from σ to η which recolors each vertex at most $k^{O(\mathrm{diam}(C)k)}$ times. Since we can treat each component of X simultaneously (there is no edge between them), the conclusion follows. □

5.5. Finishing the proof of Proposition 3.4

All the previous lemmas can be combined in order to prove Proposition 3.4, that we restate here. Note that since recoloring with $2\Delta+2$ colors is trivial, we implicitly consider that k, k' are below this value.

PROPOSITION 5.6 (Restatement of Proposition 3.4). — *Let G be a graph with $\Delta \geqslant 3$ and I be an independent set at distance 28. Let $k, k' \in \mathbb{N}$ such that $k' < k$, $k \geqslant \Delta+1$. Let μ, γ be two colorings, using respectively at most k and k' colors, that are both unfrozen on I. There is a parallel schedule from μ to γ of length at most $(k')^{O(\Delta)}$. Moreover, such a recoloring schedule can be computed in $O(\Delta)$ rounds in the LOCAL model.*

Proof of Proposition 3.4. — Let $r = 7$. Let I be a maximal independent set at distance $r' = 2r + 14$. Let $G' = G \setminus B_I$ where $B_I = \cup_{x \in I} B(x, r)$. By Lemma 5.4, there is a coloring η' which agrees with μ on $G \setminus B_I$ and a recoloring schedule from γ to η' of length at most $k^{O(rk)}$. To conclude, we only need to find a recoloring sequence from η' to μ, that is to prove that we can recolor all the balls of B_I with their target coloring μ.

For every ball B_v of radius r centered in $v \in I$, we will define a set B'_v which is an extension of B_v. We might include some vertices at distance at most $r + 5$ from v in order to satisfy the conditions of Lemma 5.5. Since I is an independent set at distance $2r + 14$, for every $v, w \in I$, the sets B'_v and B'_w will be at distance at least 4. Let $B'_I = \cup_{v \in I} B'_v$. Since the diameter of each ball B'_v for $v \in S$ is $O(r)$ and all the balls of B'_I are disjoint, we will conclude the proof of the proposition using Lemma 5.5. The schedule length ${k'}^{O(\Delta)}$ follows from the bounds in the lemma, as well as $diam(X) \leqslant r + 5 = 12$, in the notations of Lemma 5.5.

In the rest of the proof, we restrict to a single ball B_v for $v \in I$ denoted by B for simplicity.

If a vertex of B has two neighbors in $V \setminus B$ colored the same or has degree less than $k - 2$, we set $B' = B$. Otherwise, let us prove that by adding a few vertices to B and doing a few recoloring steps, we can apply the Lemma 5.5. Note that no vertex of $V \setminus B$ is colored with k in η', since it agrees with μ, which is a k'-coloring with $k' < k$ by assumption.

Let us consider a path $v_1, v_2, \ldots, v_6$ of vertices such that v_i is at distance i from B. For every $i \in \{3, 4, 5\}$, we can obtain a desired set B' if one of the following holds:

- If $\deg(v_i) < \Delta$, then we simply take $B' = B \cup \{v_j : j \leqslant i\}$ which contains a vertex of degree less than Δ.
- If $N(v_i) \setminus v_{i-1}$ is not a clique, then let a, b be two neighbors of v_i that are non-adjacent. Then, since $d(a, B)$ and $d(b, B)$ are at least two, we can recolor a and b with k in η' (the coloring is proper since color k was not used in μ by assumption). Now, in this new coloring, $B' = B \cup \{v_j : j \leqslant i\}$ satisfies the condition. (We will recolor a and b to the right color at the very end of the algorithm.)

Let us now prove that one of the conditions above must hold. Assume, for the sake of contradiction, that for every $3 \leqslant i \leqslant 5$, $N(v_i) \setminus v_{i-1}$ is a clique and that all the v_i's have degree at least Δ.

Let z be a vertex of $N(v_3)$ distinct from v_2 and v_4 (which exists since $\Delta \geqslant 3$). The vertex z is at distance at most 4 from B. Moreover, $v_4 z$ is an edge (otherwise $N(v_3) \setminus v_2$ is not a clique). Since $N(v_4) \setminus v_3$ is a clique, $z v_5$

must also be an edge. But then $N(v_5) \setminus v_4$ cannot be a clique: that would mean that z and v_6 are adjacent, and then v_6 would be at distance 5 from B, which is a contradiction.

Now, by Lemma 5.5, we can recolor all the vertices of B' with the target coloring μ in such a way that every vertex of B' is recolored at most $\Delta^{O(\Delta r)}$ times (since the diameter of B' is at most the diameter of B plus 5). We then finally recolor, if needed, the two vertices recolored k in the second item of the construction of B' with their real target color in μ.

Since all the balls B' are disjoint and do not share an edge, we can apply these steps in parallel. Moreover, since they are at distance at least 4, the fact that we recolor a vertex at distance 5 from B can also be done in parallel. This completes the proof of Proposition 3.4. □

Acknowledgments

We would like to thank Valentin Gledel for fruitful discussions at the early stage of the project, as well as Quentin Ferro, for his helpful study on the specific case of toroidal grids.

BIBLIOGRAPHY

[1] L. Barenboim & M. Elkin, *Distributed Graph Coloring: Fundamentals and Recent Developments*, Synthesis Lectures on Distributed Computing Theory, Morgan & Claypool Publishers, 2013.

[2] L. Barenboim, M. Elkin & F. Kuhn, Distributed $(\Delta + 1)$-Coloring in Linear (in Δ) Time, *SIAM J. Comput.* **43** (2014), no. 1, p. 72–95, https://doi.org/10.1137/12088848X.

[3] V. Bartier, N. Bousquet, C. Feghali, M. Heinrich, B. Moore & T. Pierron, Recoloring Planar Graphs of Girth at Least Five, *SIAM J. Discret. Math.* **37** (2023), no. 1, p. 332–350, https://doi.org/10.1137/21m1463598.

[4] M. Bonamy & N. Bousquet, Recoloring graphs via tree decompositions, *Eur. J. Comb.* **69** (2018), p. 200–213.

[5] M. Bonamy, N. Bousquet & G. Perarnau, Frozen $(\Delta+ 1)$-colourings of bounded degree graphs, *Combinatorics, Probability and Computing* **30** (2021), no. 3, p. 330–343.

[6] M. Bonamy, M. Johnson, I. Lignos, V. Patel & D. Paulusma, Reconfiguration graphs for vertex colourings of chordal and chordal bipartite graphs, *Journal of Combinatorial Optimization* (2012), p. 1–12, https://doi.org/10.1007/s10878-012-9490-y.

[7] M. Bonamy, P. Ouvrard, M. Rabie, J. Suomela & J. Uitto, Distributed Recoloring, in *32nd International Symposium on Distributed Computing, DISC 2018*, LIPIcs, vol. 121, Schloss Dagstuhl - Leibniz-Zentrum für Informatk, 2018, https://doi.org/10.4230/LIPIcs.DISC.2018.12.

[8] P. S. Bonsma & L. Cereceda, Finding Paths between graph colourings: PSPACE-completeness and superpolynomial distances, *Theor. Comput. Sci.* **410** (2009), no. 50, p. 5215–5226, https://doi.org/10.1016/j.tcs.2009.08.023.

[9] N. Bousquet & V. Bartier, Linear Transformations Between Colorings in Chordal Graphs, in *27th Annual European Symposium on Algorithms, ESA 2019*, 2019, https://doi.org/10.4230/LIPIcs.ESA.2019.24.

[10] N. Bousquet, L. Feuilloley, M. Heinrich & M. Rabie, Distributed Recoloring of Interval and Chordal Graphs, in *25th International Conference on Principles of Distributed Systems, OPODIS 2021* (Q. Bramas, V. Gramoli & A. Milani, eds.), LIPIcs, vol. 217, 2021, https://doi.org/10.4230/LIPIcs.OPODIS.2021.19.

[11] N. Bousquet & M. Heinrich, A polynomial version of Cereceda's conjecture, *Journal of Combinatorial Theory, Series B* **155** (2022), p. 1–16.

[12] N. Bousquet & G. Perarnau, Fast recoloring of sparse graphs, *Eur. J. Comb.* **52** (2016), p. 1–11.

[13] S. Cambie, W. C. v. Batenburg & D. W. Cranston, Optimally reconfiguring list and correspondence colourings, *Eur. J. Comb.* **115** (2024), article no. 103798, https://doi.org/10.1016/J.EJC.2023.103798.

[14] K. Censor-Hillel, Y. Maus, S. R. Peled & T. Tonoyan, Distributed Vertex Cover Reconfiguration, in *13th Innovations in Theoretical Computer Science Conference, ITCS 2022*, LIPIcs, vol. 215, 2022, https://doi.org/10.4230/LIPIcs.ITCS.2022.36.

[15] K. Censor-Hillel & M. Rabie, Distributed reconfiguration of maximal independent sets, *J. Comput. Syst. Sci.* **112** (2020), p. 85–96, https://doi.org/10.1016/j.jcss.2020.03.003.

[16] L. Cereceda, Mixing Graph Colourings, PhD Thesis, London School of Economics and Political Science, 2007.

[17] L. Cereceda, J. van den Heuvel & M. Johnson, Mixing 3-colourings in bipartite graphs, *Eur. J. Comb.* **30** (2009), no. 7, p. 1593–1606, https://doi.org/10.1016/j.ejc.2009.03.011.

[18] S. Chen, M. Delcourt, A. Moitra, G. Perarnau & L. Postle, Improved Bounds for Randomly Sampling Colorings via Linear Programming, in *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019*, 2019, p. 2216–2234, https://doi.org/10.1137/1.9781611975482.134.

[19] R. Diestel, *Graph Theory*, third ed., Graduate Texts in Mathematics, vol. 173, Springer-Verlag, Heidelberg, 2005.

[20] Z. Dvorák & C. Feghali, A Thomassen-type method for planar graph recoloring, *Eur. J. Comb.* **95** (2021), article no. 103319, https://doi.org/10.1016/j.ejc.2021.103319.

[21] M. Dyer, A. D. Flaxman, A. M. Frieze & E. Vigoda, Randomly coloring sparse random graphs with fewer colors than the maximum degree, *Random Structures & Algorithms* **29** (2006), no. 4, p. 450–465.

[22] C. Feghali, Paths between colourings of sparse graphs, *Eur. J. Comb.* **75** (2019), p. 169–171.

[23] C. Feghali, M. Johnson & D. Paulusma, A Reconfigurations Analogue of Brooks' Theorem and Its Consequences, *Journal of Graph Theory* **83** (2016), no. 4, p. 340–358.

[24] A. Frieze & E. Vigoda, A survey on the use of Markov chains to randomly sample colourings, *Oxford Lecture Series in Mathematics and its Applications* **34** (2007), p. 53–71.

[25] M. Ghaffari & F. Kuhn, Deterministic Distributed Vertex Coloring: Simpler, Faster, and without Network Decomposition, in *62nd IEEE Annual Symposium*

on Foundations of Computer Science, FOCS 2021, IEEE, 2021, p. 1009–1020, https://doi.org/10.1109/FOCS52979.2021.00101.

[26] S. Gupta, M. Kumar & S. Pai, Brief Announcement: Distributed Reconfiguration of Spanning Trees, in *Stabilization, Safety, and Security of Distributed Systems - 24th International Symposium, SSS 2022*, vol. 13751, 2022, p. 346–351, https://doi.org/10.1007/978-3-031-21017-4_25.

[27] J. van den Heuvel, The Complexity of change, Part of London Mathematical Society Lecture Note Series, p. 409, Part of London Mathematical Society Lecture Note Series, Cambridge University Press, S. R. Blackburn, S. Gerke, and M. Wildon ed., 2013.

[28] N. Nishimura, Introduction to reconfiguration, *Algorithms* **11** (2018), no. 4, article no. 52 (25 pages).

[29] A. Panconesi & A. Srinivasan, The local nature of Δ-coloring and its algorithmic applications, *Combinatorica* **15** (1995), no. 2, p. 255–280.

[30] D. Peleg, *Distributed computing: a locality-sensitive approach*, SIAM, 2000.

[31] J. Suomela, Survey of local algorithms, *ACM Computing Surveys (CSUR)* **45** (2013), no. 2, p. 1–40.

[32] E. Vigoda, Improved Bounds for Sampling Colorings, in *40th Annual Symposium on Foundations of Computer Science, FOCS '99, 17-18 October, 1999, New York, NY, USA*, 1999, p. 51–59, https://doi.org/10.1109/SFFCS.1999.814577.

Manuscript received 17th October 2023,
accepted 21st January 2025.

Nicolas BOUSQUET
CNRS, INSA Lyon, UCBL, LIRIS, UMR5205,
F-69621, Lyon, France
nicolas.bousquet@univ-lyon1.fr

Laurent FEUILLOLEY
CNRS, INSA Lyon, UCBL, LIRIS, UMR5205,
F-69621, Lyon, France
laurent.feuilloley@cnrs.fr

Marc HEINRICH
Leeds University, UK
m.heinrich@leeds.ac.uk

Mikaël RABIE
Université Paris Cité, CNRS, IRIF, F-75013, Paris,
France
mikael.rabie@irif.fr

CENTRE MERSENNE

Innovations in Graph Theory is a member of the Mersenne Center for Open Publishing
ISSN: 3050-743X