

Online Edge Coloring Algorithms via the Nibble Method

Sayan Bhattacharya¹, Fabrizio Grandoni², and David Wajc^{*3}

¹University of Warwick

²IDSIA, USI-SUPSI

³Stanford University

Abstract

Nearly thirty years ago, Bar-Noy, Motwani and Naor [IPL'92] conjectured that an online $(1 + o(1))\Delta$ -edge-coloring algorithm exists for n -node graphs of maximum degree $\Delta = \omega(\log n)$. This conjecture remains open in general, though it was recently proven for bipartite graphs under *one-sided vertex arrivals* by Cohen et al. [FOCS'19]. In a similar vein, we study edge coloring under widely-studied relaxations of the online model.

Our main result is in the *random-order* online model. For this model, known results fall short of the Bar-Noy et al. conjecture, either in the degree bound [Aggarwal et al. FOCS'03], or number of colors used [Bahmani et al. SODA'10]. We achieve the best of both worlds, thus resolving the Bar-Noy et al. conjecture in the affirmative for this model.

Our second result is in the adversarial online (and dynamic) model with *recourse*. A recent algorithm of Duan et al. [SODA'19] yields a $(1 + \epsilon)\Delta$ -edge-coloring with $\text{poly}(\log n/\epsilon)$ recourse. We achieve the same with $\text{poly}(1/\epsilon)$ recourse, thus removing all dependence on n .

Underlying our results is one common offline algorithm, which we show how to implement in these two online models. Our algorithm, based on the Rödl Nibble Method, is an adaptation of the distributed algorithm of Dubhashi et al. [TCS'98]. The Nibble Method has proven successful for distributed edge coloring. We display its usefulness in the context of online algorithms.

*Work done while the author was at Carnegie Mellon University.

1 Introduction

Edge coloring is the problem of assigning one of k colors to all edges of a simple graph, so that no two incident edges have the same color. The objective is to minimize the number of colors, k . The edge coloring problem goes back to the 19th century and studies of the four-color theorem [39, 41]. In 1916, König [33], in what many consider to be the birth of matching theory, proved that any bipartite graph of maximum degree Δ is colorable using Δ colors. (Clearly, no fewer colors suffice.) Nearly half a century later, Vizing [42] proved that any general graph is $(\Delta + 1)$ -edge-colorable. Vizing’s proof is algorithmic, yielding such a coloring in polynomial time. This is likely optimal, as it is NP-hard to determine if a general graph is Δ -edge-colorable [27]. Algorithms for the edge coloring problem have been studied in several different models of computation, including offline, online, distributed, parallel, and dynamic models (see, e.g., [9, 10, 11, 13, 14, 30, 38, 40] and references therein.) In this work, we study the edge coloring problem in online settings.

Online edge coloring: Here, an adversary picks an n -node graph G of maximum degree Δ (the algorithm knows n and Δ , but not G), and then reveals the edges of G one at a time. Immediately after the arrival of an edge, the algorithm must irrevocably assign a color to it, with the objective of minimizing the final number of colors used. This problem was first studied nearly thirty years ago, by Bar-Noy, Motwani, and Naor [4]. They showed that the greedy algorithm, which returns a proper $(2\Delta - 1)$ -edge coloring, is worst-case optimal among online algorithms. This might seem to be the end of the story for this line of research. However, as pointed out by Bar-Noy et al. [4], their lower bound only holds for bounded-degree graphs, with some $\Delta = O(\log n)$. This then led them to conjecture that online $(1 + o(1))\Delta$ -edge-coloring is possible for graphs with $\Delta = \omega(\log n)$. This conjecture remains wide open.

Recently, an important progress was made towards proving the Bar-Noy et al. conjecture: Cohen et al. [11] showed how to obtain a $(1 + o(1))\Delta$ -edge coloring for bipartite graphs in an online setting under *node arrivals* (together with their edges). This is a relaxation of the online edge-arrival model. Thus, this latter result can be seen as an intermediate step towards the ultimate goal of proving the Bar-Noy et al. conjecture. In a similar spirit, we consider edge coloring in two well-studied relaxations of the online model, that act as intermediate steps towards the Bar-Noy et al. conjecture, and make substantial progress on the state-of-the-art results in both these settings.

(I) Random-order online edge coloring: Here, an adversarially-chosen graph has its edges revealed to the algorithm in *uniformly random order*. Such random-order arrivals, which capture numerous stochastic arrival models, have been widely studied for many online problems. (See, e.g., [29, 31, 34, 35, 37] and the survey by Gupta and Singla [26] and references therein.) In the context of edge coloring, this model was studied by [1, 3]. Aggarwal et al. [1] were the first to show that high Δ suffices for near-ideal coloring in this model, giving a $(1 + o(1))\Delta$ -edge-coloring algorithm for *multigraphs* with $\Delta = \omega(n^2)$. Bahmani et al. [3] then breached the greedy $2\Delta - 1$ barrier for simple graphs with polylogarithmic Δ , giving a 1.43Δ -edge coloring algorithm for $\Delta = \omega(\log n)$ (improved to 1.26Δ in their journal version). This leads to the following natural open question: can one obtain “the best of both worlds” w.r.t. [1, 3]? That is, can one obtain a $(1 + o(1))\Delta$ -edge-coloring for graphs of maximum degree $\Delta = \omega(\log n)$ whose edges are presented in random order? Put another way, is the Bar-Noy et al. conjecture true for random-order edge arrivals? We answer this question in the affirmative.

Theorem 1.1. *For some absolute constant $\gamma \in (0, 1)$, there exists an online algorithm that, when given a graph G of maximum degree $\Delta = \omega(\log n)$, whose edges are presented in random order, computes a proper $(\Delta + O(\Delta^\gamma \cdot \log^{1-\gamma} n)) = (1 + o(1))\Delta$ -edge-coloring of G w.h.p.*

We complement this upper bound with a lower bound showing that, for some $\Delta = O(\log n)$, not only is it impossible to guarantee a $(1 + o(1))\Delta$ -edge-coloring under random-order arrivals, but it is even impossible to use any fewer than $2\Delta - 1$ colors: see Appendix A.

We note that previous random-order online edge coloring algorithms [1, 3] required the graph to be Δ -regular. This assumption is without loss of generality in an offline setting, but it is unclear whether the same holds in the random-order online model. Our algorithm from Theorem 1.1, however, works on any graph (including non-regular ones): this is discussed in Section 3.1.

(II) Dynamic edge coloring with recourse: Another widely-studied relaxation of online algorithms is online algorithms with *recourse*. Here, an algorithm must make immediate choices upon each arrival, but is also allowed to make a small number of changes to its solution after each arrival (referred to as *recourse*). The concept of recourse helps us understand the robustness/sensitivity of (near-)optimal solutions. Accordingly, an influential line of research in the online algorithms community is devoted to studying the tradeoffs between the solution quality and recourse for many well-known problems [6, 8, 19, 22, 23, 24, 25, 36].

Many results for bounded-recourse online algorithms hold in a more general, *dynamic* setting. In the context of edge coloring, the dynamic (oblivious) version of the problem with recourse is captured by the following scenario: The input graph G changes via a sequence of $\text{poly}(n)$ *updates* chosen in advance by an adversary, where each update consists of an edge insertion or deletion in G . The maximum degree in G remains at most Δ throughout. The algorithm maintains a proper edge coloring, while changing the colors of some edges in G after each update (the number of such changes per update is the recourse of the algorithm). The challenge is to design an algorithm that simultaneously (a) maintains a proper coloring with few colors and (b) has small recourse.

In recent years, the edge coloring problem has been extensively studied from a different, but highly related, perspective of *dynamic data structures* [5, 7, 14, 43]. Here, the goal is to maintain a proper edge coloring with few colors in a dynamic graph, taking little time after each edge update (insertion/deletion), where this time is referred to as *update time*. Note that the update time of any data structure for dynamic edge coloring upper bounds its recourse, since the data structure has to spend at least $\Omega(1)$ time per edge which changes its color after an update.

The state-of-the-art result for dynamic edge coloring with recourse follows from the work of Duan et al. [14]. In any dynamic graph with $\Delta = \Omega(\log^2 n / \epsilon^2)$, their algorithm maintains a proper $(1 + \epsilon)\Delta$ -edge coloring with $\text{poly}(\log n, 1/\epsilon)$ recourse. Given that other dynamic problems are known to admit super-constant recourse lower bounds (see, e.g., [19]), it is natural to ask if one can get a recourse bound for $(1 + \epsilon)\Delta$ -edge coloring that is independent of n . We answer this question in the affirmative.

Theorem 1.2. *There is an algorithm that maintains a proper $(1 + \epsilon)\Delta$ -edge-coloring w.h.p., with $\text{poly}(1/\epsilon)$ expected recourse in dynamic graphs of maximum degree $\Delta = \Omega(\log n / \text{poly}(\epsilon))$.*

1.1 Our Techniques

At the heart of both our results is one common algorithmic approach, inspired by the Rödl Nibble Method [2], as applied to distributed edge coloring by Dubhashi et al. [15]. This method and its variants have since found further uses in distributed settings [9, 18]. To the best of our knowledge, we are the first to export this method to online settings.

We analyze our basic algorithm, which is a variant of [15], in an offline model. We then show how to implement this algorithm in online and dynamic settings, from which we obtain our results. We now outline this basic algorithm, and the ideas needed to implement it in the models we study. For simplicity, we focus on Δ -regular graphs in this section.

The High-Level Framework: The Nibble Method in the framework of edge coloring was first used in [15] in the distributed model. Let us sketch how their algorithm would work in the offline setting. The algorithm consists of multiple rounds. In each round, each vertex v selects a random ϵ fraction of its incident uncolored edges. Each sampled edge e chooses a tentative color u.a.r. among the colors in $[\Delta]$ not yet taken by incident edges (*palette* of e). We then assign the tentative color $c(e)$ to sampled edges e for which no incident edge e' picked the same tentative color $c(e)$, else we mark e as *failed*, and leave e uncolored. It turns out that each sampled edge fails at each round with probability $O(\epsilon)$. Crucially, picking ϵ appropriately results in a number of important parameters (degrees in the uncolored subgraph, palette sizes, etc') behaving in a predictable manner, and being sharply concentrated around their mean, w.h.p. In particular, this results in the uncolored subgraph's maximum degree decreasing w.h.p. at a rate of roughly $1 - \epsilon$ per application of this subroutine, or *round*. Consequently, some $t_\epsilon = O(\log(1/\epsilon)/\epsilon)$ rounds leave an uncolored subgraph of maximum degree $\Delta' = \text{poly}(\epsilon)\Delta$ w.h.p., which can then be greedily colored using a further $2\Delta' = \text{poly}(\epsilon)\Delta$ colors. This approach therefore yields a proper $(1 + \text{poly}(\epsilon))\Delta$ edge coloring.

In part inspired by [9], we consider a slight modification of the above algorithm which is more convenient for our goals. In more detail, we make the following changes:

- (1) We do not attempt to re-color an edge e which fails in a given round in future rounds, instead leaving e to be colored greedily in the final stage. Intuitively, ignoring these edges still results in a low-degree uncolored graph after t_ϵ rounds, since few edges incident to each vertex fail.
- (2) Whenever an edge e picks a tentative color c , we remove c from the palettes of its incident edges even if e fails. Intuitively, this does not decrease the palette sizes much, again, since few edges incident to each vertex fail.
- (3) We sample each edge independently with probability ϵ in each round.

Our modifications bring two main advantages. First, the analysis can be substantially simplified: rather than using a specialized concentration inequality of Grable [21], we mostly use Hoeffding bounds for negatively-associated variables (see Appendix E). This allows us to provide a relatively concise, but complete analysis for sub-constant values of ϵ and for non-regular graphs. Second, and importantly for us, it is easier to adapt the modified algorithm to the online settings that we study.

Random-Order Online Implementation: To obtain our results for random-order arrivals, we first observe that our edge-centric sampling of modification (3) allows us to use the randomness of edge arrivals to “sample edges for us”. More formally, we implement the independent edge-sampling part of each round by considering an appropriate binomially-distributed prefix of the remaining edges (relying on our knowledge of the number of edges of the Δ -regular graph, $m = \frac{n\Delta}{2}$). This results in each remaining edge of the graph being sampled independently with probability ϵ .

For each round, we have each edge of the round sample a tentative color u.a.r. from its palette. In this online setting, however, we cannot always tell when an edge arrives whether it picked the same tentative color as its incident edges of the same round (since some of these arrive *later*). We therefore assign the tentative color $c(e)$ to sampled edges e for which no *previous* incident edge e' picked the same tentative color $c(e)$, else we mark e as *failed*. Modification (2) in the basic algorithm implies that this change still results in a feasible (partial) coloring. On the other hand, the uncolored subgraph “after” the rounds in this algorithm clearly has lower maximum degree than its counterpart in the basic algorithm, and so greedily coloring this subgraph requires fewer colors than the same stage of the basic algorithm. Finally, modification (1) of our basic algorithm, whereby we do not attempt to re-color a failed edge in “future rounds” (which would require knowledge of future arrivals), implies that we can greedily color every failed edge before the next edge arrives. So, by the analysis of our basic algorithm, we obtain Theorem 1.1 for Δ -regular graphs. In Section 3 we build on this approach to obtain our full result, for general graphs.

Low-Recourse Implementation: For our low-recourse dynamic implementation, we show how to maintain, after each update, a coloring drawn from the same distribution as that of the basic algorithm applied to the current graph. Modifications (1) and (3) of the basic algorithm imply that deciding the round in which we sample any edge can be done in advance (prior to any arrival), sampling this round number from the appropriately capped geometric distribution. The more delicate point is dealing with the choice of tentative colors. For example, when an edge is added, its tentative color is removed from the palettes of its incident edges of later rounds. Thus, some incident edges no longer have a tentative color which is u.a.r. drawn from their current palette. In Section 4 we show that a natural approach of correcting these distributions—sampling a new tentative color if the previous one is no longer valid, and switching to a new color if one samples a newly-available color—results in bounded recourse. In particular, building on our analysis of the basic algorithm, we show that the expected number of edges of round $i + 1$ whose tentative color changes due to the change of a tentative color of an edge in round $j \leq i$ is at most $O(\epsilon)$. Therefore, for any update, the number of tentative color changes is at most $(1 + O(\epsilon))^{t_\epsilon} = \text{poly}(1/\epsilon)$, from which we obtain Theorem 1.2. In Section 4 we formalize this approach and its analysis

1.2 Further Related Work

Other than [15], most closely related to our approach are other distributed edge coloring algorithms using the Nibble Method and its variants and extensions [9, 18]. While these distributed algorithms improve on [15], they require crucially that edges be tentatively colored in multiple rounds—a design pattern which seems hard to implement in online settings. Another approach is suggested by the work of Cohen et al. [11] for bipartite one-sided vertex arrivals; using an online matching algorithm of Cohen and Wajc [12] which matches each edge with probability $\frac{1-o(1)}{\Delta}$, they color roughly one edge of each maximum-degree node per round, resulting in a $(1 + o(1))\Delta$ -edge-coloring. Unfortunately, for the models we study, no such matching algorithm is known—ruling out this approach.

Returning to previous algorithms in our models, we note that the algorithm of Duan et al. [14], which uses an augmenting path based approach, has an inherent polylogarithmic recourse. On the other hand, the approaches of [1, 3] for random-order arrivals seem challenging to extend to dynamic recourse-bounded algorithms. Moreover, in the random-order online model, it is unclear how to provably achieve a $(1 + o(1))\Delta$ -edge-coloring for simple graphs with $\Delta = \omega(\log n)$ by using

the ideas in those papers. In this work we show that the Nibble Method, and in particular, a variant of the algorithm of [15], allows us to obtain such desired results in *both* these online models.

2 The Basic Algorithm

In this section, we describe our basic algorithm for near-regular graphs in the *static setting*, and state the key theorem needed for its analysis. We defer a more detailed analysis to Appendix C. Our online and dynamic algorithms in subsequent sections will be built on top of this basic algorithm.

The input to the algorithm is a graph $G = (V, E)$ with $|V| = n$ nodes, where the degree of each node lies in the interval $[(1 - \epsilon^2)\Delta, (1 + \epsilon^2)\Delta]$. The parameter ϵ satisfies the following condition:

$$1/10^4 \geq \epsilon \geq 10 \cdot (\ln n / \Delta)^{1/6}. \quad (1)$$

Note that such ϵ exist if $\Delta = \Omega(\log n)$ is large enough. The algorithm runs in two *phases*, as follows.

Phase One. In phase one, the algorithm properly colors a subset of edges of G using $(1 + \epsilon^2)\Delta$ colors, while leaving an uncolored subgraph of small maximum degree. This phase consists of $t_\epsilon - 1$ rounds $\{1, \dots, t_\epsilon - 1\}$, for

$$t_\epsilon := \lfloor \ln(1/\epsilon) / (2K\epsilon) \rfloor, \text{ and } K = 48. \quad (2)$$

Each round $i \in [t_\epsilon - 1]$ operates on a subgraph $G_i := (V, E_i)$ of the input graph (with $E_1 = E$), identifies a subset of edges $S_i \subseteq E_i$, picks a *tentative* color $c(e) \in [(1 + \epsilon^2)\Delta] \cup \{\text{null}\}$ for each edge $e \in S_i$, and returns the remaining set of edges $E_{i+1} = E_i \setminus S_i$ for the next round. Thus, we have: $E = E_1 \supseteq E_2 \supseteq \dots \supseteq E_{t_\epsilon}$. We now describe how a given round $i \in [t_\epsilon - 1]$ works. We start by defining a couple of important notations.

- (a) For all $v \in V$, let $P_i(v) := \{\chi \in [(1 + \epsilon^2)\Delta] : \chi \neq c(u, v) \text{ for all } (u, v) \in \bigcup_{j < i} S_j\}$ denote the *palette* of the node v for round i . A color $\chi \in [(1 + \epsilon^2)\Delta]$ belongs to $P_i(v)$ iff no edge incident on v has tentatively picked the color χ in previous rounds $j < i$. (b) Similarly, for all $(u, v) \in E_i$, let $P_i(u, v) := P_i(u) \cap P_i(v)$ denote the *palette* of the edge (u, v) for round i .

In round i , we first sample each edge $e \in E_i$ *independently* with probability ϵ . Let $S_i \subseteq E_i$ be the set of sampled edges. Next, each edge $e \in S_i$ with $P_i(e) \neq \emptyset$ tentatively picks a color $c(e)$ from its palette $P_i(e)$ uniformly and independently at random. We say that an edge $e \in S_i$ *failed* in round i iff either (a) $P_i(e) = \emptyset$ (in this case we set $c(e) := \text{null}$), or (b) among the edges $N(e) \subseteq E$ that are adjacent to e , there is some edge $e' \in S_i$ that tentatively picked the same color (i.e., $c(e) = c(e')$). Let $F_i \subseteq S_i$ denote the set of failed edges in round i . The remaining sampled edges $e \in S_i \setminus F_i$ are called *successful* in round i . Each such edge $e \in S_i \setminus F_i$ is *assigned* the color $c(e)$ it tentatively picked in round i . Before terminating the current round, we set $E_{i+1} := E_i \setminus S_i$ and $G_{i+1} := (V, E_{i+1})$. We remark that the color tentatively sampled by a failed edge e cannot be used by the edges incident to e in subsequent rounds. This will prove useful both for our analysis and when implementing this algorithm in other models in subsequent sections.

Phase Two. Finally, in phase two, we greedily color all edges that were not successful in phase one. That is, letting $G_F := (V, \cup_i F_i)$ be the subgraph consisting of all the edges that failed in phase one, and $G_{t_\epsilon} := (V, E_{t_\epsilon})$ be the subgraph consisting of all the edges that were never sampled in

Algorithm 1 The Basic Algorithm

- 1: $E_1 \leftarrow E$ and $G_1 \leftarrow (V, E_1)$
 - 2: **for** $i = 1, 2, \dots, t_\epsilon - 1$ **do**
 - 3: $S_i \leftarrow \emptyset$
 - 4: **for each** $e \in E_i$ **independently do**
 - 5: With probability ϵ , add e to S_i
 - 6: $P_i(e) \leftarrow [(1 + \epsilon^2)\Delta] \setminus \{c(e') \mid e' \in N(e) \cap \bigcup_{j < i} S_j\}$.
 - 7: If $P_i(e) \neq \emptyset$, sample $c(e) \sim_R P_i(e)$, else set $c(e) \leftarrow \text{null}$ \triangleright Tentative coloring of e
 - 8: Let $F_i \leftarrow \{e \in S_i \mid c(e) \in \{\text{null}\} \cup \{c(e') \mid e' \in N(e) \cap S_i\}\}$ \triangleright The set of failed edges
 - 9: Color each edge $e \in S_i \setminus F_i$ using color $c(e)$
 - 10: $E_{i+1} \leftarrow E_i \setminus S_i$ and $G_{i+1} \leftarrow (V, E_{i+1})$.
 - 11: Let $G_F := (V, \bigcup_i F_i)$ denote the subgraph of G consisting of the failed edges from phase one.
 - 12: Color $G_{t_\epsilon} \cup G_F$ greedily using colors $(1 + \epsilon^2)\Delta + 1, (1 + \epsilon^2)\Delta + 2, \dots, (1 + \epsilon^2)\Delta + 2\Delta(G_{t_\epsilon} \cup G_F) - 1$.
-

phase one, we color the edges of $G_{t_\epsilon} \cup G_F$ greedily, using a new palette of $2\Delta(G_{t_\epsilon} \cup G_F) - 1$ colors. Here $\Delta(H)$ denotes the maximum degree in any graph H .

The algorithm's pseudocode is given in Algorithm 1. We now turn to discussing its analysis.

Observation 2.1. *Algorithm 1 outputs a proper $((1 + \epsilon^2)\Delta + 2\Delta(G_{t_\epsilon} \cup G_F) - 1)$ -edge-coloring of the input graph $G = (V, E)$.*

Proof. First observe that the algorithm computes a valid partial coloring in Phase One. Indeed, any $e \in S_i \setminus F_i$ selects a color $c(e) \in P_i(e) \subseteq [(1 + \epsilon^2)\Delta]$, and the definition of $P_i(e)$ and F_i guarantees that no other edge $e' \in N(e)$ in any round of Phase One can be colored with $c(e)$. The claim follows by observing that in Phase One we use only colors from $[(1 + \epsilon^2)\Delta]$, while in Phase Two the greedy algorithm uses a disjoint set of at most $2\Delta(G_{t_\epsilon} \cup G_F) - 1$ extra colors. \square

The key property of the basic algorithm is captured in the following theorem.

Theorem 2.2. $\Delta(G_{t_\epsilon} \cup G_F) = O(\epsilon^{1/(3K)}\Delta)$ *w.h.p.*

Corollary 2.3. *The basic algorithm $(\Delta + O(\epsilon^{1/(3K)}\Delta))$ edge colors G , *w.h.p.**

Proof. Follows from Theorem 2.2 and Observation 2.1. \square

In some sense, the arguments behind the proof of Theorem 2.2 were already apparent in the work of Dubhashi et al. [15]. Consequently, we defer a complete and self-contained proof of this theorem to Appendix C. For now, we turn to exploring implications of this theorem and Algorithm 1 to online edge coloring.

3 Random-Order Online Algorithm

In this section we present algorithms which (essentially) implement Algorithm 1 in the random-order online model. We start with a warm-up case, where the input graph is near-regular, and we know the value of m (the number of edges in the final graph).

3.1 Warm-up: Near-Regular Graphs with Known m

One subroutine we rely on is the ability to use the stream's randomness to simulate independent sampling of edges. For completeness, we provide a proof of the following simple fact in Appendix B.

Fact 3.1. *Consider a universe U of n elements, and let $p \in [0, 1]$. Let $U_k \subseteq U$ denote the first k elements in a random-order stream of U , and let $X \sim \text{Bin}(n, p)$ be a binomial random variable with parameters n and p . Then the random set U_X contains every element in U independently with probability p .*

Using Fact 3.1, we simulate (a variant of) Algorithm 1 with parameter ϵ under random-order edge arrivals in a graph $G = (V, E)$ with m edges and n nodes, where the degree of each node lies in the interval $(1 \pm \epsilon^2)\Delta$, and $\Delta = \omega(\log n)$. The algorithm knows n, Δ and m (but not G).

Warm-up Algorithm: Set $\epsilon := 10 \cdot (\ln n / \Delta)^{1/6}$ (see (1)). For round $i = 1, \dots, t_\epsilon - 1$, sample an independent random variable $X_i \sim \text{Bin}(m - \sum_{j < i} X_j, \epsilon)$, and let S_i be the set of edges in G whose positions in the random-order stream lie in the interval $(\sum_{j < i} X_j, \sum_{j \leq i} X_j]$. As with Algorithm 1, each edge $e \in S_i$, upon its arrival, samples a tentative color

$$c(e) \sim_R P_i(e) := [(1 + \epsilon^2)\Delta] \setminus \{c(e') \mid e' \in N(e) \cap S_j, j < i\},$$

where we set $c(e) \leftarrow \text{null}$ if $P_i(e) = \emptyset$. Unlike in Algorithm 1, in this online setting the algorithm cannot know whether the color $c(e)$ conflicts with the tentative color of a neighboring edges $e' \in N(e) \cap S_i$ that arrives in the same round i , but *after* e in the stream. Hence, we color each edge $e \in S_i$ with its tentative color $c(e)$, unless $c(e) = \text{null}$ or some *previously-arrived* neighboring edge $e' \in N(e) \cap S_i$ also picked color $c(e') = c(e)$. In the latter case, we instead color e greedily with the first available color $j > (1 + \epsilon^2)\Delta$. We let F'_i be the edges in S_i which are colored greedily.

As we show, this online algorithm inherits the performance of the basic Algorithm 1.

Theorem 3.2. *For some absolute constant $\gamma \in (0, 1)$, the warm-up algorithm described above yields a proper $(\Delta + O(\Delta^\gamma \cdot \log^{1-\gamma} n)) = (1 + o(1))\Delta$ -edge coloring of G w.h.p.*

Proof. This algorithm outputs a valid edge coloring, as it colors every edge (due to the greedy stage) and never assigns an edge a color used by an incident edge. It remains to bound its performance.

For any $i \geq 0$, Let E_i be the set of edges whose positions in the random-order stream lie in the interval $(\sum_{j < i} X_j, m]$. By Fact 3.1, the set of edges S_i is a random subset of E_i which contains each edge in E_i independently with probability ϵ . A simple induction on i shows that the sets S_i and E_i share the same distributions as their counterparts in Algorithm 1. Next, denote by $F_i \supseteq F'_i$ the set of edges $e \in S_i$ for which $c(e) \in \{\text{null}\} \cup \{c(e') \mid e' \in N(e) \cap S_i\}$. Since each edge $e \in S_i$ picks a color uniformly at random from the set of colors not picked by any of its neighboring edges in previous rounds (including the edges in F_j for all $j < i$), a simple induction on i shows that the random variables F_i and $c(e)$ in this algorithm are distributed exactly as their counterparts in Algorithm 1. Consequently, the upper bounds on $\Delta(\bigcup_i F_i) \geq \Delta(\bigcup_i F'_i)$ and $\Delta(G_{t_\epsilon})$ of Algorithm 1 hold for this online algorithm as well. Therefore, the greedy (online) algorithm colors the uncolored edges in $G_{t_\epsilon} \cup G_F$ using at most $2 \cdot \Delta(G_{t_\epsilon} \cup G_F) - 1 = O(\epsilon^{1/(3K)} \Delta)$ colors w.h.p., by Theorem 2.2 and our choice of $\epsilon = 10 \cdot (\ln n / \Delta)^{1/6}$, as in (1). As we use $(1 + \epsilon^2)\Delta$ distinct colors for all other edges, this online algorithm uses $\Delta + O(\epsilon^{1/(3K)} \Delta)$ colors overall w.h.p. Since $\Delta = \omega(\log n)$ and K is an absolute constant (see (2)), the theorem follows from our choice of ϵ . \square

Assuming near-regularity, and known m . The assumption of near-regularity used by the above algorithm is common in the literature. Indeed, all prior random-order online edge-coloring algorithms assume perfect regularity [1, 3]. As pointed out in those papers, this assumption is without loss of generality in the offline model, where we can add dummy edges to make the graph regular. In a random-order online setting, this is problematic, however, as these dummy edges should be interspersed among the real edges to create a regular graph *presented in random order*. This last point seems impossible without prior knowledge of vertices' final degrees, and the number of edges, m , which we assume prior knowledge of. In the next section we show how to remove the assumption of near-regularity, as well as knowledge of m , while retaining the asymptotic performance of Theorem 3.2.

3.2 General Graphs

We now present and analyze our random-order online edge coloring algorithm for general graphs $G = (V, E)$ with n nodes, m edges and maximum degree $\Delta = \omega(\log n)$. In particular, we do not assume that all nodes have degree close to Δ . The algorithm knows n, Δ ; but does *not* know m nor G . Let e_1, \dots, e_m be the random stream of edges, $G^{(k)}$ be the subgraph induced by e_1, \dots, e_k , and $d^{(k)}(v)$ be the degree of node v in $G^{(k)}$. Our key insight is to observe the first few edges in the input stream until some node reaches degree $\epsilon\Delta$. This is sufficient to infer (approximately) the value of m and the degree of each node in G . At the same time, we can afford to color such initial set of edges greedily.

In more detail, our algorithm consists of 3 main steps. In Step (I), we observe the first T edges until some node v reaches the degree $d^{(T)}(v) = \epsilon\Delta$ (or we reach the end of the stream). This first set of edges is colored greedily using the first available color. Let Δ_1 be the largest color used in Step (I). The following technical lemma follows from a standard application of Chernoff bounds over sums of negatively-associated variables (proof in Appendix B).

Lemma 3.3. *Let $\epsilon \leq \frac{1}{2}$, and let $\alpha > 0$ be a constant, and assume $\Delta \geq \frac{24(\alpha+3)\ln n}{\epsilon^8}$. Then, with probability at least $1 - O(n^{-\alpha})$, the following properties hold:¹*

1. $T = \epsilon \cdot m(1 \pm \epsilon^2)$.
2. $d^{(T)}(v) = \epsilon \cdot d(v) \pm 2\epsilon^3\Delta$ for every node v .
3. Let $m' := T/(\epsilon(1 + \epsilon^2))$. Conditioned on $m' \leq m$, every node v has $d(v) - d^{(m')}(v) \leq 2\epsilon^2\Delta$.

Henceforth, we assume that all the high-probability events in Lemma 3.3 actually occur (otherwise the algorithm fails). In Step (II), we color the next $m' - T$ edges $R := \{e_{T+1}, \dots, e_{m'}\}$ using colors larger than Δ_1 , as described below.

Let $G_R = (V, R)$ denote the subgraph of G induced by the edges in R . Before processing the $(T + 1)^{th}$ update e_{T+1} , we virtually expand G_R by adding *dummy nodes* W and *dummy edges* D in the following manner. For each node $v \in V$, create Δ dummy nodes $v_1, v_2, \dots, v_\Delta$ which form a Δ -clique via dummy edges, and add extra dummy edges from v to $\max\{0, \Delta - (1/\epsilon - 1) \cdot d^{(T)}(v)\}$ of these dummy nodes $\{v_1, \dots, v_\Delta\}$. Let H be the resulting graph. Note that at this point we only know the dummy edges in H , as the edges in G_R will arrive in future.

¹We let $c = a \pm b$ denote $c \in [a - b, a + b]$.

Let \mathcal{A} denote the warm-up online algorithm from Section 3.1. In Step (II), we run this online algorithm \mathcal{A} with parameter 2ϵ on H , where the edges of H are presented to \mathcal{A} in *random order*. More precisely, initializing $j = T + 1$, $D' = D$ and $R' = R$, we perform the following operations for $|R| + |D|$ iterations.

- With probability $|D'|/(|R'| + |D'|)$, we sample a random edge e_d from D' , and feed the edge e_d to the online algorithm \mathcal{A} . We then set $D' = D' \setminus \{e_d\}$ before going to the next iteration.
- With remaining probability, we feed the edge e_j to \mathcal{A} and color e_j with the color $\chi(e_j) + \Delta_1$, where $\chi(e)$ is the color chosen by \mathcal{A} for e_j . Then we let $R' = R' \setminus \{e_j\}$ and increase j by one.

Let Δ_2 be the largest color chosen in Step (II).

Finally, in Step (III), we color the remaining edges $e_{m'+1}, \dots, e_m$ greedily with the first available color $j > \Delta_2$. Let Δ_3 be the largest color used at the end of Step (III).

We next analyze the above algorithm (assuming the occurrence of the high probability events from Lemma 3.3). Obviously this algorithm computes a feasible coloring. By definition, Step (I) uses $\Delta_1 \leq 2\epsilon\Delta$ colors. Analogously, by Item 3 of Lemma 3.3, the number of colors used in Step (III) is at most $\Delta_3 - \Delta_2 = O(\epsilon^2\Delta)$. It remains to upper bound the number of colors $\Delta_2 - \Delta_1$ used in the second step. To this end, we note that Lemma 3.3 implies that H is near-regular. More precisely, we have the following bound, whose proof is deferred to Appendix B.

Lemma 3.4. *The graph H satisfies $d_H(v) = \Delta(1 \pm 4\epsilon^2)$ for all $v \in V(H)$, w.h.p.*

It is easy to see that Step (II) implements the warm-up algorithm on H , as the edges of H are fed to this algorithm in a uniform random order. Thus, by Theorem 3.2, w.h.p. the number of colors used in Step (II) is at most $\Delta_2 - \Delta_1 \leq \Delta + O(\Delta^\gamma \cdot \log^{1-\gamma} n)$ for a constant $\gamma \in (0, 1)$. By choosing ϵ small enough so that $\epsilon\Delta \leq \Delta^\gamma \cdot \log^{1-\gamma} n$, we immediately obtain our main result.

Theorem 1.1. *For some absolute constant $\gamma \in (0, 1)$, there exists an online algorithm that, when given a graph G of maximum degree $\Delta = \omega(\log n)$, whose edges are presented in random order, computes a proper $(\Delta + O(\Delta^\gamma \cdot \log^{1-\gamma} n)) = (1 + o(1))\Delta$ -edge-coloring of G w.h.p.*

4 Low-Recourse Dynamic Algorithm

In this section, we give an implementation of Algorithm 1 in a dynamic setting with low recourse. Before describing our algorithm, we explain why in the dynamic setting we can focus our attention on near-regular graphs, as required of inputs to Algorithm 1.

(Near-)Regularizing Gadget. Consider a dynamic input graph $G = (V, E)$ on n nodes, where the degree of each node remains at most Δ all the time. In this dynamic setting, we describe a procedure to maintain a super-graph $G' = (V', E')$ of $G = (V, E)$, with $V' \supseteq V$ and $E' \supseteq E$, such that: (a) each node in G' always has degree either Δ or $\Delta - 1$, and (b) each update (edge insertion/deletion) in G results in a constant number of updates in G' . The node-set V' of G' consists of the nodes $v \in V$ of the input graph, plus Δ *dummy nodes* $v_1, v_2, \dots, v_\Delta$ for each $v \in V$.

Initially, when G is empty,² the edge-set of G' is defined as follows. For each node $v \in V$ in the input graph, the set of nodes $\{v, v_1, \dots, v_\Delta\}$ induces a $(\Delta + 1)$ -clique of *dummy edges* in the

²It is easy to extend our gadget to the setting where the input graph G is not empty at preprocessing.

supergraph G' . In other words, there is a *dummy edge* (v, v_i) for all $v \in V, i \in [\Delta]$; and a *dummy edge* (v_i, v_j) for all $v \in V, i \in [\Delta], j \in [\Delta], i \neq j$. At this point in time, the edge-set E' of G' consists only of the dummy edges.

Subsequently, whenever an edge $e = (u, v)$ is added to (resp., removed from) G during an update, we perform the following steps. We first add e to (resp., remove e from) G' . Next, let i and j be the smallest indices in $[\Delta]$ for which the dummy edges (u, u_i) and (v, v_j) currently exist (resp., do not exist) in G' : we remove (resp., add) these two dummy edges (u, u_i) and (v, v_j) .

Note that every update in G generated by an oblivious adversary results in three updates (likewise generated by an oblivious adversary) in G' . Since any k -edge-coloring of G' trivially induces a k -edge-coloring of G , our goal will be to maintain a $(1 + \epsilon)\Delta$ -edge-coloring of G' .

Accordingly, *w.l.o.g., henceforth we will assume that the input graph G given to us at preprocessing is near-regular (specifically, each of its nodes has degree either Δ or $\Delta - 1$), and that the input graph G remains near-regular throughout the sequence of updates.*

Our dynamic algorithm: At each time t (i.e., after the t -th update), our dynamic algorithm strives to assign tentative colors $c^{(t)}(e)$ to all edges in the current input graph, denoted by $G^{(t)} = (V, E^{(t)})$, as in Algorithm 1. It likewise defines failures in the same way as in Algorithm 1. Finally, it maintains a coloring of the failed and otherwise uncolored edges using a simple $O(\Delta)$ -edge-coloring, constant-recourse *dynamic* algorithm, which we denote by SIMPLECOLOR (e.g., [7, 43]).

For the rest of this section, the superscript (t) on any given notation will indicate the state of the concerned object after the t^{th} update in the input graph.³ Intuitively, the outcome of our dynamic algorithm just after the t^{th} update will be the same as the outcome we would get if we ran Algorithm 1 in the static setting with $G^{(t)}$ as input (see Lemma 4.1). We will refer to an *unordered* pair of nodes (u, v) , with $u, v \in V$, as a *potential edge*. Thus, there are $\binom{|V|}{2}$ many potential edges, and the set of potential edges do not change during the sequence of updates. In contrast, we will refer to an edge $e \in E^{(t)}$ as a *current edge* in the input graph after the t^{th} update.

Preprocessing: We start by assigning a *round* $i(u, v) \in [t_\epsilon]$ to each potential edge (u, v) , where these $i(u, v)$ values are i.i.d. samples from the capped geometric distribution $CappedGeo(\epsilon, t_\epsilon)$ with success probability ϵ and at most t_ϵ attempts.⁴ For each $j \in [t_\epsilon]$, we let $S_j := \{(u, v) \mid i(u, v) = j\}$ denote the set of all *potential* edges that are assigned to round j . Furthermore, for each $t \geq 0$, let $S_j^{(t)} := S_j \cap E^{(t)}$ denote the set of *current* edges (u, v) after the t^{th} update which have $i(u, v) = j$.

In future, throughout the sequence of updates, the random variables $\{i(u, v)\}$ will determine which sets of current edges get sampled in which round (see Line 5 of Algorithm 1). Specifically, consider any potential edge (u, v) , with $i(u, v) = j$, and any two nonnegative integers $t \neq t'$ such that $(u, v) \in E^{(t)} \cap E^{(t')}$. Then the edge (u, v) will get sampled in the same round j in both $G^{(t)}$ and $G^{(t')}$. Thus, a given edge gets assigned to the same round across all the updates.

After drawing the random variables $\{i(u, v)\}$ for all potential edges as described above, we implement Algorithm 1 on the input $G^{(0)} = (V, E^{(0)})$ given to us at the preprocessing phase.

Handling an update: For any $t \geq 1$, consider the t^{th} update which changes the input graph G from $G^{(t-1)} = (V, E^{(t-1)})$ to $G^{(t)} = (V, E^{(t)})$. Our dynamic algorithm handles this update by computing an edge-coloring for $G^{(t)}$ in three steps, as described below.

³The reader should not confuse time t in the dynamic setting with the last round t_ϵ in phase one of Algorithm 1.

⁴For a random variable $X \sim CappedGeo(\epsilon, t_\epsilon)$, we have $\Pr[X = k] = \begin{cases} \epsilon \cdot (1 - \epsilon)^{k-1} & k \in \{1, \dots, (t_\epsilon - 1)\} \\ (1 - \epsilon)^{t_\epsilon-1} & k = t_\epsilon. \end{cases}$

Step I: We perform the following operations in increasing order of $i = 1, 2, \dots, t_\epsilon - 1$:

- For every *potential* edge $e \in S_i$, we first define its palette

$$P_i^{(t)}(e) := [\Delta(1 + \epsilon^2)] \setminus \left\{ c^{(t)}(e') \mid e' \in N^{(t)}(e) \cap \bigcup_{j < i} S_j^{(t)} \right\}.$$

Next, we call Algorithm 2 to update the tentative color $c^{(t)}(e)$ of e . Note that as in Algorithm 1, if $P_i^{(t)}(e) = \emptyset$, then Algorithm 2 sets $c^{(t)}(e) \leftarrow \text{null}$.

Step II: For every round $i \in [t_\epsilon - 1]$, we now define the set of *failed* current edges

$$F_i^{(t)} := \left\{ e \in S_i^{(t)} : c^{(t)}(e) \in \{\text{null}\} \cup \{c^{(t)}(e') : e' \in N^{(t)}(e) \cap S_i^{(t)}\} \right\}.$$

Let $F^{(t)} := \bigcup_{i=1}^{t_\epsilon-1} F_i^{(t)}$ denote the set of all failed current edges across all rounds. Every current edge $e \in \left(\bigcup_{i=1}^{t_\epsilon-1} S_i^{(t)} \right) \setminus F^{(t)}$ gets colored with its tentative color $c^{(t)}(e)$.

Step III: Finally, let $G_U^{(t)} := G[F^{(t)} \cup S_{t_\epsilon}^{(t)}]$ be the subgraph of $G^{(t)}$ consisting of all the edges that are not colored using their tentative colors in Step II above. We use SIMPLECOLOR to maintain an $O(\Delta(G_U^{(t)}))$ edge coloring of $G_U^{(t)}$. In more detail, after each update, denoting by $A \oplus B := (A \setminus B) \cup (B \setminus A)$ the symmetric difference, we think of the graph $G_U^{(t)} := G[F^{(t)} \cup S_{t_\epsilon}^{(t)}]$ as having undergone $|F^{(t)} \oplus F^{(t-1)}| + |S_{t_\epsilon}^{(t)} \oplus S_{t_\epsilon}^{(t-1)}|$ updates, which we feed to algorithm SIMPLECOLOR.

This concludes the description of our dynamic algorithm.

Algorithm 2 TENTATIVELYCOLOR(e)

- 1: **if** $c^{(t-1)}(e) \in P_i^{(t-1)}(e) \setminus P_i^{(t)}(e)$ **then**
 - 2: **If** $P_i^{(t)} \neq \emptyset$ **sample** $c^{(t)}(e) \sim_R P_i^{(t)}(e)$, **else set** $c^{(t)}(e) \leftarrow \text{null}$
 - 3: **else**
 - 4: **If** $P_i^{(t)} \neq \emptyset$ **sample** $c \sim_R P_i^{(t)}(e)$, **else set** $c \leftarrow \text{null}$
 - 5: **if** $c = \text{null}$ **or** $c \in P_i^{(t)}(e) \setminus P_i^{(t-1)}(e)$ **then**
 - 6: $c^{(t)}(e) \leftarrow c$
 - 7: **else**
 - 8: $c^{(t)}(e) \leftarrow c^{(t-1)}(e)$ ▷ We keep the previous color
-

Analysis: Looking back at Algorithm 2, a moment's thought reveals that if the tentative color $c^{(t-1)}(e)$ was chosen uniformly at random from the set $P_i^{(t-1)}(e)$, then the tentative color $c^{(t)}(e)$ is also chosen uniformly at random from the set $P_i^{(t)}(e)$. This, however, is far from being sufficient for our purpose. In particular, we need a formal (and much stronger) guarantee stated below.

Lemma 4.1. *For each time $t \geq 0$, the joint distribution $\{c^{(t)}(e)\}_{e \in E^{(t)}}$ of colors sampled by the dynamic algorithm is distributed identically to $\{c(e)\}_e$ of Algorithm 1 when applied to graph $G^{(t)}$.*

We defer the proof of Lemma 4.1 to Appendix D.1. This lemma, together with our analysis of Algorithm 1, immediately leads us to the following corollary.

Corollary 4.2. For ϵ as in (1) and K as in (2), the above dynamic algorithm $\Delta(1 + O(\epsilon^{(1/3K)}))$ -edge-colors G at any time t , w.h.p.

Bounding Recourse: We now fix some $t \geq 1$, and bound the expected recourse our algorithm has to pay while handling the t^{th} update. Say that an edge $e \in E^{(t)} \cap E^{(t-1)}$ is *dirty* iff $i(e) \leq t_\epsilon - 1$ and it changes its tentative color $c(e)$ due to the t^{th} update. Let D_j denote the set of dirty edges e assigned to round $i(e) = j$. We will use the symbol $D_{<i} = \cup_{j < i} D_j$ to denote all the dirty edges at rounds $j < i$. Let $D = D_{<t_\epsilon}$ denote the set of dirty edges across all rounds.

Lemma 4.3. The recourse of the dynamic algorithm to handle the t^{th} update is $O(1 + |D|)$.

Proof. Let e^* denote the edge being inserted/deleted during the t^{th} update. The additive $+1$ term in the claimed recourse bound of $O(1 + |D|)$ comes from the edge e^* . To simplify notations, we will assume $c^{(t-1)}(e^*) = c^{(t)}(e^*)$ for the rest of this proof. Any other edge $e \in E^{(t-1)} \cap E^{(t)}$ changes its final color during the t^{th} update only if: $i(e) < t_\epsilon$, and {either (1) the edge e changes its tentative color, or (2) the edge e switches from being successful to failed (or vice versa) without changing its tentative color}. In case (1), we clearly have $e \in D$. In case (2), the edge e must have at least one neighboring edge $e_d \in (D \cup \{e^*\}) \cap N(e)$ such that $c^{(t-1)}(e) = c^{(t)}(e) \in \{c^{(t-1)}(e_d), c^{(t)}(e_d)\}$. We charge the 1 unit of recourse the algorithm has to pay for e to any one such edge e_d . A moment's thought reveals that each edge $(u, v) \in D \cup \{e^*\}$ can receive at most 4 units of charge in this scheme, one for each ordered pair $\{u, v\} \times \{c^{(t-1)}(u, v), c^{(t)}(u, v)\}$. Hence, the algorithm's recourse is at most $O(1 + |D|)$. \square

Below, we state the key lemma that contains the technical meat of our argument. Since the argument requires some nontrivial properties of Algorithm 1 which are proved only in Appendix C, we defer the proof of Lemma 4.4 to Appendix D.2.

Lemma 4.4. For every round $i \in [t_\epsilon - 1]$, we have: $\mathbb{E}[|D_i|] \leq 6\epsilon + 6\epsilon \cdot \mathbb{E}[|D_{<i}|]$.

Corollary 4.5. Our dynamic algorithm has an expected recourse of $O(1/\epsilon^{(3/K)})$ per update.

Proof. Since $|D_{<1}| = 0$ and $D = D_{<t_\epsilon}$, from Lemma 4.4 we derive that:

$$\mathbb{E}[|D|] \leq \sum_{i=0}^{t_\epsilon-1} 6\epsilon \cdot (1 + 6\epsilon)^i \leq (1 + 6\epsilon)^{t_\epsilon} \leq \exp(6\epsilon \cdot t_\epsilon) \leq \exp((3/K) \cdot \ln(1/\epsilon)) = 1/\epsilon^{(3/K)}. \quad (3)$$

The last inequality holds because of (2). The corollary now follows from Lemma 4.4. \square

From (1), (2), Corollary 4.2 and Corollary 4.5, we obtain the following theorem.

Theorem 4.6. There exists two absolute constants $\gamma, \gamma' \in (0, 1)$ such that for all $\epsilon \in [(\log n/\Delta)^{\gamma'}, \gamma]$ we can maintain, w.h.p., a proper $(1 + \epsilon)\Delta$ -edge-coloring in a dynamic graph G of maximum degree $\Delta = \omega(\log n)$, with $O(\text{poly}(1/\epsilon))$ expected recourse per update.

5 Conclusions and Open Questions

We presented one common approach for tackling edge coloring in two widely-studied relaxations of the online model of computation, making progress on (and in one case, resolving) the conjecture of Bar-Noy et al. [4] for these models. We conclude with a few interesting research directions.

Adversarial Online Arrivals. The most natural question is whether the Bar-Noy et al. conjecture holds in the strictest, adversarial edge-arrival model. This question still seems out of reach. One algorithmic approach which suggests itself is to extend the ideas in [11]. This would require some form of online dependent rounding for fractional matching under edge arrivals, generalizing [12]. Alternatively, it is not implausible that the Bar-Noy et al. conjecture is false under adversarial edge arrivals, despite being true for vertex arrivals [11]. Such a refutation of this conjecture would mirror a similar separation between these arrival models recently proven for online matching [20].

Knowledge of Δ . All our algorithms assume knowledge of the maximum degree Δ . This assumption is common to all prior best algorithms in the models we study [1, 3, 14].⁵ In fact, Cohen et al. [11] showed that not knowing Δ in their online model results in a strictly harder problem, for which no better than $\frac{e}{e-1}\Delta$ -edge-coloring algorithm exists, for any (unknown) Δ . Is the same separation between known and unknown Δ true for the models studied in this paper?

Acknowledgements. We thank Janardhan Kulkarni for many helpful discussions. The work of Sayan Bahttacharya was supported by Engineering and Physical Sciences Research Council, UK (EPSRC) Grant EP/S03353X/1. The work of Fabrizio Grandoni was supported in part by the SNF Excellence Grant 200020B_182865/1. The work of David Wajc was supported in part by NSF grants CCF-1527110, CCF-1618280, CCF-1814603, CCF-1910588, NSF CAREER award CCF-1750808 and a Sloan Research Fellowship.

Appendix

A A Lower Bound for Random Order Online Edge Coloring

Bar-Noy et al. [4] gave a simple lower bound for edge coloring under adversarial arrivals. Specifically, they showed a family of graphs \mathcal{F} with maximum degree $\Delta = O(\sqrt{\log n})$ for which any randomized online algorithm \mathcal{A} colors some graphs in \mathcal{F} with $2\Delta - 1$ colors with constant probability. Extending these ideas slightly, we show that the same holds even if the arrival order is randomized.

Lemma A.1. *There exists a distribution over n -node graphs \mathcal{G} of maximum degree $\Delta = \Omega(\sqrt{\log n})$, for which any online edge coloring algorithm \mathcal{A} must, with constant probability, use $2\Delta - 1$ colors on a graph $G \sim \mathcal{G}$ presented in random order.*

Proof. Consider a star on $\Delta - 1$ leaves. If Algorithm \mathcal{A} uses $2\Delta - 2$ or fewer colors, then it may color any such star's edges with $\binom{2\Delta-2}{\Delta-1}$ possible subsets of colors. If Δ such stars' edges are colored using the same subset $S \subseteq [2\Delta - 2]$ of $\Delta - 1$ colors and some node v neighbors the roots of these Δ stars, then the algorithm fails, as it is forced to use Δ colors outside of S for the edges of v , for a

⁵Duan et al. [14] run a logarithmic number of algorithms for unknown Δ , and switch between their colorings whenever Δ changes. This results in *fast update time*, but high recourse.

total of $2\Delta - 1$ distinct colors. We show a random graph \mathcal{G} for which this bad event happens with constant probability, even when the edges are presented in random order.

Our graph consists of independent copies of the following random graph, \mathcal{H} . The graph \mathcal{H} contains $\beta := 2\Delta \cdot \binom{2\Delta-2}{\Delta-1} \cdot \binom{2\Delta-1}{\Delta} \leq 4^{O(\Delta)}$ stars with $\Delta - 1$ leaves, and one node v which neighbors the centers of Δ randomly-chosen such stars. For any star, the probability that all $\Delta - 1$ edges of the star arrive before any of the Δ edges of v arrive is $\frac{(\Delta-1)!\Delta!}{(2\Delta-1)!} = 1/\binom{2\Delta-1}{\Delta}$. Therefore, by linearity of expectation, if we denote by X the fraction of stars in \mathcal{H} whose edges arrive before all edges of v , we have that $\mu := \mathbb{E}[X] = 1/\binom{2\Delta-1}{\Delta}$. By Markov's inequality applied to the non-negative variable $Y := 1 - X$, whose expectation is $\mathbb{E}[Y] = 1 - \mu$, we have that

$$\Pr \left[X \leq \frac{\mu}{2} \right] = \Pr \left[Y \geq 1 - \frac{\mu}{2} \right] \leq \frac{1 - \mu}{1 - \frac{\mu}{2}} = 1 - \frac{\mu}{2 - \mu} \leq 1 - \frac{\mu}{2}. \quad (4)$$

Now, if $X \geq \frac{\mu}{2} = 1/(2 \cdot \binom{2\Delta-1}{\Delta})$, then at least $\beta \cdot \frac{\mu}{2} = \Delta \cdot \binom{2\Delta-2}{\Delta-1}$ of the stars of $H \sim \mathcal{H}$ have all their edges arrive before any edge of v arrives. By pigeonhole principle, some Δ of these stars are colored with a common set of $\Delta - 1$ colors, $S \subset [2\Delta - 1]$. If v neighbors the roots of Δ such stars whose edges are colored with the colors in S , then Algorithm \mathcal{A} fails, as it must color the graph using $2\Delta - 1$ colors, as argued above. Therefore, conditioned on $X \geq \frac{\mu}{2}$, Algorithm \mathcal{A} fails when coloring $H \sim \mathcal{H}$ with probability at least

$$\Pr \left[\mathcal{A} \text{ fails on } H \sim \mathcal{H} \mid X \geq \frac{\mu}{2} \right] \geq 1/\binom{\beta}{\Delta} \geq 4^{-O(\Delta^2)}. \quad (5)$$

Consequently, combining Equation (4) and Equation (5), and using $\mu = 1/\binom{2\Delta-1}{\Delta} = 4^{-O(\Delta)}$, we find that the unconditional probability of Algorithm \mathcal{A} not failing due to H is at most

$$\begin{aligned} \Pr [\mathcal{A} \text{ does not fail on } H \sim \mathcal{H}] &\leq 1 - \Pr \left[\mathcal{A} \text{ fails on } H \sim \mathcal{H} \mid X \geq \frac{\mu}{2} \right] \cdot \Pr \left[X \geq \frac{\mu}{2} \right] \\ &\leq 1 - 4^{-O(\Delta^2)} \cdot \frac{\mu}{2} \\ &= 1 - 4^{-O(\Delta^2)}. \end{aligned}$$

As stated above, the random graph \mathcal{G} we consider consists of some γ independent copies of \mathcal{H} . For independent copies of \mathcal{H} , the above upper bound on the probability of \mathcal{A} not failing on a copy $H \sim \mathcal{H}$ holds independently of other copies' realization and coloring by \mathcal{A} . Therefore, letting \mathcal{G} consist of some sufficiently large $\gamma := 4^{\Theta(\Delta^2)}$ independent copies of \mathcal{H} , we have that

$$\Pr [\mathcal{A} \text{ does not fail on } G] \leq \left(1 - 4^{-O(\Delta^2)} \right)^\gamma \leq \frac{1}{e}.$$

Therefore, Algorithm \mathcal{A} fails on G with constant probability. The lemma follows by noting that G consists of some $n = \gamma \cdot (\beta + 1) = 4^{\Theta(\Delta^2)}$ nodes, and therefore $\Delta = \Omega(\sqrt{\log n})$. \square

B Deferred Proofs from Section 3

In this section we give the full proofs deferred from Section 3, restated below for ease of reference.

We start by proving Fact 3.1, which intuitively implies that we can use the stream's random order to sample edges independently.

Fact 3.1. Consider a universe U of n elements, and let $p \in [0, 1]$. Let $U_k \subseteq U$ denote the first k elements in a random-order stream of U , and let $X \sim \text{Bin}(n, p)$ be a binomial random variable with parameters n and p . Then the random set U_X contains every element in U independently with probability p .

Proof. For any given subset $S \subseteq U$ of size $|S| = k$, we have $U_k = S$ precisely when $X = k$ and the elements of S are the first k elements in the stream. As X is independent of the stream's randomness, this gives us:

$$\Pr[U_k = S] = \Pr[X = k] \cdot \Pr[S \text{ is a prefix of the stream}] = \Pr[X = k] / \binom{n}{k} = p^k (1-p)^{n-k}.$$

This is precisely the probability of getting a specific set S , when each of the n elements in U is sampled independently with probability p . \square

Next, we prove Lemma 3.3, which asserts that the edges until time T where some vertex reaches degree $\epsilon \cdot \Delta$ gives sharp estimates of natural graph parameters, such as the number of edges, and every vertex's degree, w.h.p.

Lemma 3.3. Let $\epsilon \leq \frac{1}{2}$, and let $\alpha > 0$ be a constant, and assume $\Delta \geq \frac{24(\alpha+3)\ln n}{\epsilon^8}$. Then, with probability at least $1 - O(n^{-\alpha})$, the following properties hold:⁶

1. $T = \epsilon \cdot m(1 \pm \epsilon^2)$.
2. $d^{(T)}(v) = \epsilon \cdot d(v) \pm 2\epsilon^3 \Delta$ for every node v .
3. Let $m' := T/(\epsilon(1 + \epsilon^2))$. Conditioned on $m' \leq m$, every node v has $d(v) - d^{(m')}(v) \leq 2\epsilon^2 \Delta$.

Proof. The proof relies on several applications of Chernoff bound and union bound, as follows. Fix some vertex v , and let X_i be an indicator variable for the i -th edge in the stream containing v . Clearly, we have that $\mathbb{E}[X_i] = \frac{d(v)}{m}$, and so by linearity of expectation

$$\mathbb{E}[d^{(k)}(v)] = k \cdot \frac{d(v)}{m}. \tag{6}$$

On the other hand, the joint distribution (X_1, \dots, X_m) is a permutation distribution, and so it is NA. We may therefore apply Chernoff bounds to sums of such variables, such as $d^{(k)}(v) = \sum_{i=1}^k X_i$. We will make use of this to prove that the three properties hold w.h.p.

We begin by proving Property 1. Fix any vertex v of degree $d(v) = \Delta$. For any $k \geq \epsilon \cdot m(1 + \epsilon^2)$, we have by Equation (6) that $\mathbb{E}[d^{(k)}(v)] \geq \epsilon \cdot \Delta(1 + \epsilon^2)$. Consequently, by Chernoff bound, we have

$$\begin{aligned} \Pr \left[d^{(k)}(v) \leq \epsilon \cdot \Delta \right] &= \Pr \left[d^{(k)}(v) \leq \frac{1}{1 + \epsilon^2} \cdot \epsilon \cdot \Delta(1 + \epsilon^2) \right] \\ &\leq \Pr \left[d^{(k)}(v) \leq (1 - \epsilon^2/2) \cdot \epsilon \cdot \Delta(1 + \epsilon^2) \right] && \frac{1}{1+x} \leq 1 - \frac{x}{2} \quad \forall x \leq 1 \\ &\leq \exp \left(\frac{-\epsilon^4/4 \cdot \epsilon \cdot \Delta(1 + \epsilon^2)}{3} \right) && \text{Lemma E.8} \\ &\leq n^{-\alpha}. && \Delta \geq \frac{12\alpha \log n}{\epsilon^5} \end{aligned}$$

⁶We let $c = a \pm b$ denote $c \in [a - b, a + b]$.

Therefore, $\Pr[T \geq \epsilon \cdot m(1 + \epsilon^2)] \leq n^{-\alpha}$.

Now, fix a vertex v . For any $k \leq \epsilon \cdot m(1 - \epsilon^2)$, we have by Equation (6) that $\mathbb{E}[d^{(k)}(v)] \leq \epsilon \cdot d(v)(1 - \epsilon^2) \leq \epsilon \cdot \Delta(1 - \epsilon^2)$. Consequently, by Chernoff bound, we have

$$\begin{aligned} \Pr \left[d^{(k)}(v) \geq \epsilon \cdot \Delta \right] &= \Pr \left[d^{(k)}(v) \geq \frac{1}{1 - \epsilon^2} \cdot \epsilon \cdot \Delta(1 - \epsilon^2) \right] \\ &\leq \Pr \left[d^{(k)}(v) \geq (1 + \epsilon^2) \cdot \epsilon \cdot \Delta(1 - \epsilon^2) \right] && 1 + x \leq \frac{1}{1 - x} \quad \forall x < 1 \\ &\leq \exp \left(\frac{-\epsilon^4 \cdot \epsilon \cdot \Delta(1 + \epsilon^2)}{3} \right) && \text{Lemma E.8} \\ &\leq n^{-(\alpha+1)}. && \Delta \geq \frac{6(\alpha + 1) \log n}{\epsilon^5} \end{aligned}$$

Taking union bound over all vertices, we find that $\Pr[T \leq \epsilon \cdot m(1 - \epsilon^2)] \leq n^{-\alpha}$, which together with the above, implies that Property 1 holds with probability at least $1 - 2n^{-\alpha}$.

We next prove that Properties 2 and 3 hold w.h.p. Consider the following property for a generic $k \leq m$:

$$d^{(k)}(v) = \epsilon \cdot d(v) \pm 2\epsilon^3 \Delta. \quad (7)$$

Let $B^{(k)}$ denote the bad event that some vertex v fails to satisfy Equation (7) for this k . We will show that the probability of any event $B^{(k)}$ to happen for $k \in [\epsilon m(1 - \epsilon^2), m]$ is at most $2n^{-\alpha}$. Observe that Property 2 and Property 3 hold if events $B^{(T)}$ and $B^{(m')}$, resp., do not happen. Assuming Property 1, both T and m' fall in the range $[\epsilon m(1 - \epsilon^2), m]$. It then follows that the 3 properties simultaneously hold with probability at least $1 - 4n^{-\alpha}$.

Equation (7) trivially holds for vertices v with $d(v) \leq \epsilon^3 \Delta$, for which $0 \leq d^{(k)}(v) \leq d(v) \leq \epsilon^3 \Delta$. On the other hand, for vertices v with $d(v) \geq \epsilon^3 \Delta$ and k as above, we have by Equation (6) that $\mathbb{E}[d^{(k)}(v)] = \epsilon \cdot \Delta(1 \pm \epsilon^2)$. Consequently, by Chernoff bound, we have

$$\begin{aligned} \Pr \left[d^{(k)}(v) \geq \epsilon \cdot d(v) + 2\epsilon^3 \Delta \right] &\leq \Pr \left[d^{(k)}(v) \geq (1 + 2\epsilon^2) \cdot \epsilon \cdot d(v) \right] && d(v) \leq \Delta \\ &\leq \Pr \left[d^{(k)}(v) \geq (1 + \epsilon^2/2) \cdot \epsilon \cdot d(v)(1 + \epsilon^2) \right] && 1 + 2\epsilon^2 \geq (1 + \epsilon^2)(1 + \epsilon^2/2) \\ &\leq \exp \left(\frac{-\epsilon^4/4 \cdot \epsilon \cdot d(v)(1 + \epsilon^2)}{3} \right) && \text{Lemma E.8} \\ &\leq \exp \left(\frac{-\epsilon^4/4 \cdot \epsilon \cdot \epsilon^3 \Delta(1 + \epsilon^2)}{3} \right) && d(v) \geq \epsilon^3 \Delta \\ &\leq n^{-(\alpha+3)}. && \Delta \geq \frac{12(\alpha + 3) \ln n}{\epsilon^8} \end{aligned}$$

Similarly, we have that

$$\begin{aligned}
\Pr \left[d^{(k)}(v) \leq \epsilon \cdot d(v) - 2\epsilon^3 \Delta \right] &\leq \Pr \left[d^{(k)}(v) \leq (1 - 2\epsilon^2) \cdot \epsilon \cdot d(v) \right] && d(v) \leq \Delta \\
&\leq \Pr \left[d^{(k)}(v) \leq (1 - \epsilon^2/2) \cdot \epsilon \cdot d(v)(1 - \epsilon^2) \right] && 1 - 2\epsilon^2 \leq (1 - \epsilon^2)(1 - \epsilon^2/2) \\
&\leq \exp \left(\frac{-\epsilon^4/4 \cdot \epsilon \cdot d(v)(1 - \epsilon^2)}{3} \right) && \text{Lemma E.8} \\
&\leq \exp \left(\frac{-\epsilon^4/4 \cdot \epsilon \cdot \epsilon^3 \Delta(1 - \epsilon^2)}{3} \right) && d(v) \geq \epsilon^3 \Delta \\
&\leq n^{-(\alpha+3)}. && \Delta \geq \frac{24(\alpha+3) \ln n}{\epsilon^8}
\end{aligned}$$

By union bound, we have that

$$\Pr[B^{(k)}] \leq \sum_v \Pr[d^{(k)}(v) \neq \epsilon \cdot d(v) \pm 2\epsilon^3 \Delta] \leq 2n^{-(\alpha+2)}.$$

The claim follows by union bound over the values $k \in [\epsilon m(1 - \epsilon^2), m]$. \square

Finally, we show that adding Δ dummy vertices per vertex v in G forming a Δ -clique, and $(\Delta - (1/\epsilon - 1) \cdot d^{(T)}(v))^+$ dummy edges⁷ from v to some of its dummy nodes, to the edges in the time range $(T, m' - T]$, yields a near-regular graph H .

Lemma 3.4. *The graph H satisfies $d_H(v) = \Delta(1 \pm 4\epsilon^2)$ for all $v \in V(H)$, w.h.p.*

Proof. By Lemma 3.3 (Item 2), the number of dummy edges of v is $\Delta - (1/\epsilon - 1) \cdot d^{(T)}(v)$, as this number is non-negative for every vertex $v \in V$ w.h.p., since

$$\Delta - (1/\epsilon - 1) \cdot d^{(T)}(v) = \Delta - (1 - \epsilon) \cdot (d(v) \pm 2\epsilon^2 \Delta) \geq \Delta(1 - (1 - \epsilon) - 2\epsilon^2) = \Delta(\epsilon - 2\epsilon^2) \geq 0.$$

Each dummy node v_i belongs to a single clique of size Δ , and possibly has another dummy edge to a single real node v , and therefore it has degree $d_H(v) \in \{\Delta - 1, \Delta\}$. As for real vertices v , again appealing to Lemma 3.3 (Item 2), combined with v having at most $\epsilon^2 \Delta$ edges in the range $(m', m]$ by Lemma 3.3 (Item 3), we find that the degree of v in H satisfies

$$\begin{aligned}
d_H(v) &= d(v) - d^{(T)}(v) + \Delta - (1/\epsilon - 1) \cdot d^{(T)}(v) \pm 2\epsilon^2 \Delta \\
&= d(v) + \Delta - (d(v) \pm 2\epsilon^2 \Delta) \pm 2\epsilon^2 \Delta \\
&= \Delta(1 \pm 4\epsilon^2). && \square
\end{aligned}$$

C Analysis of the Basic Algorithm (Proof of Theorem 2.2)

In this section, we analyze our *basic algorithm* (Algorithm 1) from Section 2. Before proceeding any further, the reader will find it useful to review Section 2. The analysis of Algorithm 1 boils down to proving that the uncolored subgraph after phase one, consisting of all the edges that either failed or were not sampled in the first place, has bounded degree. This is (re-)stated in Theorem 2.2 below.

⁷We use the symbol x^+ to denote $\max(0, x)$.

Theorem 2.2. $\Delta(G_{t_\epsilon} \cup G_F) = O(\epsilon^{1/(3K)}\Delta)$ w.h.p.

The rest of this section (Appendix C) is dedicated to the proof of Theorem 2.2, and it is organized as follows. In Appendix C.1, we give a brief, high-level and *informal* overview of the proof of Theorem 2.2. We start the formal proof of Theorem 2.2 in Appendix C.2, which defines some key random variables and events that will be extensively used in our analysis. In Appendix C.3, we show how Theorem 2.2 follows from a sequence of lemmas. The remainder of Appendix C is devoted to the proofs of these individual lemmas from Appendix C.3.

Throughout Appendix C, we use the notation $a \pm b$ to denote the interval $[a - b, a + b]$. Thus, whenever we write $x = a \pm b$ in this section, it means that $x \in [a - b, a + b]$. Similarly, whenever we write $a \pm b = a' \pm b'$, it means that $[a - b, a + b] \subseteq [a' - b', a' + b']$.

C.1 An informal overview of the proof of Theorem 2.2

For all $i \in [t_\epsilon - 1], v \in V$, let $N_i(v) := \{(u, v) \in E_i\}$ denote the edges of v that are present in $G_i = (V, E_i)$. The next lemma helps us bound the maximum degree in the subgraph G_{t_ϵ} .

Lemma C.1. $|N_i(v)| \approx (1 - \epsilon)^{(i-1)} \cdot (1 \pm \epsilon^2)\Delta$ for all $i \in [t_\epsilon], v \in V$, w.h.p.

Proof. (Sketch) The statement clearly holds for $i = 1$. The proof follows from induction on i . Condition on all the random choices made by the algorithm during rounds $\{1, \dots, i - 1\}$. Fix any node $v \in V$, and suppose that $|N_i(v)| \approx (1 - \epsilon)^{(i-1)} \cdot (1 \pm \epsilon^2)\Delta$ for some $i \in [t_\epsilon - 2]$. Each edge $e \in N_i(v)$ belongs to $N_{i+1}(v)$ independently with probability $1 - \epsilon$. From linearity of expectation, we derive that $\mathbb{E}[|N_{i+1}(v)|] = (1 - \epsilon) \cdot |N_i(v)|$. Now, a standard Hoeffding bound gives us: $|N_{i+1}(v)| = (1 - \epsilon) \cdot |N_i(v)| \pm \Theta(\sqrt{\Delta \ln n})$ w.h.p. From (1), (2) and the inductive hypothesis, we get: $\sqrt{\Delta \ln n} \ll \epsilon^2 \cdot \Delta \ll \epsilon \cdot |N_i(v)|$. This implies that $|N_{i+1}(v)| \approx (1 - \epsilon)^i \cdot |N_i(v)| \approx (1 - \epsilon)^i \cdot (1 \pm \epsilon^2)\Delta$ w.h.p. \square

The proof of Lemma C.1 gives us a glimpse as to why we need a lower bound on ϵ : Whenever we take a concentration bound during the analysis, we will end up with an additive error term of the form $\sqrt{\Delta \ln n}$. We will like this additive error term to get subsumed within $\epsilon^\beta \Delta$ for some large constant $\beta > 1$. Specifically, we will like to have $\epsilon^\beta \Delta \gg \sqrt{\Delta \ln n}$, which implies that $\epsilon \gg \left(\frac{\ln n}{\Delta}\right)^{1/(2\beta)}$. Henceforth, to convey the main ideas, we will often gloss over this issue in this section.

Corollary C.2. $\Delta(G_{t_\epsilon}) = O(\epsilon^{1/(3K)}\Delta)$, w.h.p.

Proof. (Sketch) Fix any node $v \in V$. Since $t_\epsilon \approx (\ln(1/\epsilon))/(2K\epsilon)$, Lemma C.1 gives us: $|N_{t_\epsilon}(v)| \approx (1 - \epsilon)^{(t_\epsilon-1)} \cdot \Delta \approx \exp(\epsilon \cdot (t_\epsilon - 1)) \cdot \Delta \approx \epsilon^{1/(2K)}\Delta$ w.h.p. In other words, every node in G_{t_ϵ} has degree at most $\epsilon\Delta$ w.h.p. This implies the corollary. \square

It now remains to upper bound the maximum degree of any node in the subgraph G_F . Before proceeding any further, we need to introduce the following notation. For all $i \in [t_\epsilon - 1], v \in V$ and $c \in [(1 + \epsilon^2)\Delta]$, let $N_{i,c}(v) = \{(u, v) \in N_i(v) : c \in P_i(u)\}$ denote the set of edges in $N_i(v)$ whose other endpoints have the color c in their palettes for round i . We refer to the quantity $|N_{i,c}(v)|$ as the c -degree of the node v for round i . The main challenge will be to bound the c -degrees of the nodes and the palette sizes for the edges in each round, as captured in the lemma below.

Lemma C.3. *The following guarantees hold for all rounds $i \in [t_\epsilon - 1]$, w.h.p.*

- (a) $|P_i(e)| \approx (1 - \epsilon)^{2(i-1)} \cdot (1 \pm \epsilon^2)\Delta$ for all edges $e \in E_i$.
- (b) $|N_{i,c}(v)| \approx (1 - \epsilon)^{2(i-1)} \cdot (1 \pm \epsilon^2)\Delta$ for all nodes $v \in V$ and colors $c \in [\Delta]$.

Lemma C.3 is proved via an induction on i . We skip the rather technical proof of this lemma in this overview section. Instead, here we only explain how this lemma is used to give an upper bound on the maximum degree in G_F . The following lemma will be very useful towards this end.

Lemma C.4. $|N_i(v) \cap F_i| \approx 2\epsilon \cdot |N_i(v) \cap S_i| \pm \Theta(\sqrt{\Delta \ln n})$, for all $i \in [t_\epsilon - 1], v \in V$, w.h.p.

Proof. (Sketch) Fix any round $i \in [t_\epsilon - 1]$ and any node $v \in V$. Since each edge $e \in E_i$ is sampled in S_i independently with probability ϵ , standard concentration bounds imply that:

$$|N_{i,c}(x) \cap S_i| \approx \epsilon \cdot |N_{i,c}(x)| \text{ for all colors } c \in [(1 + \epsilon^2)\Delta], \text{ w.h.p.} \quad (8)$$

Condition on all the random choices made by the algorithm during rounds $\{1, \dots, i-1\}$, as well as the random choices which determine the set S_i . Suppose that these random choices we are conditioning upon satisfy (8) and Lemma C.3 for round i (which anyway occur w.h.p.).

Fix any edge $(u, v) \in N_i(v) \cap S_i$. The probability that (u, v) belongs to the set F_i , conditioned on it having tentatively picked any color $c \in P_i(u, v)$ in round i , is given by:

$$\begin{aligned} \Pr[(u, v) \in F_i \mid c(u, v) = c] &= 1 - \prod_{e' \in (N_{i,c}(u) \cap S_i)} \left(1 - \frac{1}{|P_i(e')|}\right) \cdot \prod_{e' \in (N_{i,c}(v) \cap S_i)} \left(1 - \frac{1}{|P_i(e')|}\right) \\ &\approx 1 - \left(1 - \frac{1}{(1 - \epsilon)^{2(i-1)} \cdot (1 \pm \epsilon^2)\Delta}\right)^{\epsilon \cdot |N_{i,c}(u)| + \epsilon \cdot |N_{i,c}(v)|} \\ &\approx 1 - \left(1 - \frac{1}{(1 - \epsilon)^{2(i-1)} \cdot (1 \pm \epsilon^2)\Delta}\right)^{2\epsilon \cdot (1 - \epsilon)^{2(i-1)} \cdot (1 \pm \epsilon^2)\Delta} \\ &\approx 1 - \exp(-2\epsilon) \approx 2\epsilon. \end{aligned}$$

Since the above derivation holds for all colors $c \in P_i(u, v)$, we infer that $\Pr[(u, v) \in F_i] \approx 2\epsilon$. Now, by linearity of expectation, we get: $\mathbb{E}[|N_i(v) \cap F_i|] = \sum_{e \in N_i(v) \cap S_i} \Pr[(u, v) \in F_i] \approx 2\epsilon \cdot |N_i(v) \cap S_i|$. With some extra effort (see the full version), we can show that the value of $|N_i(v) \cap F_i|$ is tightly concentrated around $\pm \Theta(\sqrt{\Delta \ln n})$ of its expectation. \square

Corollary C.5. $\Delta(G_F) = O(\epsilon\Delta)$, w.h.p.

Proof. Consider any node $v \in V$. The degree of v in G_F is given by $\deg_v(G_F) := \sum_{i=1}^{t_\epsilon-1} |N_i(v) \cap F_i|$. Hence, w.h.p., Lemma C.4 gives us the following bound on $\deg_v(G_F)$.

$$\begin{aligned} \deg_v(G_F) &\approx \sum_{i=1}^{t_\epsilon-1} \left(2\epsilon \cdot |N_i(v) \cap S_i| \pm \Theta(\sqrt{\Delta \ln n})\right) \approx 2\epsilon \cdot \sum_{i=1}^{t_\epsilon} |N_i(v) \cap S_i| \pm \Theta(t_\epsilon \cdot \sqrt{\Delta \ln n}) \\ &\leq 2\epsilon \cdot \deg_v(G) \pm \Theta(t_\epsilon \cdot \sqrt{\Delta \ln n}) \leq 2\epsilon \cdot (1 + \epsilon^2)\Delta + \Theta(t_\epsilon \cdot \sqrt{\Delta \ln n}) = \Theta(\epsilon\Delta). \end{aligned}$$

In the above derivation, the last step follows from (1) and (2). Since every node in G_F has degree at most $O(\epsilon\Delta)$ w.h.p., we conclude that $\Delta(G_F) = O(\epsilon\Delta)$ w.h.p. \square

Theorem 2.2 follows from Corollary C.2 and Corollary C.5.

C.2 Key random variables and events

This section defines some random variables and events that will be extensively used in our analysis.

Random variables: We will need to deal with the following random variables for each $i \in [t_\epsilon]$.

- $P_i(v) \subseteq [(1 + \epsilon^2)\Delta]$: A color $c \in [(1 + \epsilon^2)\Delta]$ belongs to the set $P_i(v)$ iff there is no edge $(u, v) \in S_j$ which picked the color c at some earlier round $j < i$. We refer to the set $P_i(v)$ as the *palette* of the node $v \in V$ for round i .
- $P_i(e) \subseteq [(1 + \epsilon^2)\Delta]$: A color $c \in [(1 + \epsilon^2)\Delta]$ belongs to the set $P_i(e)$ iff $c \in P_i(u) \cap P_i(v)$, where $e = (u, v) \in E$. We refer to the set $P_i(e)$ as the *palette* of the edge $e \in E$ for round i .
- $N_i(v)$: The set of neighboring edges of $v \in V$ in G_i , that is, $N_i(v) = \{(u, v) \in E : (u, v) \in E_i\}$.
- $N_{i,c}(v)$: This is the set of neighboring edges of the node $v \in V$ in G_i whose other endpoints have the color c in their palettes for round i , that is, $N_{i,c}(v) = \{(u, v) \in N_i(v) : c \in P_i(u)\}$.

Error parameters: While analyzing the basic algorithm, we need to keep track of the amount by which the random variables $|P_i(e)|$ and $|N_{i,c}(v)|$ can deviate from their expected values. The magnitude of these deviations will be captured by the error-parameters $\{\gamma_i\}_{i \in [t_\epsilon]}$, where:

$$\gamma_1 = K\epsilon^2. \quad (9)$$

$$\gamma_{i+1} = (1 + K\epsilon)\gamma_i + K\epsilon^2 \text{ for all } i \in [t_\epsilon - 1]. \quad (10)$$

Corollary C.6. *We have $\gamma_i \leq \epsilon^{1/2}$ for all $i \in [t_\epsilon]$.*

Proof. From (9) and (10), we derive that:

$$\begin{aligned} \gamma_{t_\epsilon} &= K\epsilon^2 \cdot \sum_{i=0}^{t_\epsilon-1} (1 + K\epsilon)^i \\ &\leq (K\epsilon^2) \cdot \frac{(1 + K\epsilon)^{t_\epsilon}}{(K\epsilon)} \\ &\leq \epsilon \cdot \exp(K\epsilon t_\epsilon) \\ &\leq \epsilon \cdot \exp((1/2) \cdot \ln(1/\epsilon)) \quad (\text{by (2)}) \\ &= \epsilon^{1/2}. \end{aligned}$$

The corollary now follows from the observation that $\gamma_i \leq \gamma_{t_\epsilon}$ for all $i \in [t_\epsilon]$. □

Random events: We will track the three events $\mathcal{E}_i, \mathcal{C}_i$ and \mathcal{B}_i as defined below, for each $i \in [t_\epsilon - 1]$.

- Event \mathcal{E}_i occurs iff $|P_i(e)| = (1 - \epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta$ for all $e \in E_i$.
- Event \mathcal{C}_i occurs iff $|N_{i,c}(v)| = (1 - \epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta$ for all $v \in V$ and $c \in [(1 + \epsilon^2)\Delta]$.
- Event \mathcal{B}_i occurs iff the following conditions hold for all nodes $v \in V$ and colors $c \in [(1 + \epsilon^2)\Delta]$:

$$|S_i \cap N_i(v)| = (\epsilon \pm \epsilon^2) \cdot |N_i(v)|, \quad \text{and} \quad |S_i \cap N_{i,c}(v)| = (\epsilon \pm \epsilon^2) \cdot |N_{i,c}(v)|.$$

Remark: Since the degree of every node in G is $(1 \pm \epsilon^2)\Delta$, we have $\Pr[\mathcal{E}_1] = \Pr[\mathcal{C}_1] = 1$.

Random bits used by our algorithm: While proving Theorem 2.2, we will often need to condition upon certain critical events. It will be easier to follow the proof if we view these conditionings via the prism of a classification of random bits used by the algorithm, as described below.

During any given round $i \in [t_\epsilon - 1]$, there are two distinct tasks for which the algorithm uses randomness: (a) To determine the set of sampled edges S_i , and (b) to pick a color $c(e)$ for each sampled edge $e \in S_i$. We let $r_i^{(\text{edges})}$ and $r_i^{(\text{colors})}$ respectively denote the random bits used by the algorithm for task (a) and task (b). The random bits $r_i^{(\text{edges})}$ and $r_i^{(\text{colors})}$ are mutually independent of each other, and they are also independent of all the random bits used in the previous rounds $j < i$. We let $r_i = r_i^{(\text{edges})} \cup r_i^{(\text{colors})}$ denote all the random bits used by the algorithm in round i . Furthermore, we let $r_{<i} = \bigcup_{j=1}^{i-1} r_j$ denote the set of all random bits used by the algorithms in rounds $\{1, \dots, i-1\}$. Note that the random bits $r_{<i}$ completely determine the occurrences of the following events: $\{\mathcal{E}_j, \mathcal{C}_j, \mathcal{B}_j\}_{j < i}$ and $\{\mathcal{E}_i, \mathcal{C}_i\}$. On the other hand, the occurrence of the event \mathcal{B}_i is completely determined by the random bits $r_{<i} \cup r_i^{(\text{edges})}$.

C.3 Proof of Theorem 2.2

The main challenge is to show that all the key events defined in Appendix C.2 occur w.h.p. This is summarized in Corollary C.10, which in turn follows from Lemma C.7, Lemma C.8 and Lemma C.9. The proofs of these three crucial lemmas respectively appear in Appendix C.5, Appendix C.6 and Appendix C.7.

Lemma C.7. Consider any round $i \in [t_\epsilon - 1]$, and fix any instantiation of the bits $r_{<i}$ which ensure the occurrence of the event $\mathcal{E}_i \cap \mathcal{C}_i$. Then we have: $\Pr[\mathcal{B}_i \mid r_{<i}] \geq 1 - 1/n^{1500}$.

Lemma C.8. Consider any round $i \in [t_\epsilon - 2]$, and fix any instantiation of the bits $r_{<i} \cup r_i^{(\text{edges})}$ which ensure the occurrence of the event $\mathcal{E}_i \cap \mathcal{C}_i \cap \mathcal{B}_i$. Then we have: $\Pr[\mathcal{E}_{i+1} \mid r_{<i} \cup r_i^{(\text{edges})}] \geq 1 - 1/n^{1500}$.

Lemma C.9. Consider any round $i \in [t_\epsilon - 2]$, and fix any instantiation of the bits $r_{<i} \cup r_i^{(\text{edges})}$ which ensure the occurrence of the event $\mathcal{E}_i \cap \mathcal{C}_i \cap \mathcal{B}_i$. Then we have: $\Pr[\mathcal{C}_{i+1} \mid r_{<i} \cup r_i^{(\text{edges})}] \geq 1 - 1/n^{500}$.

Corollary C.10. We have $\Pr\left[\bigcap_{i=1}^{t_\epsilon-1} (\mathcal{E}_i \cap \mathcal{C}_i \cap \mathcal{B}_i)\right] \geq 1 - 1/n^{400}$.

Proof. First, recall that $\Pr[\mathcal{E}_1] = \Pr[\mathcal{C}_1] = 1$, and hence Lemma C.7 gives us:

$$\Pr[\mathcal{E}_1 \cap \mathcal{C}_1 \cap \mathcal{B}_1] \geq 1 - 1/n^{1500}. \quad (11)$$

Next, applying a union bound over Lemma C.8 and Lemma C.9, we get:

$$\Pr[\mathcal{E}_{i+1} \cap \mathcal{C}_{i+1} \mid \mathcal{E}_i \cap \mathcal{C}_i \cap \mathcal{B}_i] \geq 1 - 2/n^{500} \text{ for all rounds } i \in [t_\epsilon - 2]. \quad (12)$$

Accordingly, from (12) and Lemma C.7 we infer that:

$$\begin{aligned} \Pr[\mathcal{E}_{i+1} \cap \mathcal{C}_{i+1} \cap \mathcal{B}_{i+1} \mid \mathcal{E}_i \cap \mathcal{C}_i \cap \mathcal{B}_i] &= \Pr[\mathcal{E}_{i+1} \cap \mathcal{C}_{i+1} \mid \mathcal{E}_i \cap \mathcal{C}_i \cap \mathcal{B}_i] \cdot \Pr[\mathcal{B}_{i+1} \mid \mathcal{E}_{i+1} \cap \mathcal{C}_{i+1}] \\ &\geq (1 - 2/n^{500}) \cdot (1 - 1/n^{1500}) \text{ for all } i \in [t_\epsilon - 2]. \end{aligned} \quad (13)$$

Now, from (11) and (13) we derive that:

$$\begin{aligned}
\Pr \left[\bigcap_{i=1}^{t_\epsilon-1} (\mathcal{E}_i \cap \mathcal{C}_i \cap \mathcal{B}_i) \right] &= \Pr[\mathcal{E}_1 \cap \mathcal{C}_1 \cap \mathcal{B}_1] \cdot \prod_{i=1}^{t_\epsilon-2} \Pr[\mathcal{E}_{i+1} \cap \mathcal{C}_{i+1} \cap \mathcal{B}_{i+1} \mid \mathcal{E}_i \cap \mathcal{C}_i \cap \mathcal{B}_i] \\
&\geq (1 - 1/n^{1500}) \cdot (1 - 2/n^{500})^{t_\epsilon} \cdot (1 - 1/n^{1500})^{t_\epsilon} \\
&\geq (1 - 1/n^{1500}) \cdot (1 - 2t_\epsilon/n^{500}) \cdot (1 - t_\epsilon/n^{1500}) \\
&\geq 1 - 1/n^{1500} - 2t_\epsilon/n^{500} - t_\epsilon/n^{1500} \geq 1 - 1/n^{400}.
\end{aligned}$$

In the derivation above, the last inequality holds since $t_\epsilon \leq n$ and $n \geq 2$. \square

In order to prove Theorem 2.2, we need to upper bound the maximum degree of any node in the subgraph $G_{t_\epsilon} \cup G_F$. Accordingly, Corollary C.11 upper bounds the maximum degree in the subgraph G_{t_ϵ} , whereas Corollary C.13 (which follows from Lemma C.12), upper bounds the maximum degree in the subgraph G_F . Appendix C.8 contains the proof of Lemma C.12.

Corollary C.11. $\Delta(G_{t_\epsilon}) = O(\epsilon^{1/(3K)} \cdot \Delta)$ with probability at least $1 - 1/n^{400}$.

Proof. Define the event $\mathcal{B} := \bigcap_{i \in [t_\epsilon-1]} \mathcal{B}_i$. From Corollary C.10, we infer that $\Pr[\mathcal{B}] \geq 1 - 1/n^{400}$. Henceforth, we condition on the event \mathcal{B} .

Consider any node $v \in V$. Conditioned on the event \mathcal{B} , we have $|S_i \cap N_i(v)| \geq (\epsilon - \epsilon^2) \cdot |N_i(v)|$ for each round $i \in [t_\epsilon - 1]$. Since $|N_{i+1}(v)| = |N_i(v)| - |N_i(v) \cap S_i|$, we infer that:

$$|N_{i+1}(v)| \leq (1 - \epsilon + \epsilon^2) \cdot |N_i(v)| \text{ for all } i \in [t_\epsilon - 1], \text{ conditioned on the event } \mathcal{B}. \quad (14)$$

Since $|N_1(v)| \leq (1 + \epsilon^2)\Delta$, it is now easy to derive from (14) that:

$$\begin{aligned}
|N_{t_\epsilon}(v)| &\leq (1 - \epsilon + \epsilon^2)^{t_\epsilon} \cdot (1 + \epsilon^2) \cdot \Delta \\
&\leq \exp(-\epsilon(1 - \epsilon)t_\epsilon) \cdot (1 + \epsilon^2) \cdot \Delta \\
&\leq \exp\left(-\epsilon(1 - \epsilon) \cdot \left(\frac{\ln(1/\epsilon)}{2K\epsilon} - 1\right)\right) \cdot (1 + \epsilon^2) \cdot \Delta && \text{(by (1) and (2))} \\
&\leq \exp\left(-\frac{(1 - \epsilon) \ln(1/\epsilon)}{2K} + 2\epsilon\right) \cdot (1 + \epsilon^2) \cdot \Delta && \text{(by (1))} \\
&\leq \epsilon^{(1-\epsilon)/(2K)} \cdot (1 + 4\epsilon) \cdot (1 + \epsilon^2) \cdot \Delta && \text{(by (1))} \\
&\leq O(\epsilon^{1/(3K)} \cdot \Delta) \text{ for all } v \in V, \text{ conditioned on the event } \mathcal{B}. && \text{(by (1) and (2))}
\end{aligned}$$

The degree of a node $v \in V$ in G_{t_ϵ} exactly equals $|N_{t_\epsilon}(v)|$. Hence, we conclude that $\Delta(G_{t_\epsilon}) = O(\epsilon^{1/(3K)}\Delta)$, conditioned on the event \mathcal{B} . The corollary follows since $\Pr[\mathcal{B}] \geq 1 - 1/n^{400}$. \square

Lemma C.12. Consider any round $i \in [t_\epsilon - 1]$, and fix any instantiation of the bits $r_{<i} \cup r_i^{(\text{edges})}$ which ensure the occurrence of the event $\mathcal{E}_i \cap \mathcal{C}_i \cap \mathcal{B}_i$. Then $\Pr \left[\Delta(G_{F_i}) \leq 9\epsilon^2 \Delta \mid r_{<i} \cup r_i^{(\text{edges})} \right] \geq 1 - 1/n^{300}$, where $G_{F_i} = (V, F_i)$ denotes the subgraph of G consisting of the failed edges in round i .

Corollary C.13. $\Delta(G_F) = O(\epsilon \ln(1/\epsilon) \cdot \Delta)$ with probability at least $1 - 1/n^{200}$.

Proof. From Corollary C.10 and Lemma C.12, we infer that:

$$\begin{aligned}
\Pr [\Delta(G_{F_i}) = O(\epsilon^2 \cdot \Delta)] &= \Pr [\Delta(G_{F_i}) = O(\epsilon^2 \cdot \Delta) \mid \mathcal{E}_i \cap \mathcal{C}_i \cap \mathcal{B}_i] \cdot \Pr[\mathcal{E}_i \cap \mathcal{C}_i \cap \mathcal{B}_i] \\
&\geq (1 - 1/n^{300}) \cdot (1 - 1/n^{400}) \\
&\geq 1 - 1/n^{300} - 1/n^{400} \\
&\geq 1 - 1/n^{250} \text{ for each round } i \in [t_\epsilon - 1].
\end{aligned} \tag{15}$$

As $\Delta(G_F) \leq \sum_{i=1}^{t_\epsilon-1} \Delta(G_{F_i})$ and $t_\epsilon \leq n$, the corollary follows from (2) and a union bound over (15). \square

Theorem 2.2 now follows by Corollary C.11 and Corollary C.13.

C.4 A couple of important technical claims

Here, we prove two technical claims that will be used multiple times in the subsequent sections.

Claim C.14. *Fix any round $i \in [t_\epsilon - 1]$ and condition on the event $\mathcal{E}_i \cap \mathcal{C}_i$. Then we have:*

$$(\epsilon^2/2) \cdot |N_{i,c}(v)| \geq 50\sqrt{\Delta \ln n} \text{ for all nodes } v \in V \text{ and colors } c \in [(1 + \epsilon^2)\Delta]. \tag{16}$$

$$(\epsilon^2/2) \cdot |N_i(v)| \geq 50\sqrt{\Delta \ln n} \text{ for all nodes } v \in V. \tag{17}$$

$$(\epsilon^2/2) \cdot |P_i(e)| \geq 50\sqrt{\Delta \ln n} \text{ for all edges } e \in E_i. \tag{18}$$

Proof. Fix any node $v \in V$ and any color $c \in [(1 + \epsilon^2)\Delta]$. Conditioned on the event $\mathcal{E}_i \cap \mathcal{C}_i$, we get:

$$\begin{aligned}
(\epsilon^2/2) \cdot |N_{i,c}(v)| &\geq (\epsilon^2/2) \cdot (1 - \epsilon)^{2(i-1)} \cdot (1 - \gamma_i) \cdot \Delta \\
&\geq (\epsilon^2/2) \cdot (1 - \epsilon)^{2(t_\epsilon-1)} \cdot (1 - \epsilon^{1/2}) \cdot \Delta && \text{(by Corollary C.6)} \\
&\geq (\epsilon^2/2) \cdot \exp(-4\epsilon(t_\epsilon - 1)) \cdot (\Delta/2) && \text{(by (1))} \\
&\geq (\epsilon^2/2) \cdot \exp(-(2/K) \cdot \ln(1/\epsilon)) \cdot (\Delta/2) && \text{(by (2))} \\
&= (\epsilon^2/2) \cdot \epsilon^{2/K} \cdot (\Delta/2) \\
&\geq \epsilon^3 \cdot (\Delta/4) && \text{(by (1) and (2))} \\
&\geq 50\sqrt{\Delta \ln n} && \text{(by (1))}
\end{aligned}$$

Applying the same line of reasoning, one can derive that $(\epsilon^2/2) \cdot |P_i(e)| \geq 50\sqrt{\Delta \ln n}$ for all $e \in E_i$. Finally, (17) follows from (16) and the observation that $N_i(v) \supseteq N_{i,c}(v)$. \square

Claim C.15. *We have:* $\left(1 - \frac{1}{(1-\epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \Delta}\right)^{(\epsilon \pm \epsilon^2) \cdot (1-\epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \Delta} = 1 - \epsilon \pm (4\epsilon\gamma_i + \epsilon^2)$.

Proof. Let $M = \left(1 - \frac{1}{(1-\epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \Delta}\right)^{(\epsilon \pm \epsilon^2) \cdot (1-\epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \Delta}$. We first upper bound M as follows.

$$\begin{aligned}
M &\leq \left(1 - \frac{1}{(1-\epsilon)^{2(i-1)} \cdot (1 + \gamma_i) \Delta}\right)^{(\epsilon - \epsilon^2) \cdot (1-\epsilon)^{2(i-1)} \cdot (1 - \gamma_i) \Delta} \\
&\leq \exp\left(-(\epsilon - \epsilon^2) \cdot (1 - \gamma_i) \cdot (1 + \gamma_i)^{-1}\right) \\
&\leq \exp\left(-(\epsilon - \epsilon^2) \cdot (1 - 2\gamma_i)\right) && \text{(by (1) and Corollary C.6)} \\
&\leq \exp\left(-(\epsilon - 3\epsilon\gamma_i)\right) && \text{(by (1) and Corollary C.6)} \\
&\leq 1 - (\epsilon - 3\epsilon\gamma_i) + (1/2) \cdot (\epsilon - 3\epsilon\gamma_i)^2 \\
&\leq 1 - \epsilon + (4\epsilon\gamma_i + \epsilon^2). && \text{(by (1) and Corollary C.6)}
\end{aligned}$$

Next, we lower bound M as follows.

$$\begin{aligned}
M &\geq \left(1 - \frac{1}{(1-\epsilon)^{2(i-1)} \cdot (1 - \gamma_i) \Delta}\right)^{(\epsilon + \epsilon^2) \cdot (1-\epsilon)^{2(i-1)} \cdot (1 + \gamma_i) \Delta} \\
&\geq \exp\left(-\frac{(\epsilon + \epsilon^2)(1 + \gamma_i)}{(1 - \gamma_i)}\right) \cdot \left(1 - \frac{(\epsilon + \epsilon^2)(1 + \gamma_i)}{(1 - \epsilon)^{2(i-1)} \cdot (1 - \gamma_i)^2 \Delta}\right) \\
&\geq \exp\left(-\frac{(\epsilon + \epsilon^2)(1 + \gamma_i)}{(1 - \gamma_i)}\right) \cdot \left(1 - \frac{8}{(1 - \epsilon)^{2t_\epsilon} \Delta}\right) && \text{(by (1) and Corollary C.6)} \\
&\geq \exp\left(-(\epsilon + 2\epsilon\gamma_i)\right) \cdot \left(1 - \frac{8}{(1 - \epsilon)^{2t_\epsilon} \Delta}\right) && \text{(by (1) and Corollary C.6)} \\
&\geq \exp\left(-(\epsilon + 2\epsilon\gamma_i)\right) \cdot \left(1 - \frac{8 \cdot \exp(4t_\epsilon)}{\Delta}\right) && \text{(by (1))} \\
&\geq \exp\left(-(\epsilon + 2\epsilon\gamma_i)\right) \cdot \left(1 - \frac{8}{\Delta} \cdot \exp\left(4\epsilon + \frac{2 \cdot \ln(1/\epsilon)}{K}\right)\right) && \text{(by (2))} \\
&\geq \exp\left(-(\epsilon + 2\epsilon\gamma_i)\right) \cdot \left(1 - \frac{8}{\Delta} \cdot (1 + 8\epsilon) \cdot \frac{1}{\epsilon^{2/K}}\right) && \text{(by (1))} \\
&\geq \exp\left(-(\epsilon + 2\epsilon\gamma_i)\right) \cdot (1 - \epsilon^2) && \text{(by (1) and (2))} \\
&\geq (1 - \epsilon - 2\epsilon\gamma_i) \cdot (1 - \epsilon^2) \\
&\geq 1 - \epsilon - (4\epsilon\gamma_i + \epsilon^2). && \text{(by (1) and Corollary C.6)}
\end{aligned}$$

The second inequality holds since $(1 - \frac{\lambda}{x})^x \geq \exp(-\lambda) \cdot (1 - \frac{\lambda^2}{x})$ for all $0 < \lambda < x$. \square

C.5 Proof of Lemma C.7

For any node $v \in V$ and any color $c \in [(1 + \epsilon^2)\Delta]$, let $\mathcal{B}_i(v, c)$ denote the event which occurs iff:

$$|S_i \cap N_i(v)| = (\epsilon \pm \epsilon^2) \cdot |N_i(v)| \text{ and } |S_i \cap N_{i,c}(v)| = (\epsilon \pm \epsilon^2) \cdot |N_{i,c}(v)|.$$

We first focus on bounding $\Pr[\mathcal{B}_i(v, c) \mid r_{<i}]$ for a given node $v \in V$ and color $c \in [(1 + \epsilon^2)\Delta]$. We start by observing that $N_{i,c}(v) \subseteq N_i(v)$. For each edge $e \in N_i(v)$, consider an indicator

random variable $X_e \in \{0, 1\}$ that is set to one iff the edge e is sampled in round i . Thus, we have $|S_i \cap N_i(v)| = \sum_{e \in N_i(v)} X_e$ and $|S_i \cap N_{i,c}(v)| = \sum_{e \in N_{i,c}(v)} X_e$. We also have $\mathbb{E}[X_e \mid r_{<i}] = \Pr[X_e = 1 \mid r_{<i}] = \epsilon$ for all edges $e \in N_i(v)$. Hence, applying linearity of expectation, we get:

$$\begin{aligned} \mathbb{E}[|S_i \cap N_i(v)| \mid r_{<i}] &= \sum_{e \in N_i(v)} \mathbb{E}[X_e \mid r_{<i}] = \epsilon \cdot |N_i(v)|. \\ \mathbb{E}[|S_i \cap N_{i,c}(v)| \mid r_{<i}] &= \sum_{e \in N_{i,c}(v)} \mathbb{E}[X_e \mid r_{<i}] = \epsilon \cdot |N_{i,c}(v)|. \end{aligned}$$

The random variables $\{X_e\}$ are mutually independent (even after conditioning on $r_{<i}$). Since $|N_{i,c}(v)| \leq |N_i(v)| \leq (1 + \epsilon^2)\Delta$ and each X_e is a 0/1 random variable, Lemma E.1 gives us:

$$\Pr \left[|S_i \cap N_i(v)| = \epsilon \cdot |N_i(v)| \pm 50\sqrt{\Delta \ln n} \mid r_{<i} \right] \geq 1 - 1/n^{2000}. \quad (19)$$

$$\Pr \left[|S_i \cap N_{i,c}(v)| = \epsilon \cdot |N_{i,c}(v)| \pm 50\sqrt{\Delta \ln n} \mid r_{<i} \right] \geq 1 - 1/n^{2000}. \quad (20)$$

From (1) and Claim C.14, we infer that:

$$\epsilon \cdot |N_i(v)| \pm 50\sqrt{\Delta \ln n} = (\epsilon \pm \epsilon^2) \cdot |N_i(v)|. \quad (21)$$

$$\epsilon \cdot |N_{i,c}(v)| \pm 50\sqrt{\Delta \ln n} = (\epsilon \pm \epsilon^2) \cdot |N_{i,c}(v)|. \quad (22)$$

From (19) and (21), we get:

$$\Pr \left[|S_i \cap N_i(v)| = (\epsilon \pm \epsilon^2) \cdot |N_i(v)| \mid r_{<i} \right] \geq 1 - 1/n^{2000}. \quad (23)$$

Similarly, from (20) and (22), we get:

$$\Pr \left[|S_i \cap N_{i,c}(v)| = (\epsilon \pm \epsilon^2) \cdot |N_{i,c}(v)| \mid r_{<i} \right] \geq 1 - 1/n^{2000}. \quad (24)$$

Applying a union bound over (23) and (24), we get: $\Pr[\mathcal{B}_i(v, c) \mid r_{<i}] \geq 1 - 1/n^{1900}$. Since $\mathcal{B}_i = \bigcap_{v,c} \mathcal{B}_i(v, c)$, the lemma follows from one last union bound over all $v \in V$ and $c \in [(1 + \epsilon^2)\Delta]$.

C.6 Proof of Lemma C.8

Throughout this section, we fix the bits $r_{<i} \cup r_i^{(\text{edges})}$ which ensure the occurrence of the event $\mathcal{E}_i \cap \mathcal{C}_i \cap \mathcal{B}_i$. To ease notations, henceforth we refrain from repeatedly stating that we are conditioning on the bits $r_{<i} \cup r_i^{(\text{edges})}$. However, the reader should keep in mind that we are relying upon this conditioning for the rest of Appendix C.6.

We first bound the expected value of $|P_{i+1}(e)|$ for any given edge $e \in E_{i+1}$.⁸ Next, we show that w.h.p. the value of $|P_{i+1}(e)|$ does not deviate too far away from its expectation. Finally, we take a union bound over all the edges $e \in E_{i+1}$ to argue that the event \mathcal{E}_{i+1} occurs w.h.p.

Calculating the expected value of $|P_{i+1}(e)|$ for a given edge $e = (u, v) \in E_{i+1}$:

⁸Note that the subset of edges E_{i+1} is completely determined by the bits $r_{<i} \cup r_i^{(\text{edges})}$.

Since $e \in E_{i+1}$, we have $e \notin S_i$. In particular, this implies that the sets $(N_{i,c}(u) \cap S_i)$ and $(N_{i,c}(v) \cap S_i)$ are mutually disjoint. This observation will be useful in subsequent derivations.

Consider any color $c \in P_i(e)$. For any endpoint $x \in \{u, v\}$ of the edge e , let $\Gamma_{x,c}$ be the event that at least one edge $e' \in (N_{i,c}(x) \cap S_i)$ picks the color c in round i . Note that $\Pr[c \notin P_{i+1}(e)] = \Pr[\Gamma_{u,c} \cup \Gamma_{v,c}]$. Since the sets $(N_{i,c}(u) \cap S_i)$ and $(N_{i,c}(v) \cap S_i)$ are mutually disjoint, the events $\Gamma_{u,c}$ and $\Gamma_{v,c}$ are mutually independent. Hence, from the inclusion-exclusion principle we infer that:

$$\Pr[c \notin P_{i+1}(e)] = \Pr[\Gamma_{u,c}] + \Pr[\Gamma_{v,c}] - \Pr[\Gamma_{u,c}] \cdot \Pr[\Gamma_{v,c}]. \quad (25)$$

We now focus on estimating the value of $\Pr[\Gamma_{x,c}]$ for a given node $x \in \{u, v\}$. Recall that the bits $r_{<i} \cup r_i^{(\text{edges})}$ we condition upon ensure the occurrence of the event $\mathcal{E}_i \cap \mathcal{C}_i \cap \mathcal{B}_i$. Hence, we have:

$$|P_i(e')| = (1 - \epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta \text{ for all edges } e' \in E_i. \quad (26)$$

$$|N_{i,c}(x)| = (1 - \epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta. \quad (27)$$

$$|N_{i,c}(x) \cap S_i| = (\epsilon \pm \epsilon^2) \cdot (1 - \epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta. \quad (28)$$

Since the event $\Gamma_{x,c}$ occurs iff some edge $e' \in N_{i,c}(x) \cap S_i$ picks color c in round i , we infer that:

$$\begin{aligned} \Pr[\Gamma_{x,c}] &= 1 - \prod_{e' \in N_{i,c}(x) \cap S_i} \left(1 - \frac{1}{|P_i(e')|}\right) \\ &= 1 - \left(1 - \frac{1}{(1 - \epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta}\right)^{(\epsilon \pm \epsilon^2) \cdot (1 - \epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta} \\ &= \epsilon \pm (4\epsilon\gamma_i + \epsilon^2). \end{aligned} \quad (29)$$

In the derivation above, the second step follows from (26) and (28), whereas the last step follows from Claim C.15. From (25) and (29), we next infer that:

$$\Pr[c \notin P_{i+1}(e)] = 2(\epsilon \pm (4\epsilon\gamma_i + \epsilon^2)) - (\epsilon \pm (4\epsilon\gamma_i + \epsilon^2))^2 \text{ for every color } c \in P_i(e).$$

Equivalently, for every color $c \in P_i(e)$, we have:

$$\Pr[c \in P_{i+1}(e)] = 1 - \Pr[c \notin P_{i+1}(e)] = (1 - \epsilon \pm (4\epsilon\gamma_i + \epsilon^2))^2.$$

Applying linearity of expectation, we now get:

$$\mathbb{E}[|P_{i+1}(e)|] = \sum_{c \in P_i(e)} \Pr[c \in P_{i+1}(e)] = (1 - \epsilon \pm (4\epsilon\gamma_i + \epsilon^2))^2 \cdot |P_i(e)|. \quad (30)$$

Deriving a concentration bound on $|P_{i+1}(e)|$ for a given edge $e = (u, v) \in E_{i+1}$:

For each color $c \in P_i(e)$, let $X_c \in \{0, 1\}$ be an indicator random variable that is set to one iff $c \in P_{i+1}(e)$. Clearly, we have: $|P_{i+1}(e)| = \sum_{c \in P_i(e)} X_c$. We will now show that the random variables $\{X_c\}_{c \in P_i(e)}$ are negatively associated, and then apply Hoeffding bound.

Claim C.16. *The random variables $\{X_c\}_{c \in P_i(e)}$ are negatively associated.*

Proof. For each color $c \in P_i(e)$ and each edge $e' \in (N_i(u) \cap S_i) \cup (N_i(v) \cap S_i)$, define an indicator random variable $X_{c,e'} \in \{0, 1\}$ that is set to one iff the edge e' picks color c in round i . Since each edge picks at most one color in round i , Proposition E.3 implies that for each edge $e' \in (N_i(u) \cap S_i) \cup (N_i(v) \cap S_i)$, the random variables $\{X_{c,e'}\}_c$ are negatively associated. Next, note that the color picked by any edge $e' \in S_i$ in round i is independent of the color picked by a different edge $e'' \in S_i \setminus \{e'\}$ in round i . Hence, part (1) of Proposition E.5 implies that the random variables $\{X_{c,e'}\}_{c,e'}$ are also negatively associated. Finally, note that $X_c = 1 - \max_{e' \in (N_i(u) \cap S_i) \cup (N_i(v) \cap S_i)} \{X_{c,e'}\}$ for all colors $c \in P_i(e)$. Accordingly, part (2) of Proposition E.5 implies that the random variables $\{X_c\}_{c \in P_i(e)}$ are negatively associated. This concludes the proof of the claim. \square

Claim C.17. *We have:* $\Pr \left[|P_{i+1}(e)| = \mathbb{E}[|P_{i+1}(e)|] \pm 50\sqrt{\Delta \ln n} \right] \geq 1 - 1/n^{2000}$.

Proof. Note that $|P_{i+1}(e)| = \sum_{c \in P_i(e)} X_c$, where $|P_i(e)| \leq (1 + \epsilon^2)\Delta$ and each X_c is a 0/1 random variable. Since the random variables $\{X_c\}$ are negatively associated according to Claim C.16, from Lemma E.9 we now infer that $\Pr \left[|P_{i+1}(e)| = \mathbb{E}[|P_{i+1}(e)|] \pm 50\sqrt{\Delta \ln n} \right] \geq 1 - 1/n^{2000}$. \square

Corollary C.18. *We have:* $\Pr \left[|P_{i+1}(e)| = (1 - \epsilon)^{2i} \cdot (1 \pm \gamma_{i+1}) \cdot \Delta \right] \geq 1 - 1/n^{2000}$.

Proof. Consider any $M = \mathbb{E}[|P_{i+1}(e)|] \pm 50\sqrt{\Delta \ln n}$. Observe that:

$$\begin{aligned}
M &= (1 - \epsilon \pm (4\epsilon\gamma_i + \epsilon^2))^2 \cdot |P_i(e)| \pm 50\sqrt{\Delta \ln n} && \text{(by (30))} \\
&= \left((1 - \epsilon \pm (4\epsilon\gamma_i + \epsilon^2))^2 \pm \epsilon^2 \right) \cdot |P_i(e)| && \text{(by Claim C.14)} \\
&= ((1 - \epsilon)^2 + (4\epsilon\gamma_i + \epsilon^2)^2 \pm 2(4\epsilon\gamma_i + \epsilon^2) \pm \epsilon^2) \cdot |P_i(e)| \\
&= ((1 - \epsilon)^2 \pm (12\epsilon\gamma_i + 12\epsilon^2)) \cdot |P_i(e)| && \text{(by (1) and Corollary C.6)} \\
&= (1 - \epsilon)^2 \cdot (1 \pm (24\epsilon\gamma_i + 24\epsilon^2)) \cdot (1 - \epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta && \text{(by (1) and (26))} \\
&= (1 - \epsilon)^{2i} \cdot (1 \pm ((1 + 48\epsilon)\gamma_i + 48\epsilon^2)) \cdot \Delta && \text{(by (1) and Corollary C.6)} \\
&= (1 - \epsilon)^{2i} \cdot (1 \pm \gamma_{i+1}) \cdot \Delta. && \text{(by (2) and (10))}
\end{aligned}$$

To summarize, we have derived that if $|P_{i+1}(e)| = \mathbb{E}[|P_{i+1}(e)|] \pm 50\sqrt{\Delta \ln n}$, then it must be the case that $|P_{i+1}(e)| = (1 - \epsilon)^{2i} \cdot (1 \pm \gamma_{i+1}) \cdot \Delta$. The corollary now follows from Claim C.17. \square

Wrapping up the proof of Lemma C.8:

Lemma C.8 follows from Corollary C.18 and a union bound over all the edges $e \in E_{i+1}$.

C.7 Proof of Lemma C.9

Throughout this section, we fix the bits $r_{<i} \cup r_i^{(\text{edges})}$ which ensure the occurrence of the event $\mathcal{E}_i \cap \mathcal{C}_i \cap \mathcal{B}_i$. To ease notations, henceforth we refrain from repeatedly stating that we are conditioning on the bits $r_{<i} \cup r_i^{(\text{edges})}$. However, the reader should keep in mind that we are relying upon this conditioning for the rest of Appendix C.7.

We first bound the expected value of $|N_{i+1,c}(v)|$ for a given $(c, v) \in [(1 + \epsilon^2)\Delta] \times V$. Next, we show that w.h.p. the value of $|N_{i+1,c}(v)|$ does not deviate too far away from its expectation. Finally, we take a union bound over all $(c, v) \in [(1 + \epsilon^2)\Delta] \times V$ to argue that \mathcal{C}_{i+1} occurs w.h.p.

Calculating the expected value of $|N_{i+1,c}(v)|$ for a given $(c, v) \in [(1 + \epsilon^2)\Delta] \times V$:

First, note that $N_{i+1,c}(v) \subseteq N_{i,c}(v) \setminus S_i$. Consider any edge $e' = (u, v) \in N_{i,c}(v) \setminus S_i$, and let $\Gamma_{e'}$ be the event that the edge e' belongs to the set $N_{i+1,c}(v)$. Our immediate goal is to calculate the value of $\Pr[\Gamma_{e'}]$. Towards this end, we first recall that the bits $r_{<i} \cup r_i^{(\text{edges})}$ we are conditioning upon ensure the occurrence of the event $\mathcal{E}_i \cap \mathcal{C}_i \cap \mathcal{B}_i$. Hence, we have:

$$|P_i(e)| = (1 - \epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta \text{ for all edges } e \in E_i. \quad (31)$$

$$|N_{i,c}(x)| = (1 - \epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta \text{ for all nodes } x \in V. \quad (32)$$

$$|N_{i,c}(x) \cap S_i| = (\epsilon \pm \epsilon^2) \cdot (1 - \epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta \text{ for all nodes } x \in V. \quad (33)$$

The event $\Gamma_{e'}$ occurs iff no edge $(u, w) \in N_{i,c}(u) \cap S_i$ picks the color c in round i . Hence, from (31), (33) and Claim C.15, we now derive that:

$$\begin{aligned} \Pr[\Gamma_{e'}] &= \prod_{w \in N_{i,c}(u) \cap S_i} \left(1 - \frac{1}{|P_i(u, w)|} \right) \\ &= \left(1 - \frac{1}{(1 - \epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta} \right)^{(\epsilon \pm \epsilon^2) \cdot (1 - \epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta} \\ &= 1 - \epsilon \pm (4\epsilon\gamma_i + \epsilon^2) \end{aligned} \quad (34)$$

Since we have already conditioned on the event \mathcal{B}_i , we get:

$$|N_{i,c}(v) \setminus S_i| = |N_{i,c}(v)| - |N_{i,c}(v) \cap S_i| = (1 - \epsilon \pm \epsilon^2) \cdot |N_{i,c}(v)|. \quad (35)$$

From (34) and (35), together with linearity of expectation, we now derive that:

$$\begin{aligned} \mathbb{E}[|N_{i+1,c}(v)|] &= \sum_{e' \in N_{i,c}(v) \setminus S_i} \Pr[\Gamma_{e'}] \\ &= (1 - \epsilon \pm (4\epsilon\gamma_i + \epsilon^2)) \cdot |N_{i,c}(v) \setminus S_i| \\ &= (1 - \epsilon \pm (4\epsilon\gamma_i + \epsilon^2)) \cdot (1 - \epsilon \pm \epsilon^2) \cdot |N_{i,c}(v)| \\ &= (1 - \epsilon \pm (4\epsilon\gamma_i + \epsilon^2))^2 \cdot |N_{i,c}(v)|. \end{aligned} \quad (36)$$

Deriving a concentration bound on $|N_{i+1,c}(v)|$ for a given $(c, v) \in [(1 \pm \epsilon^2)\Delta] \times V$:

We first identify the sampled edges in round i that are responsible for determining which edges from $N_{i,c}(v) \setminus S_i$ will end up being included in $N_{i+1,c}(v)$. Towards this end, we define $T_i(v) := \bigcup_{u \in V: (u,v) \in N_{i,c}(v) \setminus S_i} N_{i,c}(u) \cap S_i$. Observe that if an edge $(u, u') \in T_i(v)$, where $(u, v) \in N_{i,c}(v) \setminus S_i$, picks the color c in round i , then $(u, v) \notin N_{i+1,c}(v)$. Conversely, if an edge $(u, v) \in N_{i,c}(v) \setminus S_i$ ends up not being part of $N_{i+1,c}(v)$, then some edge $(u, u') \in T_i(v)$ must pick the color c in round i . For each edge $e \in T_i(v)$, define an indicator random variable $X_e \in \{0, 1\}$ that is set to one iff the edge e picks color c in round i . Clearly, the random variables $\{X_e\}$ are mutually independent. Since $|N_{i+1,c}(v)|$ is completely determined by the random variables $\{X_e\}, e \in T_i(v)$, we write $|N_{i+1,c}(v)| := f(X)$, where $X \in \{0, 1\}^{|T_i(v)|}$ follows the joint distribution of the random variables $\{X_e\}$. We now prove a concentration bound on $f(X)$.

Claim C.19. *The function f satisfies the Lipschitz property (Definition E.10) with $d_e = 2$, $e \in T_i(v)$. Furthermore, for each edge $e \in T_i(v)$, let $X_{-\{e\}} \in \{0, 1\}^{|T_i(v)|-1}$ denote the values of all the remaining variables $\{X_{e'}\}_{e' \in T_i(v) \setminus \{e\}}$. Then for all $e \in T_i(v)$ and $X_{-\{e\}} \in \{0, 1\}^{|T_i(v)|-1}$, we have:*

$$\text{var} \left[f(X) \mid X_{-\{e\}} \right] \leq \lambda_e, \text{ where } \lambda_e := \frac{8}{(1-\epsilon)^{2(i-1)}\Delta}.$$

Proof. Throughout the proof, we fix any edge $e = (u, u') \in T_i(v)$ and the color picked by every other edge $e' \in T_i(v) \setminus \{e\}$ in round i , which determine the value of $X_{-\{e\}}$. Let $Z_e = \{(v, w) \in N_{i,c}(v) \setminus S_i : w \in \{u, u'\}\}$ denote the set of edges in $N_{i,c}(v) \setminus S_i$ that are adjacent to the edge e .

When we are trying to figure out which edges from $N_{i+1,c}(v) \setminus S_i$ will end up being part of $N_{i+1,c}(v)$, observe that the color picked by e can change the *fate* of only the edges in Z_e . Indeed, if the edge e picks the color c , then the edges in Z_e will *not* be included in $N_{i+1,c}(v)$. In contrast, if the edge e picks some color $c' \in P_i(e) \setminus \{c\}$, then the edges in Z_e can potentially be included in $N_{i+1,c}(v)$.⁹ The fate of every other edge $e'' \in (N_{i,c}(v) \setminus S_i) \setminus Z_e$ is completely determined by $X_{-\{e\}}$.

Since $|Z_e| \leq 2$, the function f satisfies the Lipschitz property with $d_e = 2$, $e \in T_i(v)$. Furthermore, since the edge e picks the color $c \in P_i(e)$ with probability $1/|P_i(e)|$, we conclude that:

$$\begin{aligned} \text{Var} \left(f(X) \mid X_{-\{e\}} \right) &\leq \frac{2^2}{|P_i(e)|} \\ &= \frac{4}{(1-\epsilon)^{2(i-1)}(1 \pm \gamma_i)\Delta} && \text{(by (31))} \\ &\leq \frac{8}{(1-\epsilon)^{2(i-1)}\Delta}. && \text{(by (1) and Corollary C.6)} \end{aligned}$$

□

Claim C.20. *We have: $\Pr \left[|N_{i+1,c}(v)| = \mathbb{E} [|N_{i+1,c}(v)|] \pm 50\sqrt{\Delta \ln n} \right] \geq 1 - 1/n^{600}$.*

Proof. We derive an upper bound on the size of the set $T_i(v)$.

$$\begin{aligned} |T_i(v)| &\leq \sum_{u \in V: (u,v) \in N_{i,c}(v) \setminus S_i} |N_{i,c}(u) \cap S_i| \\ &= |N_{i,c}(v) \setminus S_i| \cdot (\epsilon \pm \epsilon^2) \cdot (1-\epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta && \text{(by (33))} \\ &\leq |N_{i,c}(v) \setminus S_i| \cdot 4\epsilon \cdot (1-\epsilon)^{2(i-1)} \cdot \Delta && \text{(by (1))} \\ &\leq 8\epsilon \cdot (1-\epsilon)^{2(i-1)} \cdot \Delta^2. && \text{(37)} \end{aligned}$$

From (1), (37) and Claim C.19, we infer that:

$$\sum_{e \in T_i(v)} \lambda_e \leq 64\epsilon \cdot \Delta \leq \Delta. \quad (38)$$

⁹In this event, whether or not an edge in Z_e is actually included in $N_{i+1,c}(v)$ will depend on $X_{-\{e\}}$.

Recall that $|N_{i+1,c}(v)| = f(X)$, where X is drawn from the joint distribution of mutually independent random variables $\{X_e\}_{e \in T_i(v)}$. Hence, from (38), Claim C.19 and Lemma E.12, we get:

$$\begin{aligned} \Pr \left[|N_{i+1,c}(v)| = \mathbb{E} [|N_{i,c}(v)|] \pm 50\sqrt{\Delta \ln n} \right] &\geq 1 - 2 \cdot \exp \left(-\frac{2500 \cdot \Delta \log n}{2\Delta + (2/3) \cdot 100 \cdot \sqrt{\Delta \log n}} \right) \\ &\geq 1 - 2 \cdot \exp \left(-\frac{2500 \cdot \Delta \log n}{4\Delta} \right) \quad (\text{by (1)}) \\ &\geq 1 - 1/n^{600}. \end{aligned}$$

This concludes the proof of the claim. \square

Corollary C.21. *We have:* $\Pr [|N_{i+1,c}(v)| = (1 - \epsilon)^{2i} \cdot (1 \pm \gamma_{i+1}) \cdot \Delta] \geq 1 - 1/n^{600}$.

Proof. Consider any $M = \mathbb{E} [|N_{i+1,c}(v)|] \pm 50\sqrt{\Delta \ln n}$. Observe that:

$$\begin{aligned} M &= (1 - \epsilon \pm (4\epsilon\gamma_i + \epsilon^2))^2 \cdot |N_{i,c}(v)| \pm 50\sqrt{\Delta \ln n} && \text{(by (36))} \\ &= \left((1 - \epsilon \pm (4\epsilon\gamma_i + \epsilon^2))^2 \pm \epsilon^2 \right) \cdot |N_{i,c}(v)| && \text{(by Claim C.14)} \\ &= ((1 - \epsilon)^2 + (4\epsilon\gamma_i + \epsilon^2)^2 \pm 2(4\epsilon\gamma_i + \epsilon^2) \pm \epsilon^2) \cdot |N_{i,c}(v)| \\ &= ((1 - \epsilon)^2 \pm (12\epsilon\gamma_i + 12\epsilon^2)) \cdot |N_{i,c}(v)| && \text{(by (1) and Corollary C.6)} \\ &= (1 - \epsilon)^2 \cdot (1 \pm (24\epsilon\gamma_i + 24\epsilon^2)) \cdot (1 - \epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta && \text{(by (1) and (32))} \\ &= (1 - \epsilon)^{2i} \cdot (1 \pm ((1 + 48\epsilon)\gamma_i + 48\epsilon^2)) \cdot \Delta && \text{(by (1) and Corollary C.6)} \\ &= (1 - \epsilon)^{2i} \cdot (1 \pm \gamma_{i+1}) \cdot \Delta. && \text{(by (2) and (10))} \end{aligned}$$

To summarize, we have derived that if $|N_{i+1,c}(v)| = \mathbb{E} [|N_{i+1,c}(v)|] \pm 50\sqrt{\Delta \ln n}$, then it must be the case that $|N_{i+1,c}(v)| = (1 - \epsilon)^{2i} \cdot (1 \pm \gamma_{i+1}) \cdot \Delta$. The corollary now follows from Claim C.20. \square

Wrapping up the proof of Lemma C.9:

Lemma C.9 follows from Corollary C.21 and a union bound over all pairs $(c, v) \in [(1 + \epsilon^2)\Delta] \times V$.

C.8 Proof of Lemma C.12

Recall the discussion on the random bits $r_{<i}$, $r_i^{(\text{edges})}$ and $r_i^{(\text{colors})}$ from Appendix C.2. We will prove the lemma stated below. Lemma C.12 follows from Lemma C.22 and a union bound over all $v \in V$.

Lemma C.22. *Fix any instantiation of the bits $r_{<i} \cup r_i^{(\text{edges})}$ which ensure the occurrence of the event $\mathcal{E}_i \cap \mathcal{C}_i \cap \mathcal{B}_i$. Fix any node $v \in V$, and let $F_i(v) = N_i(v) \cap F_i$ denote the set of failed edges in round i that are incident on v . Then we have:* $\Pr \left[|F_i(v)| \leq 9\epsilon^2 \Delta \mid r_{<i} \cup r_i^{(\text{edges})} \right] \geq 1 - 1/n^{305}$.

The rest of Appendix C.8 is devoted to the proof of Lemma C.22. We fix the bits $r_{<i} \cup r_i^{(\text{edges})}$ which ensure the occurrence of the event $\mathcal{E}_i \cap \mathcal{C}_i \cap \mathcal{B}_i$. To ease notations, henceforth we refrain from repeatedly stating that we are conditioning on the bits $r_{<i} \cup r_i^{(\text{edges})}$. However, the reader should keep in mind that we are relying upon this conditioning for the rest of Appendix C.8.

A classification of failed edges:

Let $F_i^{(1)}(v) = \{(u, v) \in F_i(v) : c(v', v) = c(u, v) \text{ for some } (v', v) \in N_i(v) \cap S_i\}$ denote the set of edges (u, v) that fails in round i because of the following reason: Some other edge incident on v picks the same color as (u, v) in round i . Let $F_i^{(2)}(v) = F_i(v) \setminus F_i^{(1)}(v)$ denote the set of remaining edges incident on v that fails in round i . An edge $(u, v) \in N_i(v) \cap S_i$ belongs to the set $F_i^{(2)}(v)$ iff no other edge $e_v \in N_i(v) \cap S_i$ picks the same color as (u, v) in round i , and at least one edge $e_u \in N_i(u) \cap S_i$ picks the same color as (u, v) in round i . We say that a failed edge $e \in F_i(v)$ is of type-(1) iff $e \in F_i^{(1)}(v)$ and it is of type-(2) iff $e \in F_i^{(2)}(v)$. We will separately prove concentration bounds on the number of failed type-(1) and type-(2) edges incident on v . As $|F_i(v)| = |F_i^{(1)}(v)| + |F_i^{(2)}(v)|$, this will lead to the desired concentration bound on $|F_i(v)|$.

Deriving a concentration bound on $|F_i^{(1)}(v)|$:

Claim C.23 bounds the expected value of $|F_i^{(1)}(v)|$. Claim C.24 shows that w.h.p. $|F_i^{(1)}(v)|$ does not deviate too far away from its expectation. Claim C.24 follows from Claim C.23 and Claim C.24.

Claim C.23. We have $\mathbb{E} \left[|F_i^{(1)}(v)| \right] \leq 4\epsilon^2 \Delta$.

Proof. Note that $F_i^{(1)}(v) \subseteq N_i(v) \cap S_i$. Consider any edge $e = (u, v) \in N_i(v) \cap S_i$. Our immediate goal is to bound the probability that this edge e does *not* belong to $F_i^{(1)}(v)$, conditioned on it picking a given color $c \in P_i(e)$ in round i . Towards this end, we first recall that the bits $r_{<i} \cup r_i^{(\text{edges})}$ we are conditioning upon ensure the occurrence of the event $\mathcal{E}_i \cap \mathcal{C}_i \cap \mathcal{B}_i$. Hence, we have:

$$|P_i(e')| = (1 - \epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta \text{ for all edges } e' \in E_i. \quad (39)$$

$$|N_{i,c}(x)| = (1 - \epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta \text{ for all nodes } x \in V. \quad (40)$$

$$|N_{i,c}(x) \cap S_i| = (\epsilon \pm \epsilon^2) \cdot (1 - \epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta \text{ for all nodes } x \in V. \quad (41)$$

Conditioned on the edge e picking the color $c \in P_i(e)$, it does not belong to $F_i^{(1)}(v)$ iff none of the edges $e' \in (N_{i,c}(v) \cap S_i) \setminus \{e\}$ picks the same color c in round i . Hence, we derive that:

$$\begin{aligned} \Pr \left[e \notin F_i^{(1)}(v) \mid c(e) = c \right] &= \prod_{e' \in (N_{i,c}(v) \cap S_i) \setminus \{e\}} \left(1 - \frac{1}{|P_i(e')|} \right) \\ &= \left(1 - \frac{1}{(1 - \epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta} \right)^{|N_{i,c}(v) \cap S_i| - 1} \\ &\geq \left(1 - \frac{1}{(1 - \epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta} \right)^{|N_{i,c}(v) \cap S_i|} \\ &= \left(1 - \frac{1}{(1 - \epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta} \right)^{(\epsilon \pm \epsilon^2) \cdot (1 - \epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta} \\ &\geq 1 - \epsilon - (4\epsilon\gamma_i + \epsilon^2) \\ &\geq 1 - 2\epsilon. \end{aligned} \quad (42)$$

In the derivation above, the second step follows from (39), the fourth step follows from (41), the fifth step follows from Claim C.15, and the last step follows from (1) and Corollary C.6. Since (42)

holds for every color $c \in P_i(e)$, we conclude that:

$$\Pr \left[e \notin F_i^{(1)}(v) \right] \geq 1 - 2\epsilon \text{ for every edge } e \in N_i(v) \cap S_i. \quad (43)$$

Now, applying linearity of expectation, we get:

$$\begin{aligned} \mathbb{E} \left[|F_i^{(1)}(v)| \right] &= \sum_{e \in N_i(v) \cap S_i} \Pr \left[e \in F_i^{(1)}(v) \right] \\ &\leq |N_i(v) \cap S_i| \cdot (2\epsilon) && \text{(by (43))} \\ &\leq (2\epsilon\Delta) \cdot (2\epsilon) && \text{(by (1), (41) and Corollary C.6)} \\ &= 4\epsilon^2\Delta. \end{aligned}$$

This concludes the proof of the claim. \square

Claim C.24. *We have:* $\Pr \left[|F_i^{(1)}(v)| \leq \mathbb{E} \left[|F_i^{(1)}(v)| \right] + 50\sqrt{\Delta \ln n} \right] \geq 1 - 1/n^{310}$.

Proof. For each edge $e \in N_i(v) \cap S_i$, define a random variable $X_e \in P_i(e)$ whose value indicates the color picked by the edge e in round i . The quantity $|F_i^{(1)}(v)|$ is a function of the random variables $\{X_e\}, e \in N_i(v) \cap S_i$, and the random variables $\{X_e\}$ themselves are mutually independent.

We claim that the function $|F_i^{(1)}(v)|$ satisfies the Lipschitz property (see Definition E.10) with constants $d_e = 4, e \in N_i(v) \cap S_i$. To see why the claim holds, consider any given edge $e \in N_i(v) \cap S_i$ and fix the colors picked by every other edge $e' \in (N_i(v) \cap S_i) \setminus \{e\}$ in round i . Fix any two distinct colors $c_1, c_2 \in P_i(e)$. Let n_{c_1} and n_{c_2} respectively denote the number of edges $e' \in (N_i(v) \cap S_i) \setminus \{e\}$ that have picked color c_1 and color c_2 in round i . Now, consider the following two scenarios:

- (1) The edge e picks the color $c_1 \in P_i(e)$ in round i .
- (2) The edge e picks the color $c_2 \in P_i(e) \setminus \{c_1\}$ in round i .

As we switch from scenario (1) to scenario (2), the number of type-(1) failed edges in $N_i(v) \cap S_i$ that pick color c_2 increases by $\phi(n_{c_2} + 1) - \phi(n_{c_2})$; where $\phi(y) = y$ if $y \geq 2$, and $\phi(y) = 0$ otherwise. Similarly, the number of type-(1) failed edges in $N_i(v) \cap S_i$ that pick color c_1 decreases by $\phi(n_{c_1} + 1) - \phi(n_{c_1})$. In contrast, the number of type-(1) failed edges in $N_i(v) \cap S_i$ that pick any color $c \notin [\Delta] \setminus \{c_1, c_2\}$ remains unchanged. Thus, as we switch from scenario (1) to scenario (2), the absolute value of the change in $|F_i^{(1)}(v)|$ is given by:

$$\begin{aligned} |\{\phi(n_{c_2} + 1) - \phi(n_{c_2})\} - \{\phi(n_{c_1} + 1) - \phi(n_{c_1})\}| &\leq |\phi(n_{c_2} + 1) - \phi(n_{c_2})| + |\phi(n_{c_1} + 1) - \phi(n_{c_1})| \\ &\leq 2 + 2 = 4. \end{aligned}$$

We therefore conclude that $|F_i^{(1)}(v)|$ is a function of mutually independent random variables $\{X_e\}$ that satisfy the Lipschitz property (see Definition E.10) with constants $d_e = 4, e \in N_i(v) \cap S_i$. Since $\sum_{e \in N_i(v) \cap S_i} d_e^2 = 16 \cdot |N_i(v) \cap S_i| \leq 16\Delta$, applying Lemma E.11 we get:

$$\Pr \left[|F_i^{(1)}(v)| \leq \mathbb{E} \left[|F_i^{(1)}(v)| \right] + 50\sqrt{\Delta \ln n} \right] \geq 1 - 1/n^{310}.$$

This concludes the proof of the claim. \square

Corollary C.25. *We have: $\Pr \left[|F_i^{(1)}(v)| \leq 4\epsilon^2\Delta + 50\sqrt{\Delta \ln n} \right] \geq 1 - 1/n^{310}$.*

Proof. Follows from Claim C.23 and Claim C.24. \square

Deriving a concentration bound on $|F_i^{(2)}(v)|$:

While analyzing this quantity, for technical reasons we first condition upon the colors picked by all the edges incident on v that are sampled in round i . After this conditioning, we bound the expected value of $|F_i^{(2)}(v)|$ in Claim C.26, and in Claim C.27 we show that w.h.p. $|F_i^{(2)}(v)|$ does not deviate too far away from its expectation. Corollary C.28 follows from Claim C.26 and Claim C.27.

Claim C.26. *Fix any color $c^*(e) \in P_i(e)$ for every edge $e \in N_i(v) \cap S_i$. Let Γ^* be the event which occurs iff every edge $e \in N_i(v) \cap S_i$ picks the color $c^*(e)$ in round i . Then we have:*

$$\mathbb{E} \left[|F_i^{(2)}(v)| \mid \Gamma^* \right] \leq 4\epsilon^2\Delta.$$

Proof. The proof is analogous to the proof of Claim C.23. Nevertheless, for the sake of completeness, we reproduce the same chain of reasoning in its entirety.

The event Γ^* completely determines the set $F_i^{(1)}(v)$. Furthermore, we have $F_i^{(2)}(v) \subseteq (N_i(v) \cap S_i) \setminus F_i^{(1)}(v)$. Consider any edge $e = (u, v) \in (N_i(v) \cap S_i) \setminus F_i^{(1)}(v)$, which picks the color $c^*(e)$ in round i . Our immediate goal is to bound the probability that this edge e does *not* belong to the set $F_i^{(2)}(v)$. Towards this end, we first recall that the bits $r_{<i} \cup r_i^{(\text{edges})}$ we have already conditioned upon ensure the occurrence of the event $\mathcal{E}_i \cap \mathcal{C}_i \cap \mathcal{B}_i$. Hence, we have:

$$|P_i(e')| = (1 - \epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta \text{ for all edges } e' \in E_i. \quad (44)$$

$$|N_{i,c^*(e)}(x)| = (1 - \epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta \text{ for all nodes } x \in V. \quad (45)$$

$$|N_{i,c^*(e)}(x) \cap S_i| = (\epsilon \pm \epsilon^2) \cdot (1 - \epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta \text{ for all nodes } x \in V. \quad (46)$$

The edge $e = (u, v)$ does not belong to the set $F_i^{(2)}(v)$ iff no edge $e' \in (N_{i,c^*(e)}(u) \cap S_i) \setminus \{e\}$ picks the color $c(e') = c^*(e)$ in round i . Hence, we derive that:

$$\begin{aligned} \Pr \left[e \notin F_i^{(2)}(v) \mid \Gamma^* \right] &= \prod_{e' \in (N_{i,c^*(e)}(u) \cap S_i) \setminus \{e\}} \left(1 - \frac{1}{|P_i(e')|} \right) \\ &= \left(1 - \frac{1}{(1 - \epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta} \right)^{|N_{i,c^*(e)}(u) \cap S_i| - 1} \\ &\geq \left(1 - \frac{1}{(1 - \epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta} \right)^{|N_{i,c^*(e)}(u) \cap S_i|} \\ &= \left(1 - \frac{1}{(1 - \epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta} \right)^{(\epsilon \pm \epsilon^2) \cdot (1 - \epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta} \\ &\geq 1 - \epsilon - (4\epsilon\gamma_i + \epsilon^2) \\ &\geq 1 - 2\epsilon. \end{aligned} \quad (47)$$

In the derivation above, the second step follows from (44), the fourth step follows from (46), the fifth step follows from Claim C.15, the last step follows from (1) and Corollary C.6. Thus, we have:

$$\Pr \left[e \notin F_i^{(2)}(v) \mid \Gamma^* \right] \geq 1 - 2\epsilon \text{ for every edge } e \in (N_i(v) \cap S_i) \setminus F_i^{(1)}(v). \quad (48)$$

Now, applying linearity of expectation, we get:

$$\begin{aligned} \mathbb{E} \left[|F_i^{(2)}(v)| \mid \Gamma^* \right] &= \sum_{e \in (N_i(v) \cap S_i) \setminus F_i^{(1)}(v)} \Pr \left[e \in F_i^{(2)}(v) \mid \Gamma^* \right] \\ &\leq |N_i(v) \cap S_i| \cdot (2\epsilon) && \text{(by (48))} \\ &\leq (2\epsilon\Delta) \cdot (2\epsilon) && \text{(by (1), (46) and Corollary C.6)} \\ &= 4\epsilon^2\Delta. \end{aligned}$$

This concludes the proof of the claim. \square

Claim C.27. *Fix any color $c^*(e) \in P_i(e)$ for every edge $e \in N_i(v) \cap S_i$. Let Γ^* be the event which occurs iff every edge $e \in N_i(v) \cap S_i$ picks the color $c^*(e)$ in round i . Then we have:*

$$\Pr \left[|F_i^{(2)}(v)| \leq \mathbb{E} \left[|F_i^{(2)}(v)| \right] + 50\sqrt{\Delta \ln n} \mid \Gamma^* \right] \geq 1 - 1/n^{2000}.$$

Proof. Let $W_i(v) = (N_i(v) \cap S_i) \setminus F_i^{(1)}(v)$ denote the set of edges incident on v that get sampled in round i and do *not* end up being type-(1) failures under the event Γ^* . By definition, all the edges in $W_i(v)$ receive distinct colors under the event Γ^* , and we have $F_i^{(2)}(v) \subseteq W_i(v)$. Let $Z_i(v) = \{(u, u') \in S_i \setminus N_i(v) : \text{either } u \in W_i(v) \text{ or } u' \in W_i(v)\}$ denote the set of edges sampled in round i that are not themselves incident on v , but are neighbors of at least one edge in $W_i(v)$. Note that the sets $W_i(v)$ and $Z_i(v)$, along with the colors picked by the edges in $W_i(v)$, are completely determined by the bits $r_{<i} \cup r_i^{(\text{edges})}$ and the event Γ^* we are conditioning upon. On the other hand, the colors picked by the edges in $Z_i(v)$ are yet to be determined. In particular, each edge in $Z_i(v)$ will pick a color uniformly at random from the set $P_i(e)$, independently of the other edges, and these colors will uniquely determine the set $F_i^{(2)}(v)$. We can think of the edges $e \in Z_i(v)$ picking their colors in round i as a “balls and bins” process, as described below.

There is a ball for each edge $e \in Z_i(v)$, a bin for each color $c \in [(1 + \epsilon^2)\Delta]$, and an additional dummy bin \perp . Consider any edge $e \in Z_i(v)$, and let $W_i(v, e) \subseteq W_i(v)$ be the set of edges $e' \in W_i(v)$ that share an endpoint with e . Note that $|W_i(v, e)| \in \{1, 2\}$. Suppose that the edge e picks a color $c \in P_i(e)$ in round i . Then the ball for e gets thrown into the bin for c iff some edge $e' \in W_i(v, e)$ picked the same color c under the event Γ^* ; otherwise the ball for e goes to the dummy bin.

Let $\chi_i(v) = \bigcup_{e \in W_i(v)} \{c^*(e)\}$ denote the set of colors picked by the edges $e \in W_i(v)$ in round i , under the event Γ^* . For each color $c \in \chi_i(v)$, define an indicator random variable $Y_c \in \{0, 1\}$ that is set to one iff the bin for the color c is nonempty (has at least one ball in it) at the end of the balls and bins process described above. As each edge $e \in Z_i(v)$ picks a color independently of the other edges in $Z_i(v)$, the balls get thrown into the bins independently of each other. Hence, Corollary E.7 implies that the random variables $\{Y_c\}, c \in \chi_i(v)$, are negatively associated. Since $|\chi_i(v)| \leq (1 + \epsilon^2)\Delta$, from Lemma E.9 we get:

$$\Pr \left[\sum_{c \in \chi_i(v)} Y_c \leq \mathbb{E} \left[\sum_{c \in \chi_i(v)} Y_c \right] + 50\sqrt{\Delta \ln n} \mid \Gamma^* \right] \geq 1 - 1/n^{2000}. \quad (49)$$

Recall that no two edges in $W_i(v)$ pick the same color under the event Γ^* . Accordingly, there is a natural bijective mapping $g : \chi_i(v) \rightarrow W_i(v)$, where $g(c)$ is the unique edge in $W_i(v)$ that picked the color $c \in \chi_i(v)$ under the event Γ^* . For each color $c \in \chi_i(v)$, we have $Y_c = 1$ iff $g(c) \in F_i^{(2)}(v)$. Since $F_i^{(2)}(v) \subseteq W_i(v)$, we infer that $\sum_{c \in \chi_i(v)} Y_c = |F_i^{(2)}(v)|$. The claim now follows from (49). \square

Corollary C.28. *We have:* $\Pr \left[|F_i^{(2)}(v)| \leq 4\epsilon^2\Delta + 50\sqrt{\Delta \ln n} \right] \geq 1 - 1/n^{2000}$.

Proof. Fix any color $c^*(e) \in P_i(e)$ for all $e \in N_i(v) \cap S_i$. Let Γ^* be the event which occurs iff every edge $e \in N_i(v) \cap S_i$ picks the color $c^*(e)$ in round i . Claim C.26 and Claim C.27 imply that:

$$\Pr \left[|F_i^{(2)}(v)| \leq 4\epsilon^2\Delta + 50\sqrt{\Delta \ln n} \mid \Gamma^* \right] \geq 1 - 1/n^{2000}. \quad (50)$$

Since the bound in (50) holds for every possible Γ^* , the corollary follows. \square

Wrapping up the proof of Lemma C.22:

Applying a union bound over Corollary C.25 and Corollary C.28, we get:

$$\Pr \left[|F_i^{(1)}(v)| + |F_i^{(2)}(v)| \leq 8\epsilon^2\Delta + 100\sqrt{\Delta \ln n} \right] \geq 1 - 1/n^{305}.$$

Now, Lemma C.22 follows from the following two observations: (a) $|F_i(v)| = |F_i^{(1)}(v)| + |F_i^{(2)}(v)|$, and (b) $100\sqrt{\Delta \ln n} \leq \epsilon^2\Delta$ according to (1).

D Missing Proofs from Section 4

D.1 Proof of Lemma 4.1

We start by proving the key property for our dynamic algorithm, which implies that it implements the first phase of Algorithm 1. We re-state this lemma here for ease of reference.

Lemma 4.1. *For each time $t \geq 0$, the joint distribution $\{c^{(t)}(e)\}_{e \in E^{(t)}}$ of colors sampled by the dynamic algorithm is distributed identically to $\{c(e)\}_e$ of Algorithm 1 when applied to graph $G^{(t)}$.*

To prove the above, we must prove that each edge $e \in S_i^{(t)}$ has a tentative color $c^{(t)}$ sampled u.a.r. from $P_i^{(t)}(e)$ (or null if $P_i^{(t)}(e)$), independently of all other edges in $S_i^{(t)}$. To prove this, we will prove a stronger claim, whereby all edges in S_i (including those not in $E^{(t)}$) have a tentative color $c^{(t)}$ sampled u.a.r. from $P_i^{(t)}$, independent of all edges in S_i .

To prove the above, we will think of $c^{(t)}$ and $P_i^{(t)}$ restricted to S_i as $|S_i|$ -dimensional vectors, indexed by (fake or real) edges $e \in S_i$. Moreover, we let $P(e) = \{0\}$ indicate $P(e) = \emptyset$, using $c^{(t)}(e) = 0$ to mean $c^{(t)}(e) = \text{null}$. With this notation, we are ready to state our key lemma.

Lemma D.1. *For all times t , real vector \vec{c} and vector of real sets \vec{P} for which $\Pr[P_i^{(t)} = \vec{P}] \neq 0$,*

$$\Pr[c^{(t)} = \vec{c} \mid P_i^{(t)} = \vec{P}] = \begin{cases} \prod_{e \in S_i} \frac{1}{|P(e)|} & c(e) \in P(e) \quad \forall e \in S_i \\ 0 & \text{else.} \end{cases}$$

Lemma D.1 implies that the coloring of edges in $E^{(t)} \cap \bigcup_i (S_i \setminus F_i)$ shares the same joint distribution as its counterpart in $G^{(t)}$. This immediately leads to Lemma 4.1.

Proof. We prove this lemma by induction on $t \geq 0$, referring to the hypothesis we wish to prove for time t by \mathcal{H}^t . Hypothesis \mathcal{H}^0 holds trivially. We now prove that \mathcal{H}^{t-1} implies \mathcal{H}^t .

First, we consider the same probability when further conditioning on the previous palette, $P_i^{(t-1)}$, for which we have that for all vector of real sets \vec{Q} ,

$$\begin{aligned} & \Pr[c^{(t)} = \vec{c} \mid P_i^{(t)} = \vec{P}, P_i^{(t-1)} = \vec{Q}] \\ &= \sum_{\vec{d}} \Pr[c^{(t)} = \vec{c} \mid c^{(t-1)} = \vec{d}, P_i^{(t)} = \vec{P}, P_i^{(t-1)} = \vec{Q}] \cdot \Pr[c^{(t-1)} = \vec{d} \mid P_i^{(t)} = \vec{P}, P_i^{(t-1)} = \vec{Q}]. \end{aligned} \quad (51)$$

We begin by simplifying the term $\Pr[c^{(t-1)} = \vec{d} \mid P_i^{(t)} = \vec{P}, P_i^{(t-1)} = \vec{Q}]$, in the following claim.

Claim D.2. *For any real vector \vec{d} and vector of real set \vec{Q} , we have*

$$\Pr[c^{(t-1)} = \vec{d} \mid P_i^{(t)} = \vec{P}, P_i^{(t-1)} = \vec{Q}] = \begin{cases} \prod_{e \in S_i} \frac{1}{|Q(e)|} & d(e) \in Q(e) \quad \forall e \in S_i \\ 0 & \text{else,} \end{cases}$$

Proof. By the oblivious adversary assumption, the changes between $P_i^{(t)}$ and $P_i^{(t-1)}$ —the color palettes of round i of times t and $t-1$ —are independent of the candidate colors of S_i edges in time $t-1$, namely values $c^{(t-1)}$. Therefore, the joint distributions $(c^{(t-1)} \mid P_i^{(t)} = \vec{P}, P_i^{(t-1)} = \vec{Q})$ and $(c^{(t-1)} \mid P_i^{(t-1)} = \vec{Q})$ are identically distributed. The claim thus follows by Hypothesis \mathcal{H}^{t-1} . \square

By the above claim and the preceding equation, we have that

$$\begin{aligned} & \Pr[c^{(t)} = \vec{c} \mid P_i^{(t)} = \vec{P}, P_i^{(t-1)} = \vec{Q}] \\ &= \sum_{\vec{d}} \Pr[c^{(t)} = \vec{c} \mid c^{(t-1)} = \vec{d}, P_i^{(t)} = \vec{P}, P_i^{(t-1)} = \vec{Q}] \cdot \prod_{e \in S_i} \frac{1}{|Q(e)|}. \end{aligned} \quad (52)$$

On the other hand, if we denote for each edge $e \in S_i$ the color sets $Q_{\cap}(e) := Q(e) \cap P(e)$ and $Q_{\setminus}(e) := Q(e) \setminus P(e)$, then by the definition of Algorithm 2, we have that

$$\Pr[c^{(t)} = \vec{c} \mid c^{(t-1)} = \vec{d}, P_i^{(t)} = \vec{P}, P_i^{(t-1)} = \vec{Q}] = \prod_{e: d(e) \in Q_{\setminus}(e)} \frac{1}{|P(e)|} \prod_{e: c(e) = d(e) \in Q_{\cap}(e)} \frac{|Q_{\cap}(e)|}{|P(e)|}. \quad (53)$$

Summing Equation (53) over all vectors \vec{d} , Equation (52) then implies that, if we let every edge in the following equation implicitly belong to S_i , then

$$\begin{aligned} \Pr[c^{(t)} = \vec{c} \mid P_i^{(t)} = \vec{P}, P_i^{(t-1)} = \vec{Q}] &= \sum_{\vec{d}} \prod_{e: d(e) \in Q_{\setminus}(e)} \frac{1}{|P(e)|} \prod_{e: c(e) = d(e) \in Q_{\cap}(e)} \frac{|Q_{\cap}(e)|}{|P(e)|} \prod_e \frac{1}{|Q(e)|} \\ &= \prod_e \frac{|Q_{\setminus}(e)| + |Q_{\cap}(e)|}{|P(e)|} \prod_{e \in S_i} \frac{1}{|Q(e)|} \\ &= \prod_e \frac{1}{|P(e)|}, \end{aligned}$$

where the last step relied on $Q_{\cap}(e)$ and $Q_{\setminus}(e)$ being a partition of $Q(e)$, which therefore implies $|Q_{\setminus}(e)| + |Q_{\cap}(e)| = |Q(e)|$.

But then, by total probability over $P_i^{(t-1)}$, we therefore have that Hypothesis \mathcal{H}^t holds, as

$$\Pr[c^{(t)} = \vec{c} \mid P_i^{(t)} = \vec{P}] = \sum_{\vec{Q}} \Pr[c^{(t)} = \vec{c} \mid P_i^{(t)} = \vec{P}, P_i^{(t-1)} = \vec{Q}] \cdot \Pr[P_i^{(t-1)} = \vec{Q}] = \prod_{e \in S_i} \frac{1}{|P(e)|}. \square$$

D.2 Proof of Lemma 4.4

In this section, we will need to use some key concepts from the analysis of Algorithm 1. In particular, before proceeding any further, the reader will find it useful to review all of Appendix C.2, and the statement of Corollary C.10 from Appendix C.3.

Throughout this section, we use the symbol $\mathcal{E}_i^{(t)}$ to denote the random event \mathcal{E}_i for the output of the dynamic algorithm at the end of the t^{th} update. To be very specific, the event $\mathcal{E}_i^{(t)}$ occurs iff $|P_i^{(t)}(e)| = (1 - \epsilon)^{2(i-1)} \cdot (1 \pm \gamma_i) \cdot \Delta$ for all edges $e \in E_i^{(t)}$. Similarly, we will use the notations $\mathcal{C}_i^{(t)}$ and $\mathcal{B}_i^{(t)}$ to respectively denote the corresponding events \mathcal{C}_i and \mathcal{B}_i at the end of the t^{th} update.

We now define the following random event that will play an important role in this section.

$$\mathcal{Z} := \mathcal{E}_i^{(t-1)} \cap \mathcal{C}_i^{(t-1)} \cap \mathcal{B}_i^{(t-1)} \cap \mathcal{E}_i^{(t)} \cap \mathcal{C}_i^{(t)} \cap \mathcal{B}_i^{(t)}.$$

Lemma D.3. *We have $\mathbb{E}[|D_i| \mid \mathcal{Z}] \leq 5\epsilon \cdot (1 + \mathbb{E}[|D_{<i}| \mid \mathcal{Z}])$.*

The complete proof of Lemma D.3 appears in Appendix D.2.1. For now, we focus on showing how Lemma D.3 almost immediately leads us to the proof of Lemma 4.4.

Lemma D.4. *We have $\mathbb{E}[|D_i| \mid \neg\mathcal{Z}] \cdot \Pr[\neg\mathcal{Z}] \leq \epsilon$.*

Proof. Corollary C.10 and Lemma 4.1 imply that:

$$\Pr[\mathcal{E}_i^{(t-1)} \cap \mathcal{C}_i^{(t-1)} \cap \mathcal{B}_i^{(t-1)}] \geq 1 - 1/n^{400}. \quad (54)$$

$$\Pr[\mathcal{E}_i^{(t)} \cap \mathcal{C}_i^{(t)} \cap \mathcal{B}_i^{(t)}] \geq 1 - 1/n^{400}. \quad (55)$$

Taking a union bound over (54) and (55), we get:

$$\Pr[\neg\mathcal{Z}] \leq 2/n^{400}. \quad (56)$$

Since $|D_i| \leq n^2$ with probability one, (56) implies that: $\mathbb{E}[|D_i| \mid \neg\mathcal{Z}] \cdot \Pr[\neg\mathcal{Z}] \leq n^2 \cdot (2/n^{400}) \leq \epsilon$. \square

Proof of Lemma 4.4: From Lemma D.3 and Lemma D.4, we now derive that:

$$\begin{aligned} \mathbb{E}[|D_i|] &= \mathbb{E}[|D_i| \mid \mathcal{Z}] \cdot \Pr[\mathcal{Z}] + \mathbb{E}[|D_i| \mid \neg\mathcal{Z}] \cdot \Pr[\neg\mathcal{Z}] \\ &\leq 5\epsilon \cdot (1 + \mathbb{E}[|D_{<i}| \mid \mathcal{Z}]) \cdot \Pr[\mathcal{Z}] + \epsilon \\ &= 5\epsilon \cdot \Pr[\mathcal{Z}] + 5\epsilon \cdot \mathbb{E}[|D_{<i}| \mid \mathcal{Z}] \cdot \Pr[\mathcal{Z}] + \epsilon \\ &\leq 6\epsilon + 6\epsilon \cdot \mathbb{E}[|D_{<i}|]. \end{aligned}$$

In the above derivation, the first inequality follows from Lemma D.3 and Lemma D.4.

D.2.1 Proof of Lemma D.3

Let e^* be the edge that gets inserted/deleted during the t^{th} update, that is, the sets $E^{(t)}$ and $E^{(t-1)}$ differ only in the edge e^* . Now, define the set $D_{<i}^*$ as follows.

$$D_{<i}^* := \begin{cases} D_{<i} \cup \{e^*\} & \text{if } i(e^*) < i; \\ D_{<i} & \text{otherwise.} \end{cases}$$

For the rest of the proof, we will fix (condition upon) the following “critical” random variables:

- The set of edges $S_{\leq i}^* = \{e \in E^{(t)} \cup E^{(t-1)} : i(e) \leq i\}$ that get assigned to levels at most i .
- The tentative colors $c^{(t-1)}(e), c^{(t)}(e)$ for every edge $e \in S_{\leq i-1}^* = \{e' \in S_{\leq i}^* : i(e') \leq i-1\}$.

Note that once we fix the critical random variables, the occurrence of the event \mathcal{Z} and the contents of the sets $D_{<i}^*, D_{<i}$ are completely determined. Our main goal will be to show that:

$$\mathbb{E}[|D_i|] \leq 5\epsilon \cdot |D_{<i}^*|, \text{ if we fix the critical random variables in such a way that event } \mathcal{Z} \text{ occurs.} \quad (57)$$

Since $|D_{<i}^*| \leq 1 + |D_{<i}|$, Lemma D.3 follows from (57) if we take expectations on both sides while still conditioning upon the event \mathcal{Z} . Accordingly, for the rest of the proof, we fix the critical random variables in such a way that the event \mathcal{Z} occurs, and focus on proving (57). For ease of exposition, however, from now on we will refrain from explicitly stating again and again that we have conditioned on the critical random variables.

Our first task is to identify, for each edge $e \in D_i$, the set of edges in $D_{<i}^*$ that are responsible for e changing its tentative color during the t^{th} update. Recall that an edge $e \in D_i$ changes its tentative color either in line (2) of Algorithm 2 (in this case we say that the edge e is of type-(a)), or in line (6) of Algorithm 2 (in this case we say that the edge e is of type-(b)). We now define:

$$R(e) := \{e' \in N(e) \cap D_{<i}^* : c^{(t)}(e') = c^{(t-1)}(e)\} \text{ for every edge } e \in D_i \text{ of type-(a).} \quad (58)$$

$$R(e) := \{e' \in N(e) \cap D_{<i}^* : c^{(t-1)}(e') = c^{(t)}(e)\} \text{ for every edge } e \in D_i \text{ of type-(b).} \quad (59)$$

To appreciate the rationale behind (58), consider any edge $e \in D_i$ of type-(a). For such an edge e , line (1) of Algorithm 2 implies that $c^{(t-1)}(e) \in P_i^{(t-1)}(e) \setminus P_i^{(t)}(e)$. This happens only if e has some neighboring edges e' in $D_{<i}^*$ with $c^{(t)}(e') = c^{(t-1)}(e)$. These neighboring edges are precisely the ones that are *responsible* for e becoming part of the set D_i .

Similarly, to appreciate the rationale behind (59), consider any edge $e \in D_i$ of type-(b). For such an edge e , line (5) of Algorithm 2 implies that $c^{(t)}(e) \in P_i^{(t)}(e) \setminus P_i^{(t-1)}(e)$.¹⁰ This happens only if e has some neighboring edges e' in $D_{<i}^*$ with $c^{(t-1)}(e') = c^{(t)}(e)$. These neighboring edges are precisely the ones that are *responsible* for e becoming part of the set D_i .

The discussion above leads us to the following observation.

¹⁰Since conditioned on \mathcal{Z} , the event $\mathcal{E}_i^{(t)}$ occurs with probability one, we have $P_i^{(t)}(e) \neq \emptyset$ and hence $c^{(t)}(e) \neq \text{null}$.

Observation D.5. We have $R(e) \neq \emptyset$ for all edges $e \in D_i$.

The sets $\{R(e)\}_{e \in D_i}$ tell us, for every $e \in D_i$, which edges in $D_{<i}^*$ are responsible for e changing its tentative color during the t^{th} update. We now want to look at the same picture, but from the point of view of the edges in $D_{<i}^*$. Accordingly, we define:

$$R^{-1}(e') = \{e \in D_i : e' \in R(e)\} \text{ for all } e' \in D_{<i}^*. \quad (60)$$

$$R_a^{-1}(e') = \{e \in D_i : e \text{ is of type-(a) and } e' \in R(e)\} \text{ for all } e' \in D_{<i}^*. \quad (61)$$

$$R_b^{-1}(e') = \{e \in D_i : e \text{ is of type-(b) and } e' \in R(e)\} \text{ for all } e' \in D_{<i}^*. \quad (62)$$

We are now derive a useful claim.

Claim D.6. We have: $\mathbb{E}[|D_i|] \leq \sum_{e' \in D_{<i}^*} (\mathbb{E}[|R_a^{-1}(e')|] + \mathbb{E}[|R_b^{-1}(e')|])$.

Proof. Observation D.5 implies that $D_i = \bigcup_{e' \in D_{<i}^*} R^{-1}(e')$. Since each $R^{-1}(e')$ is partitioned into two subsets $R_a^{-1}(e')$ and $R_b^{-1}(e')$, we get:

$$|D_i| \leq \sum_{e' \in D_{<i}^*} (|R_a^{-1}(e')| + |R_b^{-1}(e')|).$$

The claim follows if we take expectations on both sides of the above inequality. \square

Claim D.6 suggests the following natural approach for coming up an upper bound on $\mathbb{E}[|D_i|]$: Separately upper bound $\mathbb{E}[|R_a^{-1}(e')|]$ and $\mathbb{E}[|R_b^{-1}(e')|]$ for every edge $e' \in D_{<i}^*$, and then sum up the resulting inequalities. We implement this approach in the next two claims.

Claim D.7. We have $\mathbb{E}[|R_a^{-1}(e')|] \leq (5/2) \cdot \epsilon$ for every edge $e' \in D_{<i}^*$.

Proof. Fix any edge $e' = (u, v) \in D_{<i}^*$, and let $c = c^{(t)}(e')$ denote its tentative color in $G^{(t)}$. Note that for an edge $e \in E^{(t-1)} \cap E^{(t)}$ to be included in the set $R_a^{-1}(e')$, it needs to: (1) be a neighbor of e' with $i(e) = i$ and (2) have the color c in its palette for round i in $G^{(t-1)}$. This happens iff $e \in \left((N_{i,c}^{(t-1)}(u) \cap S_i^{(t-1)}) \cup (N_{i,c}^{(t-1)}(v) \cap S_i^{(t-1)}) \right) \setminus \{e^*\}$.¹¹ For such an edge e , we have:

$$\begin{aligned} \Pr[e \in R_a^{-1}(e')] &= \Pr[c^{(t-1)}(e) = c] = \frac{1}{|P_i^{(t-1)}(e)|} \\ &\leq \frac{1}{(1 - \epsilon)^{2(i-1)} \cdot (1 - \gamma_i) \cdot \Delta} \end{aligned} \quad (63)$$

In the above derivation, the second equality holds because of Lemma 4.1. The last inequality holds since conditioned on \mathcal{Z} , the event $\mathcal{E}_i^{(t-1)}$ occurs with probability one. Now, summing (63) over all

¹¹Note that this specific set is completely determined once we condition upon the critical random variables.

the relevant edges e , we get:

$$\begin{aligned}
\mathbb{E}[|R_a^{-1}(e')|] &= \sum_e \Pr[e \in R_a^{-1}(e')] \\
&\leq \frac{|N_{i,c}^{(t-1)}(u) \cap S_i^{(t-1)}| + |N_{i,c}^{(t-1)}(v) \cap S_i^{(t-1)}|}{(1-\epsilon)^{2(i-1)} \cdot (1-\gamma_i) \cdot \Delta} \\
&\leq \frac{2 \cdot (\epsilon + \epsilon^2) \cdot (1-\epsilon)^{2(i-1)} \cdot (1+\gamma_i) \cdot \Delta}{(1-\epsilon)^{2(i-1)} \cdot (1-\gamma_i) \cdot \Delta} \\
&\leq 2(\epsilon + \epsilon^2)(1+3\gamma_i) \leq (5/2) \cdot \epsilon.
\end{aligned}$$

The second inequality follows from the fact that conditioned on \mathcal{Z} , the event $\mathcal{B}_i^{(t-1)} \cap \mathcal{C}_i^{(t-1)}$ occurs with probability one. The last two inequalities follow from (1) and Corollary C.6. \square

Claim D.8. *We have $\mathbb{E}[|R_b^{-1}(e')|] \leq (5/2) \cdot \epsilon$ for every edge $e' \in D_{<i}^*$.*

Proof. Fix any edge $e' = (u, v) \in D_{<i}^*$, and let $c = c^{(t-1)}(e')$ denote its tentative color in $G^{(t-1)}$. For an edge $e \in E^{(t-1)} \cap E^{(t)}$ to be included in the set $R_b^{-1}(e')$, it needs to be a neighbor of e' with $i(e) = i$ and have the color c in its palette for round i in $G^{(t)}$. This happens iff $e \in \left((N_{i,c}^{(t)}(u) \cap S_i^{(t)}) \cup (N_{i,c}^{(t)}(v) \cap S_i^{(t)}) \right) \setminus \{e^*\}$.¹² For such an edge e , we have:

$$\begin{aligned}
\Pr[e \in R_b^{-1}(e')] &= \Pr[c^{(t)}(e) = c] = \frac{1}{|P_i^{(t)}(e)|} \\
&\leq \frac{1}{(1-\epsilon)^{2(i-1)} \cdot (1-\gamma_i) \cdot \Delta}.
\end{aligned} \tag{64}$$

In the above derivation, the second equality holds because of Lemma 4.1. The last inequality holds since conditioned on \mathcal{Z} , the event $\mathcal{E}_i^{(t)}$ occurs with probability one. Now, summing (64) over all the relevant edges e' , we get:

$$\begin{aligned}
\mathbb{E}[|R_b^{-1}(e')|] &= \sum_e \Pr[e \in R_b^{-1}(e')] \\
&\leq \frac{|N_{i,c}^{(t)}(u) \cap S_i^{(t)}| + |N_{i,c}^{(t)}(v) \cap S_i^{(t)}|}{(1-\epsilon)^{2(i-1)} \cdot (1-\gamma_i) \cdot \Delta} \\
&\leq \frac{2 \cdot (\epsilon + \epsilon^2) \cdot (1-\epsilon)^{2(i-1)} \cdot (1+\gamma_i) \cdot \Delta}{(1-\epsilon)^{2(i-1)} \cdot (1-\gamma_i) \cdot \Delta} \\
&\leq 2(\epsilon + \epsilon^2)(1+3\gamma_i) \leq (5/2) \cdot \epsilon.
\end{aligned}$$

The second inequality follows from the fact that conditioned on \mathcal{Z} , the event $\mathcal{B}_i^{(t)} \cap \mathcal{C}_i^{(t)}$ occurs with probability one. The last two inequalities follow from (1) and Corollary C.6. \square

Proof of Lemma D.3: Recall that in order to prove the lemma, all we needed to do was to prove (57). Now, (57) follows from Claim D.6, Claim D.7 and Claim D.8.

¹²Note that this specific set is completely determined once we condition upon the critical random variables.

E Useful concentration inequalities

We start by describing the standard Hoeffding bounds for mutually independent random variables.

Lemma E.1 (Hoeffding bounds). *Let X be the sum of m mutually independent random variables X_1, \dots, X_m with $X_i \in [a_i, b_i]$ for each $i \in [m]$. Then for all $t > 0$, we have:*

$$\Pr[X \geq \mathbb{E}[X] + t] \leq \exp\left(-\frac{2t^2}{\sum_i (b_i - a_i)^2}\right),$$

$$\Pr[X \leq \mathbb{E}[X] - t] \leq \exp\left(-\frac{2t^2}{\sum_i (b_i - a_i)^2}\right).$$

Throughout this paper, we denote by $x \sim_R S$ a (uniformly) random sample x from the set S . One tool we will rely on crucially for our analysis is concentration inequalities of *dependent* random variables. Specifically, we will study concentration of sums of *negatively associated* variables.

Definition E.2 ([28, 32]). *We say a joint distribution (X_1, \dots, X_n) is negatively associated (NA), or that the variables X_1, \dots, X_n are NA, if every two monotone increasing functions f and g defined on disjoint subsets of the variables in \vec{X} are negatively correlated. That is,*

$$\mathbb{E}[f \cdot g] \leq \mathbb{E}[f] \cdot \mathbb{E}[g]. \quad (65)$$

A trivial example of NA is *independent* random variables (for which (65) holds with equality.) A more interesting, useful example of NA for our use is given by the following two propositions.

Proposition E.3 (0-1 Principle [16]). *Let $X_1, \dots, X_n \in \{0, 1\}$ be binary random variables such that $\sum_i X_i \leq 1$ always. Then, the joint distribution (X_1, \dots, X_n) is NA.*

Proposition E.4 (Permutation Distributions are NA ([28])). *Let x_1, \dots, x_n be n values and let X_1, \dots, X_n be random variables taking on all permutations of (x_1, \dots, x_n) with equal probability. Then X_1, \dots, X_n are NA.*

Permutation distributions prove useful in our context due to our study of random-order streams.

More elaborate NA distributions can be obtained from simple NA distributions as those given by propositions E.3 and E.4 via the following closure properties.

Proposition E.5 (NA Closure Properties [16, 28, 32]).

1. Independent union. *If the joint distributions (X_1, \dots, X_n) and (Y_1, \dots, Y_m) are both NA and independent of each other, then the joint distribution $(X_1, \dots, X_n, Y_1, \dots, Y_m)$ is also NA.*
2. Function composition. *Let f_1, \dots, f_k be monotone (all increasing or all decreasing) functions defined on disjoint subsets of the variables in \vec{X} . Then the joint distribution $(f_1(\vec{X}), \dots, f_k(\vec{X}))$ is NA.*

An example NA distribution obtained using these closure properties are balls and bins processes.

Proposition E.6 (Balls and Bins is NA). *Suppose m balls are thrown independently into one of n bins (not necessarily u.a.r., and not necessarily i.i.d). Let B_i be the number of balls placed in bin i in this process. Then the joint distribution (B_1, B_2, \dots, B_n) is NA.*

Proof. For each $b \in [m]$ and $i \in [n]$, let $X_{b,i}$ be an indicator variable for ball b landing in bin i . By the 0-1 Principle (Proposition E.3), the variables $\{X_{b,i} \mid i \in [n]\}$ are NA. By closure of NA under independent union (Proposition E.5.1), as each ball is placed independently of all other balls, $\{X_{b,i} \mid b \in [m], i \in [n]\}$ are NA. Finally, by closure of NA under monotone increasing functions on disjoint subsets (Proposition E.5.2), the variables $B_i = \sum_b X_{b,i}$ are indeed NA. \square

As the variables $N_i = \min\{1, B_i\}$, indicating whether bin i is non-empty, are monotone increasing functions depending on disjoint subsets of the B_i variables (specifically, singletons), we obtain the following corollary.

Corollary E.7. *The indicator variables N_i for bins being non-empty in a balls and bins process as in Proposition E.6, are NA.*

A particularly useful property of NA variables is the applicability of Chernoff-Hoeffding type concentration inequalities to sums of NA variables X_1, X_2, \dots, X_n . This follows from monotonicity of the exponential function implying (by induction, using Definition E.2) that $\mathbb{E}[\exp(\lambda \cdot \sum_i X_i)] \leq \prod_i \mathbb{E}[\exp(\lambda \cdot x_i)]$ (see [16]). This is the crucial first step of proofs of such tail bounds. In particular, we will make use of the following tail bounds.

Lemma E.8 (Chernoff bounds for NA variables [16]). *Let X be the sum of NA random variables X_1, \dots, X_m with $X_i \in [0, 1]$ for each $i \in [m]$. Then for all $\delta \in (0, 1)$, and $\kappa \geq \mathbb{E}[X]$,*

$$\Pr[X \leq (1 - \delta) \cdot \mathbb{E}[X]] \leq \exp\left(-\frac{\mathbb{E}[X] \cdot \delta^2}{2}\right),$$

$$\Pr[X \geq (1 + \delta) \cdot \kappa] \leq \exp\left(-\frac{\kappa \cdot \delta^2}{3}\right).$$

Lemma E.9 (Hoeffding bounds for NA variables [16]). *Let X be the sum of m NA random variables X_1, \dots, X_m with $X_i \in [a_i, b_i]$ for each $i \in [m]$. Then for all $t > 0$, we have:*

$$\Pr[X \geq \mathbb{E}[X] + t] \leq \exp\left(-\frac{2t^2}{\sum_i (b_i - a_i)^2}\right),$$

$$\Pr[X \leq \mathbb{E}[X] - t] \leq \exp\left(-\frac{2t^2}{\sum_i (b_i - a_i)^2}\right).$$

We next describe a concentration inequality which is known as the *method of bounded differences*.

Definition E.10. [17] *Consider n sets A_1, \dots, A_n and a real-valued function $f : A_1 \times \dots \times A_n \rightarrow \mathbf{R}$. The function f satisfies the Lipschitz property with constants $\{d_i\}, i \in [n]$, iff $|f(\mathbf{a}) - f(\mathbf{a}')| \leq d_i$ whenever \mathbf{a} and \mathbf{a}' differ only in the i^{th} co-ordinate, for all $i \in [n]$.*

Lemma E.11. [17] *Consider a function $f(X_1, \dots, X_n)$ of n mutually independent random variables X_1, \dots, X_n that satisfy the Lipschitz property with $\{d_i \mid i \in [n]\}$. Then for all $t > 0$,*

$$\Pr[f \geq \mathbb{E}[f] + t] \leq \exp\left(-\frac{2t^2}{\sum_{i=1}^n d_i^2}\right),$$

$$\Pr[f \leq \mathbb{E}[f] - t] \leq \exp\left(-\frac{2t^2}{\sum_{i=1}^n d_i^2}\right).$$

Lemma E.12. Consider a function $f(X_1, \dots, X_n)$ of n mutually independent 0/1 random variables X_1, \dots, X_n that satisfy the Lipschitz property with $\{d_i\}, i \in [n]$. For each $i \in [n]$, let $X_{-i} \in \{0, 1\}^{n-1}$ denote the values taken up by every other random variable $\{X_j\}, j \in [n] \setminus \{i\}$. Furthermore, suppose that $\text{var}[f \mid X_{-i}] \leq \lambda_i$ for all $i \in [n]$ and all $X_{-i} \in \{0, 1\}^{n-1}$. Let $\lambda = \sum_{i=1}^n \lambda_i$, and $d = \max_{i \in [n]} \{d_i\}$. Then for all $t > 0$, we have:

$$\begin{aligned} \Pr[f \geq \mathbb{E}[f] + t] &\leq \exp\left(-\frac{t^2}{2\lambda + (2/3) \cdot td}\right), \\ \Pr[f \leq \mathbb{E}[f] - t] &\leq \exp\left(-\frac{t^2}{2\lambda + (2/3) \cdot td}\right). \end{aligned}$$

Proof. The lemma follows from the method of bounded variances, as explained in Chapter 8.1 of [17] (in particular, the lemma follows from equation (8.5) in this chapter). \square

References

- [1] AGGARWAL, G., MOTWANI, R., SHAH, D., AND ZHU, A. 2003. Switch scheduling via randomized edge coloring. In *Proceedings of the 44th Symposium on Foundations of Computer Science (FOCS)*. 502–512.
- [2] ALON, N. AND SPENCER, J. H. 2004. *The probabilistic method*. John Wiley & Sons.
- [3] BAHMANI, B., MEHTA, A., AND MOTWANI, R. 2012. Online graph edge-coloring in the random-order arrival model. *Theory of Computing (conference version appeared in SODA 2010)* 8, 1, 567–595.
- [4] BAR-NOY, A., MOTWANI, R., AND NAOR, J. 1992. The greedy algorithm is optimal for on-line edge coloring. *Information Processing Letters (IPL)* 44, 5, 251–253.
- [5] BARENBOIM, L. AND MAIMON, T. 2017. Fully-dynamic graph algorithms with sublinear time inspired by distributed computing. *Procedia Computer Science* 108, 89–98.
- [6] BERNSTEIN, A., HOLM, J., AND ROTENBERG, E. 2018. Online bipartite matching with amortized replacements. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, A. Czumaj, Ed. 947–959.
- [7] BHATTACHARYA, S., CHAKRABARTY, D., HENZINGER, M., AND NANONGKAI, D. 2018. Dynamic algorithms for graph coloring. In *Proceedings of the 29th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 1–20.
- [8] BOSEK, B., LENIOWSKI, D., SANKOWSKI, P., AND ZYCH, A. 2014. Online bipartite matching in offline time. In *55th IEEE Annual Symposium on Foundations of Computer Science (FOCS)*. 384–393.
- [9] CHANG, Y.-J., HE, Q., LI, W., PETTIE, S., AND UITTO, J. 2018. The complexity of distributed edge coloring with small palettes. In *Proceedings of the 29th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 2633–2652.

- [10] CHARIKAR, M. AND LIU, P. 2021. Improved algorithms for edge colouring in the w-streaming model. In *Proceedings of the 4th Symposium on Simplicity in Algorithms (SOSA)*. To appear.
- [11] COHEN, I. R., PENG, B., AND WAJC, D. 2019. Tight bounds for online edge coloring. In *Proceedings of the 60th Symposium on Foundations of Computer Science (FOCS)*. 1–25.
- [12] COHEN, I. R. AND WAJC, D. 2018. Randomized online matching in regular graphs. In *Proceedings of the 29th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 960–979.
- [13] COLE, R., OST, K., AND SCHIRRA, S. 2001. Edge-coloring bipartite multigraphs in $O(E \log D)$ time. *Combinatorica* 21, 1, 5–12.
- [14] DUAN, R., HE, H., AND ZHANG, T. 2019. Dynamic edge coloring with improved approximation. In *Proceedings of the 30th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 1937–1945.
- [15] DUBHASHI, D., GRABLE, D. A., AND PANCONESI, A. 1998. Near-optimal, distributed edge colouring via the nibble method. *Theor. Comput. Sci.* 203, 2, 225–252.
- [16] DUBHASHI, D. AND RANJAN, D. 1996. Balls and bins: A study in negative dependence. *BRICS Report Series* 3, 25.
- [17] DUBHASHI, D. P. AND PANCONESI, A. 2009. *Concentration of Measure for the Analysis of Randomized Algorithms*.
- [18] ELKIN, M., PETTIE, S., AND SU, H.-H. 2014. $(2\delta-1)$ -edge-coloring is much easier than maximal matching in the distributed setting. In *Proceedings of the 26th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 355–370.
- [19] FELDKORD, B., FELDOTTO, M., GUPTA, A., GURUGANESH, G., KUMAR, A., RIECHERS, S., AND WAJC, D. 2018. Fully-dynamic bin packing with little repacking. In *Proceedings of the 45th International Colloquium on Automata, Languages and Programming (ICALP)*. 51:1–51:24.
- [20] GAMLATH, B., KAPRALOV, M., MAGGIORI, A., SVENSSON, O., AND WAJC, D. 2019. Online matching with general arrivals. In *Proceedings of the 60th Symposium on Foundations of Computer Science (FOCS)*. 26–37.
- [21] GRABLE, D. A. 1998. A large deviation inequality for functions of independent, multi-way choices. *Combinatorics Probability and Computing* 7, 1, 57–63.
- [22] GU, A., GUPTA, A., AND KUMAR, A. 2013. The power of deferral: maintaining a constant-competitive steiner tree online. In *ACM Symposium on Theory of Computing (STOC)*. 525–534.
- [23] GUPTA, A., KRISHNASWAMY, R., KUMAR, A., AND PANIGRAHI, D. 2017. Online and dynamic algorithms for set cover. In *Proceedings of the 49th Annual ACM Symposium on Theory of Computing (STOC)*. 537–550.

- [24] GUPTA, A. AND KUMAR, A. 2014. Online steiner tree with deletions. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, C. Chekuri, Ed. 455–467.
- [25] GUPTA, A., KUMAR, A., AND STEIN, C. 2014. Maintaining assignments online: Matching, scheduling, and flows. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, (SODA)*. 468–479.
- [26] GUPTA, A. AND SINGLA, S. 2020. Random-order models. *arXiv preprint arXiv:2002.12159*.
- [27] HOLYER, I. 1981. The np-completeness of edge-coloring. *SIAM Journal on Computing (SICOMP)* 10, 4, 718–720.
- [28] JOAG-DEV, K. AND PROSCHAN, F. 1983. Negative association of random variables with applications. *The Annals of Statistics*, 286–295.
- [29] KARANDE, C., MEHTA, A., AND TRIPATHI, P. 2011. Online bipartite matching with unknown distributions. In *Proceedings of the 43rd Annual ACM Symposium on Theory of Computing (STOC)*. 587–596.
- [30] KARLOFF, H. J. AND SHMOYS, D. B. 1987. Efficient parallel algorithms for edge coloring problems. *J. Algorithms* 8, 1, 39–52.
- [31] KESSELHEIM, T., TÖNNIS, A., RADKE, K., AND VÖCKING, B. 2014. Primal beats dual on online packing lps in the random-order model. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing (STOC)*. 303–312.
- [32] KHURSHIED, A. AND LAI SAXENA, K. 1981. Positive dependence in multivariate distributions. *Communications in Statistics - Theory and Methods* 10, 12, 1183–1196.
- [33] KÖNIG, D. 1916. Über graphen und ihre anwendung auf determinantentheorie und mengenlehre. *Mathematische Annalen* 77, 4, 453–465.
- [34] KORULA, N., MIRROKNI, V., AND ZADIMOGHADDAM, M. 2018. Online submodular welfare maximization: Greedy beats 1/2 in random order. *SIAM Journal on Computing (SICOMP)* 47, 3, 1056–1086.
- [35] MAHDIAN, M. AND YAN, Q. 2011. Online bipartite matching with random arrivals: an approach based on strongly factor-revealing lps. In *Proceedings of the 43rd Annual ACM Symposium on Theory of Computing (STOC)*. 597–606.
- [36] MEGOW, N., SKUTELLA, M., VERSCHAE, J., AND WIESE, A. The power of recourse for online MST and TSP. In *Proceedings of 39th International Colloquium on Automata, Languages, and Programming, (ICALP)*. Vol. 7391. 689–700.
- [37] MEYERSON, A. 2001. Online facility location. In *Proceedings of the 42nd Symposium on Foundations of Computer Science (FOCS)*. 426–431.
- [38] MOTWANI, R., NAOR, J. S., AND NAOR, M. 1994. The probabilistic method yields deterministic parallel algorithms. *Journal of Computer and System Sciences* 49, 3, 478–516.

- [39] PETERSEN, J. 1898. Sur le théoreme de tait. *L'intermédiaire des Mathématiciens* 5, 225–227.
- [40] SU, H. AND VU, H. T. 2019. Towards the locality of vizing's theorem. In *Proceedings of the 51st Annual ACM Symposium on Theory of Computing (STOC)*. 355–364.
- [41] TAIT, P. 1880. Remarks on the colourings of maps. *Proc. R. Soc. Edinburgh* 10, 729.
- [42] VIZING, V. G. 1964. On an estimate of the chromatic class of a p-graph. *Diskret analiz* 3, 25–30.
- [43] WAJC, D. 2020. Rounding dynamic matchings against an adaptive adversary. In *Proceedings of the 52nd Annual ACM Symposium on Theory of Computing (STOC)*. 194–207.