On a Decentralized (Δ+1)-Graph Coloring Algorithm

Deeparnab Chakrabarty* Dartmouth

Paul de Supinski† Facebook

Abstract

We consider a decentralized graph coloring model where each vertex only knows its own color and whether some neighbor has the same color as it. The networking community has studied this model extensively due to its applications to channel selection, rate adaptation, etc. Here, we analyze variants of a simple algorithm of Bhartia et al. [Proc., ACM MOBIHOC, 2016]. In particular, we introduce a variant which requires only $O(n \log \Delta)$ expected recolorings that generalizes the coupon collector problem. Finally, we show that the $O(n\Delta)$ bound Bhartia et al. achieve for their algorithm still holds and is tight in adversarial scenarios.

1 Introduction

It is well known that an undirected graph $G = (V, E)$ with maximum degree Δ can be properly vertex-colored using $(\Delta+1)$ colors. The simple "greedy" algorithm makes one pass over the nodes, giving each node one of the colors not currently used by its neighbors.

Motivated by applications to channel selection for access points, Bhartia et al. [3] investigate highly constrained decentralized algorithms for graph coloring. In their setting, the only information a vertex knows at any time is its own color and whether at least one adjacent vertex has the same color. In the case of networking applications, nodes correspond to access points, colors correspond to transmission channels, and edges correspond to whether two access points interfere with each other when transmitting in the same channel. Fittingly, an access point only knows its own channel and whether some neighbor is using the same channel, which can be inferred from the resulting packet loss. Accordingly, the networking community has studied this model extensively [3, 7, 9–12, 15, 20]. This model is sometimes called the *conflict detection model* [20].

In the stylized setting below, we describe the algorithm proposed by Bhartia et al. [3], arguably the simplest and most natural one for the model. This algorithm proceeds over time and maintains a coloring $\chi_t : V \to \{1, 2, \ldots, \Delta + 1\}$. A vertex v is *conflicted* at time t if there is some neighbor u of v such that $\chi_t(u) = \chi_t(v)$.

Decentralized Coloring $(G = (V, E))$

1. Initially, every vertex v chooses a color $\chi_0(v)$ at random from $\{1, 2, \ldots, \Delta + 1\}$.
2. At each time t, a vertex v is chosen uniformly at random among all conflicted vertices.
3. v changes its color to a random color in $\{1, 2, \ldots, \Delta + 1\}$.
4. Steps 2 and 3 repeat until there are no conflicted vertices.

In the decentralized model, Bhartia et al. [3] implement this algorithm by having vertices wait random amounts of time between recolors (Steps 3 and 2). They also prove that the algorithm converges to a proper $(\Delta+1)$-coloring in $O(n\Delta)$ expected recolorings. However, our results, which now summarize, strongly suggest that this bound is *not* tight. As an introduction, consider the special case when the graph is a clique, which turns out to be trivial.

Example. Let $H_k := \sum_{i=1}^{k} \frac{1}{i}$ be the k^{th} harmonic number. On K_n, the clique of n vertices, Decentralized Coloring converges to a $(\Delta+1)$-coloring in exactly $nH_n = \Theta(n \log n)$ expected recolorings.

To see this, observe that Decentralized Coloring is essentially the coupon collector process on cliques. That is, all n vertices require different colors, and no color once in the graph can ever be fully removed from the graph. Hence, the process terminates when

*Supported in part by CCF-1813053

†Work performed as part of Dartmouth undergraduate honors thesis [8].

all n colors have been chosen exactly once. Thus, the number of recolorings (including the initial n from Step 1) is precisely the number of draws to obtain all n coupons, whose expected value is well known to be nH_n. ∎

Our first contribution is to introduce a variant of Decentralized Coloring which is easier to analyze. The sole difference is the while-loop of Step 3

Persistent Decentralized Coloring $(G = (V, E))$

1. Initially, every vertex v chooses a color $\chi_0(v)$ at random from $\{1, 2, \ldots, \Delta + 1\}$.
2. At each time t, a vertex v is chosen uniformly at random among all conflicted vertices.
3. While v is conflicted, it keeps changing to a random color in $\{1, 2, \ldots, \Delta + 1\}$.
4. Steps 2 and 3 repeat until there are no conflicted vertices.

As our second contribution, we prove the following theorem in Section 2.2.

THEOREM 1.1. *The* Persistent Decentralized Coloring *algorithm converges to a proper* $(\Delta+1)$*-coloring in* $O(n \log \Delta)$ *expected recolorings.*

For our final two contributions, we analyze adversarial variants of the two algorithms. For either algorithm, if we allow an adversary to choose the initial coloring χ_0 in Step 1, we say the algorithm uses an *adversarial start*. Similarly, if we allow an adversary to choose the conflicted vertex in Step 2, we say the algorithm uses an *adversarial order*.

REMARK 1.1. *In adversarial order* Decentralized Coloring*, the adversary could choose vertices so as to mimic* Persistent Decentralized Coloring*. Therefore, a lower bound for* Persistent Decentralized Coloring *implies a lower bound for adversarial order* Decentralized Coloring*.*

It may be interesting to ponder whether the algorithms even converge given one or both modifications.

As our third contribution, in Section 3 we show that in fact Decentralized Coloring still only requires $O(n\Delta)$ expected recolorings in the adversarial start, adversarial order case.

THEOREM 1.2. *Even with an adversarial initial coloring* χ_0 *in Step 1 and an adversarial choice of conflicted vertices in Step 2, the* Decentralized Coloring *algorithm converges to a proper* $(\Delta+1)$*-coloring in* $O(n\Delta)$ *expected recolorings.*

In other words, we achieve the same bound as Bhartia et al. [3], whose proof would only yield an $O(n\Delta^2)$ bound in this case, while forgoing the randomness from all but Step 3.

Encouraged by Theorem 1.1 and the clique example, one may simply conjecture that all variants require only $O(n \log \Delta)$ recolorings. However, our fourth contribution is a counterexample we give in Section 2.1 showing that Theorem 1.2 is tight in certain cases, even when the order of vertices is still random.

THEOREM 1.3. *With an adversarial initial coloring* χ_0 *in Step 1 and random choice of vertices in Step 2,* Persistent Decentralized Coloring *requires* $\Omega(n\Delta)$ *expected recolorings in the worst case.*

Notably, this counterexample will not apply to random order Decentralized Coloring. So, finally, we offer the following conjecture, which motivated this research, but whose proof eludes us.

CONJECTURE 1.1. *The* Decentralized Coloring *algorithm finds a proper* $(\Delta+1)$*-coloring in* $O(n \log \Delta)$ *recolorings.*

2 Persistent Decentralized Coloring

2.1 Adversarial Start, Random order

As a warmup, we begin with the counterexample proving Theorem 1.3.

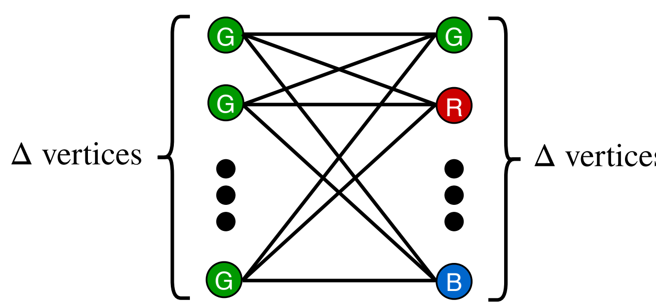

Figure 1: A bad initial coloring for Persistent Decentralized Coloring

Proof. Consider the complete bipartite graph $K_{\Delta,\Delta}$. Suppose we initially color every left side vertex with the same color, green, and use Δ colors on the right half, including the color green (see Figure 1). In this configuration, every left side vertex is conflicted, and there is only one conflicted right side vertex v. Furthermore, the left side vertices have only one free color, and hence would each expect to recolor $\Delta+1$ times if selected.

On average, we recolor half of the left side vertices before fixing the right side vertex v (at which point the process terminates). Hence, the total expected run time is $\Omega(n\Delta)$. ∎

2.2 Random Start, Random Order

In this section, we prove Theorem 1.1, restated here for convenience.

THEOREM 2.1. *The* Persistent Decentralized Coloring *algorithm converges to a proper* $(\Delta+1)$*-coloring in* $O(n \log \Delta)$ *expected recolorings.*

Our strategy is to localize our analysis to an arbitrary vertex v and then proceed by coupling. In particular, we couple v with an arbitrary vertex from the clique of size $\deg(v)$, whose behavior we understand from the coupon collector coupling.

Now, fix v to be an arbitrary vertex.

LEMMA 2.1. *The expected number of recolorings of* v *is* $\leq H_{\deg(v)}$. *The expectation is over (a) the random initial coloring* χ_0 *in Step 1, (b) the random order in which conflicted vertices are picked in Step 2, and (c) the randomness in the recoloring in Step 3.*

Lemma 2.1 immediately implies Theorem 1.1 because the expected number of recolorings of any vertex is thus $\leq H_\Delta = O(\log \Delta)$. Again, it is essential to take the expectation over the random initial colorings χ_0, otherwise Figure 1 would act as a counterexample.

Proof. We begin by setting some notation. As usual, we let $\Gamma(v)$ denote the neighborhood of v, not including v itself. For simplicity let $d := \deg(v)$, and $D := \Delta + 1$. Next, observe that Step 2 in Persistent Decentralized Coloring can be simulated by first selecting a random permutation π of the vertices, and then selecting the conflicted vertices in the π order. This is valid because no vertex can ever be chosen twice in Step 2 (unlike in Decentralized Coloring). Recall that χ_0 is the initial coloring of the graph.

Given π and χ_0, we define $\text{recolors}_{\pi,\chi_0}(v)$ to be the random variable indicating the number of times v recolors given that the initial coloring was χ_0 and the order of vertices was π. Note that the randomness of $\text{recolors}_{\pi,\chi_0}(v)$ arises solely from Step 3 of Persistent Decentralized Coloring. Our goal is to bound $\mathbb{E}_{\pi,\chi_0}\mathbb{E}[\text{recolors}_{\pi,\chi_0}(v)]$, the expected number of recolors of v, averaged over all π and χ_0.

Given π, let $B_\pi(v)$ and $A_\pi(v)$ denote the subsets of $\Gamma(v)$ which come *before* and *after* v in the permutation π, respectively. Let $\text{free}_{\pi,\chi_0}(v)$ be the random variable denoting the number of colors *not* used by $\Gamma(v)$ when v begins recoloring, given that the initial coloring is χ_0 and the order is π.

We now proceed with a coupling argument. Let $v_1, v_2, \ldots, v_d$ be an arbitrary labeling of $\Gamma(v)$. Let K denote the $(d+1)$-sized clique with vertices arbitrarily labeled $w, w_1, \ldots, w_d$. We now consider the Persistent Decentralized Coloring process on K, but using D colors even if $d < \Delta$. We couple the initial coloring χ_0' with χ_0, and the order π' with π. In particular, let $\chi_0'(w) = \chi_0(v)$ and $\chi_0'(w_i) = \chi_0(v_i)$ for all $1 \leq i \leq d$. Let the order π' of $\{w, w_1, \ldots, w_d\}$ be equal to the order π restricted to $\{v, v_1, \ldots, v_d\}$, with the same vertex pairings as before. To be clear, $A_{\pi'}(w) = \{w_i : v_i \in A_\pi(v)\}$, and $B_{\pi'}(w) = \{w_i : v_i \in B_\pi(v)\}$. Finally, $\text{free}_{\pi',\chi_0'}(w)$ is the number of colors *not* used by $\Gamma(w)$ when w begins recoloring.

LEMMA 2.2. *For any* χ_0 *and* π, *we have* $\mathbb{E}[\text{recolors}_{\pi,\chi_0}(v)] \leq \mathbb{E}[\text{recolors}_{\pi',\chi_0'}(w)]$.

Proof. Observe that v has to recolor iff[1] $\chi_0(v) \in \chi_0(A_\pi(v))$. This is because each vertex of $B_\pi(v)$ necessarily fixes to a different color than v's color (which is still $\chi_0(v)$) prior to v's turn. Thus, the only way v could be still be conflicted is if $\chi_0(v) \in \chi_0(A_\pi(v))$. Similarly, w has to recolor iff $\chi_0'(w) \in \chi_0'(A_{\pi'}(w))$. But $\chi_0(A_\pi(v)) = \chi_0'(A_{\pi'}(w))$ and $\chi_0(v) = \chi_0'(w)$, so $\text{recolors}(v) > 0$ iff $\text{recolors}(w) > 0$ under our coupling.

Next, observe that $\text{free}_{\pi,\chi_0}(v) \geq \text{free}_{\pi',\chi_0'}(w)$ under our coupling. To see this, note that $\text{free}_{\pi,\chi_0}(v)$ is equivalently the total number of colors, D, minus the number of different colors used by $\Gamma(v)$ when v begins recoloring. In the case of the $(d+1)$-clique K, when w begins recoloring, we can guarantee that the set of colors used by $B_{\pi'}(w)$ has size $|B_{\pi'}(w)|$ and is disjoint from the set of colors used by $A_{\pi'}(w)$, because the vertices are all connected. So $A_\pi(v)$ and $A_{\pi'}(w)$ use the same colors, and $B_{\pi'}(w)$ uses at least as many additional colors as $B_\pi(v)$.

Finally, note that $\text{recolors}_{\pi,\chi_0}(v)$ is either 0 or the geometric random variable whose probability

[1] We use the notation $\chi_0(S) := \{\chi_0(z) : z \in S\}$.

parameter is $\text{free}_{\pi,\chi_0}(v)/D$, with a similar statement for w. Thus,

$$\begin{aligned}
&\mathbb{E}\left[\text{recolors}_{\pi,\chi_0}(v)\right] \\
&= \mathbb{E}\left[\mathbb{1}_{\{\text{recolors}_{\pi,\chi_0}(v)>0\}} \cdot \frac{D}{\text{free}_{\pi,\chi_0}(v)}\right] \\
&\leq \mathbb{E}\left[\mathbb{1}_{\{\text{recolors}_{\pi',\chi_0'}(w)>0\}} \cdot \frac{D}{\text{free}_{\pi',\chi_0'}(w)}\right] \\
&= \mathbb{E}\left[\text{recolors}_{\pi',\chi_0'}(w)\right].
\end{aligned}$$

∎

LEMMA 2.3. *For any $1 \leq i \leq d$, we have*

$$\mathbb{E}_{\pi',\chi_0'}\mathbb{E}[\text{recolors}_{\pi',\chi_0'}(w)] = \mathbb{E}_{\pi',\chi_0'}\mathbb{E}[\text{recolors}_{\pi',\chi_0'}(w_i)]$$

Proof. This is by symmetry of the clique. It may be instructive to point out that the randomness of χ_0' and π' are both necessary. For example, if we fix an initial coloring χ_0', then no initially happy vertex ever recolors. If we fix an ordering π', then the last vertex never recolors. ∎

Lemma 2.2 and Lemma 2.3 together imply that

$$\begin{aligned}
&\mathbb{E}_{\pi,\chi_0}\mathbb{E}[\text{recolors}_{\pi,\chi_0}(v)] \\
&\quad \leq \mathbb{E}_{\pi',\chi_0'}\mathbb{E}[\text{recolors}_{\pi',\chi_0'}(w)] \\
&\quad = \frac{1}{d+1}\sum_{x\in K}\mathbb{E}_{\pi',\chi_0'}\mathbb{E}[\text{recolors}_{\pi',\chi_0'}(x)]. \qquad (2.1)
\end{aligned}$$

We already know how to bound the sum in Equation (2.1). It is precisely the expected total number of recolors of Persistent Decentralized Coloring on a $(d{+}1)$-clique, but with D colors available. In fact, for *any* π' and χ_0', we know that

$$\sum_{x\in K}\mathbb{E}[\text{recolors}_{\pi',\chi_0'}(x)]$$
$$\leq \frac{D}{D-1} + \cdots + \frac{D}{D-d} \qquad (2.2)$$
$$\leq (d+1)H_d. \qquad (2.3)$$

This follows from another coupon collector argument. Here, Equation (2.2) represents the time to collect $d{+}1$ out of D coupons, and Equation (2.3) represents the time to collect $d{+}1$ out of $d{+}1$ coupons. (We can omit the leading 1 in both sums because the initial coloring includes at least a first color.) Simple manipulation shows that $\frac{D}{D-i} \leq \frac{(d+1)}{d+1-i}$ when $i \geq 0$, because $D \geq d{+}1$. Thus, the second inequality holds. Together, Equation (2.1) and Equation (2.3) imply Lemma 2.1. ∎

Because v was an arbitrary vertex, Theorem 1.1 follows from Lemma 2.1 and linearity of expectation.

3 Decentralized Coloring with Adversarial Start and Order

In this section, we prove Theorem 1.2, restated here for convenience.

THEOREM 3.1. *Even with an adversarial initial coloring χ_0 in Step 1 and an adversarial choice of conflicted vertices in Step 2, the* Decentralized Coloring *algorithm converges to a proper $(\Delta{+}1)$-coloring in $O(n\Delta)$ expected recolorings.*

Recall that $D := \Delta{+}1$. Before we begin, observe that this bound is fairly trivial for Persistent Decentralized Coloring: in Step 3, there is always at least a $\frac{1}{D}$ chance that recoloring satisfies the chosen vertex, implying that each vertex recolors at most D times in expectation. However, in Decentralized Coloring, there is no similar concept of vertices becoming fixed. Instead, our strategy is to analyze the rate at which Decentralized Coloring drifts toward convergence.

One way to analyze drift is with a potential argument. This entails defining a potential function which monotonically changes in expectation with each iteration of the algorithm.

In our case, we define a potential function Φ on graph colorings χ such that χ is valid iff $\Phi(\chi)$ is some value λ. Then, we show that $\mathbb{E}[\Phi(\chi_t)]$ converges toward λ monotonically in expectation at a bounded rate as t increases. Indeed, this is the approach of Bhartia et al. [3], who choose $\Phi(\chi)$ to denote the number of conflicted edges in G with respect to χ, in which case $\lambda = 0$. (An edge is conflicted iff its end points have the same color.) It is easy to show that the expected number of conflicted edges decreases by at least $1/D$ with each recoloring. If the initial coloring χ_0 is random, then it is easy to see that $\mathbb{E}[\Phi(\chi_0)] = O(n)$, which in turn implies[2] that the expected number of recolorings is $O(n\Delta)$. Unfortunately, with an adversarial start, this particular argument only implies an $O\left(n\Delta^2\right)$ bound because there can be $\Omega\left(n\Delta\right)$ conflicted edges initially.

The other obvious choice for $\Phi\left(\chi\right)$ is the number of conflicted vertices in G under χ. However, we can concoct examples where we would actually expect $\Phi\left(\chi_t\right)$ to increase, given an adversarial selection. For example, if we recolor v in Figure 2, the number of conflicted vertices increases (additively) by 1/4, on average.

[2] To make this formal, one needs to use a stopping theorem which Bhartia et al. [3] do not explicitly mention. We prove and use such a theorem explicitly.

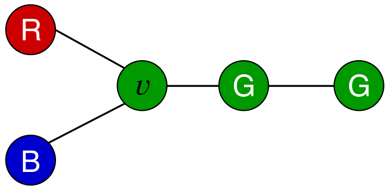

Figure 2: An example of a graph in which the number of conflicted vertices would be expected to increase, given an adversarial selection. There are 4 colors available: R, G, B and Y. If v, whose color is G, recolors to G, then the number of conflicted vertices remains the same. If v recolors to Y, then the number decreases by 1. However, if v recolors to R or B, then the number *increases* by 1.

To achieve our $O(n\Delta)$ upper bound, we define Φ such that $\Phi(\chi)$ is the number of *monochromatic connected components* in G under χ. That is, $\Phi(\chi)$ is the number of connected components induced by the vertices of the same color, taken over all colors. For example, in Figure 2, there are three monochromatic components. Note that χ is a proper coloring iff $\Phi(\chi) = n$, because the monochromatic components all need to be singletons.

LEMMA 3.1. *For $t > 0$, let χ_t be the coloring after the t^{th} recoloring in* Decentralized Coloring. *Then,*

$$\mathbb{E}[\Phi(\chi_t) - \Phi(\chi_{t-1}) \mid \chi_{t-1} \text{ invalid }] \geq \frac{1}{D},$$

where the expectation is over the randomness in Step 3 for the vertex chosen at the t^{th} recoloring.

Proof. Let v be the vertex which recolors at time t (so that χ_t has the new color of v), where v could be chosen adversarially. Given a color $c \in \{1, 2, \ldots, D\}$ define $m_t(c)$ to be the number of monochromatic components with respect to χ_t of color c which contain *at least* one vertex from $\Gamma(v) \cup \{v\}$. We exclude components of color c which have no vertices in $\Gamma(v) \cup \{v\}$ because v's recoloring cannot affect those components. Similarly define $m_{t-1}(c)$. Observe that

$$\Phi(\chi_t) - \Phi(\chi_{t-1}) = \sum_{c\in[D]} m_t(c) - m_{t-1}(c). \tag{3.4}$$

With this, the proof follows from three observations. First, $m_t(\chi_t(v)) = 1$ because all adjacent components of v's color are connected through v. Second, for any *other* color $c \neq \chi_t(v)$, we have $m_t(c) - m_{t-1}(c) \geq 0$ (meaning improvement). This is because recoloring v to a color besides c cannot possibly create any new paths of color c. Third, $\sum_{c\in[D]} m_{t-1}(c) \leq \Delta$. This is because v is conflicted with respect to χ_{t-1} and hence has the same color as one of its neighbors.

Therefore,

$$\begin{aligned}
&\mathbb{E}\left[\Phi(\chi_t) - \Phi(\chi_{t-1}) \mid \chi_{t-1} \text{ invalid}\right] \\
&\quad = \mathbb{E}\left[\sum_{c\in[D]} m_t(c) - m_{t-1}(c) \,\middle|\, \chi_{t-1}\right] \\
&\quad = \sum_{c\in[D]} \Pr[\chi_t(v) = c] \cdot (m_t(c) - m_{t-1}(c)) \qquad (3.5)\\
&\qquad + \Pr[\chi_t(v) \neq c] \cdot (m_t(c) - m_{t-1}(c)) \\
&\quad \geq \sum_{c\in[D]} \Pr[\chi_t(v) = c] \cdot (1 - m_{t-1}(c)) \qquad (3.6)\\
&\quad = \frac{1}{D} \cdot \sum_{c\in[D]} (1 - m_{t-1}(c)) \\
&\quad \geq 1 - \frac{\Delta}{D} = \frac{1}{D}. \qquad (3.7)
\end{aligned}$$

Recall that the expectation is over the random recoloring of v at time t. We lose the conditioning on χ_{t-1} in Equation (3.5) because the new color is independently random. Also note that m_{t-1} is fixed once we know χ_{t-1}. Equation (3.5) follows from Equation (3.4). Equation (3.6) follows from the first two observations mentioned after Equation (3.4). Equation (3.7) follows from the third observation and the fact that $D = \Delta + 1$. ∎

Using Lemma 3.1 to prove that the expected stopping time $\tau(G)$ of our process is $O(n\Delta)$ requires one more theorem. In particular, we use the following adjusted version[3] of Wald's equation [22]. We also note that this allows one to provide a formal proof of the Bhartia et al. [3] claim.

LEMMA 3.2. *Let Φ be a real-valued function of colorings for a graph G such that χ is valid iff $\Phi(\chi) = \lambda$, for some constant λ. Let χ_t be the state of χ after t recolorings. Let $\tau(G)$ be the random number of recolorings required to produce a valid coloring of G. If $\mathbb{E}\left[|\lambda - \Phi(\chi_{t-1})| - |\lambda - \Phi(\chi_t)| \,\middle|\, \chi_{t-1} \text{ invalid}\right] \geq C$ for some positive constant C, then $\mathbb{E}[\tau(G)] \leq \mathbb{E}[|\lambda - \Phi(\chi_0)|]/C$.*

We first show how Lemma 3.1 and Lemma 3.2 imply Theorem 1.2. Recall that χ is valid iff $\Phi(\chi) =$

[3] We don't claim novelty here; a theorem similar to Lemma 3.2 may exist elsewhere.

n. We have

$$\begin{aligned}&\mathbb{E}\left[|n-\Phi(\chi_{t-1})|-|n-\Phi(\chi_t)| \;\middle|\; \chi_{t-1} \text{ invalid}\right]\\ &\quad=\mathbb{E}\left[\Phi(\chi_t)-\Phi(\chi_{t-1}) \;\middle|\; \chi_{t-1} \text{ invalid}\right]\\ &\quad\geq \frac{1}{D}.\end{aligned}$$

Because $1 \leq \Phi(\chi) \leq n$, we have $\mathbb{E}[|n-\Phi(\chi_0)|] \leq n-1$. Hence, $\mathbb{E}[\tau(G)] \leq (n-1)D$. This completes the proof of Theorem 1.2. □

We now prove Lemma 3.2.

Proof. For each $t \in \mathbb{Z}^+$, let

$$Z_t := \begin{cases} |\lambda-\Phi(\chi_{t-1})|-|\lambda-\Phi(\chi_t)| & t \leq \tau(G) \\ 0 & \text{otherwise.}\end{cases}$$

By assumption, $\Phi(\chi_{\tau(G)}) = \lambda$. Hence, $\sum_{t=1}^{\tau(G)} Z_t$ telescopes to $|\lambda-\Phi(\chi_0)|$. From here, we see that

$$\begin{aligned}&\mathbb{E}[|\lambda-\Phi(\chi_0)|]\\ &\quad=\mathbb{E}\left[\sum_{t=1}^{\tau(G)} Z_t\right]\\ &\quad=\mathbb{E}\left[\sum_{t=1}^{\infty} Z_t \cdot \mathbb{1}_{\{\tau(G)\geq t\}}\right]. \qquad (3.8)\end{aligned}$$

Next, we prepare to apply the following analogue of linearity of expectation for infinite sums.

THEOREM 3.2.
(INFINITE LINEARITY OF EXPECTATION [18])

Let $X_1, X_2, \ldots$ be random variables. If $\sum_{t=1}^{\infty}\mathbb{E}[|X_t|]$ converges, then

$$\mathbb{E}\left[\sum_{t=1}^{\infty} X_t\right]=\sum_{t=1}^{\infty}\mathbb{E}[X_t].$$

To apply Theorem 3.2, we need to show that $\sum_{t=1}^{\infty}\mathbb{E}\left[\left|Z_t \cdot \mathbb{1}_{\{\tau(G)\geq t\}}\right|\right]$ converges. Observe that Φ must be bounded, because it is real-valued and there are only finitely many possible colorings of G. Thus, $|Z_t| \leq \rho$ for some constant ρ. Hence,

$$\begin{aligned}(3.9)\qquad &\sum_{t=1}^{\infty}\mathbb{E}\left[\left|Z_t \cdot \mathbb{1}_{\{\tau(G)\geq t\}}\right|\right]\\ &\quad=\sum_{t=1}^{\infty}\mathbb{E}\left[|Z_t| \cdot \mathbb{1}_{\{\tau(G)\geq t\}}\right]\\ &\quad\leq \rho\sum_{t=1}^{\infty}\mathbb{E}\left[\mathbb{1}_{\{\tau(G)\geq t\}}\right]\\ &\quad=\rho\cdot\mathbb{E}[\tau(G)].\end{aligned}$$

Trivially, we can bound $\mathbb{E}[\tau(G)]$ by nD^n, because at worst we need to select the lone satisfying color for n consecutive vertices, which can be cast as a geometric random variable with probability $\left(\frac{1}{D}\right)^n$ that uses at most n recolors per trial. Hence, $\mathbb{E}[\tau(G)]$ is finite.

Because Equation (3.9) has positive terms and is bounded above, it indeed converges. Thus, we have

$$\begin{aligned}&\mathbb{E}\left[\sum_{t=1}^{\infty} Z_t \cdot \mathbb{1}_{\{\tau(G)\geq t\}}\right]\\ &\quad=\sum_{t=1}^{\infty}\mathbb{E}\left[Z_t \cdot \mathbb{1}_{\{\tau(G)\geq t\}}\right] \qquad \textit{Theorem 3.2}\\ &\quad=\sum_{t=1}^{\infty}\mathbb{E}[Z_t \mid \tau(G)\geq t]\cdot\Pr[\tau(G)\geq t]\\ &\quad\geq C\sum_{t=1}^{\infty}\Pr[\tau(G)\geq t]\\ &\quad=C\cdot\mathbb{E}[\tau(G)].\end{aligned}$$

With Equation (3.8), we have $\mathbb{E}[\tau(G)] \leq \mathbb{E}[|\lambda-\Phi(\chi_0)|]/C$, which completes the proof of Lemma 3.2. □

4 Related Work

The decentralized model of graph coloring has been studied extensively [3, 7, 9–12, 15, 20]. Most of this work appears in the networking literature, motivated by the need to minimize the communication between nodes. We mention two works which are most similar to our work, and comment on the differences. The first work is by Motskin et al. [20] which considers an algorithm very similar to Decentralized Coloring, except that all conflicted vertices simultaneously randomly recolor. It is not too hard to show that a $\frac{1}{\Delta+1}$-fraction of the nodes become happy in each round, and therefore, $O(\Delta \log n)$-rounds suffice with high probability. Still, this leads to $O(n\Delta)$ recolorings, which is no better than what Bhartia et al. [3] achieve (note that the first random recoloring constitutes a random start). The second work is by Checco and Leith [7], which itself generalizes works by Barcelo et al. [2] and Duffy et al. [9, 10], where again all conflicted vertices recolor simultaneously, but according to a distribution that evolves with time. Their algorithms, which also converge in $O(n \log n)$ rounds, are robust to changes in the graph.

In this paragraph, we describe other decentralized models which are unrelated to the model we study but may be interesting to the reader. Synchronous graph coloring in minimal number of rounds arises in *distributed computing*. Unlike our decentralized setting, this model, first defined in Linial's seminal paper [16], allows nodes to pass messages among each other, and the number of rounds is one key complexity parameter. Johansson [21] obtains an $(\Delta+1)$-coloring in $O(\log n)$ rounds using a simple algorithm is similar to, and indeed inspired by, Luby's MIS algorithm [17]. This was recently improved by Harris, Schneider and Su [13] to a $O(\sqrt{\log n})$-round algorithm, and more recently to a $O(\text{polyloglog } n)$-round algorithm by Chang et al. [6]. $(\Delta+1)$-colorings have also recently been considered in the streaming model by Assadi et al. [1] where edges stream in and there is only $\tilde{O}(n)$-space available to help maintain a $(\Delta+1)$-coloring. The same paper also gives algorithms in the graph-query and MPC (massively parallel computation) models. $(\Delta+1)$-coloring has also recently been considered in the *dynamic graph model* where edges may be added or deleted and the objective is to maintain a $(\Delta+1)$-coloring with quick updates. Bhattacharya et al. [4] describe a randomized algorithm with $O(\log n)$ amortized update time which has very recently been improved upon by Henzinger and Peng [14], and Bhattacharya et al. [5].

5 Conclusion and Discussion

In this paper we considered variants of two decentralized graph coloring algorithms: Decentralized Coloring, introduced by Bhartia et al. [3], and Persistent Decentralized Coloring, our proposed modification. Beyond the Persistent Decentralized Coloring algorithm itself, we produced three primary contributions:

- Adversarial start, random order Persistent Decentralized Coloring requires $\Omega(n\Delta)$ expected recolorings in the worst case.
- Random start, random order Persistent Decentralized Coloring requires only $O(n \log \Delta)$ expected recolorings.
- Adversarial start, adversarial order Decentralized Coloring requires only $O(n\Delta)$ recolorings.

We proved the first result with a counterexample involving a bipartite graph, the second through coupling and generalization to the coupon collector problem, and the third through analysis of an interesting potential function. We also note that our stopping theorem may be extensible to other stochastic processes that tend to drift toward convergence. Lastly, we formalized the conjecture that the $O(n \log \Delta)$ bound holds for random start, random order Decentralized Coloring.

It is perhaps instructive to remark that Decentralized Coloring resembles, in spirit, the celebrated Moser-Tardos [19] randomized algorithm for finding satisfying assignments to CSPs obeying the Lovász Local Lemma. Indeed, if we tweak the Decentralized Coloring algorithm so that we pick a conflicted *edge* in every round and randomly recolor its endpoints, we get the Moser-Tardos algorithm. However, $(\Delta+1)$-coloring is outside the LLL regime: the probability of a bad event (of getting an unhappy edge) is $p = \frac{1}{\Delta+1}$, and the degree d of the dependency graph is $2\Delta-2$ (a single edge is independent of all but its $2\Delta-2$ neighboring edges), giving $pd \approx 2$, which causes the general analysis of [19] to break down.

Our hope is that our new understanding of Persistent Decentralized Coloring may lead to a proof of Conjecture 1.1. A first question to answer may be whether Persistent Decentralized Coloring always requires more expected recolorings than Decentralized Coloring on any particular graph. We also wonder whether understanding random start, adversarial order variants could help solve the mystery.

References

[1] Sepehr Assadi, Yu Chen, and Sanjeev Khanna. Sublinear algorithms for $(\Delta + 1)$ vertex coloring. In *Proc., ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 767–786, 2019.

[2] J. Barcelo, B. Bellalta, C. Cano, and M. Oliver. Learning-BEB: Avoiding collisions in WLANs. *Carrier Sense Multiple Access with Enhanced Collision Avoidance*, page 23, 2009.

[3] Apurv Bhartia, Deeparnab Chakrabarty, Krishna Chintalapudi, Lili Qiu, Bozidar Radunovic, and Ramachandran Ramjee. IQ-hopping: Distributed oblivious channel selection for wireless networks.

In *Proc., ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC)*, pages 81–90, 2016.

[4] Sayan Bhattacharya, Deeparnab Chakrabarty, Monika Henzinger, and Danupon Nanongkai. Dynamic algorithms for graph coloring. In *Proc., ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1–20, 2018.

[5] Sayan Bhattacharya, Fabrizio Grandoni, Janardhan Kulkarni, Quanquan C. Liu, and Shay Solomon. Fully dynamic $(\delta + 1)$-coloring in constant update time. *https://arxiv.org/abs/1910.02063*, 2019.

[6] Yi-Jun Chang, Wenzheng Li, and Seth Pettie. An optimal distributed $(\Delta + 1)$-coloring algorithm? In *Proc., ACM Symposium on Theory of Computing (STOC)*, pages 445–456, 2018.

[7] Alessandro Checco and Doug J. Leith. Fast, responsive decentralized graph coloring. *IEEE/ACM Trans. Netw.*, 25(6):3628–3640, December 2017.

[8] Paul B. de Supinski. Convergence times of decentralized graph coloring algorithms. Technical Report TR2019-864, Dartmouth College, Computer Science, Hanover, NH, May 2019.

[9] Ken R. Duffy, Charles Bordenave, and Douglas J. Leith. Decentralized constraint satisfaction. *IEEE/ACM Trans. Netw.*, 21(4):1298–1308, August 2013.

[10] Ken R. Duffy, Neil O'Connell, and Artëm Sapozhnikov. Complexity analysis of a decentralised graph colouring algorithm. *Inform. Process. Lett.*, 107(2):60–63, July 2008.

[11] Stephen Fitzpatrick and Lambert Meertens. Soft, real-time, distributed graph coloring using decentralized, synchronous, stochastic, iterative-repair, anytime algorithms. *Kestrel Institute Technical Report*, May 2001.

[12] Severino F. Galán. Simple decentralized graph coloring. *Computational Optimization and Applications*, 66(1):163–185, Jan 2017.

[13] David G. Harris, Johannes Schneider, and Hsin-Hao Su. Distributed $(\Delta + 1)$-coloring in sublogarithmic rounds. In *Proc., ACM Symposium on Theory of Computing (STOC)*, pages 465–478, 2016.

[14] Monika Henzinger and Pan Peng. Constant-time dynamic $(\Delta + 1)$-coloring and weight approximation for minimum spanning forest: Dynamic algorithms meet property testing. *arXiv preprint arXiv:1907.04745*, 2019.

[15] Douglas Leith and Peter Clifford. Convergence of distributed learning algorithms for optimal wireless channel allocation. In *Proc., IEEE Conference on Decision and Control (CDC)*, pages 2980 – 2985, 01 2007.

[16] Nathan Linial. Locality in distributed graph algorithms. *SIAM Journal on Computing (SICOMP)*, 21(1):193–201, 1992.

[17] Michael Luby. A simple parallel algorithm for the maximal independent set problem. *SIAM Journal on Computing (SICOMP)*, 15(4):1036–1053, 1986.

[18] Michael Mitzenmacher and Eli Upfal. *Probability and Computing: Randomized Algorithms and Probabilistic Analysis*. Cambridge University Press, New York, NY, USA, 2005.

[19] Robin A. Moser and Gábor Tardos. A constructive proof of the general Lovász local lemma. *Journal of the ACM*, 57(2):11, 2010.

[20] Arik Motskin, Tim Roughgarden, Primoz Skraba, and Leonidas Guibas. Lightweight coloring and desynchronization for networks. In *Proc., IEEE INFOCOM*, pages 2383–2391. IEEE, 2009.

[21] Öjvind Johansson. Simple distributed $(\Delta+1)$-coloring of graphs. *Inform. Process. Lett.*, 70:229–232, 1999.

[22] Abraham Wald. Some generalizations of the theory of cumulative sums of random variables. *Ann. Math. Statist.*, 16(3):287–293, 09 1945.